

ECMA-376, 4th Edition

**Office Open XML File Formats — Transitional
Migration Features**

December 2012

Table of Contents

Foreword	xiii
Introduction	xiv
1. Scope	1
2. Conformance	2
2.1 Document Conformance.....	2
2.2 Application Conformance	2
3. Normative References	4
4. Terms and Definitions	8
5. Notational Conventions	11
6. Acronyms and Abbreviations	12
7. General Description	13
8. Additional Shared Parts	15
8.1 VML Drawing Part	15
9. WordprocessingML	17
9.1 Part Summary (Part 1, §11.3)	17
9.1.1 Alternative Format Import Part (Part 1, §11.3.1)	17
9.1.2 Comments Part (Part 1, §11.3.2)	17
9.1.3 Document Settings Part (Part 1, §11.3.3)	17
9.1.4 Endnotes Part (Part 1, §11.3.4).....	17
9.1.5 Fonts Table Part (Part 1, §11.3.5)	17
9.1.6 Footer Part (Part 1, §11.3.6)	17
9.1.7 Footnotes Part (Part 1, §11.3.7).....	18
9.1.8 Glossary Document Part (Part 1, §11.3.8)	18
9.1.9 Header Part (Part 1, §11.3.9)	18
9.1.10 Main Document Part (Part 1, §11.3.10).....	18
9.1.11 Numbering Definitions Part (Part 1, §11.3.11)	18
9.1.12 Style Definitions Part (Part 1, §11.3.12).....	18
9.1.13 Web Settings Part (Part 1, §11.3.13).....	18
9.2 Document Template (Part 1, §11.4)	19
9.3 Framesets (Part 1, §11.5).....	19
9.4 Master Documents and Subdocuments (Part 1, §11.6)	19
9.5 Mail Merge Data Source (Part 1, §11.7)	19
9.6 Mail Merger Header Data Source (Part 1, §11.8)	19
9.7 XSL Transformation (Part 1, §11.9).....	19
10. SpreadsheetML	20
10.1 Part Summary (Part 1, §12.3)	20
10.1.1 Calculation Chain Part (Part 1, §12.3.1).....	20
10.1.2 Chartsheet Part (Part 1, §12.3.2)	20
10.1.3 Comments Part (Part 1, §12.3.3)	20

10.1.4	Connections Part (Part 1, §12.3.4).....	20
10.1.5	Custom Property Part (Part 1, §12.3.5).....	20
10.1.6	Custom XML Mappings Part (Part 1, §12.3.6).....	21
10.1.7	Dialogsheet Part (Part 1, §12.3.7).....	21
10.1.8	Drawings Part (Part 1, §12.3.8).....	21
10.1.9	External Workbook References Part (Part 1, §12.3.9).....	21
10.1.10	Metadata Part (Part 1, §12.3.10).....	21
10.1.11	Pivot Table Part (Part 1, §12.3.11).....	21
10.1.12	Pivot Table Cache Definition Part (Part 1, §12.3.12).....	21
10.1.13	Pivot Table Cache Records Part (Part 1, §12.3.13).....	22
10.1.14	Query Table Part (Part 1, §12.3.14).....	22
10.1.15	Shared Strings Table Part (Part 1, §12.3.15).....	22
10.1.16	Shared Workbook Revision Headers Part (Part 1, §12.3.16).....	22
10.1.17	Shared Workbook Revision Log Part (Part 1, §12.3.17).....	22
10.1.18	Shared Workbook User Data part (Part 1, §12.3.18).....	22
10.1.19	Single Cell Table Definitions Part (Part 1, §12.3.19).....	22
10.1.20	Styles Part (Part 1, §12.3.20).....	23
10.1.21	Table Definition Part (Part 1, §12.3.21).....	23
10.1.22	Volatile Dependencies Part (Part 1, §12.3.22).....	23
10.1.23	Workbook Part (Part 1, §12.3.23).....	23
10.1.24	Worksheet Part (Part 1, §12.3.24).....	23
10.2	External Workbooks (Part 1, §12.4).....	23
11.	PresentationML.....	24
11.1	Part Summary (Part 1, §13.3).....	24
11.1.1	Comment Authors Part (Part 1, §13.3.1).....	24
11.1.2	Comments Part (Part 1, §13.3.2).....	24
11.1.3	Handout Master Part (Part 1, §13.3.3).....	24
11.1.4	Notes Master Part (Part 1, §13.3.4).....	24
11.1.5	Notes Slide Part (Part 1, §13.3.5).....	24
11.1.6	Presentation Part (Part 1, §13.3.6).....	25
11.1.7	Presentation Properties Part (Part 1, §13.3.7).....	25
11.1.8	Slide Part (Part 1, §13.3.8).....	25
11.1.9	Slide Layout Part (Part 1, §13.3.9).....	25
11.1.10	Slide Master Part (Part 1, §13.3.10).....	25
11.1.11	Slide Synchronization Data Part (Part 1, §13.3.11).....	25
11.1.12	User Defined Tags Part (Part 1, §13.3.12).....	25
11.1.13	View Properties Part (Part 1, §13.3.13).....	26
11.2	HTML Publish Location (Part 1, §13.4).....	26
11.3	Slide Synchronization Server Location (Part 1, §13.5).....	26
12.	DrawingML.....	27
12.1	Part Summary (Part 1, §14.2).....	27
12.1.1	Chart Part (Part 1, §14.2.1).....	27
12.1.2	Chart Drawing Part (Part 1, §14.2.2).....	27
12.1.3	Diagram Colors Part (Part 1, §14.2.3).....	27
12.1.4	Diagram Data Part (Part 1, §14.2.4).....	27
12.1.5	Diagram Layout Definition Part (Part 1, §14.2.5).....	27
12.1.6	Diagram Style Part (Part 1, §14.2.6).....	28

12.1.7	Theme Part (Part 1, §14.2.7).....	28
12.1.8	Theme Override Part (Part 1, §14.2.8).....	28
12.1.9	Table Styles Part (Part 1, §14.2.9).....	28
13.	Shared MLs.....	29
13.1	Part Summary (Part 1, §15.2).....	29
13.1.1	Additional Characteristics Part (Part 1, §15.2.1).....	29
13.1.2	Audio Part (Part 1, §15.2.2).....	29
13.1.3	Bibliography Part (Part 1, §15.2.3).....	29
13.1.4	Content Part (Part 1, §15.2.4).....	29
13.1.5	Custom XML Data Storage Part (Part 1, §15.2.5).....	29
13.1.6	Custom XML Data Storage Properties Part (Part 1, §15.2.6).....	29
13.1.7	Embedded Control Persistence Part (Part 1, §15.2.9).....	30
13.1.8	Embedded Object Part (Part 1, §15.2.10).....	30
13.1.9	Embedded Package Part (Part 1, §15.2.11).....	30
13.1.10	Core File Properties Part (Part 1, §15.2.12.1).....	30
13.1.11	Custom File Properties Part (Part 1, §15.2.12.2).....	30
13.1.12	Extended File Properties Part (Part 1, §15.2.12.3).....	30
13.1.13	Font Part (Part 1, §15.2.13).....	30
13.1.14	Image Part (Part 1, §15.2.14).....	30
13.1.15	Printer Settings Part (Part 1, §15.2.15).....	30
13.1.16	Thumbnail Part (Part 1, §15.2.16).....	31
13.1.17	Video Part (Part 1, §15.2.17).....	31
13.2	Hyperlinks Part (Part 1, §15.3).....	31
14.	WordprocessingML Reference Material.....	32
14.1	Table of Contents.....	32
14.2	Paragraphs and Rich Formatting.....	35
14.2.1	Paragraphs.....	35
14.2.2	Run Content.....	37
14.3	Tables.....	39
14.3.1	left (Table Cell Leading Edge Border).....	39
14.3.2	left (Table Leading Edge Border).....	39
14.3.3	left (Table Cell Leading Margin Exception).....	39
14.3.4	left (Table Cell Leading Margin Default).....	40
14.3.5	right (Table Cell Trailing Edge Border).....	40
14.3.6	right (Table Trailing Edge Border).....	40
14.3.7	right (Table Cell Trailing Margin Default).....	40
14.3.8	right (Table Cell Trailing Margin Exception).....	40
14.3.9	Additional attribute for cnfStyle element (Part 1, §17.4.7).....	40
14.3.10	Additional attribute for cnfStyle element (Part 1, §17.4.8).....	42
14.3.11	Additional attribute for tblLook element (Part 1, §17.4.55).....	43
14.3.12	Additional attribute for tblLook element (Part 1, §17.4.56).....	43
14.3.13	hMerge (Horizontally Merged Cell).....	44
14.4	Fonts.....	46
14.4.1	Elements.....	46
14.5	Numbering.....	48
14.5.1	pict (Picture Numbering Symbol Properties).....	48
14.5.2	legacy (Legacy Numbering Level Properties).....	49

14.6	Annotations	50
14.6.1	Revisions	50
14.7	Settings	59
14.7.1	Legacy Password Hash Algorithm	59
14.7.2	Document Settings.....	66
14.7.3	Compatibility Settings	80
14.7.4	Web Page Settings	159
14.8	Miscellaneous Topics.....	160
14.8.1	Text Box Content.....	160
14.9	Fields and Hyperlinks.....	162
14.9.1	Syntax.....	162
14.9.2	Legacy language references.....	163
14.9.3	Use of DOS File Paths.....	170
14.9.4	Field definitions.....	170
14.9.5	fldData (Custom Field Data).....	179
14.9.6	fldData (Custom Field Data).....	180
14.9.7	hyperlink (Hyperlink) (Part 1, §17.16.22).....	181
14.10	Simple Types.....	181
14.10.1	Additional member types for the union in ST_DecimalNumberOrPercent (Part 1, §17.18.11) ...	181
14.10.2	Additional enumeration values for ST_Jc (Part 1, §17.18.44).....	181
14.10.3	Additional enumeration values for ST_JcTable (Part 1, §17.18.45).....	181
14.10.4	Additional enumeration values for ST_NumberFormat (Part 1, §17.18.59)	182
14.10.5	Additional enumeration values for ST_StyleSort (Part 1, §17.18.82)	182
14.10.6	Additional enumeration values for ST_TabJc (Part 1, §17.18.84).....	183
14.10.7	Additional enumeration values for ST_TextDirection (Part 1, §17.18.93).....	183
14.10.8	Additional member types for the union in ST_TextScale (Part 1, §17.18.95)	183
14.10.9	ST_Cnf (Conditional Formatting Bitmask).....	183
14.10.10	ST_UnqualifiedPercentage (Percentage Value Without Percent Sign).....	185
14.10.11	ST_TextScaleDecimal (Text Expansion/Compression Percentage).....	185
14.11	Changed attributes	186
14.11.1	Changed attribute for contentPart element (Part 1, §17.3.3.2).....	186
14.11.2	Changed attribute for control element (Part 1, §17.3.3.3).....	187
14.11.3	Changed attribute for movie element (Part 1, §17.3.3.17)	187
14.11.4	Changed attribute for objectEmbed element (Part 1, §17.3.3.20).....	188
14.11.5	Changed attribute for objectLink element (Part 1, §17.3.3.21).....	189
14.11.6	Changed attribute for bottom element (Part 1, §17.6.2)	189
14.11.7	Changed attribute for left element (Part 1, §17.6.7).....	190
14.11.8	Changed attribute for printerSettings element (Part 1, §17.6.14)	191
14.11.9	Changed attribute for right element (Part 1, §17.6.15).....	192
14.11.10	Changed attribute for top element (Part 1, §17.6.21).....	192
14.11.11	Changed attribute for embedBold element (Part 1, §17.8.3.3).....	194
14.11.12	Changed attribute for embedBoldItalic element (Part 1, §17.8.3.4).....	194
14.11.13	Changed attribute for embedItalic element (Part 1, §17.8.3.5)	195
14.11.14	Changed attribute for embedRegular element (Part 1, §17.8.3.6).....	196
14.11.15	Changed attribute for footerReference element (Part 1, §17.10.2).....	196
14.11.16	Changed attribute for headerReference element (Part 1, §17.10.5)	197
14.11.17	Changed attribute for dataSource element (Part 1, §17.14.9).....	198
14.11.18	Changed attribute for headerSource element (Part 1, §17.14.16).....	198

14.11.19	Changed attribute for recipientData element (Part 1, §17.14.28)	199
14.11.20	Changed attribute for src element (Part 1, §17.14.30)	200
14.11.21	Changed attribute for attachedTemplate element (Part 1, §17.15.1.6)	200
14.11.22	Changed attribute for saveThroughXslt element (Part 1, §17.15.1.76)	201
14.11.23	Changed attribute for longDesc element (Part 1, §17.15.2.23)	201
14.11.24	Changed attribute for sourceFileName element (Part 1, §17.15.2.39)	202
14.11.25	Changed attribute for subDoc element (Part 1, §17.17.1.1)	203
14.11.26	Changed attribute for altChunk element (Part 1, §17.17.2.1)	203
15.	SpreadsheetML Reference Material	205
15.1	Table of Contents	205
15.2	Workbook	206
15.2.1	Additional attribute for fileSharing element (Part 1, §18.2.12)	206
15.2.2	Additional attribute for webPublishing element (Part 1, §18.2.24)	206
15.2.3	Additional attributes for workbookProtection element (Part 1, §18.2.29)	206
15.2.4	Modified content for Date Conversion for Serial Date-Times (Part 1, §18.17.4.1)	212
15.3	Worksheets	213
15.3.1	Worksheets	213
15.3.2	AutoFilter Settings	216
15.4	Styles	217
15.4.1	left (Leading Edge Border)	217
15.4.2	right (Trailing Edge Border)	217
15.5	Pivot Tables	217
15.5.1	Pivot Tables	217
15.6	External Data Connections	218
15.6.1	Additional attribute for textPr element (Part 1, §18.13.12)	218
15.7	Simple Types	218
15.7.1	Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)	218
15.7.2	ST_UnsignedShortHex (Unsigned Short Hex)	218
15.7.3	Removed enumeration values for ST_CellType (Part 1, §18.18.11)	219
15.8	Formulas	219
15.8.1	Attribute synonym for c element (Part 1, §18.6.1)	219
15.8.2	Additional representation for dates and times (Part 1, §18.17.4)	219
15.9	Changed attributes	219
15.9.1	Changed attribute for externalReference element (Part 1, §18.2.8)	219
15.9.2	Changed attribute for pivotCache element (Part 1, §18.2.17)	219
15.9.3	Changed attribute for sheet element (Part 1, §18.2.19)	220
15.9.4	Changed attribute for control element (Part 1, §18.3.1.19)	220
15.9.5	Changed attribute for controlPr element (Part 1, §18.3.1.20)	220
15.9.6	Changed attribute for customPr element (Part 1, §18.3.1.22)	221
15.9.7	Changed attribute for dataRef element (Part 1, §18.3.1.30)	221
15.9.8	Changed attribute for drawing element (Part 1, §18.3.1.36)	221
15.9.9	Changed attribute for drawingHF element (Part 1, §18.3.1.37)	221
15.9.10	Changed attribute for hyperlink element (Part 1, §18.3.1.47)	222
15.9.11	Changed attribute for objectPr element (Part 1, §18.3.1.56)	222
15.9.12	Changed attribute for oleObject element (Part 1, §18.3.1.59)	222
15.9.13	Changed attribute for pageSetup element (Part 1, §18.3.1.63)	223
15.9.14	Changed attribute for pageSetup element (Part 1, §18.3.1.64)	223

15.9.15	Changed attribute for picture element (Part 1, §18.3.1.67).....	223
15.9.16	Changed attribute for pivotSelection element (Part 1, §18.3.1.69)	223
15.9.17	Changed attribute for tablePart element (Part 1, §18.3.1.94)	223
15.9.18	Changed attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)	224
15.9.19	Changed attribute for rangeSet element (Part 1, §18.10.1.79).....	224
15.9.20	Changed attribute for worksheetSource element (Part 1, §18.10.1.95)	224
15.9.21	Changed attribute for header element (Part 1, §18.11.1.1).....	224
15.9.22	Changed attribute for externalBook element (Part 1, §18.14.7).....	225
15.9.23	Changed attribute for oleLink element (Part 1, §18.14.11).....	225
16.	PresentationML Reference Material	226
16.1	Table of Contents.....	226
16.2	Presentation.....	227
16.2.1	Presentation Properties	227
16.3	Slides	235
16.3.1	Embedded Objects.....	235
16.4	Simple Types	236
16.4.1	ST_WebColorType (HTML Slide Navigation Control Colors).....	236
16.4.2	ST_WebEncoding (Web Encoding).....	236
16.4.3	ST_WebScreenSize (HTML/Web Screen Size Target)	236
16.5	Changed attributes	237
16.5.1	Changed attribute for bold element (Part 1, §19.2.1.1)	237
16.5.2	Changed attribute for boldItalic element (Part 1, §19.2.1.2)	238
16.5.3	Changed attribute for font element (Part 1, §19.2.1.13)	238
16.5.4	Changed attribute for handoutMasterId element (Part 1, §19.2.1.14).....	240
16.5.5	Changed attribute for italic element (Part 1, §19.2.1.16).....	240
16.5.6	Changed attribute for notesMasterId element (Part 1, §19.2.1.20)	241
16.5.7	Changed attribute for notesSz element (Part 1, §19.2.1.22).....	241
16.5.8	Changed attribute for regular element (Part 1, §19.2.1.29).....	242
16.5.9	Changed attribute for sld element (Part 1, §19.2.1.31).....	242
16.5.10	Changed attribute for sldId element (Part 1, §19.2.1.33).....	242
16.5.11	Changed attribute for sldMasterId element (Part 1, §19.2.1.36)	242
16.5.12	Changed attribute for SmartTags element (Part 1, §19.2.1.40)	243
16.5.13	Changed attribute for gridSpacing element (Part 1, §19.2.2.3).....	243
16.5.14	Changed attribute for origin element (Part 1, §19.2.2.9).....	244
16.5.15	Changed attribute for sld element (Part 1, §19.2.2.14).....	244
16.5.16	Changed attribute for bgRef element (Part 1, §19.3.1.3).....	245
16.5.17	Changed attribute for blipFill element (Part 1, §19.3.1.4).....	245
16.5.18	Changed attribute for clrMap element (Part 1, §19.3.1.6).....	245
16.5.19	Changed attribute for cNvPicPr element (Part 1, §19.3.1.11)	247
16.5.20	Changed attribute for cNvPr element (Part 1, §19.3.1.12).....	247
16.5.21	Changed attribute for cNvSpPr element (Part 1, §19.3.1.13).....	249
16.5.22	Changed attribute for contentPart element (Part 1, §19.3.1.14)	249
16.5.23	Changed attribute for custData element (Part 1, §19.3.1.17).....	250
16.5.24	Changed attribute for grpSpPr element (Part 1, §19.3.1.23)	250
16.5.25	Changed attribute for sldLayoutId element (Part 1, §19.3.1.40).....	250
16.5.26	Changed attribute for spPr element (Part 1, §19.3.1.44)	250
16.5.27	Changed attribute for tags element (Part 1, §19.3.1.47)	251

16.5.28	Changed attribute for xfrm element (Part 1, §19.3.1.53).....	251
16.5.29	Changed attribute for control element (Part 1, §19.3.2.1).....	252
16.5.30	Changed attribute for oleObj element (Part 1, §19.3.2.4).....	252
16.5.31	Changed attribute for pos element (Part 1, §19.4.5)	252
16.5.32	Changed attribute for snd element (Part 1, §19.5.68)	253
16.5.33	Changed attribute for sndTgt element (Part 1, §19.5.70)	253
17.	DrawingML - Framework Reference Material.....	254
17.1	DrawingML - Main	254
17.1.1	Table of Contents.....	254
17.1.2	Simple Types	254
17.2	DrawingML - Legacy Compatibility	258
17.2.1	Table of Contents	258
17.2.2	Basics.....	259
17.3	Changed attributes	260
17.3.1	Changed attribute for hlinkHover element (Part 1, §20.1.2.2.23).....	260
17.3.2	Changed attribute for snd element (Part 1, §20.1.2.2.32)	260
17.3.3	Changed attribute for audioFile element (Part 1, §20.1.3.2)	260
17.3.4	Changed attribute for quickTimeFile element (Part 1, §20.1.3.4).....	261
17.3.5	Changed attribute for videoFile element (Part 1, §20.1.3.6).....	261
17.3.6	Changed attribute for wavAudioFile element (Part 1, §20.1.3.7)	261
17.3.7	Changed attribute for blip element (Part 1, §20.1.8.13)	261
17.3.8	Changed attribute for blipFill element (Part 1, §20.2.2.1).....	262
17.3.9	Changed attribute for cNvPicPr element (Part 1, §20.2.2.2)	262
17.3.10	Changed attribute for cNvPr element (Part 1, §20.2.2.3).....	263
17.3.11	Changed attribute for spPr element (Part 1, §20.2.2.6)	264
17.3.12	Changed attribute for docPr element (Part 1, §20.4.2.5).....	265
17.3.13	Changed attribute for extent element (Part 1, §20.4.2.7).....	266
17.3.14	Changed attribute for lineTo element (Part 1, §20.4.2.9)	267
17.3.15	Changed attribute for simplePos element (Part 1, §20.4.2.13).....	267
17.3.16	Changed attribute for start element (Part 1, §20.4.2.14).....	268
17.3.17	Changed attribute for blipFill element (Part 1, §20.5.2.2).....	269
17.3.18	Changed attribute for cNvPicPr element (Part 1, §20.5.2.7)	269
17.3.19	Changed attribute for cNvPr element (Part 1, §20.5.2.8).....	270
17.3.20	Changed attribute for cNvSpPr element (Part 1, §20.5.2.9).....	271
17.3.21	Changed attribute for contentPart element (Part 1, §20.5.2.12).....	272
17.3.22	Changed attribute for ext element (Part 1, §20.5.2.14)	272
17.3.23	Changed attribute for grpSpPr element (Part 1, §20.5.2.18)	273
17.3.24	Changed attribute for pos element (Part 1, §20.5.2.26)	273
17.3.25	Changed attribute for spPr element (Part 1, §20.5.2.30)	274
17.3.26	Changed attribute for xfrm element (Part 1, §20.5.2.36).....	274
18.	DrawingML - Components Reference Material.....	276
18.1	DrawingML - Charts	276
18.1.1	Table of Contents.....	276
18.1.2	Elements	277
18.1.3	Simple Types	277
18.2	Changed attributes	281
18.2.1	Changed attribute for hlinkClick element (Part 1, §21.1.2.3.5).....	281

18.2.2	Changed attribute for hlinkMouseOver element (Part 1, §21.1.2.3.6)	281
18.2.3	Changed attribute for chart element (Part 1, §21.2.2.26)	281
18.2.4	Changed attribute for clrMapOvr element (Part 1, §21.2.2.30)	281
18.2.5	Changed attribute for externalData element (Part 1, §21.2.2.63)	283
18.2.6	Changed attribute for spPr element (Part 1, §21.2.2.197)	283
18.2.7	Changed attribute for userShapes element (Part 1, §21.2.2.221)	284
18.2.8	Changed attribute for blipFill element (Part 1, §21.3.2.2)	284
18.2.9	Changed attribute for cNvPicPr element (Part 1, §21.3.2.6)	284
18.2.10	Changed attribute for cNvPr element (Part 1, §21.3.2.7)	285
18.2.11	Changed attribute for cNvSpPr element (Part 1, §21.3.2.8)	287
18.2.12	Changed attribute for ext element (Part 1, §21.3.2.10)	287
18.2.13	Changed attribute for grpSpPr element (Part 1, §21.3.2.14)	288
18.2.14	Changed attribute for spPr element (Part 1, §21.3.2.23)	288
18.2.15	Changed attribute for xfrm element (Part 1, §21.3.2.28)	289
18.2.16	Changed attribute for rellds element (Part 1, §21.4.2.22)	289
18.2.17	Changed attribute for shape element (Part 1, §21.4.2.27)	290
18.2.18	Changed attribute for spPr element (Part 1, §21.4.3.7)	291
18.2.19	Changed attribute for sp3d element (Part 1, §21.4.5.6)	291
19.	VML Reference Material	294
19.1	VML	294
19.1.1	Table of Contents	295
19.1.2	Elements	296
19.1.3	Simple Types	671
19.2	VML - Office Drawing	678
19.2.1	Table of Contents	679
19.2.2	Elements	680
19.2.3	Simple Types	801
19.3	VML - WordprocessingML Drawing	811
19.3.1	Table of Contents	811
19.3.2	Elements	812
19.3.3	Simple Types	820
19.4	VML - SpreadsheetML Drawing	826
19.4.1	Table of Contents	827
19.4.2	Elements	829
19.4.3	Simple Types	855
19.5	VML - PresentationML Drawing	857
19.5.1	Table of Contents	857
19.5.2	Elements	858
20.	Shared MLs Reference Material	860
20.1	Shared Simple Types	860
20.1.1	Table of Contents	860
20.1.2	Simple Types	860
20.2	Extended Properties (Part 1, §22.2)	864
20.3	Custom Properties (Part 1, §22.3)	864
20.4	Changed attributes	864
20.4.1	Changed attribute for sources element (Part 1, §22.6.2.60)	864

Annex A. (normative) Schemas – W3C XML Schema	867
A.1 WordprocessingML	867
A.2 SpreadsheetML	936
A.3 PresentationML	1021
A.4 DrawingML - Framework	1053
A.4.1 DrawingML - Main	1053
A.4.2 DrawingML - Picture	1112
A.4.3 DrawingML - Locked Canvas	1112
A.4.4 DrawingML - WordprocessingML Drawing	1113
A.4.5 DrawingML - SpreadsheetML Drawing	1118
A.5 DrawingML - Components	1122
A.5.1 DrawingML - Charts	1122
A.5.2 DrawingML - Chart Drawings	1150
A.5.3 DrawingML - Diagrams	1153
A.6 VML	1174
A.6.1 VML	1174
A.6.2 VML - Office Drawing	1185
A.6.3 VML - WordprocessingML Drawing	1195
A.6.4 VML - SpreadsheetML Drawing	1197
A.6.5 VML - PresentationML Drawing	1199
A.7 Shared MLs	1199
A.7.1 Math	1199
A.7.2 Extended Properties	1210
A.7.3 Custom Properties	1211
A.7.4 Variant Types	1213
A.7.5 Custom XML Data Properties	1216
A.7.6 Bibliography	1217
A.7.7 Additional Characteristics	1220
A.7.8 Office Document Relationships	1220
A.7.9 Shared Simple Types	1221
A.8 Custom XML Schema References	1224
Annex B. (informative) Schemas – RELAX NG	1225
B.1 WordprocessingML	1225
B.1.1 Part Schemas	1270
B.2 SpreadsheetML	1276
B.2.1 Part Schemas	1364
B.3 PresentationML	1371
B.3.1 Part Schemas	1394
B.4 DrawingML - Framework	1398
B.4.1 DrawingML - Main	1398
B.4.2 DrawingML - Picture	1443
B.4.3 DrawingML - Locked Canvas	1443
B.4.4 DrawingML - Wordprocessing Drawing	1443
B.4.5 DrawingML - Spreadsheet Drawing	1447
B.5 DrawingML - Components	1449
B.5.1 DrawingML - Chart	1449
B.5.2 DrawingML - Chart Drawing	1468

B.5.3	DrawingML - Diagrams.....	1470
B.6	VML.....	1486
B.6.1	VML - Main.....	1486
B.6.2	VML - Office Drawing	1493
B.6.3	VML - Wordprocessing Drawing	1501
B.6.4	VML - Spreadsheet Drawing	1503
B.6.5	VML - Presentation Drawing.....	1505
B.6.6	Part Schemas.....	1505
B.7	Shared MLs	1506
B.7.1	Math.....	1506
B.7.2	Extended Properties.....	1512
B.7.3	Custom Properties	1513
B.7.4	Variant Types	1514
B.7.5	Custom XML Data Properties.....	1517
B.7.6	Bibliography	1518
B.7.7	Additional Characteristics	1520
B.7.8	Office Document Relationships	1521
B.7.9	Shared Simple Types	1522
B.8	Custom XML Schema References	1523
B.9	Additional Resources	1524
B.9.1	Any	1524
B.9.2	XML	1524
Annex C. (informative) Namespace Prefix Mapping in Examples.....		1525
Annex D. (informative) Differences Between ECMA-376:2012 and ECMA-376:2006		1527
D.1	WordprocessingML.....	1527
D.2	SpreadsheetML.....	1529
D.3	PresentationML	1530
D.4	DrawingML.....	1530
D.4.1	DrawingML – Main.....	1530
D.4.2	DrawingML – Chart	1531
D.4.3	DrawingML – Diagrams.....	1531
D.4.4	DrawingML – Spreadsheet Drawing	1531
D.5	VML.....	1532
D.5.1	VML	1532
D.5.2	VML – Office Drawing	1532
D.5.3	VML – Spreadsheet Drawing.....	1532
D.6	Shared.....	1532
D.6.1	Shared – Bibliography	1532
D.6.2	Shared – Custom Properties Variant Types	1532
D.6.3	Shared – Math.....	1532
D.6.4	Shared Simple Types	1533
D.7	Custom XML Schema References	1533
Bibliography		1534

Foreword

Changes from the 3rd edition were made to align this 4th edition Standard with ISO/IEC 29500:2012. Both this 4th edition and ISO/IEC 29500:2012 refer to the 1st edition. As such, this 4th edition does not cancel or replace the 1st edition. This 4th edition does, however, cancel and replace the 3rd edition.

Some important differences between ECMA-376:2012 and ECMA-376:2006 are given in Annex D.

ECMA-376 consists of the following parts:

- *Part 1: Fundamentals and Markup Language Reference*
- *Part 2: Open Packaging Conventions*
- *Part 3: Markup Compatibility and Extensibility*
- *Part 4: Transitional Migration Features*

Annex A forms a normative part of this Part of ECMA-376. Annexes B, C, and D are for information only.

This Part of ECMA-376 includes two annexes (Annex A and Annex B) that refer to data files provided in electronic form.

The document representation formats defined by this Part are different from the formats defined in the corresponding Part of ECMA-376:2006. Some of the differences are reflected in schema changes, as shown in Annex D of this Part.

Introduction

ECMA-376 specifies a family of XML schemas, collectively called *Office Open XML*, which define the XML vocabularies for word-processing, spreadsheet, and presentation documents, as well as the packaging of documents that conform to these schemas.

The goal is to enable the implementation of the Office Open XML formats by the widest set of tools and platforms, fostering interoperability across office productivity applications and line-of-business systems, as well as to support and strengthen document archival and preservation, all in a way that is fully compatible with the existing corpus of Microsoft Office documents.

The intent of this Part of ECMA-376 is to enable a transitional period during which existing binary documents being migrated to ECMA-376 can make use of legacy features to preserve their fidelity, while noting that new documents should not use them. Part 1, §2.4, “Document Conformance”, notes that WML Strict, SML Strict and PML Strict documents do not use any of the features defined in Part 4.

This Part of ECMA-376 is normative for the current edition of ECMA-376, but is not guaranteed to be included in future revisions of that Standard. The intent is to enable the group responsible for maintenance of ECMA-376 to choose, at a later date, to remove this set of features from a revised version of that Standard.

In general, this Part of ECMA-376 augments Part 1, and inherits the provisions of that Part. Exceptions to this are indicated explicitly.

The following organizations have participated in the creation of ECMA-376 and their contributions are gratefully acknowledged:

Apple, Barclays Capital, BP, The British Library, Essilor, Intel, Microsoft, NextPage, Novell, Statoil, Toshiba, and the United States Library of Congress.

1. Scope

ECMA-376 defines a set of XML vocabularies for representing word-processing documents, spreadsheets and presentations. On the one hand, the goal of ECMA-376 is to represent faithfully the existing corpus of word-processing documents, spreadsheets and presentations that have been produced by Microsoft Office applications (from Microsoft Office 97 to Microsoft Office 2008, inclusive). It also specifies requirements for Office Open XML consumers and producers. On the other hand, the goal is to facilitate extensibility and interoperability by enabling implementations by multiple vendors and on multiple platforms.

This Part of ECMA-376 defines features for backward-compatibility and that are useful for high-quality migration of existing binary documents to ECMA-376. These features are used only by documents of conformance class WML Transitional (§2.1), SML Transitional (§2.1), or PML Transitional (§2.1). These features are sometimes needed for high-quality migration of existing binary documents to ECMA-376.

2. Conformance

2.1 Document Conformance

A document of conformance class Office Open XML Transitional shall be a package of conformance class OPC, as specified in ECMA-376-2, for which all the following shall hold:

- The document obeys all constraints specified in this Part of ECMA-376
- The document is of category Wordprocessing, Spreadsheet, or Presentation. These categories are defined in ECMA-376-1:2011 §4
- VML Drawing Parts (§8.1) are of conformance class MCE, as specified in ECMA-376-3. Any child elements of the root element of VMLDrawing Parts are valid against the VML schema shown in A.6, “VML”, after the removal of any extensions specified using the mechanisms in ECMA-376-3. VML Drawing Parts obey all constraints specified in this Part of ECMA-376
- For each OPC Part of the document of the types listed in §9.1 or ECMA-376-1:2011 §11.3, §12.3, §13.3, §14.2, and §15.2, all the following shall hold:
 - i. The part is of conformance class MCE, as specified in ECMA-376-3
 - ii. After the removal of any extensions using the mechanisms in ECMA-376-3, the part is valid against the Transitional W3C XML Schema (Annex A)

This Part of ECMA-376 uses the following further terms to refer to documents of conformance class Office Open XML Transitional:

- *WML Transitional*, if the document is of category Wordprocessing
- *SML Transitional*, if the document is of category Spreadsheet
- *PML Transitional*, if the document is of category Presentation

2.2 Application Conformance

Application conformance incorporates both syntax and semantics.

- A conforming consumer shall not reject any conforming documents of at least one document conformance class.
- A conforming producer shall be able to produce conforming documents of at least one document conformance class.
- A conforming application shall treat the information in Office Open XML documents in a manner consistent with the semantic definitions given in ECMA-376. An application's intended behavior need not require that application to process all of the information in an Office Open XML document. However, the information that it does process shall be processed in a manner that is consistent with the semantic definitions given in ECMA-376.

[Note: This note illustrates the third bullet above. Conforming applications might serve various functions. Examples include a viewer, an editor, and a back-end processor. Here is an illustration of how the third bullet applies to each of those examples:

- If a conforming viewer supports a given feature, then when it displays information using that feature, it respects the semantics of that feature as described in the Standard.
- If a conforming editor supports a given feature, then when it provides its user with an interface for manipulating information using that feature, it respects the semantics of that feature as described in the Standard.
- If a conforming back-end processor supports a given feature, then when that processor transforms or assembles information involving that feature, that processor respects the semantics of that feature as described in the Standard.

end note]

This Part of ECMA-376 defines the following application conformance classes:

- *WML Transitional*, if the application is a conforming application that is a consumer or producer of documents having conformance class WML Transitional.
- *SML Transitional*, if the application is a conforming application that is a consumer or producer of documents having conformance class SML Transitional.
- *PML Transitional*, if the application is a conforming application that is a consumer or producer of documents having conformance class PML Transitional.

3. Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ANSI X3.4-1986, *American Standard Code for Information Interchange (ASCII)*

Bureau of Standards, Metrology and Inspection of the Ministry of Economic Affairs, *CNS 7648: Data Elements and Interchange Formats — Information Interchange — Representation of Dates and Times*

Calendar Reform Committee, *Indian Ephemeris and Nautical Almanac*. 1957

Stokes, M., M. Anderson, S. Chandrasekar, and R. Motta. *A Standard Default color Space for the Internet. Vers. 1.10*. November 5, 1996. <http://www.w3.org/Graphics/Color/sRGB>

Har'El, Zvi, *Gauss Formula for the Julian Date of Passover*. Department of Mathematics, Technion, Israel Institute of Technology, Haifa 32000, Israel, 2005, 6

Duerst, M, and M Suignard. *Internationalized Resource Identifiers (IRIs)*. IETF. January 2005.
<http://tools.ietf.org/html/rfc3987>

IANA, *Character Sets from IANA*, as specified at <http://www.iana.org/assignments/character-sets>

IANA. *MIME Media Types*. Internet Assigned Numbers Authority. <http://www.iana.org/assignments/media-types/>

IEC 60559:1989, *Binary Floating-Point Arithmetic for Microprocessor Systems*

ISO/IEC 2382-1:1993, *Information technology — Vocabulary — Part 1: Fundamental terms*

ISO 8601:2004, *Data elements and interchange formats — Information interchange — Representation of dates and times*

ISO/IEC 8859-1:1998, *Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1* (referred to in ECMA-376 as the ANSI character set)

ISO/IEC 9075-1, *Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework)*

ISO/IEC 10118-3:2004, *Information technology — Security techniques — Hash-functions — Part 3: Dedicated hash-functions*.

ISO/IEC 10646, *Information technology — Universal Coded Character Set (UCS)*.

ISO/IEC 14496-22:2009, *Information technology — Coding of audio-visual objects — Part 22: Open Font Format*

ECMA-376-1:2012, *Information technology — Document description and processing languages — Office Open XML File Formats, Part 1: Fundamentals and Markup Language Reference*.

ECMA-376-3:2012, *Information technology — Document description and processing languages — Office Open XML File Formats — Part 3: Markup Compatibility and Extensibility*

Japanese Industrial Standard, JIS X 0301: *Data elements and interchange formats — Information interchange — Representation of dates and times*. Japan, 2002.

Kingdom of Saudi Arabia, Ministry of Islamic Affairs, Endowments, Da'wah and Guidance.

Korean Law Enactment No. 4, 1961.

Faure, D. (n.d.). *Creating and Using Components (KParts)*. <http://techbase.kde.org/Projects/Documentation>.

Maimon, Rabbi Moshe ben, *Complete Restatement of the Oral Law (Mishneh Torah)*.

Ausbrooks, Ron, et al. *Mathematical Markup Language (MathML) Version 2.0 (Second Edition)*. October 21, 2003. <http://www.w3.org/TR/MathML/>.

Kaliski, B. *The MD2 Message-Digest Algorithm*. April 1992. <http://www.ietf.org/rfc/rfc1319.txt>

Rivest, R. *The MD4 Message-Digest Algorithm*. April 1992. <http://www.ietf.org/rfc/rfc1320.txt>

The MD5 Message-Digest Algorithm. April 1992. <http://www.ietf.org/rfc/rfc1321.txt>

National Measurement Regulations 1999, Commonwealth of Australia
<http://www.comlaw.gov.au/Details/F2011C00445>

NIST Guide to SI Units, <http://physics.nist.gov/Pubs/SP811/appenB9.html>

QuickTime File Format Specification (2007-09-04 version)
<http://developer.apple.com/standards/classicquicktime.html>

Resource Description Framework (RDF), <http://www.w3.org/RDF/>

RFC 822, *Standard for ARPA Internet Text Messages* (<http://www.ietf.org/rfc/rfc0822.txt>)

RFC 2045, Borenstein, N., and N. Freed. *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*. The Internet Society. 1996. <http://www.ietf.org/rfc/rfc2045.txt>

RFC 2119, Bradner, Scott, 1997: *Key words for use in RFCs to Indicate Requirement Levels*.
<http://www.ietf.org/rfc/rfc2119.txt>

RFC 2616, Berners-Lee, T., R. Fielding, H. Frystyk, J. Gettys, P. Leach, L. Masinter, and J. Mogul. *Hypertext Transfer Protocol—HTTP/1.1*. The Internet Society. 1999. <http://www.ietf.org/rfc/rfc2616.txt>

RFC 3066, Alvestrand, H. *Tags for the Identification of Languages*. The Internet Society. 2001.

<http://www.ietf.org/rfc/rfc3066.txt>

RFC 3339, Klyne, G. and C. Newman. *Date and Time on the Internet: Timestamps*. The Internet Society. 2002.

<http://www.ietf.org/rfc/rfc3339.txt>

RFC 3629, Yergeau, F. *UTF-8, a transformation format of ISO 10646*. The Internet Society. 2003.

<http://www.ietf.org/rfc/rfc3629.txt>

RFC 3986, Berners-Lee, T., R. Fielding, and L. Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. The Internet Society. 2005. <http://www.ietf.org/rfc/rfc3986.txt>

Simple Object Access Protocol (SOAP), <http://www.w3.org/TR/soap12>

SMIL, Bulterman, D., Grassel, G., Jansen, J., Koivisto, A., Layaïda, N., Michel, T., et al. (2005, December 13).

Synchronized Multimedia Integration Language (SMIL 2.1). Retrieved from W3C: <http://www.w3.org/TR/SMIL/>

SVG, Andersson, O., Armstrong, P., Axelsson, H., Berjon, R., Bézaire, B., Bowler, J., et al. (2003, January 14).

Scalable Vector Graphics (SVG) 1.1 Specification. Retrieved from W3C - World Wide Web Consortium:

<http://www.w3.org/TR/SVG/>

The GNOME Project. (2003, December 12). *Component Model - Bonobo Document Model*. Retrieved from The

GNOME Development Site: <http://developer.gnome.org/arch/gnome/componentmodel/bonobo.htm>

The Unicode Consortium. *The Unicode Standard*, <http://www.unicode.org/standard/standard.html>

Unicode Technical Report #25, <http://www.unicode.org/reports/tr25/>

Unicode Technical Note #28, *Nearly Plain-Text Encoding of Mathematics*. August 29, 2006,

<http://www.unicode.org/notes/tn28>

United States Postal Service. *Domestic Mail Manual*. United States Postal Service. November 8, 2007.

<http://pe.usps.com/cpim/ftp/manuals/dmm300/Full/MailingStandards.pdf>

The Units of Measurement Regulations 1995, United Kingdom

http://www.opsi.gov.uk/si/si1995/Uksi_19951804_en_2.htm

Universal Postal Union. *POST*CODE: Postal addressing systems*. Berne: UPU Publications, 2006, ISBN 92-95025-37-7, ISSN 1020-6019

Web Accessibility Initiative (WAI), <http://www.w3.org/WAI/>

XSLT, Clark, James, *XSL Transformations (XSLT) Version 1.0*, World Wide Web Consortium Recommendation.

1999. <http://www.w3.org/TR/xslt>

XML, Tim Bray, Jean Paoli, Eve Maler, C. M. Sperberg-McQueen, and François Yergeau (editors). *Extensible Markup Language (XML) 1.0*, Fourth Edition.1 World Wide Web Consortium. 2006.

<http://www.w3.org/TR/2006/REC-xml-20060816/> [Implementers should be aware that a further correction of the normative reference to XML to refer to the 5th Edition will be necessary when the related Reference Specifications to which this International Standard also makes normative reference and which also depend upon XML, such as XSLT, XML Namespaces and XML Base, are all aligned with the 5th Edition.]

XML Base, Marsh, Jonathan. *XML Base*. World Wide Web Consortium. 2001. <http://www.w3.org/TR/2001/REC-xmlbase-20010627/>

XML Namespaces, Tim Bray, Dave Hollander, Andrew Layman, and Richard Tobin (editors). *Namespaces in XML 1.0 (Third Edition)*, 8 December 2009. World Wide Web Consortium. <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

XPATH, Clark, James; DeRose, Steve. *XML Path Language (XPath) Version 1.0*, World Wide Web Consortium Recommendation. 1999. <http://www.w3.org/TR/xpath>.

XML Schema Part 0: Primer (Second Edition), W3C Recommendation 28 October 2004, <http://www.w3.org/TR/xmlschema-0/>

XML Schema Part 1: Structures (Second Edition), W3C Recommendation 28 October 2004, <http://www.w3.org/TR/xmlschema-1/>

XML Schema Part 2: Datatypes (Second Edition), W3C Recommendation 28 October 2004, <http://www.w3.org/TR/xmlschema-2/>

.ZIP File Format Specification from PKWARE, Inc., version 6.2.0 (2004), as specified in http://www.pkware.com/documents/APPNOTE/APPNOTE_6.2.0.

4. Terms and Definitions

For the purposes of this document, the following terms and definitions apply. Other terms are defined where they appear in *italic* typeface, on the left side of a syntax rule, or within subclauses of language-specific grammars. Terms explicitly defined in this Part of ECMA-376 are not to be presumed to refer implicitly to similar terms defined elsewhere. [*Note: This part uses OPC-related terms, which are defined in ECMA-376-2. end note*]

application — A consumer or producer.

behavior — External appearance or action.

behavior, implementation-defined — Unspecified behavior where each implementation is expected to document that behavior, which would thereby promote predictability and reproducibility within any given implementation. (This term is sometimes called “application-defined behavior”.)

behavior, locale-specific — Behavior that depends on local conventions of nationality, culture, and language.

behavior, unspecified — Behavior where ECMA-376 makes no recommendations. [*Note: To add an extension, an implementer must use the extensibility mechanisms described by ECMA-376 rather than trying to do so by giving meaning to otherwise unspecified behavior. end note*]

byte — A sequence of 8 bits treated as a unit.

comment — A note that an author or reviewer attaches to content in a document. Although a consumer might choose to display comments, they are not considered part of the body of the document. A comment might include the text of the note, the comment author's name and initials, and date of creation, among other things.

consumer — A piece of software or a device that reads packages through a package implementer. A consumer is often designed to consume packages only for a specific physical package format.

content type — Describes the content stored in a part. Content types define a media type, a subtype, and an optional set of parameters, as defined in RFC 2616.

document category — One of the three categories of Office Open XML documents: Wordprocessing, Spreadsheet, and Presentation, defined as follows:

- A document whose package-relationship item contains a relationship to a Main Document part (Part 1, §11.3.10) is a document of category Wordprocessing.
- A document whose package-relationship item contains a relationship to a Workbook part (Part 1, §12.3.23) is a document of category Spreadsheet.

- A document whose package-relationship item contains a relationship to a Presentation part (Part 1, §13.3.6) is a document of category Presentation.

An Office Open XML document can contain one or more embedded Office Open XML packages (Part 1, §15.2.11) with each embedded package having any of the three document categories. However, the presence of these embedded packages does not change the category of the document.

DOS file path — A legacy file naming scheme which used a file name of at most eight characters, followed by a period ("."), followed by a filename extension of at most three characters. This name may be preceded by a slash-delimited path, and the combined structure may be preceded by a drive letter specifier. The grammar for DOS file paths is defined as follows:

```
filepath = [drive] [ folder] [filename]
filename = corefilename "." (0 * 3 validchar)
corefilename = (1 * 8 validchar)
validchar = uppercaseletter | decimaldigit | "!" | "#" | "$" | "%" | "&"
           | "'" | "\" | "-" | "@" | "^" | "_" | "`" | "{" | "}" | "~"
corefoldername = (1 * 8 validchar) | "." | ".."
folder = 1 * (corefoldername "\\") | "\\"
drive = uppercaseletter ":\\\\"
uppercaseletter = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J"
               | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V"
               | "W" | "X" | "Y" | "Z"
decimaldigit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

DrawingML — A set of conventions for specifying the location and appearance of drawing elements in an Office Open XML document.

extension — Any XML element, XML attribute, relationship, or part not explicitly included in ECMA-376, but that uses the extensibility mechanisms described by ECMA-376.

Office Open XML document — A rendition of a data stream formatted using the wordprocessing, spreadsheet, or presentation ML and its related MLs as described in ECMA-376-1 and ECMA-376-4. Such a document is represented as a package as described in ECMA-376-2.

package — A ZIP archive that conforms to the Open Packaging Conventions specification defined in ECMA-376-2.

package, embedded — A package that has been stored as the target of an Embedded Package relationship (Part 1, §15.2.11) in an Office Open XML document

PresentationML — A set of conventions for representing an Office Open XML document of category Presentation.

producer — A piece of software or a device that writes packages through a package implementer. A producer is often designed to produce packages according to a particular physical package format specification.

relationship —The kind of connection between a source part and a target part in a package. Relationships make the connections between parts directly discoverable without looking at the content in the parts, and without altering the parts themselves. (See also Package Relationships.)

relationships part — A part containing an XML representation of relationships.

relationship, explicit — A relationship in which a resource is referenced from a source part's XML using the Id attribute of a Relationship tag.

relationship, implicit — A relationship that is not explicit.

SpreadsheetML — A set of conventions for representing an Office Open XML document of category Spreadsheet.

WordprocessingML — A set of conventions for representing an Office Open XML document of category Wordprocessing.

5. Notational Conventions

The following typographical conventions are used in this Part of ECMA-376:

1. The first occurrence of a new term is written in italics. [*Example*: The text in ECMA-376 is divided into *normative* and *informative* categories. *end example*]
2. In each definition of a term in §4 (Terms and Definitions), the term is written in bold. [*Example*: **behavior** — External appearance or action. *end example*]
3. The tag name of an XML element is written using a distinct style and typeface. [*Example*: The `bookmarkStart` and `bookmarkEnd` elements specify ... *end example*]
4. The name of an XML attribute is written using a distinct style and typeface. [*Example*: The `dropCap` attribute specifies ... *end example*]
5. The value of an XML attribute is written using a constant-width style. [*Example*: The attribute value of `auto` specifies ... *end example*]
6. The qualified or unqualified name of a simple type, complex type, or base datatype is written using a distinct style and typeface. [*Example*: The possible values for this attribute are defined by the `ST_HexColor` simple type. *end example*]

6. Acronyms and Abbreviations

This clause is informative.

The following acronyms and abbreviations are used throughout ECMA-376:

IEC — the International Electrotechnical Commission

ISO — the International Organization for Standardization

W3C — World Wide Web Consortium

End of informative text.

7. General Description

This Part of ECMA-376 is divided into the following subdivisions:

1. Front matter (clauses 1–7);
2. Main body (clauses 8–20);
3. Annexes

Examples are provided to illustrate possible forms of the constructions described. References are used to refer to related clauses. Notes are provided to give advice or guidance to implementers or programmers. Rationale provides explanatory material as to why something is or is not in ECMA-376. Annexes provide additional information or summarize the information contained in ECMA-376.

The following form the normative pieces of this Part of ECMA-376:

- Introduction
- Clauses 1–5, 7, and 8–20
- Annex A

The following form the informative pieces of this Part of ECMA-376:

- Clause 6
- Annex B–Annex D
- All notes
- All examples

Except for whole clauses or annexes that are identified as being informative, informative text that is contained within normative text is indicated in the following ways:

1. [*Example*: code fragment, possibly with some narrative ... *end example*]
2. [*Note*: narrative ... *end note*]
3. [*Rationale*: narrative ... *end rationale*]
4. [*Guidance*: narrative ... *end guidance*]

Unless stated otherwise in this Part, the functionality defined in Part 1 is applicable to Part 4. However, Part 4 uses namespaces that are different from those used by Part 1. As such, when examples in Part 1 are read in the context of Part 4, they should be understood in the context of the corresponding Part 4 namespaces.

8. Additional Shared Parts

8.1 VML Drawing Part

Content Type:	application/vnd.openxmlformats-officedocument.vmlDrawing
Root Namespace:	not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/vmlDrawing

An instance of this part type contains markup in the Vector Markup Language (VML) syntax, which is used to provide an alternative image representation of objects stored in a SpreadsheetML or PresentationML document.

[*Note:* The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a deprecated format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. *end note*]

A package is permitted to contain zero or more VML Drawing parts, each of which shall be the target of an explicit relationship in a Handout Master (Part 1, §13.3.3), Notes Slide (Part 1, §13.3.5), Notes Master (Part 1, §13.3.4), Slide (Part 1, §13.3.8), Slide Layout (Part 1, §13.3.9), or Slide Master (Part 1, §13.3.10) part in a PresentationML document; or a Dialogsheet (Part 1, §12.3.7) or Worksheet part (Part 1, §12.3.24) in a SpreadsheetML document.

[*Example:* The following SpreadsheetML's package-relationship item contains one relationship, for the VML Drawing part stored in the ZIP item ../drawings/drawing1.vml:

```
<Relationships xmlns="...">
  <Relationship Id="rId8"
    Type="http://.../vmlDrawing" Target="../drawings/drawing1.vml"/>
</Relationships>
```

end example]

The root element for a part of this content type shall be xml in the null namespace, encapsulating an arbitrary amount of VML markup as defined by ECMA-376.

[*Example*: Consider the following VML Drawing part:

```
<xml>
  <v:shape ...>
    ...
  </v:shape>
  ...
</xml>
```

end example]

A VML Drawing part shall be located within the package containing the relationships part (expressed syntactically, the TargetMode attribute of the Relationship element shall be Internal).

A VML Drawing part is permitted to have explicit relationships to the following parts defined by ECMA-376:

- Image (Part 1, §15.2.14)

A VML Drawing part shall not have implicit or explicit relationships to any other part defined by ECMA-376.

9. WordprocessingML

The following parts, which are defined in subclauses within Part 1, §11, “WordprocessingML”, have different source relationships and/or root namespaces when used in documents of the Transitional conformance class:

9.1 Part Summary (Part 1, §11.3)

9.1.1 Alternative Format Import Part (Part 1, §11.3.1)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/aFChunk
----------------------	---

9.1.2 Comments Part (Part 1, §11.3.2)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

9.1.3 Document Settings Part (Part 1, §11.3.3)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/settings

9.1.4 Endnotes Part (Part 1, §11.3.4)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/endnotes

9.1.5 Fonts Table Part (Part 1, §11.3.5)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/fontTable

9.1.6 Footer Part (Part 1, §11.3.6)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
-----------------	---

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer
----------------------	---

9.1.7 Footnotes Part (Part 1, §11.3.7)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/footnotes

9.1.8 Glossary Document Part (Part 1, §11.3.8)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/glossaryDocument

9.1.9 Header Part (Part 1, §11.3.9)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/header

9.1.10 Main Document Part (Part 1, §11.3.10)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

9.1.11 Numbering Definitions Part (Part 1, §11.3.11)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/numbering

9.1.12 Style Definitions Part (Part 1, §11.3.12)

Root Namespace:	http://schemas.openxmlformats.org/wordprocessingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles

9.1.13 Web Settings Part (Part 1, §11.3.13)

Root	http://schemas.openxmlformats.org/wordprocessingml/2006/main
------	---

Namespace:	
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/webSettings

9.2 Document Template (Part 1, §11.4)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/attachedTemplate
----------------------	---

9.3 Framesets (Part 1, §11.5)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/frame
----------------------	---

9.4 Master Documents and Subdocuments (Part 1, §11.6)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/subDocument
----------------------	---

9.5 Mail Merge Data Source (Part 1, §11.7)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/mailMergeSource
----------------------	---

9.6 Mail Merger Header Data Source (Part 1, §11.8)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/mailMergeHeaderSource
----------------------	---

9.7 XSL Transformation (Part 1, §11.9)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/transform
----------------------	---

10. SpreadsheetML

The following parts, which are defined in subclauses within Part 1, §12, “SpreadsheetML”, have different source relationships and/or root namespaces when used in documents of the Transitional conformance class:

10.1 Part Summary (Part 1, §12.3)

10.1.1 Calculation Chain Part (Part 1, §12.3.1)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/calcChain

10.1.2 Chartsheet Part (Part 1, §12.3.2)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chartsheet

10.1.3 Comments Part (Part 1, §12.3.3)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

10.1.4 Connections Part (Part 1, §12.3.4)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/connections

10.1.5 Custom Property Part (Part 1, §12.3.5)

Root Namespace:	Not applicable
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customProperty

10.1.6 Custom XML Mappings Part (Part 1, §12.3.6)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/xmlMaps

10.1.7 Dialogsheet Part (Part 1, §12.3.7)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/dialogsheet

10.1.8 Drawings Part (Part 1, §12.3.8)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/drawing

10.1.9 External Workbook References Part (Part 1, §12.3.9)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/externalLink

10.1.10 Metadata Part (Part 1, §12.3.10)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/sheetMetadata

10.1.11 Pivot Table Part (Part 1, §12.3.11)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/pivotTable

10.1.12 Pivot Table Cache Definition Part (Part 1, §12.3.12)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/pivotCacheDefinition

10.1.13 Pivot Table Cache Records Part (Part 1, §12.3.13)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/pivotCacheRecords

10.1.14 Query Table Part (Part 1, §12.3.14)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/queryTable

10.1.15 Shared Strings Table Part (Part 1, §12.3.15)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/sharedStrings

10.1.16 Shared Workbook Revision Headers Part (Part 1, §12.3.16)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/revisionHeaders

10.1.17 Shared Workbook Revision Log Part (Part 1, §12.3.17)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/revisionLog

10.1.18 Shared Workbook User Data part (Part 1, §12.3.18)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/usernames

10.1.19 Single Cell Table Definitions Part (Part 1, §12.3.19)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableSingleCells

10.1.20 Styles Part (Part 1, §12.3.20)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles

10.1.21 Table Definition Part (Part 1, §12.3.21)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/table

10.1.22 Volatile Dependencies Part (Part 1, §12.3.22)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/volatileDependencies

10.1.23 Workbook Part (Part 1, §12.3.23)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

10.1.24 Worksheet Part (Part 1, §12.3.24)

Root Namespace:	http://schemas.openxmlformats.org/spreadsheetml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/worksheet

10.2 External Workbooks (Part 1, §12.4)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/externalLinkPath
----------------------	---

11. PresentationML

The following parts, which are defined in subclauses within Part 1, §13, “PresentationML”, have different source relationships and/or root namespaces when used in documents of the Transitional conformance class:

11.1 Part Summary (Part 1, §13.3)

11.1.1 Comment Authors Part (Part 1, §13.3.1)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/commentAuthors

11.1.2 Comments Part (Part 1, §13.3.2)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/comments

11.1.3 Handout Master Part (Part 1, §13.3.3)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/handoutMaster

11.1.4 Notes Master Part (Part 1, §13.3.4)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/notesMaster

11.1.5 Notes Slide Part (Part 1, §13.3.5)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/notesSlide

11.1.6 Presentation Part (Part 1, §13.3.6)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument

11.1.7 Presentation Properties Part (Part 1, §13.3.7)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/presProps

11.1.8 Slide Part (Part 1, §13.3.8)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slide

11.1.9 Slide Layout Part (Part 1, §13.3.9)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideLayout

11.1.10 Slide Master Part (Part 1, §13.3.10)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideMaster

11.1.11 Slide Synchronization Data Part (Part 1, §13.3.11)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideUpdateInfo

11.1.12 User Defined Tags Part (Part 1, §13.3.12)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tags

11.1.13 View Properties Part (Part 1, §13.3.13)

Root Namespace:	http://schemas.openxmlformats.org/presentationml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/viewProps

11.2 HTML Publish Location (Part 1, §13.4)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/htmlPubSaveAs
----------------------	---

11.3 Slide Synchronization Server Location (Part 1, §13.5)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideUpdateUrl
----------------------	---

12. DrawingML

The following parts, which are defined in subclauses within Part 1, §14, “DrawingML”, have different source relationships and/or root namespaces when used in documents of the Transitional conformance class:

12.1 Part Summary (Part 1, §14.2)

12.1.1 Chart Part (Part 1, §14.2.1)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/chart
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chart

12.1.2 Chart Drawing Part (Part 1, §14.2.2)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/chart
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/chartUserShapes

12.1.3 Diagram Colors Part (Part 1, §14.2.3)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramColors

12.1.4 Diagram Data Part (Part 1, §14.2.4)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramData

12.1.5 Diagram Layout Definition Part (Part 1, §14.2.5)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramLayout

12.1.6 Diagram Style Part (Part 1, §14.2.6)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/diagram
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramQuickStyle

12.1.7 Theme Part (Part 1, §14.2.7)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme

12.1.8 Theme Override Part (Part 1, §14.2.8)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/themeOverride

12.1.9 Table Styles Part (Part 1, §14.2.9)

Root Namespace:	http://schemas.openxmlformats.org/drawingml/2006/main
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableStyles

13. Shared MLs

The following parts, which are defined in subclauses within Part 1, §15, “Shared”, have different source relationships and/or root namespaces when used in documents of the Transitional conformance class:

13.1 Part Summary (Part 1, §15.2)

13.1.1 Additional Characteristics Part (Part 1, §15.2.1)

Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/additionalCharacteristics
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml

13.1.2 Audio Part (Part 1, §15.2.2)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/audio
----------------------	---

13.1.3 Bibliography Part (Part 1, §15.2.3)

Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/bibliography
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml

13.1.4 Content Part (Part 1, §15.2.4)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml
----------------------	---

13.1.5 Custom XML Data Storage Part (Part 1, §15.2.5)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXml
----------------------	---

13.1.6 Custom XML Data Storage Properties Part (Part 1, §15.2.6)

Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/customXmlDataProps
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/customXmlProps

13.1.7 Embedded Control Persistence Part (Part 1, §15.2.9)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/control
----------------------	---

13.1.8 Embedded Object Part (Part 1, §15.2.10)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/oleObject
----------------------	---

13.1.9 Embedded Package Part (Part 1, §15.2.11)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/package
----------------------	---

13.1.10 Core File Properties Part (Part 1, §15.2.12.1)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/metadata/core-properties
----------------------	---

13.1.11 Custom File Properties Part (Part 1, §15.2.12.2)

Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/custom-properties
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties

13.1.12 Extended File Properties Part (Part 1, §15.2.12.3)

Root Namespace:	http://schemas.openxmlformats.org/officeDocument/2006/extended-properties
Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/extended-properties

13.1.13 Font Part (Part 1, §15.2.13)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/font
----------------------	---

13.1.14 Image Part (Part 1, §15.2.14)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/image
----------------------	---

13.1.15 Printer Settings Part (Part 1, §15.2.15)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings
----------------------	---

13.1.16 Thumbnail Part (Part 1, §15.2.16)

Source Relationship:	http://schemas.openxmlformats.org/package/2006/relationships/metadata/thumbnail
----------------------	---

13.1.17 Video Part (Part 1, §15.2.17)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/video
----------------------	---

13.2 Hyperlinks Part (Part 1, §15.3)

Source Relationship:	http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink
----------------------	---

14. WordprocessingML Reference Material

[Note: For further information on the mapping of elements and attributes to OPC parts, see the Bibliography entry, “Information on elements, attributes, and OPC parts in ECMA-376 (OOXML)”. *end note*]

14.1 Table of Contents

This subclause is informative.

14.2	Paragraphs and Rich Formatting	35
14.2.1	Paragraphs	35
14.2.1.1	Additional attribute for cnfStyle element (Part 1, §17.3.1.8)	35
14.2.1.2	Additional attributes for ind element (Part 1, §17.3.1.12)	36
14.2.2	Run Content	37
14.2.2.1	control (Floating Embedded Control)	37
14.2.2.2	pict (VML Object)	39
14.3	Tables	39
14.3.1	left (Table Cell Leading Edge Border)	39
14.3.2	left (Table Leading Edge Border)	39
14.3.3	left (Table Cell Leading Margin Exception)	39
14.3.4	left (Table Cell Leading Margin Default)	40
14.3.5	right (Table Cell Trailing Edge Border)	40
14.3.6	right (Table Trailing Edge Border)	40
14.3.7	right (Table Cell Trailing Margin Default)	40
14.3.8	right (Table Cell Trailing Margin Exception)	40
14.3.9	Additional attribute for cnfStyle element (Part 1, §17.4.7)	40
14.3.10	Additional attribute for cnfStyle element (Part 1, §17.4.8)	42
14.3.11	Additional attribute for tblLook element (Part 1, §17.4.55)	43
14.3.12	Additional attribute for tblLook element (Part 1, §17.4.56)	43
14.3.13	hMerge (Horizontally Merged Cell)	44
14.4	Fonts	46
14.4.1	Elements	46
14.4.1.1	Additional attribute for charset element (Part 1, §17.8.3.2)	46
14.5	Numbering	48
14.5.1	pict (Picture Numbering Symbol Properties)	48
14.5.2	legacy (Legacy Numbering Level Properties)	49
14.6	Annotations	50
14.6.1	Revisions	50
14.6.1.1	numberingChange (Previous Numbering Field Properties)	50
14.6.1.2	numberingChange (Previous Paragraph Numbering Properties)	53
14.7	Settings	59

14.7.1 Legacy Password Hash Algorithm	59
14.7.2 Document Settings.....	66
14.7.2.1 hdrShapeDefaults (Default Properties for VML Objects in Header and Footer)	66
14.7.2.2 shapeDefaults (Default Properties for VML Objects in Main Document)	67
14.7.2.3 Additional attributes for documentProtection element (Part 1, §17.15.1.29).....	67
14.7.2.4 Additional attribute for stylePaneFormatFilter element (Part 1, §17.15.1.85).....	73
14.7.2.5 Additional attributes for writeProtection element (Part 1, §17.15.1.93).....	74
14.7.3 Compatibility Settings.....	80
14.7.3.1 alignTablesRowByRow (Align Table Rows Independently)	81
14.7.3.2 allowSpaceOfSameStyleInTable (Allow Contextual Spacing of Paragraphs in Tables)	82
14.7.3.3 autofitToFirstFixedWidthCell (Allow Table Columns To Exceed Preferred Widths of Constituent Cells)	84
14.7.3.4 autoSpaceLikeWord95 (Incorrectly Adjust Text Spacing for Specific Unicode Ranges)	86
14.7.3.5 cachedColBalance (Use Cached Paragraph Information for Column Balancing).....	87
14.7.3.6 convMailMergeEsc (Treat Backslash Quotation Delimiter as Two Quotation Marks)	88
14.7.3.7 displayHangulFixedWidth (Always Use Fixed Width for Hangul Characters)	89
14.7.3.8 doNotAutofitConstrainedTables (Do Not AutoFit Tables To Fit Next To Wrapped Objects).....	90
14.7.3.9 doNotBreakConstrainedForcedTable (Don't Break Table Rows Around Floating Tables).....	91
14.7.3.10 doNotBreakWrappedTables (Do Not Allow Floating Tables To Break Across Pages).....	93
14.7.3.11 doNotSnapToGridInCell (Do Not Snap to Document Grid in Table Cells with Objects)	94
14.7.3.12 doNotSuppressIndentation (Do Not Ignore Floating Objects When Calculating Paragraph Indentation)	95
14.7.3.13 doNotSuppressParagraphBorders (Do Not Suppress Paragraph Borders Next To Frames).....	97
14.7.3.14 doNotUseEastAsianBreakRules (Do Not Compress Compressible Characters When Using Document Grid).....	98
14.7.3.15 doNotUseHTMLParagraphAutoSpacing (Use Fixed Paragraph Spacing for HTML Auto Setting) ...	99
14.7.3.16 doNotUseIndentAsNumberingTabStop (Ignore Hanging Indent When Creating Tab Stop After Numbering)	101
14.7.3.17 doNotVertAlignCellWithSp (Don't Vertically Align Cells Containing Floating Objects)	102
14.7.3.18 doNotVertAlignInTxbx (Ignore Vertical Alignment in Textboxes).....	104
14.7.3.19 doNotWrapTextWithPunct (Do Not Allow Hanging Punctuation With Character Grid)	106
14.7.3.20 footnoteLayoutLikeWW8 (Ignore Page Break from Continuous Section Break)	107
14.7.3.21 forgetLastTabAlignment (Ignore Width of Last Tab Stop When Aligning Paragraph If It Is Not Left Aligned)	110
14.7.3.22 growAutofit (Allow Tables to AutoFit Into Page Margins)	112
14.7.3.23 layoutRawTableWidth (Ignore Space Before Table When Deciding If Table Should Wrap Floating Object)	113
14.7.3.24 layoutTableRowsApart (Allow Table Rows to Wrap Inline Objects Independently)	115
14.7.3.25 lineWrapLikeWord6 (Ignore Compression of Full-Width Punctuation Ending a Line)	116
14.7.3.26 mwSmallCaps (Use Specific Small Caps Algorithm)	117
14.7.3.27 noColumnBalance (Do Not Balance Text Columns within a Section)	118
14.7.3.28 noExtraLineSpacing (Do Not Center Content on Lines With Exact Line Height).....	120
14.7.3.29 noLeading (Do Not Add Leading Between Lines of Text).....	121
14.7.3.30 noSpaceRaiseLower (Do Not Increase Line Height for Raised/Lowered Text)	123
14.7.3.31 noTabHangInd (Do Not Create Custom Tab Stop for Hanging Indent).....	123
14.7.3.32 printBodyTextBeforeHeader (Print Body Text before Header/Footer Contents).....	125
14.7.3.33 printColBlack (Print Colors as Black And White without Dithering)	126
14.7.3.34 selectFldWithFirstOrLastChar (Select Field When First or Last Character Is Selected)	126

14.7.3.35	shapeLayoutLikeWW8 (Ignore Text Wrapping around Objects at Bottom of Page)	127
14.7.3.36	showBreaksInFrames (Display Page/Column Breaks Present in Frames)	130
14.7.3.37	spacingInWholePoints (Only Expand/Condense Text By Whole Points)	132
14.7.3.38	splitPgBreakAndParaMark (Always Move Paragraph Mark to Page after a Page Break)	134
14.7.3.39	subFontBySize (Require Exact Size During Font Substitution)	135
14.7.3.40	suppressBottomSpacing (Ignore Exact Line Height for Last Line on Page)	136
14.7.3.41	suppressSpacingAtTopOfPage (Ignore Minimum Line Height for First Line on Page)	138
14.7.3.42	suppressSpBfAfterPgBrk (Do Not Use Space Before On First Line After a Page Break)	140
14.7.3.43	suppressTopSpacing (Ignore Minimum and Exact Line Height for First Line on Page)	142
14.7.3.44	suppressTopSpacingWP (Use Static Text Leading)	143
14.7.3.45	swapBordersFacingPages (Swap Paragraph Borders on Odd Numbered Pages)	143
14.7.3.46	truncateFontHeightsLikeWP6 (Use Truncated Integer Division For Font Calculation)	146
14.7.3.47	underlineTabInNumList (Underline Following Character Following Numbering)	147
14.7.3.48	useAltKinsokuLineBreakRules (Use Alternate Set of East Asian Line Breaking Rules)	148
14.7.3.49	useAnsiKerningPairs (Use ANSI Kerning Pairs from Fonts)	149
14.7.3.50	useFELayout (Do Not Bypass East Asian/Complex Script Layout Code)	149
14.7.3.51	useNormalStyleForList (Do Not Automatically Apply List Paragraph Style To Bulleted/Numbered Text)	150
14.7.3.52	usePrinterMetrics (Use Printer Metrics To Display Documents)	151
14.7.3.53	useSingleBorderforContiguousCells (Use Simplified Rules For Table Border Conflicts)	152
14.7.3.54	useWord2002TableStyleRules (Incorrectly Display Top Border of Conditional Columns)	153
14.7.3.55	useWord97LineBreakRules (Use Incorrect Inter-Character Spacing Rules)	155
14.7.3.56	wpJustification (Fit To Expanded Width When Performing Full Justification)	157
14.7.3.57	wpSpaceWidth (Use Specific Space Width)	158
14.7.3.58	wrapTrailSpaces (Line Wrap Trailing Spaces)	158
14.7.4	Web Page Settings	159
14.7.4.1	relyOnVML (Utilize VML When Saving as Web Page)	159
14.8	Miscellaneous Topics	160
14.8.1	Text Box Content	160
14.8.1.1	txbxContent (Rich Text Box Content Container)	160
14.9	Fields and Hyperlinks	162
14.9.1	Syntax	162
14.9.2	Legacy language references	163
14.9.3	Use of DOS File Paths	170
14.9.4	Field definitions	170
14.9.4.1	AUTONUM	170
14.9.4.2	AUTONUMLGL	171
14.9.4.3	AUTONUMOUT	172
14.9.4.4	BARCODE	173
14.9.4.5	BIDIOUTLINE	175
14.9.4.6	EQ	175
14.9.4.7	INFO	178
14.9.4.8	QUOTE	179
14.9.5	fldData (Custom Field Data)	179
14.9.6	fldData (Custom Field Data)	180
14.9.7	hyperlink (Hyperlink) (Part 1, §17.16.22)	181

14.10	Simple Types	181
14.10.1	Additional member types for the union in ST_DecimalNumberOrPercent (Part 1, §17.18.11)	181
14.10.2	Additional enumeration values for ST_Jc (Part 1, §17.18.44)	181
14.10.3	Additional enumeration values for ST_JcTable (Part 1, §17.18.45)	181
14.10.4	Additional enumeration values for ST_NumberFormat (Part 1, §17.18.59)	182
14.10.5	Additional enumeration values for ST_StyleSort (Part 1, §17.18.82)	182
14.10.6	Additional enumeration values for ST_TabJc (Part 1, §17.18.84)	183
14.10.7	Additional enumeration values for ST_TextDirection (Part 1, §17.18.93)	183
14.10.8	Additional member types for the union in ST_TextScale (Part 1, §17.18.95)	183
14.10.9	ST_Cnf (Conditional Formatting Bitmask)	183
14.10.10	ST_UnqualifiedPercentage (Percentage Value Without Percent Sign)	185

End of informative text.

14.2 Paragraphs and Rich Formatting

14.2.1 Paragraphs

14.2.1.1 [Additional attribute for cnfStyle element \(Part 1, §17.3.1.8\)](#)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
val (Conditional Formatting Bit Mask)	<p>Specifies the set of conditional formatting properties that have been applied to this object.</p> <p>These properties are expressed using a string serialization of a binary bitmask for each of the following properties (reading from the first character position right):</p> <ul style="list-style-type: none"> • First Row - Is this the first row of the table? • Last Row - Is this the last row of the table? • First Column - Does this belong to the first column of the table? • Last Column - Does this belong to the last column of the table? • Band 1 Vertical - Does this belong to a column which should receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered columns (e.g. 1,3,5,...) • Band 2 Vertical - Does this belong to a column which should receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered columns (e.g. 2,4,6,...) • Band 1 Horizontal - Does this receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered rows (e.g. 1,3,5,...) • Band 2 Horizontal - Does this receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered rows (e.g. 2,4,6,...) • NE Cell - Is this part of the top-right corner of the table?

Attributes	Description
	<ul style="list-style-type: none"> NW Cell - Is this part of the top-left corner of the table? SE Cell - Is this part of the bottom-right corner of the table? SW Cell - Is this part of the bottom-left corner of the table? <p>For each of these properties, a value of 1 in the specified character position in the string means that the value is true, a value of 0 means false. All values shall be specified.</p> <p>[<i>Example:</i> Consider a paragraph in the top right corner of a table with a table style applied. This paragraph would need to specify the following WordprocessingML:</p> <pre><w:p> <w:pPr> <w:cnfStyle w:val="101000000100" /> ... </w:pPr> ... </w:p></pre> <p>This paragraph specifies that it has the conditional properties from the table style for the first column, first row, and the NW corner of the parent table by setting the appropriate bits in the val attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Cnf simple type (§14.10.8).</p>

14.2.1.2 Additional attributes for ind element (Part 1, §17.3.1.12)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
left (Start Indentation)	<p>Semantically equivalent to the start attribute.</p> <p>The possible values for this attribute are defined by the ST_SignedTwipsMeasure simple type (Part 1, §17.18.81).</p>
leftChars (Start Indentation in Character Units)	<p>Semantically equivalent to the startChars attribute.</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
right (End Indentation)	<p>Semantically equivalent to the end attribute.</p> <p>The possible values for this attribute are defined by the ST_SignedTwipsMeasure simple type (Part 1, §17.18.81).</p>
rightChars (End Indentation in Character Units)	<p>Semantically equivalent to the endChars attribute.</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>

14.2.2 Run Content

14.2.2.1 control (Floating Embedded Control)

This element specifies that the parent VML object is a representation of an embedded control at the current location in the document. This element shall be used to associate the VML data with the appropriate embedded control settings and properties when the document is displayed.

If the embedded control is not present, cannot be loaded due to application settings, or is not supported, then the VML data shall be used to provide an image representation of the control at the appropriate location in the document.

[*Example:* Consider a run which consists of an embedded control. That run would be specified using the following WordprocessingML:

```
<w:r>
  <w:pict>
    ...
    <w:control r:id="rId99" w:shapeid="shape01" ... />
  </w:pict>
</w:r>
```

The control element indicates that the parent VML object contains the positioning and last known image representation of an embedded control, whose settings and properties are stored on this element. *end example]*

Attributes	Description
id (Embedded Control Properties Relationship Reference) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID for the relationship which contains the properties for this embedded control. This property bag is contained in a separate part within the Office Open XML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/control</code> or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then the embedded control shall be given no property bag when instantiated.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for an embedded control in a document:</p> <pre><w:control r:id="rId5" w:name="CheckBox1" w:shapeid="_x0000_s1027" /></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID <code>rId5</code> must contain the property data for this embedded control. <i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).
name (Unique Name for Embedded Control)	<p>Specifies a unique name for this embedded control. This name shall be unique across all controls in this document.</p> <p><i>[Example: Consider the following WordprocessingML markup for an embedded control in a document:</i></p> <pre><w:control r:id="rId5" w:name="CheckBox1" w:shapeid="_x0000_s1027" /></pre> <p>The name attribute specifies that the unique name for this control must be CheckBox1. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
shapeid (Shape Reference)	<p>Specifies the shape ID for a shape which shall be used to define the presentation and location of this embedded control within the document if the control is floating using the DrawingML syntax.</p> <p><i>[Note: This positioning data is sufficient to display the control in any case where:</i></p> <ul style="list-style-type: none"> • The embedded control is not on the current machine • Embedded controls are disabled • Embedded controls of this control type are not supported <p><i>end note]</i></p> <p>This shape ID reference is resolved by looking for a DrawingML object whose id attribute matches the value specified within this attribute. If no such shape exists, then the control shall be rendered inline in the document content at the current run content location.</p> <p>If this attribute is omitted, then this embedded control shall be displayed inline in the current location in the parent run.</p> <p><i>[Example: Consider the following WordprocessingML markup for an embedded control in a document:</i></p> <pre><w:control r:id="rId5" w:name="CheckBox1" w:shapeid="10" /></pre> <p>The shapeid attribute specifies that the DrawingML object with an id attribute value of 10 must contain the positioning data for this embedded control. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Control](#)) is located in §A.1. *end note*]

14.2.2.2 pict (VML Object)

This element specifies that an object is located at this position in the run's contents. The layout properties of this object are specified using the VML syntax (§19.1).

[*Example:* Consider a run which consists of an object specified using VML. That run would be specified using the following WordprocessingML:

```
<w:r>
  <w:pict>
    ...
  </w:pict>
</w:r>
```

The pict element indicates that an object specified in VML is located at the current position in the run (e.g. a floating embedded control). *end example*]

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Picture](#)) is located in §A.1. *end note*]

14.3 Tables

14.3.1 left (Table Cell Leading Edge Border)

This element is semantically equivalent to start (Part 1, §17.4.34), specified above.

For tables which have the bidiVisual property (Part 1, §17.4.1) applied, this border is applied to the right edge of the cell.

This element's content model is defined by the common border properties definition in Part 1, §17.3.4.

14.3.2 left (Table Leading Edge Border)

This element is semantically equivalent to start (Part 1, §17.4.37), specified above.

For tables which have the bidiVisual property (Part 1, §17.4.1) applied, this border is applied to the right edge of the table.

This element's content model is defined by the common border properties definition in Part 1, §17.3.4.

14.3.3 left (Table Cell Leading Margin Exception)

This element is semantically equivalent to start (Part 1, §17.4.36), specified above.

For tables which have the bidiVisual property (Part 1, §17.4.1) applied, this cell margin is applied to the right edge of the cell.

This element's content model is defined by the common table measurement definition in Part 1, §17.4.88.

14.3.4 left (Table Cell Leading Margin Default)

This element is semantically equivalent to start (Part 1, §17.4.35), specified above.

For tables which have the `bidiVisual` property (Part 1, §17.4.1) applied, this cell margin is applied to the right edge of the cell.

This element's content model is defined by the common table measurement definition in Part 1, §17.4.88.

14.3.5 right (Table Cell Trailing Edge Border)

This element is semantically equivalent to end (Part 1, §17.4.12), specified above.

For tables which have the `bidiVisual` property (Part 1, §17.4.1) applied, this border is applied to the left edge of the cell.

This element's content model is defined by the common border properties definition in Part 1, §17.3.4.

14.3.6 right (Table Trailing Edge Border)

This element is semantically equivalent to end (Part 1, §17.4.13), specified above.

For tables which have the `bidiVisual` property (Part 1, §17.4.1) applied, this border is applied to the left edge of the table.

This element's content model is defined by the common border properties definition in Part 1, §17.3.4.

14.3.7 right (Table Cell Trailing Margin Default)

This element is semantically equivalent to end (Part 1, §17.4.11), specified above.

For tables which have the `bidiVisual` property (Part 1, §17.4.1) applied, this cell margin is applied to the left edge of the cell.

This element's content model is defined by the common table measurement definition in Part 1, §17.4.88.

14.3.8 right (Table Cell Trailing Margin Exception)

This element is semantically equivalent to end (Part 1, §17.4.10), specified above.

For tables which have the `bidiVisual` property (Part 1, §17.4.1) applied, this cell margin is applied to the left edge of the cell.

This element's content model is defined by the common table measurement definition in Part 1, §17.4.88.

14.3.9 Additional attribute for `cnfStyle` element (Part 1, §17.4.7)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
val (Conditional Formatting Bit Mask)	<p>Specifies the set of conditional formatting properties that have been applied to this object.</p> <p>These properties are expressed using a string serialization of a binary bitmask for each of the following properties (reading from the first character position right):</p> <ul style="list-style-type: none"> • First Row - Is this the first row of the table? • Last Row - Is this the last row of the table? • First Column - Does this belong to the first column of the table? • Last Column - Does this belong to the last column of the table? • Band 1 Vertical - Does this belong to a column which should receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered columns (e.g. 1,3,5,...) • Band 2 Vertical - Does this belong to a column which should receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered columns (e.g. 2,4,6...) • Band 1 Horizontal - Does this receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered rows (e.g. 1,3,5,...) • Band 2 Horizontal - Does this receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered rows (e.g. 2,4,6...) • NE Cell - Is this part of the top-right corner of the table? • NW Cell - Is this part of the top-left corner of the table? • SE Cell - Is this part of the bottom-right corner of the table? • SW Cell - Is this part of the bottom-left corner of the table? <p>For each of these properties, a value of 1 in the specified character position in the string means that the value is true, a value of 0 means false. All values shall be specified.</p> <p>[<i>Example:</i> Consider a paragraph in the top right corner of a table with a table style applied. This paragraph would need to specify the following WordprocessingML:</p> <pre> <w:p> <w:pPr> <w:cnfStyle w:val="101000000100" /> ... </w:pPr> ... </w:p> </pre> <p>This paragraph specifies that it has the conditional properties from the table style for the first column, first row, and the NW corner of the parent table by setting the appropriate bits in the val attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Cnf simple type (§14.10.8).</p>

14.3.10 Additional attribute for cnfStyle element (Part 1, §17.4.8)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
val (Conditional Formatting Bit Mask)	<p>Specifies the set of conditional formatting properties that have been applied to this object.</p> <p>These properties are expressed using a string serialization of a binary bitmask for each of the following properties (reading from the first character position right):</p> <ul style="list-style-type: none"> • First Row - Is this the first row of the table? • Last Row - Is this the last row of the table? • First Column - Does this belong to the first column of the table? • Last Column - Does this belong to the last column of the table? • Band 1 Vertical - Does this belong to a column which should receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered columns (e.g. 1,3,5,...) • Band 2 Vertical - Does this belong to a column which should receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered columns (e.g. 2,4,6,...) • Band 1 Horizontal - Does this receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered rows (e.g. 1,3,5,...) • Band 2 Horizontal - Does this receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered rows (e.g. 2,4,6,...) • NE Cell - Is this part of the top-right corner of the table? • NW Cell - Is this part of the top-left corner of the table? • SE Cell - Is this part of the bottom-right corner of the table? • SW Cell - Is this part of the bottom-left corner of the table? <p>For each of these properties, a value of 1 in the specified character position in the string means that the value is true, a value of 0 means false. All values shall be specified.</p> <p>[Example: Consider a paragraph in the top right corner of a table with a table style applied. This paragraph would need to specify the following WordprocessingML:</p> <pre> <w:p> <w:pPr> <w:cnfStyle w:val="101000000100" /> ... </w:pPr> ... </w:p> </pre> <p>This paragraph specifies that it has the conditional properties from the table style for the</p>

Attributes	Description
	<p>first column, first row, and the NW corner of the parent table by setting the appropriate bits in the val attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Cnf simple type (§14.10.8).</p>

14.3.11 Additional attribute for tblLook element (Part 1, §17.4.55)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
val (Bitmask of Table Conditional Formatting)	<p>Specifies a hexadecimal code containing a bitmask of options, interpreted as follows:</p> <ul style="list-style-type: none"> • 0x0020=Apply first row conditional formatting • 0x0040=Apply last row conditional formatting • 0x0080=Apply first column conditional formatting • 0x0100=Apply last column conditional formatting • 0x0200=Do not apply row banding conditional formatting • 0x0400=Do not apply column banding conditional formatting <p>If omitted, the bitmask of table style options on the current table shall be assumed to be 0000.</p> <p>[<i>Example:</i> Consider a table which must use the following conditional formatting properties from the referenced table style:</p> <ul style="list-style-type: none"> • First row conditional formatting • Last row conditional formatting <p>This table would then apply the following portions of the bitmask:</p> <ul style="list-style-type: none"> • 0x0020=Apply first row conditional formatting • 0x0040=Apply last row conditional formatting • 0x0200=Do not apply row banding conditional formatting • 0x0400=Do not apply column banding conditional formatting <p>The resulting WordprocessingML would be specified as follows:</p> <pre><w:tblPr> <w:tblLook w:val="0660"/> </w:tblPr></pre> <p>The val attribute specifies a bitmask which determines the components of the table style applied to the current table. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_ShortHexNumber simple type (Part 1, §17.18.79).</p>

14.3.12 Additional attribute for tblLook element (Part 1, §17.4.56)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
val (Bitmask of Table Conditional Formatting)	<p>Specifies a hexadecimal code containing a bitmask of options, interpreted as follows:</p> <ul style="list-style-type: none"> • 0x0020=Apply first row conditional formatting • 0x0040=Apply last row conditional formatting • 0x0080=Apply first column conditional formatting • 0x0100=Apply last column conditional formatting • 0x0200=Do not apply row banding conditional formatting • 0x0400=Do not apply column banding conditional formatting <p>If omitted, the bitmask of table style options on the current table shall be assumed to be 0000.</p> <p>[<i>Example:</i> Consider a table which must use the following conditional formatting properties from the referenced table style:</p> <ul style="list-style-type: none"> • First row conditional formatting • Last row conditional formatting <p>This table would then apply the following portions of the bitmask:</p> <ul style="list-style-type: none"> • 0x0020=Apply first row conditional formatting • 0x0040=Apply last row conditional formatting • 0x0200=Do not apply row banding conditional formatting • 0x0400=Do not apply column banding conditional formatting <p>The resulting WordprocessingML would be specified as follows:</p> <pre><w:tblPr> <w:tblLook w:val="0660"/> </w:tblPr></pre> <p>The val attribute specifies a bitmask which determines the components of the table style applied to the current table. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_ShortHexNumber simple type (Part 1, §17.18.79).</p>

14.3.13 hMerge (Horizontally Merged Cell)

This element specifies that this cell is part of a horizontally merged set of cells in a table. The val attribute on this element determines how this cell is defined with respect to the previous cell in the table (i.e., whether this cell continues the horizontal merge or starts a new merged group of cells).

[*Note:* This property is maintained for compatibility with legacy word processing documents that defined tables in this manner. Whenever possible, this form or horizontal merges should not be produced, and should be translated to the appropriate gridSpan (Part 1) settings on the table cells instead. *end note*]

If this element is omitted, then this cell shall not be part of any horizontally merged grouping of cells, and any horizontal merge group in the preceding cells shall be closed.

[*Example:* Consider a table with one row and three columns with the last two columns horizontally merged:

--	--	--

The second cell in the first row starts a merge that is completed in the right adjacent cell, resulting in the following WordprocessingML:

```
<w:tbl>
...
<w:tr>
  <w:tc>
    ...
  </w:tc>
  <w:tc>
    <w:tcPr>
      <w:hMerge w:val="restart"/>
    </w:tcPr>
    ...
  </w:tc>
  <w:tc>
    <w:tcPr>
      <w:hMerge/>
    </w:tcPr>
    ...
  </w:tc>
</w:tr>
</w:tbl>
```

The hMerge element defines the cells that are to be horizontally merged, and how each group is merged together. *end example*]

Attributes	Description
val (Horizontal Merge Type)	<p>Specifies how the table cell is part of a horizontally merged region. This determines whether the cell should join onto an existing grouping of merged cells if any exist, or start a new group of merged cells. Refer to the simple type definition for a full description of each type.</p> <p>If this attribute is omitted, its value shall be assumed to be continue.</p> <p>[<i>Example:</i> Consider a table cell where a horizontal cell merge begins represented as the following WordprocessingML:</p> <pre><w:tcPr> <w:hMerge w:val="restart"/> </w:tcPr></pre> <p>The attribute value of restart specifies that this element must start a new horizontally merged region in this table. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Merge simple type (Part 1, §17.18.57).</p>

[*Note:* The W3C XML Schema definition of this element’s content model (CT_HMerge) is located in §A.1. *end note*]

14.4 Fonts

14.4.1 Elements

14.4.1.1 Additional attribute for charset element (Part 1, §17.8.3.2)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description								
val (Value)	<p>Specifies a value specified as single octet (two-digit) hexadecimal number whose contents are interpreted based on the context of the parent XML element.</p> <p>If this attribute is not present, then the character set for this font shall be assumed to be ISO/IEC 8859-1.</p> <p>The value of this attribute shall be interpreted as follows:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>0x00</td><td>Specifies a Latin character set. (IANA name iso-8859-1)</td></tr><tr><td>0x01</td><td>Specifies the default character set.</td></tr><tr><td>0x02</td><td>Specifies the Symbol character set. This value specifies that the</td></tr></table>	Value	Description	0x00	Specifies a Latin character set. (IANA name iso-8859-1)	0x01	Specifies the default character set.	0x02	Specifies the Symbol character set. This value specifies that the
Value	Description								
0x00	Specifies a Latin character set. (IANA name iso-8859-1)								
0x01	Specifies the default character set.								
0x02	Specifies the Symbol character set. This value specifies that the								

Attributes	Description	
		characters in the Unicode private use area (U+FF00 to U+FFFF) of the font should be used to display the corresponding characters in the range U+0000 to U+00FF.
	0x4D	Specifies a Macintosh (Standard Roman) character set. (IANA name <code>macintosh</code>)
	0x80	Specifies the JIS character set. (IANA name <code>shift_jis</code>)
	0x81	Specifies the Hangul character set. (IANA name <code>ks_c-5601-1987</code>)
	0x82	Specifies a Johab character set. (IANA name <code>KS_C-5601-1992</code>)
	0x86	Specifies the GB-2312 character set. (IANA name <code>GBK</code>)
	0x88	Specifies the Chinese Big Five character set. (IANA name <code>Big5</code>)
	0xA1	Specifies a Greek character set. (IANA name <code>windows-1253</code>)
	0xA2	Specifies a Turkish character set. (IANA name <code>iso-8859-9</code>)
	0xA3	Specifies a Vietnamese character set. (IANA name <code>windows-1258</code>)
	0xB1	Specifies a Hebrew character set. (IANA name <code>windows-1255</code>)
	0xB2	Specifies an Arabic character set. (IANA name <code>windows-1256</code>)
	0xBA	Specifies a Baltic character set. (IANA name <code>windows-1257</code>)
	0xCC	Specifies a Russian character set. (IANA name <code>windows-1251</code>)
	0xDE	Specifies a Thai character set. (IANA name <code>windows-874</code>)
	0xEE	Specifies an Eastern European character set. (IANA name <code>windows-1250</code>)
	0xFF	Specifies an OEM character set not defined by ECMA-376.
	Any other value	Application-defined, can be ignored.
<p>[<i>Example:</i> Consider the following value for an attribute of type <code>ST_UCharHexNumber</code>:</p> <pre><... w:val="BE"/></pre> <p>This value is permitted, as it contains two hexadecimal digits, an encoding of an octet of the actual decimal number value. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_UCharHexNumber</code> simple type (Part 1, §17.18.98).</p>		

14.5 Numbering

14.5.1 pict (Picture Numbering Symbol Properties)

This element specifies the properties for a picture which shall be used as a picture numbering symbol in a given document, using the VML syntax.

[*Example:* Consider the WordprocessingML below illustrating the usage of the pict element in a document containing a single picture numbering symbol:

```
<w:numPicBullet w:numPicBulletId="0">
  <w:pict>
    <v:shapetype id="_x0000_t75" coordsize="21600,21600" o:spt="75"
o:preferrelative="t" path="m@4@5l@4@11@9@11@9@5xe" filled="f" stroked="f">
      <v:stroke joinstyle="miter" />
      <v:formulas>
        <v:f eqn="if lineDrawn pixelLineWidth 0" />
        <v:f eqn="sum @0 1 0" />
        <v:f eqn="sum 0 0 @1" />
        <v:f eqn="prod @2 1 2" />
        <v:f eqn="prod @3 21600 pixelWidth" />
        <v:f eqn="prod @3 21600 pixelHeight" />
        <v:f eqn="sum @0 0 1" />
        <v:f eqn="prod @6 1 2" />
        <v:f eqn="prod @7 21600 pixelWidth" />
        <v:f eqn="sum @8 21600 0" />
        <v:f eqn="prod @7 21600 pixelHeight" />
        <v:f eqn="sum @10 21600 0" />
      </v:formulas>
      <v:path o:extrusionok="f" gradientshapeok="t" o:connecttype="rect" />
      <o:lock v:ext="edit" aspectratio="t" />
    </v:shapetype>
    <v:shape id="_x0000_i1029" type="#_x0000_t75"
style="width:11.25pt;height:11.25pt" o:bullet="t">
      <v:imagedata r:id="rId1" o:title="sample picture" />
    </v:shape>
  </w:pict>
</w:numPicBullet>
```

end example]

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Picture](#)) is located in §A.1. *end note]*

14.5.2 legacy (Legacy Numbering Level Properties)

This element specifies that a given numbering level is from an earlier word processing application which did not support the full richness of the numbering properties supported by WordprocessingML.

These properties shall be used to render any numbered paragraph which references this numbering level if the legacy attribute is set. [Note: Using this element in generated WordprocessingML documents is not recommended, as updated numbering structures in WordprocessingML should be used in its place. This element is provided solely to save and roundtrip the numbering properties of legacy word processing products in WordprocessingML such that they are recreated if the document is resaved in an older word processor format. *end note*]

[Example: Consider the following WordprocessingML numbering level:

```
<w:lvl w:ilvl="0">
...
<w:legacy w:legacySpace="820" w:legacyIndent="960" />
<w:lvlJc w:val="start" />
<w:pPr>
  <w:ind w:start="360" w:hanging="360" />
</w:pPr>
</w:lvl>
```

This level has the legacy element present, therefore the legacy numbering level properties must be used to format all paragraphs which reference this level. *end example*]

Attributes	Description
legacy (Use Legacy Numbering Properties)	<p>Specifies whether the legacy numbering properties present for this numbering level shall be used to format the numbering for any paragraph which references it.</p> <p>A value of on, 1, or true for this attribute value specifies that the legacy numbering properties shall be applied. This is the default value for this attribute, and is implied when the attribute is omitted.</p> <p>A value of off, 0, or false for this attribute value specifies that the legacy numbering properties shall not be used, and shall be explicitly turned off.</p> <p>[Example: For example, consider the set of legacy numbering properties from a document:</p> <pre><w:legacy w:legacy="off" w:legacySpace="820" w:legacyIndent="960" /></pre> <p>This set of legacy properties are explicitly not used when processing the numbering level via the fact that the legacy attribute is turned off for this example. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_OnOff simple type (Part 1, §22.9.2.7).</p>

Attributes	Description
legacyIndent (Legacy Indent)	<p>Specifies the indentation which shall be applied to a legacy numbering symbol from the text margin of the document. This value is specified in twentieths of a point.</p> <p>If this attribute is not present, then no indentation shall be applied with respect to the margin.</p> <p>[<i>Example:</i> For example, consider the set of legacy numbering properties from a document:</p> <pre data-bbox="451 470 1227 499"><w:legacy w:legacySpace="820" w:legacyIndent="960" /></pre> <p>This set of legacy properties specify that there must be exactly 960 twentieths of a point (½ of an inch) between the text margin and the start of the numbering on the paragraph. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_SignedTwipsMeasure simple type (Part 1, §17.18.81).</p>
legacySpace (Legacy Spacing)	<p>Specifies the indentation which shall be applied between a legacy numbering symbol and the accompanying text of the associated paragraph in the document. This value is specified in twentieths of a point.</p> <p>If this attribute is not present, then no indentation shall be applied with respect to the paragraph text.</p> <p>[<i>Example:</i> For example, consider the set of legacy numbering properties from a document:</p> <pre data-bbox="451 999 1227 1029"><w:legacy w:legacySpace="820" w:legacyIndent="960" /></pre> <p>This set of legacy properties specify that there must be exactly 860 twentieths of a point between the end of the numbering on the paragraph and the associated paragraph text. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TwipsMeasure simple type (Part 1, §22.9.2.14).</p>

[*Note:* The W3C XML Schema definition of this element's content model (CT_LvlLegacy) is located in §A.1. *end note*]

14.6 Annotations

14.6.1 Revisions

14.6.1.1 numberingChange (Previous Numbering Field Properties)

This element specifies the previous state of the numbering displayed by a LISTNUM field (Part 1, §17.16.5.33) within a WordprocessingML document when additional LISTNUM fields are added and revisions are being tracked.

[*Rationale:* The legacy numbering mechanism provided by the LISTNUM field relies on the presence of fields in the run content of the document, rather than being a paragraph property (as numbering typically is represented). For this reason, these fields must store their previous state as a unique revision type on the field character of the numbering field. *end rationale*]

If this element is supplied for a field which is not of type LISTNUM as defined by its field codes (Part 1, §17.16.5), then this property shall be ignored.

[Example: Consider the following paragraph containing a single LISTNUM field, as follows:

Some 1. text

If another LISTNUM field is added before it in the document, resulting in its evaluation to a different number, as follows:

Some ~~1~~.2. text

This revision to the field result would be stored as follows in the WordprocessingML:

```
<w:fldChar w:fldCharType="begin">
  <w:numberingChange w:id="0" ... w:original="1." />
</w:fldChar>
<w:r>
  <w:instrText>LISTNUM</w:instrText>
</w:r>
<w:fldChar w:fldCharType="separate"/>
<w:r>
  <w:t>2.</w:t>
</w:r>
<w:fldChar w:fldCharType="end" />
```

The numberingChange element specifies that the numbering resulting from this LISTNUM field was modified and this change was tracked as a revision. The previous numbering result of 1. is cached in the original attribute. *end example*]

For numbering fields, the original attribute shall specify the previous numbering displayed by the parent LISTNUM field within a WordprocessingML document. This information is a performance-enhancing cache of the state of the numbering before the revision to allow applications to show the previous state without having to recalculate all of the LISTNUM fields in the document.

If this attribute is omitted, then no previous numbering value is implied and applications can choose to calculate this value, or display no previous numbering value.

[Example: Consider the following paragraph containing a single LISTNUM field with a revision, as follows:

Some ~~1~~.2. text

This revision to the field result would be stored as follows in the WordprocessingML:

```

<w:fldChar w:fldCharType="begin">
  <w:numberingChange w:id="0" ... w:original="1." />
</w:fldChar>

```

The original attribute specifies that the previous numbering value of the field was 1. *end example*

Attributes	Description
author (Annotation Author)	<p>Specifies the author for an annotation within a WordprocessingML document.</p> <p>If this attribute is omitted, then no author shall be associated with the parent annotation type.</p> <p>[<i>Example</i>: Consider a comment represented using the following WordprocessingML fragment:</p> <pre> <... w:id="1" w:author="Example Author"> ... </...> </pre> <p>The author attribute specifies that the author of the current annotation is Example Author, which can be used as desired. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
date (Annotation Date)	<p>Specifies the date information for an annotation within a WordprocessingML document. The use of this information is outside of the scope of ECMA-376.</p> <p>If this attribute is omitted, then no date information shall be associated with the parent annotation type.</p> <p>[<i>Example</i>: Consider a comment represented using the following WordprocessingML fragment:</p> <pre> <... w:id="1" w:date="2006-01-01T10:00:00"> ... </...> </pre> <p>The date attribute specifies that the date of the current annotation is January 1st 2006 at 10:00 AM, which can be used as desired. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_DateTime simple type (Part 1, §17.18.9).</p>
id (Annotation Identifier)	<p>Specifies a unique identifier for an annotation within a WordprocessingML document. The restrictions on the id attribute, if any, are defined by the parent XML element.</p> <p>If this attribute is omitted, then the document is non-conformant.</p>

Attributes	Description
	<p>[<i>Example</i>: Consider an annotation represented using the following WordprocessingML fragment:</p> <pre><... w:id="1" ... > ... </...></pre> <p>The id attribute specifies that the ID of the current annotation is 1. This value is used to uniquely identify this annotation within the document content. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
original (Previous Numbering Value)	<p>Specifies the previous numbering displayed by the parent numbering change revision. Its format is specified by the parent element.</p> <p>If this attribute is omitted, then no previous numbering value is implied and applications can choose to calculate this value, or display no previous numbering value.</p> <p>[<i>Example</i>: Consider the following paragraph containing a single LISTNUM field with a revision, as follows:</p> <p style="text-align: center;">Some 1<ins>2</ins>. text</p> <p>This revision to the field result would be stored as follows in the WordprocessingML:</p> <pre><w:fldChar w:fldCharType="begin"> <w:numberingChange w:id="0" ... w:original="1." /> </w:fldChar></pre> <p>The original attribute specifies that the previous numbering value of the field was 1. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>

[*Note*: The W3C XML Schema definition of this element's content model ([CT_TrackChangeNumbering](#)) is located in §A.1. *end note*]

14.6.1.2 numberingChange (Previous Paragraph Numbering Properties)

This element specifies the previous state of the numbering on a paragraph when revisions are being tracked.

[*Rationale*: This mechanism is simply used to provide storage for revisions to numbering produced by legacy word processing applications, and applications are encouraged to use the pPrChange element to store these changes as changes to the paragraph properties instead. *end rationale*]

[Example: Consider the following list using Arabic numerals as the numbering, as follows:

1. one
2. two
3. three

Consider a revision where the numbering definition is changed from Arabic numerals to Roman numerals, as follows:

- 1.i. one
- 2.ii. two
- 3.iii. three

This revision to the numbering definition would be stored as follows in the WordprocessingML:

```
<w:p>
  <w:pPr>
    <w:numPr>
      <w:ilvl w:val="0" />
      <w:numId w:val="1" />
      <w:numberingChange w:id="0" ... w:original="%1:1:0:." />
    </w:numPr>
  </w:pPr>
  <w:r>
    <w:t>one</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:numPr>
      <w:ilvl w:val="0" />
      <w:numId w:val="1" />
      <w:numberingChange w:id="1" ... w:original="%1:2:0:." />
    </w:numPr>
  </w:pPr>
  <w:r>
    <w:t>two</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:numPr>
      <w:ilvl w:val="0" />
      <w:numId w:val="1" />
```



```

        <w:numberingChange w:id="2" ... w:original="%1:3:0:." />
    </w:numPr>
</w:pPr>
<w:r>
    <w:t>three</w:t>
</w:r>
</w:p>

```

The numberingChange element specifies that the numbering definition was modified and this change was tracked as a revision. The previous Arabic numeral numbering definition is cached in the original attribute. *end example]*

For paragraph numbering, the original attribute shall specify the previous numbering definition for an individual paragraph of text within a WordprocessingML document while revisions are being tracked.

The value of original is represented as separate numbering level definitions defined as follows:

```

<%[numbering level]:[nfc value]:[numbering format]:[separator]>[repeat if more
than one level]

```

where

- **numbering level** – The level for which the numbering definition is defined
- **nfc value** – The value of the numbering style at the specific numbering level
- **numbering format** – The nfc value of the numbering format, as referenced in the table below.
- **separator** – The separator used to separate the numbering level definitions

The numbering format values are mapped as follows:

nfc Value	ST_NumberFormat enumeration equivalent
0	decimal
1	upperRoman
2	lowerRoman
3	upperLetter
4	lowerLetter
5	ordinal
6	cardinalText
7	ordinalText
8	hex
9	chicago
10	ideographDigital
11	japaneseCounting
12	Aiueo

nfc Value	ST_NumberFormat enumeration equivalent
13	Iroha
14	decimalFullWidth
15	decimalHalfWidth
16	japaneseLegal
17	japaneseDigitalTenThousand
18	decimalEnclosedCircle
19	decimalFullWidth2
20	aiueoFullWidth
21	irohaFullWidth
22	decimalZero
23	bullet
24	ganada
25	chosung
26	decimalEnclosedFullstop
27	decimalEnclosedParen
28	decimalEnclosedCircleChinese
29	ideographEnclosedCircle
30	ideographTraditional
31	ideographZodiac
32	ideographZodiacTraditional
33	taiwaneseCounting
34	ideographLegalTraditional
35	taiwaneseCountingThousand
36	taiwaneseDigital
37	chineseCounting
38	chineseLegalSimplified
39	chineseCountingThousand
40	Application-defined. Can be ignored.
41	koreanDigital
42	koreanCounting
43	koreanLegal
44	koreanDigital2
45	hebrew1
46	arabicAlpha
47	hebrew2
48	arabicAbjad
49	hindiVowels
50	hindiConsonants

nfc Value	ST_NumberFormat enumeration equivalent
51	hindiNumbers
52	hindiCounting
53	thaiLetters
54	thaiNumbers
55	thaiCounting
56	vietnameseCounting
57	numberInDash
58	russianLower
59	russianUpper
60 or above	Application-defined. Can be ignored.

[*Example:* Consider the following numbered paragraph where the numbering definition has changed while revisions are being tracked, as follows:

~~1.1.1.~~ Three

This revision to the numbered paragraph would be stored as follows in the WordprocessingML:

```
<w:numPr>
...
  <w:numberingChange ... w:original="%1:1:0:.%2:1:2:.%3:1:0:." />
</w:numPr>
```

In the above example there are three levels in the original numbering definition, thus three numbering level definitions are needed to represent the original numbering definition.

The first level is specified by %1, and says that it was number value 1 in the nfc format 0 (arabic).

The original attribute specifies that the previous numbering definition was made up of three levels whose value was 1.i.1.. *end example*]

Attributes	Description
author (Annotation Author)	<p>Specifies the author for an annotation within a WordprocessingML document.</p> <p>If this attribute is omitted, then no author shall be associated with the parent annotation type.</p> <p>[<i>Example:</i> Consider a comment represented using the following WordprocessingML fragment:</p> <pre><... w:id="1" w:author="Example Author"> ...</pre>

Attributes	Description
	<p data-bbox="456 249 521 281"></...></p> <p data-bbox="415 317 1403 386">The author attribute specifies that the author of the current annotation is Example Author, which can be used as desired. <i>end example</i></p> <p data-bbox="415 422 1435 491">The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
date (Annotation Date)	<p data-bbox="415 510 1468 579">Specifies the date information for an annotation within a WordprocessingML document. The use of this information is outside of the scope of ECMA-376.</p> <p data-bbox="415 615 1459 684">If this attribute is omitted, then no date information shall be associated with the parent annotation type.</p> <p data-bbox="415 720 1411 789"><i>[Example: Consider a comment represented using the following WordprocessingML fragment:</i></p> <p data-bbox="456 825 1110 926"> <pre data-bbox="456 825 1110 926"><... w:id="1" w:date="2006-01-01T10:00:00"> ... </...></pre> </p> <p data-bbox="415 961 1484 1031">The date attribute specifies that the date of the current annotation is January 1st 2006 at 10:00 AM, which can be used as desired. <i>end example</i></p> <p data-bbox="415 1066 1459 1136">The possible values for this attribute are defined by the ST_DateTime simple type (Part 1, §17.18.9).</p>
id (Annotation Identifier)	<p data-bbox="415 1161 1435 1230">Specifies a unique identifier for an annotation within a WordprocessingML document. The restrictions on the id attribute, if any, are defined by the parent XML element.</p> <p data-bbox="415 1266 1200 1297">If this attribute is omitted, then the document is non-conformant.</p> <p data-bbox="415 1333 1443 1402"><i>[Example: Consider an annotation represented using the following WordprocessingML fragment:</i></p> <p data-bbox="456 1438 696 1539"> <pre data-bbox="456 1438 696 1539"><... w:id="1" ... > ... </...></pre> </p> <p data-bbox="415 1575 1459 1644">The id attribute specifies that the ID of the current annotation is 1. This value is used to uniquely identify this annotation within the document content. <i>end example</i></p> <p data-bbox="415 1680 1476 1749">The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
original (Previous Numbering Value)	<p data-bbox="415 1776 1476 1845">Specifies the previous numbering displayed by the parent numbering change revision. Its format is specified by the parent element.</p>

Attributes	Description
	<p>If this attribute is omitted, then no previous numbering value is implied and applications can choose to calculate this value, or display no previous numbering value.</p> <p>[<i>Example:</i> Consider the following paragraph containing a single LISTNUM field with a revision, as follows:</p> <p style="text-align: center;">Some 1<ins>2</ins>. text</p> <p>This revision to the field result would be stored as follows in the WordprocessingML:</p> <pre><w:fldChar w:fldCharType="begin"> <w:numberingChange w:id="0" ... w:original="1." /> </w:fldChar></pre> <p>The original attribute specifies that the previous numbering value of the field was 1. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_TrackChangeNumbering](#)) is located in §A.1. *end note*]

14.7 Settings

14.7.1 Legacy Password Hash Algorithm

When a password hash value is stored using the transitional hashing mechanism described in the following subclause, that process shall be done in two stages:

The following steps assume that all words are unsigned, the word size is two bytes, and that bit-level SHL/SHR operations shift in the direction of the highest-order and lowest-order bit, respectively. [*Example:* 0x61 SHR 1 is 0xC2, as 01100001 shifted one position in the direction of its highest-order bit is 11000010. *end example*]

The UTF-16LE encoded password shall be hashed using the following algorithm (if there is a leading BOM character (U+FEFF) in the encoded password it is removed before hash calculation):

- Passwords of 15 or fewer characters shall be used in the hash without further change; passwords longer than 15 characters shall be truncated to 15 characters.
- Construct a new NULL-terminated string consisting of single-byte values using the algorithm described by the following bullet. The input to this step should be the series of UTF-16 characters defined above:
 - Get the single-byte values by iterating through the Unicode characters of the truncated password. For each character, if the low byte is not equal to 0, take it. Otherwise, take the high byte.

- From now on, the single-byte character string is used.
- If the password is empty, return 0.
- Compute the high-order word of the new key:
 - Initialize from the initial code array (see below), depending on the password's length. For each character in the password:
 - For every bit in the character, starting with the least significant and progressing to (but excluding) the most significant, if the bit is set, XOR the key's high-order word with the corresponding word from the encryption matrix
- Compute the low-order word of the new key:
 - Initialize with 0
 - For each character in the password, going backwards, low-order word = (((low-order word SHR 14) AND 0x0001) OR (low-order word SHL 1) AND 0x7FFF) XOR character
 - Lastly, low-order word = (((low-order word SHR 14) AND 0x0001) OR (low-order word SHL 1) AND 0x7FFF) XOR password length XOR 0xCE4B.

Initial code array

The initial code array contains the initial values for the key's high-order word. The initial value depends on the length of the password, as follows:

Password length	Initial value for the key's high-order word
1	0xE1F0
2	0x1D0F
3	0xCC9C
4	0x84C0
5	0x110C
6	0x0E10
7	0xF1CE
8	0x313E
9	0x1872
10	0xE139
11	0xD40F
12	0x84F9
13	0x280C
14	0xA96A
15	0x4EC3

Encryption matrix

The encryption matrix contains codes used during the calculation of the key's high-order word. As described in the algorithm above, for every bit of the password's characters, if the bit is set, a corresponding value is taken from this encryption matrix and is used to XOR the key's high-order word with it. Each row in the encryption matrix corresponds to a single character from the password, and each of the seven columns corresponds to a particular bit (0-6) in this character.

The values are taken in such a way so that the last character of the password uses the last row in the encryption matrix. The next-to-last character uses the next-to-last row in the matrix, and so on. This means that the beginning of the matrix might be unused, depending on the length of the password.

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
Last-14	0xAEFC	0x4DD9	0x9BB2	0x2745	0x4E8A	0x9D14	0x2A09
Last-13	0x7B61	0xF6C2	0xFDA5	0xEB6B	0xC6F7	0x9DCF	0x2BBF
Last-12	0x4563	0x8AC6	0x05AD	0x0B5A	0x16B4	0x2D68	0x5AD0
Last-11	0x0375	0x06EA	0x0DD4	0x1BA8	0x3750	0x6EA0	0xDD40
Last-10	0xD849	0xA0B3	0x5147	0xA28E	0x553D	0xAA7A	0x44D5
Last-9	0x6F45	0xDE8A	0xAD35	0x4A4B	0x9496	0x390D	0x721A
Last-8	0xEB23	0xC667	0x9CEF	0x29FF	0x53FE	0xA7FC	0x5FD9
Last-7	0x47D3	0x8FA6	0x0F6D	0x1EDA	0x3DB4	0x7B68	0xF6D0
Last-6	0xB861	0x60E3	0xC1C6	0x93AD	0x377B	0x6EF6	0xDDEC
Last-5	0x45A0	0x8B40	0x06A1	0x0D42	0x1A84	0x3508	0x6A10
Last-4	0xAA51	0x4483	0x8906	0x022D	0x045A	0x08B4	0x1168
Last-3	0x76B4	0xED68	0xCAF1	0x85C3	0x1BA7	0x374E	0x6E9C
Last-2	0x3730	0x6E60	0xDCC0	0xA9A1	0x4363	0x86C6	0x1DAD
Last-1	0x3331	0x6662	0xCCC4	0x89A9	0x0373	0x06E6	0x0DCC
Last	0x1021	0x2042	0x4084	0x8108	0x1231	0x2462	0x48C4

[*Example:* Consider a password which has been supplied - the string "Example". It is already under 15 characters, so truncation does not affect it. It is then converted to a string of single-byte characters.

- The password is 7 characters long, so, from the initial code array, the initial value for the key's high-order word is 0xF1CE.
- The key's high-order word is then computed further depending on the password's characters:
 - The first character is 'E' (0x45). This is the first character of a 7-character password, so its corresponding row in the encryption matrix is "Last-6".

- Bit 0 is set, therefore the key's high-order word is combined (via XOR) with the corresponding value for Bit 0 on row "Last-6", which is 0xB861. The new result is 0xF1CE XOR 0xB861 = 0x49AF.
- Bit 2 is set, so the key's high-order word is XOR-ed with the corresponding value for Bit 2 on row "Last-6", which is 0xC1C6. The new result is 0x49AF XOR 0xC1C6 = 0x8869.
- This process is repeated for each bit.
- The next character is 'x' (0x78). Its corresponding row in the encryption matrix is "Last-5".
 - Bit 3 is set. The value for Bit 3 on row "Last-5" in the encryption matrix is 0x0D42. The current value for the key's high-order byte is 0x5585, so the new one should be 0x5585 XOR 0x0D42 = 0x58C7.
 - This process is repeated for each bit.
- This process is repeated for all characters.
- After the last character has been processed, the above step produced 0x64CE for the key's high-order word. Now the low-order word needs to be calculated:
 - The initial value is 0.
 - It is then calculated using the password:
 - The last character of the password is 'e' (0x65), so, by the formula, low-order word = (((low-order word SHR 14) AND 0x0001) OR ((low-order word SHL 1) AND 0x7FFF)) XOR 'e' = (((0 SHR 14) AND 0x0001) OR ((0 SHL 1) AND 0x7FFF)) XOR 0x65 = 0x0065.
 - The next to last character of the password is 'l' (0x6C). Again, by the formula, (((0x0065 SHR 14) AND 0x0001) OR ((0x0065 SHL 1) AND 0x7FFF)) XOR 0x6C = (0x0000 OR 0x00CA) XOR 0x6C = 0x00CA XOR 0x6C = 0x00A6.
 - This process is repeated for each character.
 - After the password's first character has been processed, we have 0x1199 for the key's low-order word. Lastly, the password's length is combined into it: low-order word = (((0x1199 SHR 14) AND 0x0001) OR ((0x1199 SHL 1) AND 0x7FFF)) XOR 0x0007 XOR 0xCE4B = 0x2332 XOR 0x0007 XOR 0xCE4B = 0x2335 XOR 0xCE4B = 0xED7E.
- The end result for the key is 0x64CEED7E.

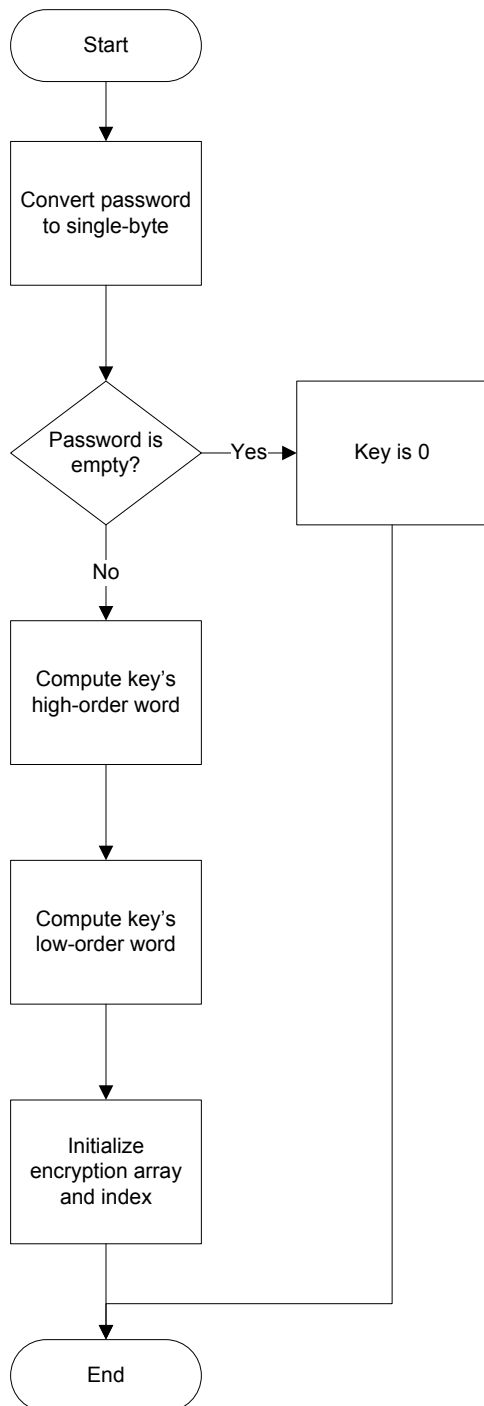
end example]

[*Rationale:* This pre-processing step is necessary for compatibility with legacy word processing applications which hashed their password solely using this mechanism. *end rationale]*

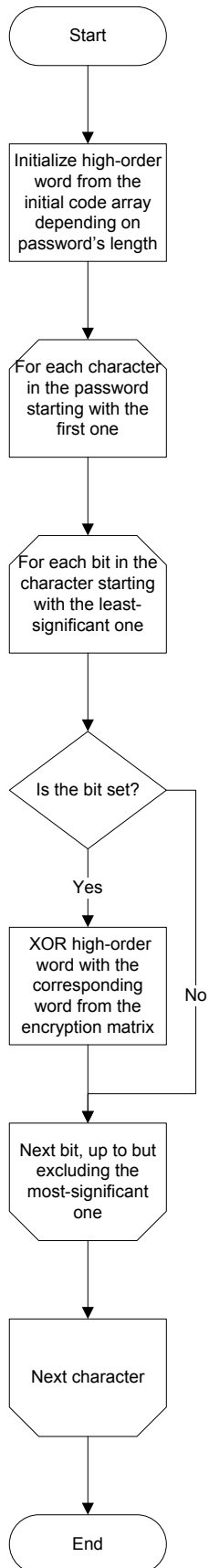
Second, the byte order of the result shall be reversed [*Example:* 0x64CEED7E becomes 7EEDCE64. *end example]*, and that value shall be hashed as defined by the attribute values.

[*Note:* The algorithm above can be stated as follows using diagrams:

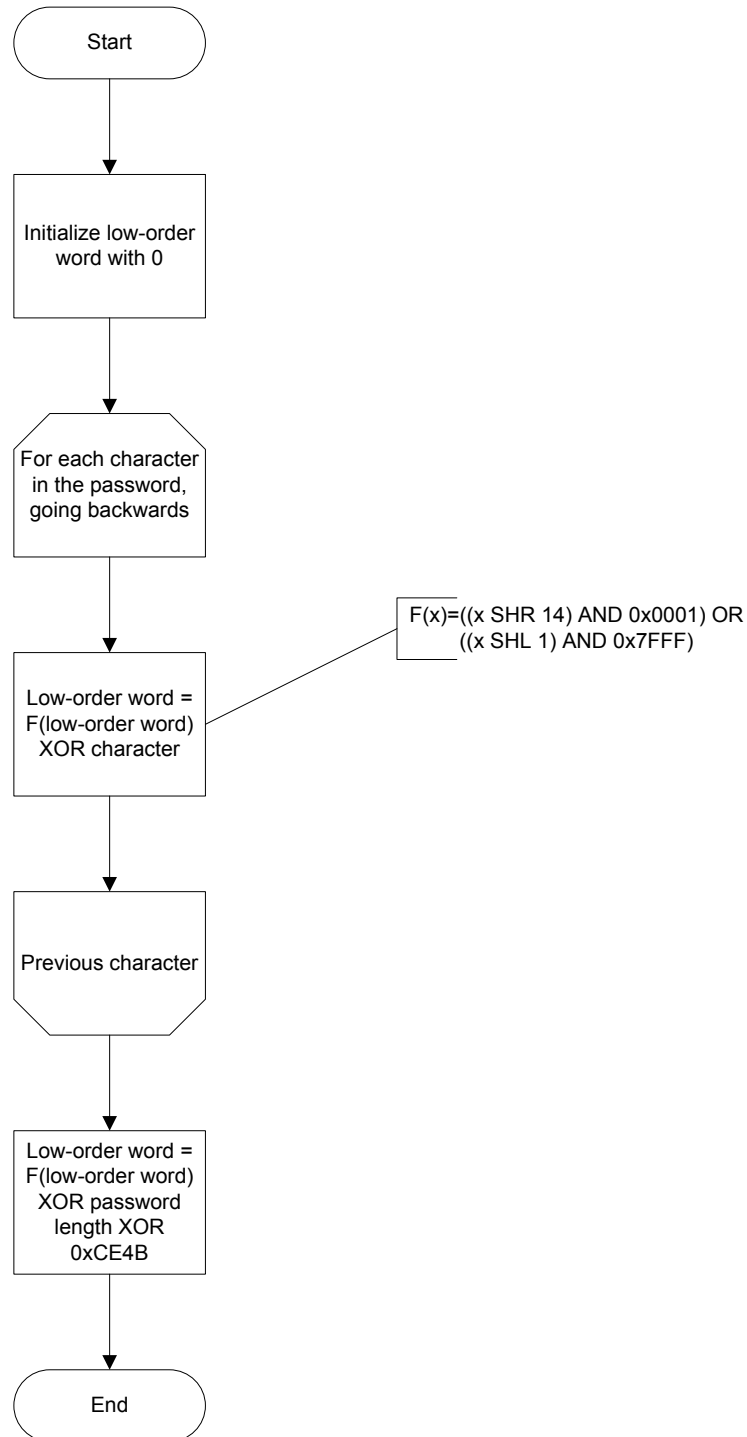
Calculate Key



Compute Key's High-Order Word



Compute Key's Low-Order Word



end note]

[*Example:* Consider a WordprocessingML document which specifies that applications must not allow any modifications to this document other than the addition of comments. This requirement would be specified using the following WordprocessingML in the document settings:

```
<w:documentProtection w:edit="comments" w:enforcement="true" ...
  w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny"
  w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" />
```

The documentProtection element has an edit attribute value of comments, specifying that the only modification allowed should be comments, the enforcement attribute has a value of true, specifying that the document protection specified is to be enforced on the given document. Finally, in order for the hosting application to stop enforcement of the document protection applied to the document, the hosting application would have to be provided with a password that the hosting application would then hash, compare to the value of the hash attribute (9oN7nWkCAyEZib1RomSJTjmPpCY=), and if the two values matched, halt enforcement of any document protection. *end example*]

14.7.2 Document Settings

14.7.2.1 hdrShapeDefaults (Default Properties for VML Objects in Header and Footer)

This element specifies the default parameters for object using the VML syntax (§19.1) inserted in the header and footer of a WordprocessingML document. The definition and semantics of these parameters is described in the VML - Office Drawing subclause (§19.2) of ECMA-376.

If this element is omitted, then no default properties are applied to VML objects in the header and footer of this document.

[*Example:* Consider a WordprocessingML document whose document settings contain the following markup:

```
<w:hdrShapeDefaults>
  <o:shapedefaults v:ext="edit" spidmax="2050" fillcolor="none [3207]"
strokecolor="none [3041]">
    <v:fill color="none [3207]" />
    <v:stroke color="none [3041]" weight="3pt" />
    <v:shadow on="t" type="perspective" color="none [1607]" opacity=".5"
offset="1pt" offset2="1pt" />
  </o:shapedefaults>
  <o:shapelayout v:ext="edit">
    <o:idmap v:ext="edit" data="2" />
  </o:shapelayout>
</w:hdrShapeDefaults>
```

The hdrShapeDefaults element specifies a set of shape defaults which must be applied to the set of all shapes present in the header and footer of this document. *end example*]

[Note: The W3C XML Schema definition of this element's content model ([CT_ShapeDefaults](#)) is located in §A.1. *end note*]

14.7.2.2 [shapeDefaults \(Default Properties for VML Objects in Main Document\)](#)

This element specifies the default parameters for object using the VML syntax (§19.1) inserted in the body (the main document story, comments, footnotes, and endnotes) of the WordprocessingML document. The definition and semantics of these parameters is described in the VML - Office Drawing subclause (§19.2) of ECMA-376.

If this element is omitted, then no default properties are applied to VML objects in the body of this document.

[Example: Consider a WordprocessingML document whose document settings contain the following markup:

```
<w:shapeDefaults>
  <o:shapedefaults v:ext="edit" spidmax="1026" />
  <o:shapelayout v:ext="edit">
    <o:idmap v:ext="edit" data="1" />
  </o:shapelayout>
</w:shapeDefaults>
```

The shapeDefaults element specifies a set of shape defaults which must be applied to the set of all shapes present in the body document. *end example*]

[Note: The W3C XML Schema definition of this element's content model ([CT_ShapeDefaults](#)) is located in §A.1. *end note*]

14.7.2.3 [Additional attributes for documentProtection element \(Part 1, §17.15.1.29\)](#)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
algIdExt (Cryptographic Algorithm Extensibility)	<p>Specifies that a cryptographic algorithm which was not defined by ECMA-376 has been used to generate the hash value stored with this document.</p> <p>This value, when present, shall be interpreted based on the value of the algIdExtSource attribute in order to determine the algorithm used, which shall be application-defined. [Rationale: This extensibility affords the fact that with exponentially increasing computing power, documents created in the future might need to utilize as yet undefined hashing algorithms in order to remain secure. <i>end rationale</i>]</p> <p>If this value is present, the cryptAlgorithmClass, cryptAlgorithmType, and cryptAlgorithmSid attribute values shall be ignored in favor of the algorithm defined by this attribute.</p> <p>[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:algIdExt="0000000A"</pre>

Attributes	Description
	<p><code>w:algIdExtSource="futureCryptography"</code> <code>w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></code></p> <p>The <code>algIdExt</code> attribute value of <code>0000000A</code> specifies that the algorithm with hex code A must be used as defined by the <code>futureCryptography</code> application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_LongHexNumber</code> simple type (Part 1, §17.18.50).</p>
<p><code>algIdExtSource</code> (Algorithm Extensibility Source)</p>	<p>Specifies the application which defined the algorithm value specified by the <code>algIdExt</code> attribute.</p> <p>[<i>Example</i>: Consider a <code>WordprocessingML</code> document with the following information stored in one of its protection elements:</p> <p><code><... w:algIdExt="0000000A"</code> <code>w:algIdExtSource="futureCryptography"</code> <code>w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></code></p> <p>The <code>algIdExtSource</code> attribute value of <code>futureCryptography</code> specifies that the algorithm used here was published by the <code>futureCryptography</code> application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_String</code> simple type (Part 1, §22.9.2.13).</p>
<p><code>cryptAlgorithmClass</code> (Cryptographic Algorithm Class)</p>	<p>Specifies the class of cryptographic algorithm used by this protection. [<i>Note</i>: The initial version of ECMA-376 only supports a single version - <code>hash</code> - but future versions can expand this as necessary. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a <code>WordprocessingML</code> document with the following information stored in one of its protection elements:</p> <p><code><... w:cryptAlgorithmClass="hash"</code> <code>w:cryptAlgorithmType="typeAny"</code> <code>w:cryptAlgorithmSid="1"</code> <code>w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></code></p> <p>The <code>cryptAlgorithmClass</code> attribute value of <code>hash</code> specifies that the algorithm used for the password is a hashing algorithm. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_AlgClass</code> simple type (§20.1.2.1).</p>
<p><code>cryptAlgorithmSid</code> (Cryptographic Hashing Algorithm)</p>	<p>Specifies the specific cryptographic hashing algorithm which shall be used along with the salt attribute and user-supplied password in order to compute a hash value for comparison.</p> <p>The possible values for this attribute shall be interpreted as follows:</p>

Attributes	Description																																
	<table border="1" data-bbox="415 245 1349 1056"> <thead> <tr> <th data-bbox="415 245 586 294">Value</th><th data-bbox="586 245 1349 294">Algorithm</th></tr> </thead> <tbody> <tr><td data-bbox="415 294 586 342">1</td><td data-bbox="586 294 1349 342">MD2</td></tr> <tr><td data-bbox="415 342 586 390">2</td><td data-bbox="586 342 1349 390">MD4</td></tr> <tr><td data-bbox="415 390 586 438">3</td><td data-bbox="586 390 1349 438">MD5</td></tr> <tr><td data-bbox="415 438 586 487">4</td><td data-bbox="586 438 1349 487">SHA-1</td></tr> <tr><td data-bbox="415 487 586 535">5</td><td data-bbox="586 487 1349 535">MAC</td></tr> <tr><td data-bbox="415 535 586 583">6</td><td data-bbox="586 535 1349 583">RIPEMD</td></tr> <tr><td data-bbox="415 583 586 632">7</td><td data-bbox="586 583 1349 632">RIPEMD-160</td></tr> <tr><td data-bbox="415 632 586 680">8</td><td data-bbox="586 632 1349 680">Undefined. Shall not be used.</td></tr> <tr><td data-bbox="415 680 586 728">9</td><td data-bbox="586 680 1349 728">HMAC</td></tr> <tr><td data-bbox="415 728 586 777">10</td><td data-bbox="586 728 1349 777">Undefined. Shall not be used.</td></tr> <tr><td data-bbox="415 777 586 825">11</td><td data-bbox="586 777 1349 825">Undefined. Shall not be used.</td></tr> <tr><td data-bbox="415 825 586 873">12</td><td data-bbox="586 825 1349 873">SHA-256</td></tr> <tr><td data-bbox="415 873 586 921">13</td><td data-bbox="586 873 1349 921">SHA-384</td></tr> <tr><td data-bbox="415 921 586 970">14</td><td data-bbox="586 921 1349 970">SHA-512</td></tr> <tr><td data-bbox="415 970 586 1056">Any other value</td><td data-bbox="586 970 1349 1056">Undefined. Shall not be used.</td></tr> </tbody> </table> <p data-bbox="415 1094 1401 1161">[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre data-bbox="451 1199 1127 1335"> <... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="4" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /> </pre> <p data-bbox="415 1373 1443 1440">The cryptAlgorithmSid attribute value of 4 specifies that the SHA-1 hashing algorithm must be used to generate a hash from the user-defined password. <i>end example</i>]</p> <p data-bbox="415 1478 1471 1545">The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>	Value	Algorithm	1	MD2	2	MD4	3	MD5	4	SHA-1	5	MAC	6	RIPEMD	7	RIPEMD-160	8	Undefined. Shall not be used.	9	HMAC	10	Undefined. Shall not be used.	11	Undefined. Shall not be used.	12	SHA-256	13	SHA-384	14	SHA-512	Any other value	Undefined. Shall not be used.
Value	Algorithm																																
1	MD2																																
2	MD4																																
3	MD5																																
4	SHA-1																																
5	MAC																																
6	RIPEMD																																
7	RIPEMD-160																																
8	Undefined. Shall not be used.																																
9	HMAC																																
10	Undefined. Shall not be used.																																
11	Undefined. Shall not be used.																																
12	SHA-256																																
13	SHA-384																																
14	SHA-512																																
Any other value	Undefined. Shall not be used.																																
cryptAlgorithmType (Cryptographic Algorithm Type)	<p data-bbox="415 1562 1443 1667">Specifies the type of cryptographic algorithm used by this protection. [Note: The initial version of ECMA-376 only supports a single algorithm type - typeAny - but future versions can expand this as necessary. <i>end note</i>]</p> <p data-bbox="415 1705 1401 1772">[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre data-bbox="451 1810 964 1879"> <... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" </pre>																																

Attributes	Description
	<pre>w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmType attribute value of typeAny specifies that any type of algorithm might have been used for the password. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_AlgType simple type (§20.1.2.2).</p>
cryptProvider (Cryptographic Provider)	<p>Specifies the cryptographic provider which was used to generate the hash value stored in this document. If the user provided a cryptographic provider which was not the system's built-in provider, then that provider shall be stored here so it can subsequently be used if available.</p> <p>If this attribute is omitted, then the built-in cryptographic provider on the system shall be used.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptProvider="Krista'sProvider" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProvider attribute value of Krista'sProvider specifies that the cryptographic provider with name "Krista's Provider" must be used if available. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
cryptProviderType (Cryptographic Provider Type)	<p>Specifies the type of cryptographic provider to be used.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptProviderType="rsaAES" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProviderType attribute value of rsaAES specifies that the cryptographic provider type must be an Advanced Encryption Standard provider. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_CryptProv simple type (§20.1.2.4).</p>
cryptProviderType Ext (Cryptographic Provider Type Extensibility)	<p>Specifies that a cryptographic provider type which was not defined by ECMA-376 has been used to generate the hash value stored with this document.</p> <p>This value, when present, shall be interpreted based on the value of the cryptProviderTypeExtSource attribute in order to determine the provider type used,</p>

Attributes	Description
	<p>which shall be application-defined. [<i>Rationale</i>: This extensibility affords the fact that with exponentially increasing computing power, documents created in the future might need to utilize as yet undefined cryptographic provider types in order to remain secure. <i>end rationale</i>]</p> <p>If this value is present, the cryptProviderType attribute value shall be ignored in favor of the provider type defined by this attribute.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptProviderTypeExt="00A5691D" w:cryptProvideTypeExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProviderTypeExt attribute value of 00A5691D specifies that the provider type associated with hex code A5691D must be used as defined by the futureCryptography application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_LongHexNumber simple type (Part 1, §17.18.50).</p>
cryptProviderTypeExtSource (Provider Type Extensibility Source)	<p>Specifies the application which defined the provider type value specified by the cryptProviderTypeExt attribute.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptProviderTypeExt="00A5691D" w:cryptProvideTypeExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProvideTypeExtSource attribute value of futureCryptography specifies that the provider type used here was published by the futureCryptography application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
cryptSpinCount (Iterations to Run Hashing Algorithm)	<p>Specifies the number of times the hashing function shall be iteratively run (runs using each iteration's result plus a 4 byte value (0-based, little endian) containing the number of the iteration as the input for the next iteration) when attempting to compare a user-supplied password with the value stored in the hash attribute. [<i>Rationale</i>: Running the algorithm many times increases the cost of exhaustive search attacks correspondingly. Storing this value allows for the number of iterations to be increased over time to accommodate faster hardware (and hence the ability to run more iterations in less time). <i>end rationale</i>]</p>

Attributes	Description
	<p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptSpinCount="100000" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptSpinCount attribute value of 100000 specifies that the hashing function must be run one hundred thousand times to generate a hash value for comparison with the hash attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
hash (Password Hash)	<p>Specifies the hash value for the password stored with this document. This value shall be compared with the resulting hash value after hashing the user-supplied password using the algorithm specified by the preceding attributes and parent XML element, and if the two values match, the protection shall no longer be enforced.</p> <p>If this value is omitted, then no password shall be associated with the protection, and it can be turned off without supplying any password.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The hash attribute value of 9oN7nWkCAyEZib1RomSJTjmPpCY= specifies that the user-supplied password must be hashed using the pre-processing defined by the parent element (if any) followed by the SHA-1 algorithm (specified via the cryptAlgorithmSid attribute value of 1) and that the resulting hash value must be 9oN7nWkCAyEZib1RomSJTjmPpCY= for the protection to be disabled. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>
salt (Salt for Password Verifier)	<p>Specifies the salt which was prepended to the user-supplied password before it was hashed using the hashing algorithm defined by the preceding attribute values to generate the hash attribute, and which shall also be prepended to the user-supplied password before attempting to generate a hash value for comparison. A <i>salt</i> is a random string which is added to a user-supplied password before it is hashed in order to prevent a malicious party from pre-calculating all possible password/hash combinations and simply using those precalculated values (often referred to as a "dictionary attack").</p> <p>If this attribute is omitted, then no salt shall be prepended to the user-supplied password before it is hashed for comparison with the stored hash value.</p>

Attributes	Description
	<p>[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:salt="ZUdHa+D8F/OAKP3I7ssUnQ==" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The salt attribute value of ZUdHa+D8F/OAKP3I7ssUnQ== specifies that the user-supplied password must have this value prepended before it is run through the specified hashing algorithm to generate a resulting hash value for comparison. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>

14.7.2.4 [Additional attribute for stylePaneFormatFilter element \(Part 1, §17.15.1.85\)](#)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description																						
val (Bitmask of Suggested Filtering Options)	<p>Specifies a bitmask of the following filtering options:</p> <table> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>0x0001</td><td>Specifies that all styles present in the styles part should be displayed in the list of document styles.</td></tr> <tr> <td>0x0002</td><td>Specifies that only styles with the customStyle attribute should be displayed in the list of document styles.</td></tr> <tr> <td>0x0004</td><td>Specifies that all latent styles should be displayed in the list of document styles.</td></tr> <tr> <td>0x0008</td><td>Specifies that only styles used in the document should be displayed in the list of document styles.</td></tr> <tr> <td>0x0010</td><td>Undefined. Shall not be used.</td></tr> <tr> <td>0x0020</td><td>Specifies that heading styles (styles with a styleId of Heading1 to Heading9) should be displayed in the list of document styles when the previous style is used in the document and/or is present in the styles part.</td></tr> <tr> <td>0x0040</td><td>Specifies that numbering styles should be displayed in the list of document styles.</td></tr> <tr> <td>0x0080</td><td>Specifies that table styles should be displayed in the list of document styles.</td></tr> <tr> <td>0x0100</td><td>Specifies that all unique forms of run-level direct formatting should be displayed in the list of document styles as though they were each a unique style.</td></tr> <tr> <td>0x0200</td><td>Specifies that all unique forms of paragraph-level direct formatting should</td></tr> </table>	Value	Description	0x0001	Specifies that all styles present in the styles part should be displayed in the list of document styles.	0x0002	Specifies that only styles with the customStyle attribute should be displayed in the list of document styles.	0x0004	Specifies that all latent styles should be displayed in the list of document styles.	0x0008	Specifies that only styles used in the document should be displayed in the list of document styles.	0x0010	Undefined. Shall not be used.	0x0020	Specifies that heading styles (styles with a styleId of Heading1 to Heading9) should be displayed in the list of document styles when the previous style is used in the document and/or is present in the styles part.	0x0040	Specifies that numbering styles should be displayed in the list of document styles.	0x0080	Specifies that table styles should be displayed in the list of document styles.	0x0100	Specifies that all unique forms of run-level direct formatting should be displayed in the list of document styles as though they were each a unique style.	0x0200	Specifies that all unique forms of paragraph-level direct formatting should
Value	Description																						
0x0001	Specifies that all styles present in the styles part should be displayed in the list of document styles.																						
0x0002	Specifies that only styles with the customStyle attribute should be displayed in the list of document styles.																						
0x0004	Specifies that all latent styles should be displayed in the list of document styles.																						
0x0008	Specifies that only styles used in the document should be displayed in the list of document styles.																						
0x0010	Undefined. Shall not be used.																						
0x0020	Specifies that heading styles (styles with a styleId of Heading1 to Heading9) should be displayed in the list of document styles when the previous style is used in the document and/or is present in the styles part.																						
0x0040	Specifies that numbering styles should be displayed in the list of document styles.																						
0x0080	Specifies that table styles should be displayed in the list of document styles.																						
0x0100	Specifies that all unique forms of run-level direct formatting should be displayed in the list of document styles as though they were each a unique style.																						
0x0200	Specifies that all unique forms of paragraph-level direct formatting should																						

Attributes	Description
	be displayed in the list of document styles as though they were each a unique style.
	0x0400 Specifies that all unique forms of direct formatting of numbering data should be displayed in the list of document styles as though they were each a unique style.
	0x0800 Specifies that all unique forms of direct formatting of tables should be displayed in the list of document styles as though they were each a unique style.
	0x1000 Specifies that a style should be present which removes all formatting and styles from text.
	0x2000 Specifies that heading styles with a styleId of Heading1 to Heading3 should always be displayed in the list of document styles.
	0x4000 Specifies that styles should only be shown the semiHidden element (Part 1, §17.7.4.16) is false and the hidden element (Part 1, §17.7.4.4) is false.
	0x8000 Specifies that primary names for styles should not be shown if an alternate name using the name element (Part 1, §17.7.4.9) exists.
	Any other value Undefined. Shall not be used.
<p>[Example: Consider a document with the following value in its document settings:</p> <pre><w:stylePaneFormatFilter w:val="2002" /></pre> <p>The val attribute specifies two suggested filter options for the list of document styles:</p> <ul style="list-style-type: none"> • Only custom styles should be shown (0002) • Heading styles with a styleId of Heading1 to Heading3 should always be displayed in the list (2000) <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ShortHexNumber simple type (Part 1, §17.18.79).</p>	

14.7.2.5 Additional attributes for writeProtection element (Part 1, §17.15.1.93)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
algIdExt (Cryptographic Algorithm)	Specifies that a cryptographic algorithm which was not defined by ECMA-376 has been used to generate the hash value stored with this document.

Attributes	Description
Extensibility)	<p>This value, when present, shall be interpreted based on the value of the algIdExtSource attribute in order to determine the algorithm used, which shall be application-defined. <i>[Rationale: This extensibility affords the fact that with exponentially increasing computing power, documents created in the future might need to utilize as yet undefined hashing algorithms in order to remain secure. end rationale]</i></p> <p>If this value is present, the cryptAlgorithmClass, cryptAlgorithmType, and cryptAlgorithmSid attribute values shall be ignored in favor of the algorithm defined by this attribute.</p> <p><i>[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</i></p> <pre><... w:algIdExt="0000000A" w:algIdExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The algIdExt attribute value of 0000000A specifies that the algorithm with hex code A must be used as defined by the futureCryptography application. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_LongHexNumber simple type (Part 1, §17.18.50).</p>
algIdExtSource (Algorithm Extensibility Source)	<p>Specifies the application which defined the algorithm value specified by the algIdExt attribute.</p> <p><i>[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</i></p> <pre><... w:algIdExt="0000000A" w:algIdExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The algIdExtSource attribute value of futureCryptography specifies that the algorithm used here was published by the futureCryptography application. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
cryptAlgorithmClass (Cryptographic Algorithm Class)	<p>Specifies the class of cryptographic algorithm used by this protection. <i>[Note: The initial version of ECMA-376 only supports a single version - hash - but future versions can expand this as necessary. end note]</i></p> <p><i>[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</i></p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny"</pre>

Attributes	Description																																
	<pre>w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmClass attribute value of hash specifies that the algorithm used for the password is a hashing algorithm. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_AlgClass simple type (§20.1.2.1).</p>																																
cryptAlgorithmSid (Cryptographic Hashing Algorithm)	<p>Specifies the specific cryptographic hashing algorithm which shall be used along with the salt attribute and user-supplied password in order to compute a hash value for comparison.</p> <p>The possible values for this attribute shall be interpreted as follows:</p> <table data-bbox="415 716 1349 1528"> <tr> <th>Value</th><th>Algorithm</th></tr> <tr><td>1</td><td>MD2</td></tr> <tr><td>2</td><td>MD4</td></tr> <tr><td>3</td><td>MD5</td></tr> <tr><td>4</td><td>SHA-1</td></tr> <tr><td>5</td><td>MAC</td></tr> <tr><td>6</td><td>RIPEMD</td></tr> <tr><td>7</td><td>RIPEMD-160</td></tr> <tr><td>8</td><td>Undefined. Shall not be used.</td></tr> <tr><td>9</td><td>HMAC</td></tr> <tr><td>10</td><td>Undefined. Shall not be used.</td></tr> <tr><td>11</td><td>Undefined. Shall not be used.</td></tr> <tr><td>12</td><td>SHA-256</td></tr> <tr><td>13</td><td>SHA-384</td></tr> <tr><td>14</td><td>SHA-512</td></tr> <tr><td>Any other value</td><td>Undefined. Shall not be used.</td></tr> </table> <p>[Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="4" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmSid attribute value of 4 specifies that the SHA-1 hashing algorithm</p>	Value	Algorithm	1	MD2	2	MD4	3	MD5	4	SHA-1	5	MAC	6	RIPEMD	7	RIPEMD-160	8	Undefined. Shall not be used.	9	HMAC	10	Undefined. Shall not be used.	11	Undefined. Shall not be used.	12	SHA-256	13	SHA-384	14	SHA-512	Any other value	Undefined. Shall not be used.
Value	Algorithm																																
1	MD2																																
2	MD4																																
3	MD5																																
4	SHA-1																																
5	MAC																																
6	RIPEMD																																
7	RIPEMD-160																																
8	Undefined. Shall not be used.																																
9	HMAC																																
10	Undefined. Shall not be used.																																
11	Undefined. Shall not be used.																																
12	SHA-256																																
13	SHA-384																																
14	SHA-512																																
Any other value	Undefined. Shall not be used.																																

Attributes	Description
	<p>must be used to generate a hash from the user-defined password. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
<p>cryptAlgorithmType (Cryptographic Algorithm Type)</p>	<p>Specifies the type of cryptographic algorithm used by this protection. [<i>Note</i>: The initial version of ECMA-376 only supports a single algorithm type - typeAny - but future versions can expand this as necessary. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmType attribute value of typeAny specifies that any type of algorithm might have been used for the password. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_AlgType simple type (§20.1.2.2).</p>
<p>cryptProvider (Cryptographic Provider)</p>	<p>Specifies the cryptographic provider which was used to generate the hash value stored in this document. If the user provided a cryptographic provider which was not the system's built-in provider, then that provider shall be stored here so it can subsequently be used if available.</p> <p>If this attribute is omitted, then the built-in cryptographic provider on the system shall be used.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptProvider="Krista'sProvider" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProvider attribute value of Krista'sProvider specifies that the cryptographic provider with name "Krista's Provider" must be used if available. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
<p>cryptProviderType (Cryptographic Provider Type)</p>	<p>Specifies the type of cryptographic provider to be used.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p>

Attributes	Description
	<p><... w:cryptProviderType="rsaAES" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></p> <p>The cryptProviderType attribute value of rsaAES specifies that the cryptographic provider type must be an Advanced Encryption Standard provider. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_CryptProv simple type (§20.1.2.4).</p>
cryptProviderTypeExt (Cryptographic Provider Type Extensibility)	<p>Specifies that a cryptographic provider type which was not defined by ECMA-376 has been used to generate the hash value stored with this document.</p> <p>This value, when present, shall be interpreted based on the value of the cryptProviderTypeExtSource attribute in order to determine the provider type used, which shall be application-defined. [<i>Rationale:</i> This extensibility affords the fact that with exponentially increasing computing power, documents created in the future might need to utilize as yet undefined cryptographic provider types in order to remain secure. <i>end rationale]</i></p> <p>If this value is present, the cryptProviderType attribute value shall be ignored in favor of the provider type defined by this attribute.</p> <p>[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <p><... w:cryptProviderTypeExt="00A5691D" w:cryptProvideTypeExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></p> <p>The cryptProviderTypeExt attribute value of 00A5691D specifies that the provider type associated with hex code A5691D must be used as defined by the futureCryptography application. <i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_LongHexNumber simple type (Part 1, §17.18.50).</p>
cryptProviderTypeExtSource (Provider Type Extensibility Source)	<p>Specifies the application which defined the provider type value specified by the cryptProviderTypeExt attribute.</p> <p>[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <p><... w:cryptProviderTypeExt="00A5691D" w:cryptProvideTypeExtSource="futureCryptography" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></p> <p>The cryptProvideTypeExtSource attribute value of futureCryptography specifies that the provider type used here was published by the futureCryptography application. <i>end</i></p>

Attributes	Description
	<p><i>example]</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>
<p>cryptSpinCount (Iterations to Run Hashing Algorithm)</p>	<p>Specifies the number of times the hashing function shall be iteratively run (runs using each iteration's result plus a 4 byte value (0-based, little endian) containing the number of the iteration as the input for the next iteration) when attempting to compare a user-supplied password with the value stored in the hash attribute. [<i>Rationale</i>: Running the algorithm many times increases the cost of exhaustive search attacks correspondingly. Storing this value allows for the number of iterations to be increased over time to accommodate faster hardware (and hence the ability to run more iterations in less time). <i>end rationale</i>]</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptSpinCount="100000" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptSpinCount attribute value of 100000 specifies that the hashing function must be run one hundred thousand times to generate a hash value for comparison with the hash attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>
<p>hash (Password Hash)</p>	<p>Specifies the hash value for the password stored with this document. This value shall be compared with the resulting hash value after hashing the user-supplied password using the algorithm specified by the preceding attributes and parent XML element, and if the two values match, the protection shall no longer be enforced.</p> <p>If this value is omitted, then no password shall be associated with the protection, and it can be turned off without supplying any password.</p> <p>[<i>Example</i>: Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The hash attribute value of 9oN7nWkCAyEZib1RomSJTjmPpCY= specifies that the user-supplied password must be hashed using the pre-processing defined by the parent element (if any) followed by the SHA-1 algorithm (specified via the cryptAlgorithmSid attribute value of 1) and that the resulting hash value must be 9oN7nWkCAyEZib1RomSJTjmPpCY= for the protection to be disabled. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.
salt (Salt for Password Verifier)	<p>Specifies the salt which was prepended to the user-supplied password before it was hashed using the hashing algorithm defined by the preceding attribute values to generate the hash attribute, and which shall also be prepended to the user-supplied password before attempting to generate a hash value for comparison. A <i>salt</i> is a random string which is added to a user-supplied password before it is hashed in order to prevent a malicious party from pre-calculating all possible password/hash combinations and simply using those precalculated values (often referred to as a "dictionary attack").</p> <p>If this attribute is omitted, then no salt shall be prepended to the user-supplied password before it is hashed for comparison with the stored hash value.</p> <p>[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:salt="ZUDHa+D8F/OAKP3I7ssUnQ==" w:hash="9oN7nWkCAYeZib1RomSJTjmPpCY=" /></pre> <p>The salt attribute value of ZUDHa+D8F/OAKP3I7ssUnQ== specifies that the user-supplied password must have this value prepended before it is run through the specified hashing algorithm to generate a resulting hash value for comparison. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>

14.7.3 Compatibility Settings

The last group of settings stored in WordprocessingML is compatibility settings. *Compatibility Settings* are optional settings used to preserve visual fidelity of documents created in earlier word processing applications. Some of these settings provide affordance for specific behaviors, described in detail below; and others simply instruct applications to mimic the behavior of an existing word processing application.

If compatibility settings are needed, they are stored in the Document Settings part.

It is important to note that all compatibility settings are optional in nature - applications can freely ignore all behaviors described within this section and these settings should not be added unless compatibility is specifically needed in one or more cases. The compatibility settings are provided for backward compatibility with documents created in legacy applications. As such, a number of the settings reference specific applications and specific versions of those applications. This is solely for backward compatibility reasons, and any of those settings are not intended for use by typical applications.

[*Note:* These settings can also be expressed using the generic compatSetting element defined in ECMA-376-1. *end note*]

[*Example:* Consider the following WordprocessingML fragment for the compatibility settings in a WordprocessingML document:

```
<w:settings>
...
<w:compat>
  <w:noTabHangInd />
</w:compat>
</w:settings>
```

The compat element contains all of the document settings for this document. In this case, the single setting applied is the suppression of a tab stop when using a hanging indent using the noTabHangInd element (§14.7.3.31). *end example*]

14.7.3.1 alignTablesRowByRow (Align Table Rows Independently)

This element specifies whether applications shall align each row within a table independently based on the alignment setting of the jc element (Part 1, §17.4.28) when displaying the contents of a table in a WordprocessingML document.

When the justification of a table using the jc element is typically applied, that alignment is applied to the contents of the table (the table is centered, left justified, or right-aligned), and then individual rows are laid out based on the resulting table's position. This element, when present with a val attribute value of true (or equivalent), specifies that each table row shall be independently aligned based on the table alignment setting, ignoring the placement of all other rows.

[*Example:* Consider a WordprocessingML document with a single centered table, whose second row is defined such that one-half of an inch is left before the row begins, as follows:

```
<w:tbl>
  <w:tblPr>
    <w:jc w:val="center" />
  </w:tblPr>
  <w:tr>
    ...
  </w:tr>
  <w:tr>
    <w:trPr>
      <w:gridBefore w:val="1" />
```

```
<w:wBefore w:w="720" w:type="dxa" />
</w:trPr>
...
</w:tr>
<w:tr>
...
</w:tr>
</w:tbl>
```

The default presentation would have the entire table centered, then the second row indented beyond that by 720 points:

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:alignTablesRowByRow />
</w:compat>
```

Then that second row would instead be centered on the page independently of the other table rows, resulting in the following output:

In this case, the wBefore element's value is ignored, since the row was centered on the line as a row, and there is no table to be indented relative to. *end example]*

This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.2 allowSpaceOfSameStyleInTable (Allow Contextual Spacing of Paragraphs in Tables)

This element specifies whether the suppression of additional space (contextual spacing) defined using the contextualSpacing element (Part 1, §17.3.1.9) shall be applied to paragraphs contained within tables.

Typically, the rules for the removal of additional paragraph spacing via the contextualSpacing element are applied to all paragraphs in a WordprocessingML document. This element, when present with a val attribute value of true (or equivalent), specifies that this setting shall always be ignored for paragraphs in table cells (and additional spacing shall be allowed).

[Example: Consider a WordprocessingML document with a default paragraph style with additional spacing after and contextual spacing set, as follows:

```
<w:style w:name="Normal" w:default="1">
...
<w:pPr>
  <w:spacing w:after="200" />
  <w:contextualSpacing />
</w:pPr>
</w:style>
```

The default presentation would have the spacing suppressed between all paragraphs, since they are all of the default paragraph style defined above (contextual spacing applies):

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:allowSpaceOfSameStyleInTable />
</w:compat>
```

Then the paragraphs in the table never have their spacing suppressed, resulting in the following output:

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.3 `autofitToFirstFixedWidthCell` (Allow Table Columns To Exceed Preferred Widths of Constituent Cells)

This element specifies that when performing an AutoFit on a table in a WordprocessingML document in order to display it, applications shall alter that logic slightly in order to mimic the behavior of a previous word processing application.

Normally, the AutoFit behavior of a table is as is described in the associated simple type. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that this logic shall be changed as follows:

- If the width of a grid column in a table has been set by a preferred table cell width, then that column's width can be enlarged by the content of cells which themselves do not have a preferred width (in contrast, the normal logic never allows the content of cells to override a preferred width on a grid column).

[*Example:* Consider a WordprocessingML table with only one preferred cell width, a width of 720 points on the second cell in the first column, as follows:

```
<w:tbl>
...
<w:tr>
  <w:tc>
    <w:p/>
```

```

    </w:tc>
    <w:tc>
      <w:p/>
    </w:tc>
  </w:tr>
  <w:tr>
    <w:tc>
      <w:tcPr>
        <w:tcW w:w="720" w:type="dxa" />
      </w:tcPr>
      <w:p/>
    </w:tc>
    <w:tc>
      <w:p/>
    </w:tc>
  </w:tr>
</w:tbl>

```

The default presentation would have the first column constrained to 720 points by the preferred width of the second cell in the first column:

This is an example of a cell with lots of content.	

However, if this compatibility setting is turned on:

```

<w:compat>
  <w:autoFitToFirstFixedWidthCell />
</w:compat>

```

Then the column would be resized proportionally based on the content (ignoring the preferred width in that row), resulting in the following output:

This is an example of a cell with lots of content.	

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.4 autoSpaceLikeWord95 (Incorrectly Adjust Text Spacing for Specific Unicode Ranges)

This element specifies adjustments (detailed below) which should be applied to the spacing between adjoining regions of non-ideographic and ideographic text when the autoSpaceDE (Part 1, §17.3.1.2) and autoSpaceDN (Part 1, §17.3.1.3) elements have a value of true (or equivalent). This algorithm typically results in the following:

- An increase in the inter-character spacing added between non-ideographic and/or number characters and certain full-width characters
- No inter-character spacing between non-ideographic and/or number characters and certain half-width characters

Typically, applications apply additional spacing between ideographic and non-ideographic characters/numeric characters when the autoSpaceDE / autoSpaceDN properties are applied. This element, when present with a val attribute value of true (or equivalent), specifies that applications shall apply the following adjustments to this logic:

- Characters in the following Unicode ranges should be treated as ideographic, even though those characters are full-width forms of non-ideographic text: U+FF10–U+FF19, U+FF21–U+FF3A, and U+FF41–U+FF5A. [*Note*: This results in the unnecessary addition of space. *end note*]
- Characters in the following Unicode ranges should be treated as non-ideographic, even though those characters are ideographic: U+FF66–U+FF9F. [*Note*: This results in the omission of the intended additional space. *end note*]

[*Example*: Consider a WordprocessingML document with two paragraphs containing a mix of East Asian and Latin characters:

```
<w:p>
  <w:r>
    <w:t>ab</w:t>
  </w:r>
  <w:r>
    <w:t>ㄗ</w:t>
  </w:r>
  <w:r>
    <w:t>ㄗ</w:t>
  </w:r>
  <w:r>
    <w:t>cd</w:t>
  </w:r>
</w:p>
<w:p>
  <w:r>
```



```

    <w:t>ab</w:t>
  </w:r>
  <w:r>
    <w:t> 2</w:t>
  </w:r>
  <w:r>
    <w:t> 2</w:t>
  </w:r>
  <w:r>
    <w:t>cd</w:t>
  </w:r>
</w:p>

```

The first paragraph contains characters with Unicode value U+FF66 (㐦). The second paragraph contains characters with Unicode value U+FF12 (2). If autoSpaceDE is true, spacing is added in the first paragraph (between the ideographs and the non-ideographic characters), but not in the second (all four characters are not ideographs):

ab 㐦㐦 cd
 ab 2 2 cd

If this compatibility setting is turned on:

```

  <w:compat>
    <w:autoSpaceLikeWord95 />
  </w:compat>

```

Then, although it appears incorrect, applications should not add space in the first paragraph and should apply it in the second:

ab㐦㐦cd
 ab 2 2 cd

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.5 [cachedColBalance \(Use Cached Paragraph Information for Column Balancing\)](#)

This element specifies whether applications shall incorrectly calculate the height of a paragraph for the purposes of column balancing when rendering WordprocessingML documents. Specifically, this element specifies that when a paragraph's lines have differing heights, an application shall treat this paragraph as though it had only one line equaling the full paragraph height, regardless of the actual number of lines in the paragraph.

[*Guidance*: It is recommended that applications not intentionally replicate this behavior; it is maintained only for compatibility with existing documents from a legacy application. *end guidance*]

Typically, lines are correctly measured for their height when balancing columns as part of a WordprocessingML document. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall perform the incorrect calculation in the conditions described above.

[*Example*: Consider a WordprocessingML document with two columns of text which shall be balanced.

If this compatibility setting is turned on:

```
<w:compat>
  <w:cachedColBalance />
</w:compat>
```

Then applications should perform the calculation described above to balance the columns, as needed. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.6 `convMailMergeEsc` (Treat Backslash Quotation Delimiter as Two Quotation Marks)

This element specifies whether applications should perform a conversion of the contents of a mail merge data source when reading those contents in order to perform a mail merge operation with their contents.

Typically, the contents of a mail merge data source are read in exactly as specified when performing a mail merge with the contents of a data source. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall interpret delimiters composed of a backslash and quotation mark (`\"`) as two quotation marks (`""`), within external data sources to be connected to via a mail merge.

[*Example*: Consider a WordprocessingML document with the following content in its data source:

```
This is a \"test\".
```

The default presentation would have the resulting merged data read in just as it appears:

```
This is a \"test\".
```

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:convMailMergeEsc />
</w:compat>
```

Then instances of a backslash and quotation mark would be converted, resulting in the following output:

```
This is a ""test"".
```

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.7 `displayHangulFixedWidth` (Always Use Fixed Width for Hangul Characters)

This element specifies whether applications should assume that all characters in the Hangul Syllables Unicode sub range (character values between 0xAC00 and 0xD7FF) are of a single fixed width or shall use the characters widths defined by the font in use (typical for a proportional width font).

Typically, applications shall retrieve the character width for any character in a document from the associated font, allowing each character to be of its own width (a proportional width character). This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall instead assume a single fixed width for all characters in the Hangul Syllables sub range, by reading the width of Unicode character 0x4E00 from the associated font and using that width for all Hangul characters (or, if that character is not present, the next available character in the font).

[Example: Consider a WordprocessingML document with three Hangul characters:



The default presentation would have each of those characters using the widths defined by the font (the highlighting indicates that each character has its own width):



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:displayHangulFixedWidth />
</w:compat>
```

Then all three characters are forced to the fixed width of character 0x4E00 from the font (or, in this case, the next available character), resulting in the characters in the font being forced to that fixed width, which results in the following output:



Notice from the highlighting that the characters have been compressed to the width of the single character and displayed at that fixed width. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.8 doNotAutofitConstrainedTables (Do Not AutoFit Tables To Fit Next To Wrapped Objects)


This element specifies whether applications shall allow tables to be resized to the remaining available line width when they are using the AutoFit algorithm and part of that line is filled by a shape with a wrapping type with a value of square or tight.

Typically, a table which is AutoFit and has a preferred width shall have its width reduced in order to allow a floating shape to wrap around its contents within the document, as that shape simply reduces the width of the line and the AutoFit algorithm applies to the remaining line width. This element, when present with a val attribute value of true (or equivalent), specifies that tables shall never have any preferred width overridden to allow them to wrap around that floating object, and shall instead be pushed to the next full width line in the document to be displayed.

[*Example:* Consider a WordprocessingML document with a floating shape centered in the document, followed by a table with preferred cell widths of 2.22", as follows:

This is some text.

This is some text.



This is some text.

The diagram illustrates the default presentation of a document. It shows three lines of text: "This is some text.", "This is some text.", and "This is some text.". Between the second and third lines of text, there is a table with three columns and three rows. To the right of the table, there is a rectangular floating shape. The table is positioned such that it fits on the line next to the floating shape, demonstrating the effect of the doNotAutofitConstrainedTables property.

The default presentation of this document overrides the preferred cell widths to force the table to fit on the line next to the floating shape with tight wrapping.


However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotAutofitConstrainedTables />
</w:compat>
```

Then that table is not resized, so it cannot fit and must be pushed to the next full width line, resulting in the following output:

This is some text.

This is some text.



This is some text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.9 `doNotBreakConstrainedForcedTable` (Don't Break Table Rows Around Floating Tables)

This element specifies whether applications shall allow a table row to be split in two when its contents are displayed under the following circumstances:

- The table row exceeds one page in height (it shall be split into two pages)
- The table row would need to be split in order to accommodate a floating table also on the page (tables which have been set to floating using the `tblpPr` element (Part 1, §17.4.58))

Typically, assuming the `cantSplit` property (Part 1, §17.4.6) is not set, a table row which cannot fit on one single page shall be split as needed around any floating table on a page, in order to allow its contents to be fully displayed across two or more pages. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that table rows which exceed one page in height shall never be split around floating tables in the document, and shall instead be displayed on the first page below the floating table, even if that means that part of the table row is clipped by the edge of the page.

[*Example:* Consider a WordprocessingML document with a long single table row which must be split across two separate pages in the document, in order to accommodate a floating table anchored in the footer, as follows:

[illegible]

The default presentation of this document forces that row to be split as needed around that floating table.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotBreakConstrainedForcedTable />
</w:compat>
```

Then that table row is never split around the floating table, so it is always placed below that floating table on the page, and allowed to flow off the page as needed, resulting in the following output:

[1]	[2]
This is a single table cell in the footer	This is a single table cell in the footer

This example, while extreme, shows how the row is placed below the floating table, rather than breaking around it. *end example*]

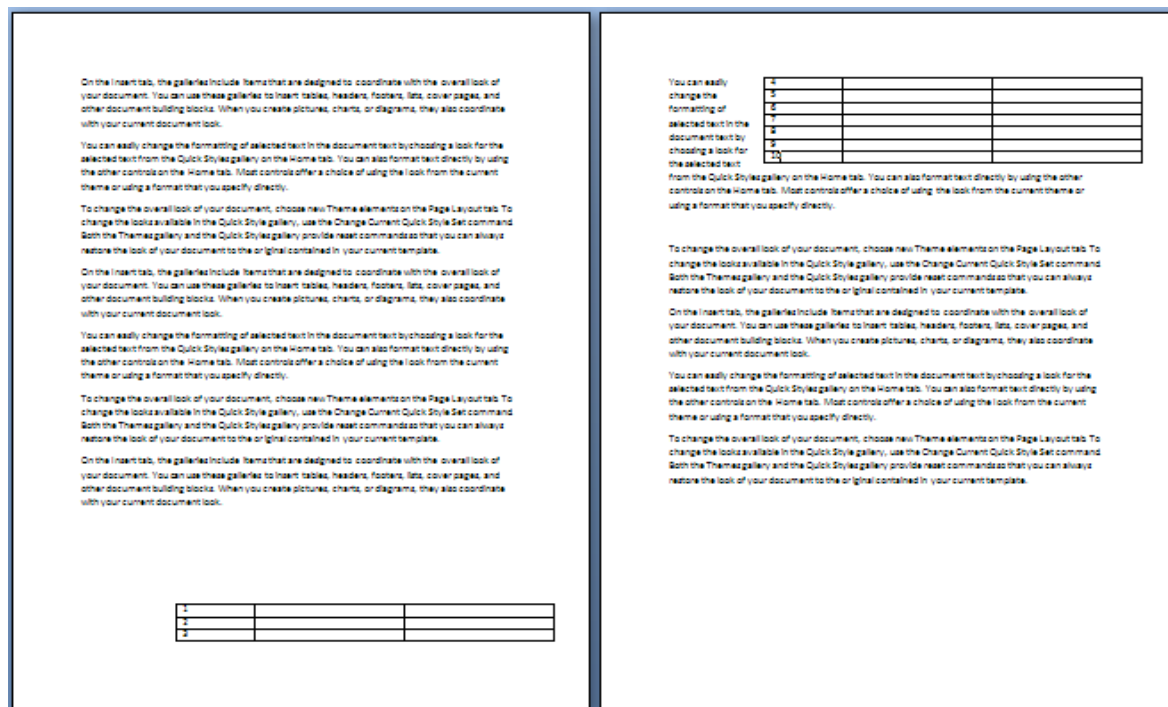
This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.10 doNotBreakWrappedTables (Do Not Allow Floating Tables To Break Across Pages)

This element specifies whether applications shall allow tables which have been set to floating using the `tblPr` element (Part 1, §17.4.58) shall be allowed to break across multiple pages when needed.

Typically, a table whose contents cannot all be displayed on one page is broken as needed across multiple pages in order to preserve the location of the table (just as a paragraph of multiple lines is broken across pages as needed). This element, when present with a `val` attribute value of `true` (or equivalent), specifies that floating tables shall never be broken across pages, and shall instead be put on the first page by adjusting the starting position of the table as needed to fit on that single page.

[Example: Consider a WordprocessingML document with a floating table positioned at the bottom of a page , as follows:



The default presentation of this document results in that table being broken across two pages of content.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotBreakWrappedTables />
</w:compat>
```

Then that table is not broken across the page boundary, so it must be moved further up on the first page to accommodate its entire size, resulting in the following output:

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the look available in the Quick Styles gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the look available in the Quick Styles gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

To change the overall look of your document, choose

1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

new Theme elements on the Page Layout tab. To change the look available in the Quick Styles gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.

To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the look available in the Quick Styles gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.

Notice that the table now flows into the page margins in order to keep it on one page. *end example*]

This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.11 doNotSnapToGridInCell (Do Not Snap to Document Grid in Table Cells with Objects)

This element specifies whether a document grid defined using the docGrid element (Part 1, §17.6.5) shall be applied to the contents of table cells in that section which also contain floating objects. Note that the floating object shall be part of the cell, and simply not displayed over the cell due to its anchoring relative to another part of the document.


Typically, if a floating object is present in a table cell, then that setting shall have no impact on whether East Asian text in that cell is snapped to the document grid (as text is always snapped to the grid). This element, when present with a val attribute value of true (or equivalent), specifies that whenever a floating object is present in a table cell, that the cell's contents shall not be snapped to the document grid.

[Example: Consider a WordprocessingML document consisting of a single section, whose document grid settings specify that each page must be exactly 10 characters wide, as follows:

```
<w:sectPr>
  <w:docGrid w:type="snapToChars" w:charSpace="146636" />
</w:sectPr>
```


If this document contains a table with a single cell, containing some text and a single floating shape, the contents of the cell are still snapped to the 10 characters per line character grid, as follows:

ト	リ	ス	タ	ン	ト	リ	ス	タ	ン
ト	リ	ス	タ	ン					ン
リ	ス	タ	ン	ト					リ
ス	タ	ン	ト	リ					ス
タ	ン	ト	リ	ス	タ	ン	ト	リ	ス
タ	ン								




However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotSnapToGridInCell />
</w:compat>
```

Then the presence of a floating object in each cell must result in the document grid setting being ignored, resulting in the following output:

ト	リ	ス	タ	ン	ト	リ	ス	タ	ン
ト	リ		ス	タ	ン				ン
リ	ス	ス	タ	ン	ト				リ
ス	タ	ン	ト	リ	ス				ス
タ	ン	ト	リ	ス	タ	ン	ト	リ	ス
タ	ン								



The additional character pitch was still added to each character on the line, but those characters are no longer snapped to the document grid. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.12 doNotSuppressIndentation (Do Not Ignore Floating Objects When Calculating Paragraph Indentation)

This element specifies whether applications should ignore the presence of floating objects when calculating the starting position of paragraphs which are wrapped around floating objects.

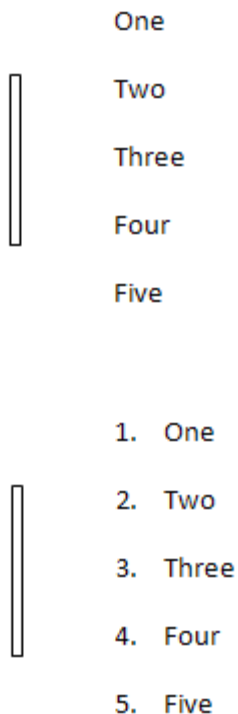
Typically, the presence of a floating object on the same line or lines as a paragraph shall only affect the text when the floating object occurs where that text would normally be presented. [*Example*: Text at a 1" indentation would only be displaced by a floating object that appears at that position and not one that appears from 0" to 0.5" on the same line. *end example*].

This element, when present with a `val` attribute value of `true` (or equivalent), specifies that floating objects shall always impact paragraphs on the same line in two ways:

- If the paragraph is not numbered, then it shall tightly wrap any floating object which precedes it on the same line, ignoring its own indentation settings. [*Example: A paragraph with a 1" left indent shall tightly wrap a floating object which appears at only 0.25" on the same line. end example*]
- If the paragraph is numbered using the numPr element (Part 1, §17.3.1.19), then it shall calculate and use its full indent relative to the edge of the floating object, not relative to the edge of the page. [*Example: A numbered paragraph with a 1" left indent must appear 1.5" into the page if it is preceded by a floating object which appears at 0.5" on the same line. end example*]

[*Example: Consider a WordprocessingML document with a narrow floating object at 0.5" on the page, surrounded by both numbered and unnumbered paragraphs.*

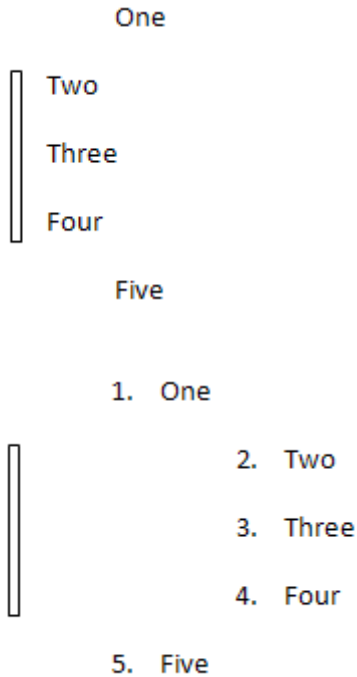
The default presentation would have no impact on the paragraphs based on that floating object, since the two do not intersect:



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotSuppressIndentation />
</w:compat>
```

Then the two alternate rules defined above would apply, resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

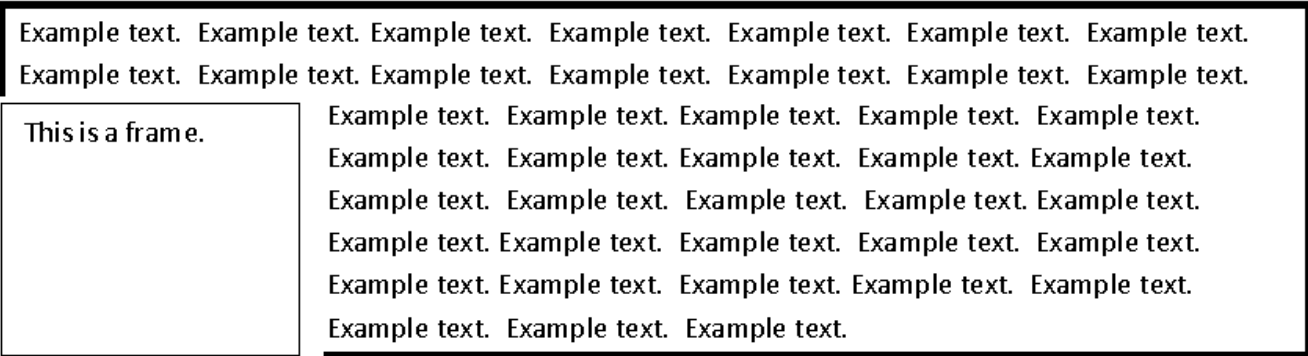
14.7.3.13 `doNotSuppressParagraphBorders` (Do Not Suppress Paragraph Borders Next To Frames)

This element specifies whether applications should suppress paragraph borders defined using the `pBdr` element (Part 1, §17.3.1.24) when those borders would be displayed next to the contents of paragraphs which have been defined as frames using the `framePr` element (Part 1, §17.3.1.11).

Typically, when a paragraph's borders appear next to a frame, those borders are suppressed to avoid having two borders in close proximity. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that those borders shall not be suppressed.

[*Example:* Consider a WordprocessingML document with a paragraph with a paragraph border that is bounded on its bottom left side by a text frame.

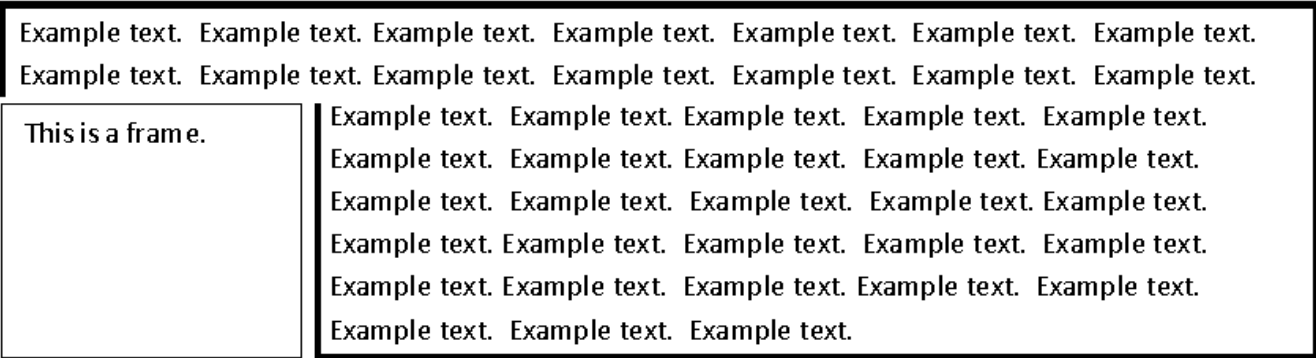
The default presentation would suppress the borders which intersect the frame (in this case, the right border of lines three through eight):



However, if this compatibility setting is turned on:

```
<w:compat>  
  <w:doNotSuppressParagraphBorders />  
</w:compat>
```

Then no border suppression must take place, resulting in the following output:



end example]

This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.14 [doNotUseEastAsianBreakRules \(Do Not Compress Compressible Characters When Using Document Grid\)](#)

This element specifies whether applications should compress characters with identical compression rules when the document grid has been defined using the docGrid element (Part 1, §17.6.5). *Compression rules* refer to the additional bearing on the left and/or right side of a typical character, which can be compressed as needed without modifying the actual width of the character (its breadth).

Typically, punctuation characters with an identical set of compression rules are compressed when the contents of a document are displayed. This element, when present with a val attribute value of true (or equivalent), specifies that if a document grid is defined for the current section, compression shall never be performed on any character - all compressible characters shall be individually snapped to the document grid.

[*Example:* Consider a WordprocessingML document with a document grid set to allow 10 characters per line:

```
<w:sectPr>
  <w:docGrid w:type="snapToChars" w:charSpace="146636" ... />
</w:sectPr>
```

The default presentation would allow characters with identical compression rules to compress and utilize a single slot on the document grid (notice that the four parenthesis on the first line are combined since they can be compressed identically, while the two parenthesis with different compression on line two are not):

```
あ あ あ あ ) ) ) ) v あ あ あ あ
あ あ あ あ a あ あ ) (
```

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotUseEastAsianBreakRules />
</w:compat>
```

Then no character with compression is compressed and instead are snapped to the grid individually, resulting in the following output:

```
あ あ あ あ ) ) ) ) v あ
あ あ あ あ あ あ あ a あ あ )
(
```

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.15 `doNotUseHTMLParagraphAutoSpacing` (Use Fixed Paragraph Spacing for HTML Auto Setting)

This element specifies whether applications should use a fixed definition when interpreting automatic paragraph spacing defined by a value of true (or equivalent) on the `beforeAutospacing` and/or `afterAutospacing` attributes on the spacing element (Part 1, §17.3.1.33).

Typically, applications shall interpret these settings to match the behavior of most HTML user agents, mimicking the default spacing above and below an HTML `p` element without additional spacing information. This element, when present with a `val` attribute value of true (or equivalent), specifies that those two attributes shall result in the following settings for each value:

- beforeAutospacing = 5 points of spacing before
- afterAutospacing = 10 points of spacing after

[Example: Consider a WordprocessingML document with a three paragraphs using HTML autospacing, as follows:

```
<w:p>
  <w:pPr>
    <w:spacing w:beforeAutospacing="true" w:afterAutospacing="true" />
  </w:pPr>
  <w:r>
    <w:t>Paragraph One</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:spacing w:beforeAutospacing="true" w:afterAutospacing="true" />
  </w:pPr>
  <w:r>
    <w:t>Paragraph Two</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:spacing w:beforeAutospacing="true" w:afterAutospacing="true" />
  </w:pPr>
  <w:r>
    <w:t>Paragraph Three</w:t>
  </w:r>
</w:p>
```

The default presentation would result in output designed to match that of all common HTML user agents:

Paragraph One.

Paragraph Two.

Paragraph Three.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotUseHTMLParagraphAutoSpacing />
</w:compat>
```

Then the paragraphs has exact spacing of 5 points before and 10 points after, resulting in the following output:

Paragraph One.

Paragraph Two.

Paragraph Three.

Notice that the paragraphs are more condensed in the second example. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.16 `doNotUseIndentAsNumberingTabStop` (Ignore Hanging Indent When Creating Tab Stop After Numbering)

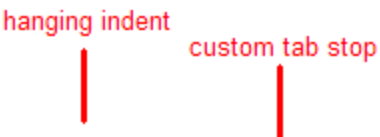
This element specifies whether applications shall use the custom tab stop generated by the hanging indent (if any) when advancing the text after the numbering for a numbered paragraph.

Typically, a hanging indent on a paragraph creates a virtual custom tab stop at that location, and therefore a tab added after the numbering on a numbered paragraph by the `suff` element (Part 1, §17.9.29) shall advance to that tab stop, so that the text of the numbered paragraph begins at that location. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that a tab stop added as the suffix to the numbering of a numbered paragraph shall ignore that virtual custom tab stop and shall instead advance to the next real tab stop (custom or automatic) on the current line.

[*Example:* Consider a WordprocessingML document with numbering, whose first level of numbering specifies a tab stop suffix, a hanging indent at 1", and a custom tab stop at 2":

```
<w:abstractNum w:abstractNumId="0">
  ...
  <w:lvl w:ilvl="0">
    <w:suff w:val="tab" />
    <w:pPr>
      <w:ind w:left="1440" w:hanging="1440" />
      <w:tabs>
        <w:tab w:val="2880" />
      </w:tabs>
    </w:pPr>
  </w:lvl>
</w:abstractNum>
```

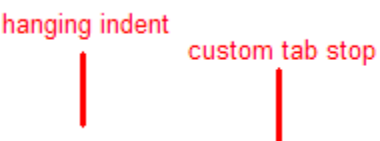
The default presentation of this document results in the tab stop generated by the numbering advancing to the virtual tab stop generated by the hanging indent at 1", as follows:

1.  This is numbered text. There is a hanging indent at 1" and a custom tab stop at 2"

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotUseIndentAsNumberingTabStop />
</w:compat>
```

Then that tab suffix ignores the virtual tab stop of the hanging indent, so it must advance to the next custom tab stop on the line (at 2"), resulting in the following output:

1.  This is numbered text. There is a hanging indent at 1" and a custom tab stop at 2".

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

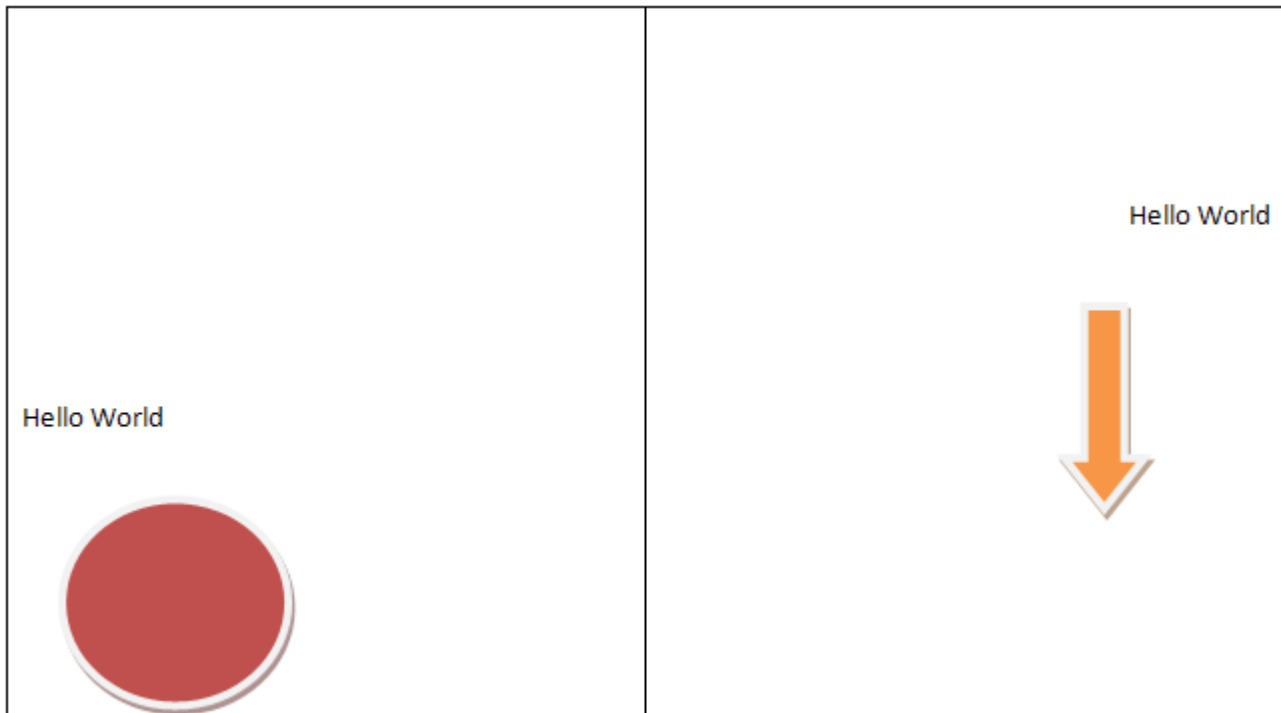
14.7.3.17 [doNotVertAlignCellWithSp \(Don't Vertically Align Cells Containing Floating Objects\)](#)

This element specifies whether applications shall vertically align the contents of a table cell, even when the contents of that table cell include one or more floating objects. Note that the floating object shall be part of the cell, and simply not displayed over the cell due to its anchoring relative to another part of the document.

Typically, if the alignment of a table cell in a WordprocessingML document is specified, then the entire contents of that cell are aligned as specified [*Example*: The entire contents of the cell are centered vertically and moved right-aligned horizontally at that point. *end example*]. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that whenever a floating object is present in a table cell, that no vertical alignment shall be applied to the contents of that cell, and the contents of the cell shall instead always be top aligned to the cell's contents.

[*Example*: Consider a WordprocessingML table with two cells, each containing some text and a single floating shape. The first cell is vertically aligned to the bottom of the cell, and the second cell is vertically aligned to the center of the cell.

The default presentation of this document results in each cell (including the extents of the floating objects) being vertically aligned as specified, as follows:



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotVertAlignCellWithSp />
</w:compat>
```

Then the presence of a floating object in each cell must result in the vertical alignment setting being ignored (each vertical alignment must be top-aligned relative to the cell), resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.18 `doNotVertAlignInTxbx` (Ignore Vertical Alignment in Textboxes)

This element specifies whether applications shall allow text within text boxes to be vertically aligned when the `v-text-anchor` property is set within the parent VML shape.

Typically, if when the `v-text-anchor` property is set within the parent VML shape, then based on the value of that property, the text is top, center, or bottom aligned appropriately. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the property shall be ignored, and instead the contents of the table shall always be top-aligned.

[Example: Consider a WordprocessingML table with a single center-aligned text box:

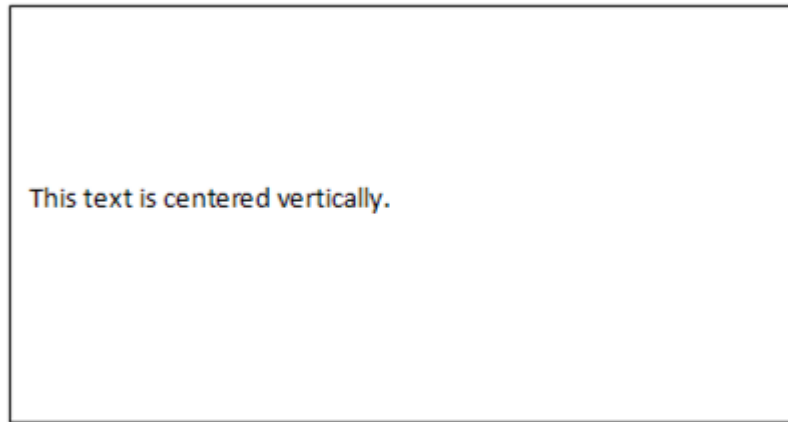
```
<v:shape id="_x0000_s1026" type="#_x0000_t202" style="v-text-anchor:middle">
  <v:textbox>
    <w:txbxContent>
      <w:p>
        <w:r>
          <w:t>This text is centered vertically.</w:t>
        </w:r>
      </w:p>
    </w:txbxContent>
  </v:textbox>
</v:shape>
```

```

    </w:txbxContent>
  </v:textbox>
</v:shape>

```

The default presentation of this document results in the contents of the text box being center aligned, as follows:



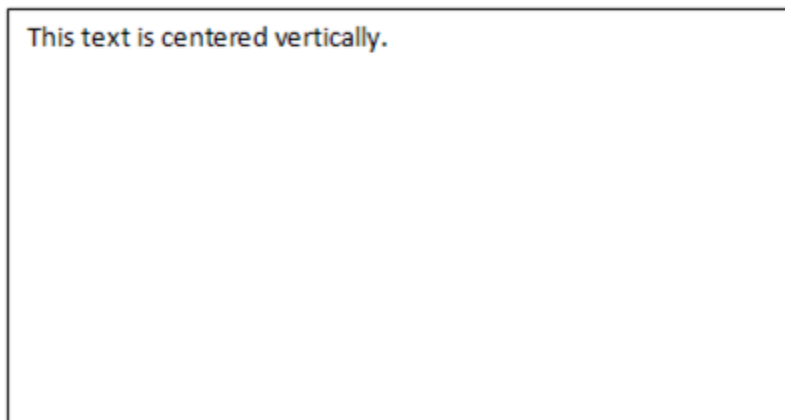
However, if this compatibility setting is turned on:

```

<w:compat>
  <w:doNotVertAlignInTxbx />
</w:compat>

```

Then the text must always be top aligned, regardless of the -text-anchor property, resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.19 `doNotWrapTextWithPunct` (Do Not Allow Hanging Punctuation With Character Grid)

This element specifies whether applications shall allowing hanging punctuation when:

- The `overflowPunct` element (Part 1, §17.3.1.21) is turned on for a paragraph
- A document grid is defined using the `docGrid` element (Part 1, §17.6.5) which defines the number of characters per line

Typically, paragraphs which allow hanging punctuation shall allow the number of characters on a line as specified by the document grid to be exceeded by one in order to allow for hanging punctuation. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the document grid shall never be exceeded for hanging punctuation.

[*Example:* Consider a WordprocessingML document with a document grid set to allow 10 characters per line:

```
<w:sectPr>
  <w:docGrid w:type="snapToChars" w:charSpace="146636" ... />
</w:sectPr>
```

If the eleventh character on the line was a punctuation characters, the default presentation would allow that character to behave as hanging punctuation on the first line:

“ 言 葉 が 言 葉 が 言 葉 が ”
 言 葉 が 言 葉

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:doNotwrapTextWithPunct />
</w:compat>
```

Then the character grid cannot be exceeded even for the hanging punctuation, resulting in the following output:

“ 言 葉 が 言 葉 が 言 葉
 が ” 言 葉 が 言 葉

The hanging punctuation was disallowed, moving it (and the character before it, since that character cannot begin a line) to the following line. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.20 footnoteLayoutLikeWW8 (Ignore Page Break from Continuous Section Break)

This element specifies that applications should override the default behaviour for a continuous section break when one or more footnotes are present on the page with the footnote. This override typically results in text being displayed on the same page as a continuous section break (after the break, which would normally move all following text to the next page).

Typically, applications render a continuous section break as a page break when one or more footnoteRef elements (Part 1, §17.11.13) occur on that page before the break, as described in Part 1, §17.18.77. This element, when present with a val attribute value of true (or equivalent), specifies that applications should allow any paragraph after the section break that contains no footnoteRef elements (Part 1, §17.11.13) to be displayed on the same page. If the resulting content reaches the page extents, the section's page break is ignored.

[*Example:* Consider a WordprocessingML document with two footnotes contained in two sections, separated by a continuous section break:

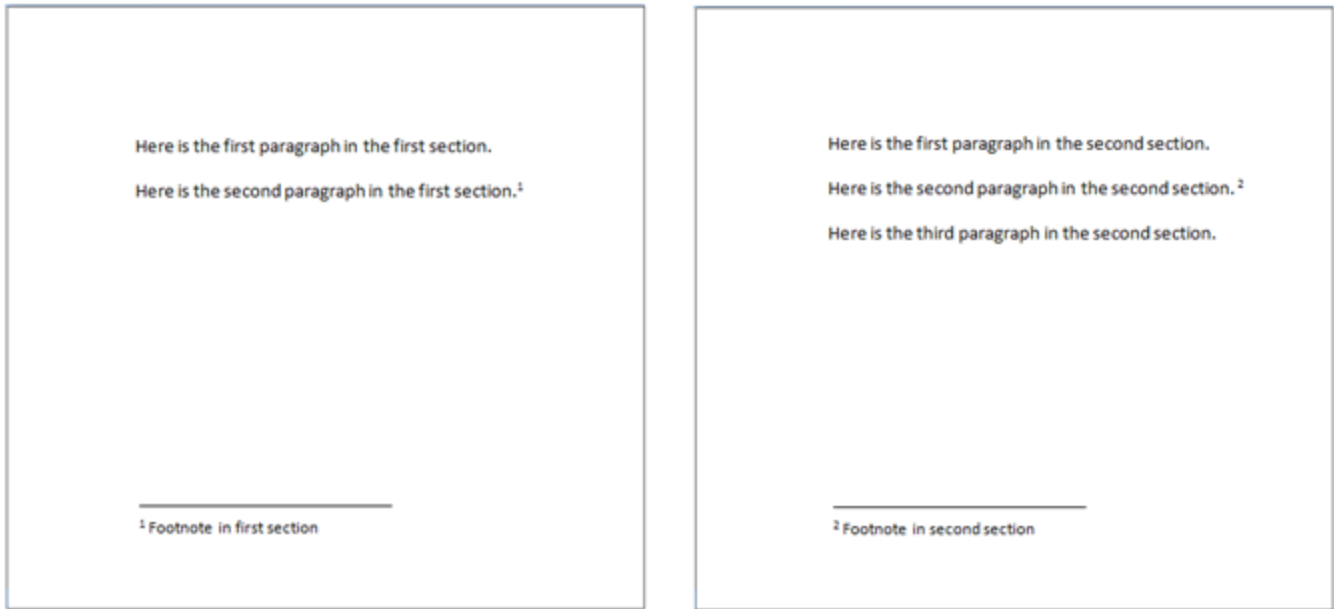
```
<w:p>
  <w:r>
    <w:t xml:space="preserve">Here is the first paragraph in the first
section.</w:t>
  </w:r>
</w:p>
<w:p>
  <w:r>
    <w:t>Here is the second paragraph in the first section.</w:t>
  </w:r>
  <w:r>
    <w:rPr>
      <w:rStyle w:val="FootnoteReference" />
    </w:rPr>
    <w:footnoteReference w:id="2" />
  </w:r>
</w:p>
<w:p/>
<w:p>
  <w:pPr>
    <w:sectPr>
      ...
    </w:sectPr>
  </w:pPr>
```

```

    </w:p>
<w:p>
  <w:r>
    <w:t>Here is the first paragraph in the second section.</w:t>
  </w:r>
</w:p>
<w:p>
  <w:r>
    <w:t xml:space="preserve">Here is the second paragraph in the second
section.</w:t>
  </w:r>
  <w:r>
    <w:rPr>
      <w:rStyle w:val="FootnoteReference" />
    </w:rPr>
    <w:footnoteReference w:id="3" />
  </w:r>
</w:p>
<w:p>
  <w:r>
    <w:t xml:space="preserve">Here is the third paragraph in the second section.
</w:t>
  </w:r>
</w:p>
<w:sectPr>
  <w:type w:val="continuous" />
  ...
</w:sectPr>

```

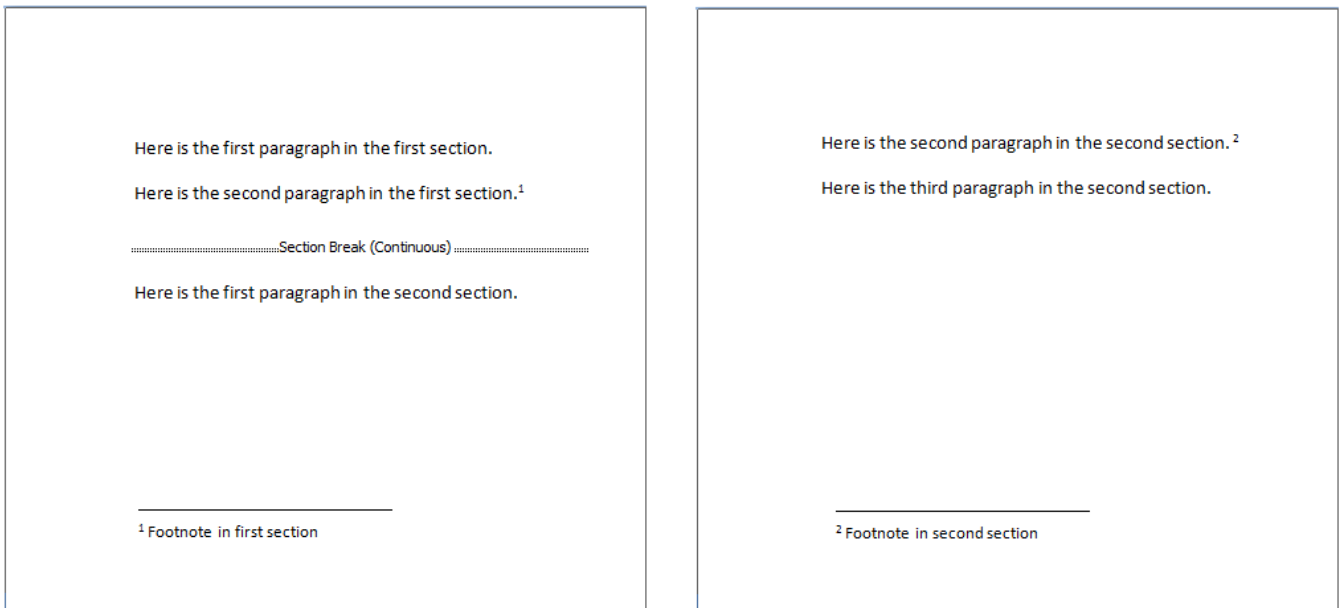
The default rendering of such a document results in the continuous section break as a page break, resulting in the following two page document:



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:footnoteLayoutLikeWW8 />
</w:compat>
```

Then the first paragraph following the section break (not having any footnote references) is displayed on the same page, despite the section break, resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.21 `forgetLastTabAlignment` (Ignore Width of Last Tab Stop When Aligning Paragraph If It Is Not Left Aligned)

This element specifies how applications should handle the final tab stop on a line when aligning the contents of a paragraph as specified by the `jc` element (Part 1, §17.3.1.13) in the paragraph's properties.

Typically, aligning the contents of a paragraph involves the following:

- Determining the layout of that line before the alignment (including all tab stops)
- Aligning the resulting contents of the line

This is done to ensure that tab stops on a line do not change when the contents of the paragraph are aligned (i.e. the tab stops should not have to take into account the paragraph alignment).

This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall ignore the additional line width generated by the last tab stop (and only the last tab stop) when the alignment of the tab stop as defined by the `val` attribute on the `tab` element (Part 1, §17.3.1.37) is not `left` (or `bar`, which as defined by ECMA-376, is not a tab stop per se) when determining the width of the line. The resulting full line shall then be aligned at the position where the line would have been aligned without that tab stop.

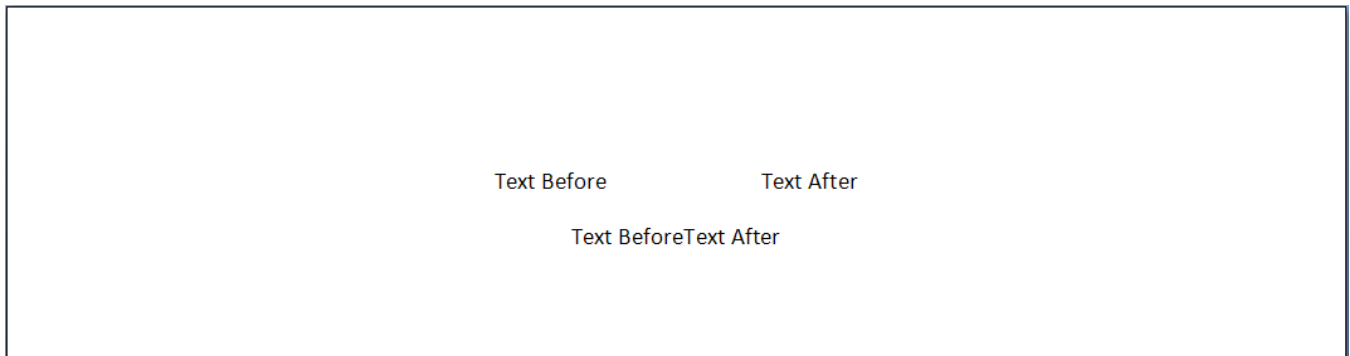
[*Example:* Consider a WordprocessingML document with two center aligned paragraphs of text - the first also containing a centered tab stop positioned at 2":


```

<w:p>
  <w:pPr>
    <w:tabs>
      <w:tab w:val="center" w:pos="2880" />
    </w:tabs>
    <w:jc w:val="center" />
  </w:pPr>
  <w:r>
    <w:t>Text Before</w:t>
    <w:tab/>
    <w:t>Text After</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:jc w:val="center" />
  </w:pPr>
  <w:r>
    <w:t>Text BeforeText After</w:t>
  </w:r>
</w:p>

```

The default presentation would determine the full width of each line including the tab stops, finally aligning the resulting text to the center position as requested by the jc element:



However, if this compatibility setting is turned on:

```

<w:compat>
  <w:forgetLastTabAlignment />
</w:compat>

```

Then the width added to the line by the last tab is ignored when centering the paragraph because that tab is a center aligned tab stop, resulting in the following output:

Text Before Text After

Text BeforeText After

In the resulting output, the starting location of both lines is at the same place on the page, as the resulting width of both lines is identical when the tab stop is removed from the line width calculation. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.22 growAutofit (Allow Tables to AutoFit Into Page Margins)

This element specifies whether applications shall allow a table which is using the AutoFit table layout algorithm to extend beyond the margins of the page if the minimum width of each table cell would result in an overall table width which is wider than those page margins.

Typically, if a table is using the AutoFit layout algorithm, then based on the definition of that logic, each column in the table shall be increased to the minimum width of its contents (e.g. the longest non-breaking run of text contained within it and/or the width of an inline image contained in one of its cells) until the overall width of the table reaches that of the text extents on the page, at which point text shall be broken and images shall be clipped as needed to maintain the width of the table at the page width (i.e. the page width is an immutable maximum width for the table). This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the minimum width of the cells shall not be constrained by the page width, and instead the table shall be allowed to extend into the page margins as needed in order to meet the minimum widths of each of its cells.

[*Example:* Consider a WordprocessingML table with three cells in each row. If the contents of each cell in that first row each contain a long non-breaking string (such that the minimum widths of each cell's contents exceed the page width), then the rules for table AutoFit specify that each cell must be broken proportionally when the overall width of the table reaches the page width.

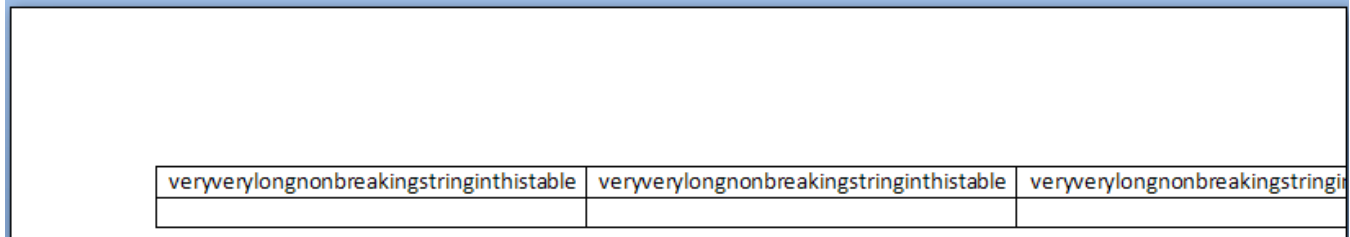
The default presentation of this document results in each cell being broken as needed to maintain the table width, as follows:

veryverylongnonbreakingstringin thistable	veryverylongnonbreakingstringin thistable	veryverylongnonbreakingstringin thistable

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:growAutofit />
</w:compat>
```

Then the presence of those long non-breaking strings (and the resulting large minimum widths for each table cell) must result in a table width which is then allowed to override the page margins, resulting in the following output:



veryverylongnonbreakingstringinthistable	veryverylongnonbreakingstringinthistable	veryverylongnonbreakingstringinthistable

The resulting table is clipped by the edge of the page on its right side, but the minimum widths of each cell are maintained as defined by the long non-breaking string contents of each. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.23 `layoutRawTableWidth` (Ignore Space Before Table When Deciding If Table Should Wrap Floating Object)

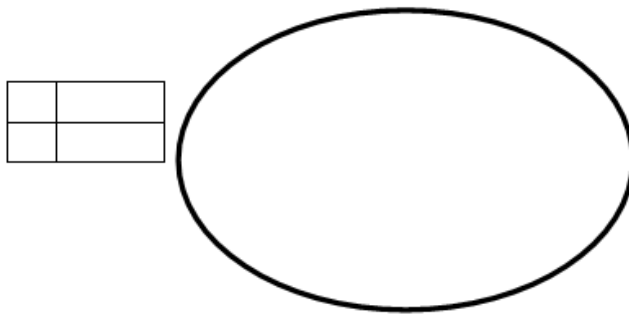
This element specifies how tables which have been indented from the margin using the `tblInd` element (Part 1, §17.4.51) shall be wrapped around floating objects.

Typically, when a table is positioned next to a floating object, the table shall only remain next to the object if it can fit in the remaining space on the line when considering the full width needed for the table: the space before the table, plus the width of the table. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the calculation determining whether the table shall fit next to the object shall not include the space before the table, even if that means that the table is actually clipped by the object.

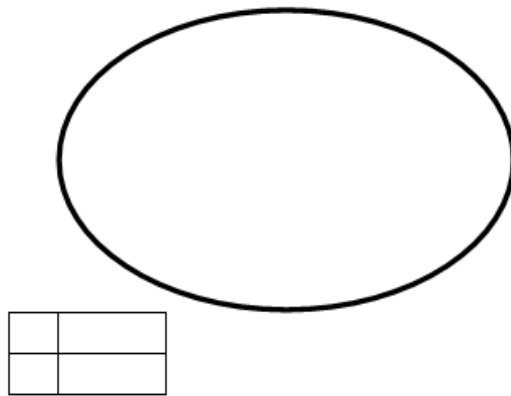
[*Example*: Consider a WordprocessingML document with a floating shape using square wrapping, next to a table which has been indented one inch from the left margin:

```
<w:tbl>
  <w:tblPr>
    <w:tblInd w:w="1440" w:type="dxa" />
  </w:tblPr>
  ...
</w:tbl>
```

The resulting presentation would place the table next to the object:



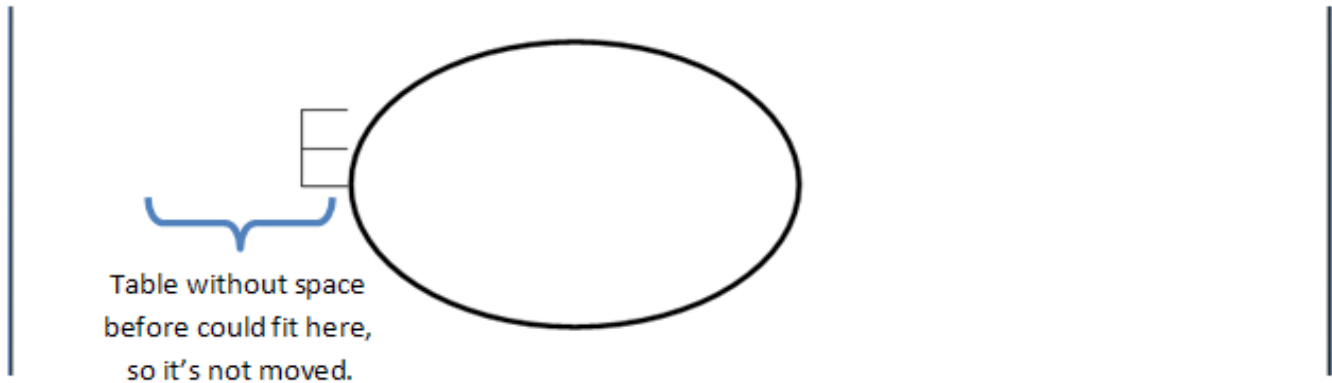
If this object is then moved to the left, such that it would clip the table, the default presentation would have the entire table moved below the shape, since it does not fit in the remaining space on the line:



However, if this compatibility setting is turned on:

```
<w:compat>  
  <w:layoutRawTableWidth />  
</w:compat>
```

Then the determination to move the table is done ignoring the spaced needed before the table, resulting in the following output:



The resulting table is clipped behind the object, as the fit calculation ignores the space needed before the table.
end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.24 `layoutTableRowsApart` (Allow Table Rows to Wrap Inline Objects Independently)

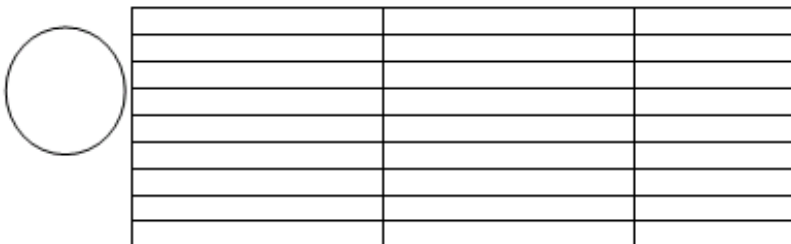
This element specifies whether tables which are wrapping around floating objects shall wrap around the object as a whole, or if each table row shall individually wrap the object as needed (causing a more stuttered, yet tighter, wrapping of the object).

Typically, when a table wraps around a floating object, the table shall wrap the object as a unit (i.e. the whole table square wraps the object). This element, when present with a `val` attribute value of `true` (or equivalent), specifies that wrapping is applied to each row in the table one by one, even if its means that each row has a different resulting position with respect to the table.

[*Example:* Consider a WordprocessingML document with a floating shape using square wrapping.

The default presentation would have the entire table wrapping around that shape:

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.

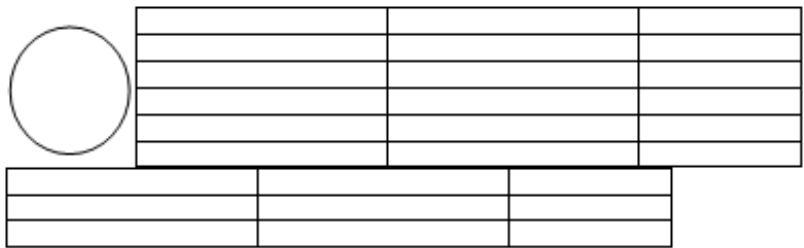


However, if this compatibility setting is turned on:

```
<w:compat>
  <w:layoutTableRowsApart />
</w:compat>
```

Then each row would wrap around the shape one by one, resulting in the following output:

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.25 [lineWrapLikeWord6 \(Ignore Compression of Full-Width Punctuation Ending a Line\)](#)

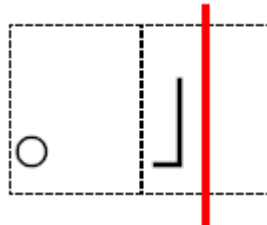
This element specifies that applications should ignore the character compression settings specified by the `characterSpacingControl` element (Part 1, §17.15.1.18) when determining if one more character fits within the text margins on each line of the document. This setting typically results in a character being pushed to the following line, ignoring the fact that the character compression settings would have allowed it to fit within the text boundaries.

Typically, an application would check the character compression settings, and apply any character-level whitespace compression before attempting to fit the last character on the line. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall ignore that compression and fit the character as if it should be displayed at its full width, regardless of whether the compression settings are applied.

[*Example:* Consider a paragraph which ends with the following two characters (with each character's bounding box outlined for illustrative purposes:



If the document's character compression settings were not set to `doNotCompress` and text extent fell at the location identified by this red line:



The last character would have compression applied to its blank half, and would fit on the line.

If this compatibility setting is turned on:

```
<w:compat>
  <w:lineWrapLikeWord6 />
</w:compat>
```

Then applications should compress the character, but should treat the character as full width when determining if it fits on the line; in this case, the second character would be displayed on the following line. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.26 mwSmallCaps (Use Specific Small Caps Algorithm)

This element specifies that applications should use a specific algorithm to determine the font size of small caps (the formatting resulting from the use of the `smallCaps` element (Part 1, §17.3.2.33). This emulation typically results in small caps which are smaller than typical small caps at most font sizes.

Typically, applications can utilize any algorithm that results in small caps formatting. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications should determine the font size for small caps using the following algorithm:

- If $font \leq 7$, then the font size for small caps is 7 points.
- Otherwise, sequentially iterate through *sequence* until $sequence[i] \leq font \leq sequence[i + 1]$, at which point the font size for small caps is $sequence[i]$ points.

where

- *sequence* is an array defined as follows:
 $\{7, 9, 10, 12, 14, 18, 24, 36, 48, 60, 72, 80, x_1, x_2, \dots, x_n\}$ where $x_n = 80 + 10 * n$.
- *font* is an integer calculated as follows:
 The font size of the run to which small caps formatting is applied (in points).

[*Example*: Consider a WordprocessingML document with small caps on its text contents.

If this compatibility setting is turned on:

```
<w:compat>
  <w:mwSmallCaps />
</w:compat>
```

And the font size for a single run is 16 points, and performing the algorithm above would result in 14 points as the calculated font size for small caps. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.27 noColumnBalance (Do Not Balance Text Columns within a Section)

[*Note*: Typically, a continuous section break (Part 1, §17.18.77) balances the content of the previous section, unless the "noColumnBalance" compatibility option is given. *end note*]

This element specifies whether the contents of sections with multiple columns defined using the cols element (Part 1, §17.6.4) should automatically be balanced. In terms of column layout, *balancing* is the act of attempting to ensure that the number of lines in each column is equivalent (rather than completely filling one column before populating the next).

Typically, column balancing is automatically performed on the contents of sections with multiple columns. This element, when present with a val attribute value of true (or equivalent), specifies that column balancing shall not occur, and each column shall be filled individually until the end of the current page, until all text has been displayed, even if this means one or more columns are unused.

[*Example*: Consider a WordprocessingML document with an initial section with three columns, defined by the following section properties:

```
<w:sectPr>
  <w:cols w:num="3" w:space="720" />
</w:sectPr>
```

The default presentation would have the text in that section balanced between those three columns:

This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in

a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.

This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three

columns. This is some text in
a section of three columns.

This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns. This is some
text in a section of three
columns. This is some text in
a section of three columns.
This is some text in a section
of three columns. This is
some text in a section of
three columns.

This is text in a section with one column. This is text in a section with one column. This is text in a section with one column. This is text in a section with one column. This is text in a section with one column. This is text in a section with one column.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:noColumnBalance />
</w:compat>
```

Then the columns are not balanced, and the contents of the section are used to fill each column to the bottom of the current page in succession, resulting in the following output:

[illegible]

The next section is now forced to begin on the next page, as the columns on page one extend to the bottom of that page. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.28 noExtraLineSpacing (Do Not Center Content on Lines With Exact Line Height)

This element specifies whether an exact line height using the spacing element (Part 1, §17.3.1.33) in the paragraph's properties, each line shall not be automatically centered within the given amount of line spacing.

Typically, if the exact amount of spacing allotted to a line via the paragraph properties exceeds the amount of space required by that line, then the line of text shall be automatically centered when the text of the document is displayed. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that all additional spacing shall instead be placed below the normal layout of the line of text.

[Example: Consider a WordprocessingML document with a line with an exact height of 32 points:

```

<w:p>
  <w:pPr>
    <w:spacing w:line="640" w:lineRule="exact" />
  </w:pPr>
  <w:r>
    <w:t>This is text on a line that's exactly 32 points high.</w:t>
  </w:r>
</w:p>

```

The default presentation would have the resulting text centered on that line:

This is text

This is text on a line that's exactly 32 points high.

This is text.

However, if this compatibility setting is turned on:

```

<w:compat>
  <w:noExtraLineSpacing />
</w:compat>

```

Then all line spacing is added after the text, resulting in the following output:

This is text

This is text on a line that's exactly 32 points high.

This is text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.29 noLeading (Do Not Add Leading Between Lines of Text)

This element specifies whether the additional leading specified by the current font face shall be added between each line of text when that text is displayed. *Leading* refers to the additional spacing requested by a particular font in order to ensure that letters on subsequent lines do not display in a fashion where they are positioned too closely together.

Typically, leading should be added as specified by the associated font. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the additional leading specified by the font shall never be output when the text is displayed.

[*Example:* Consider a WordprocessingML document with three lines of text. The default presentation would have the text displayed as follows:

EXAMPLE TEXT

Some text.
Some text.
Some text.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:noLeading />
</w:compat>
```

Then no leading is added between lines, resulting in the following output:

EXAMPLE TEXT

Some text.
Some text.
Some text.

This adjustment is usually very minute in nature; therefore the result is better illustrated by showing how the characters were pushed out due to the leading added to that text:

EXAMPLE TEXT

Some text.
Some text.
Some text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.30 `noSpaceRaiseLower` (Do Not Increase Line Height for Raised/Lowered Text)

This element specifies whether the height which is allotted to any given line of text when the contents of this document are displayed shall include additional spacing in order to ensure that all raised and/or lowered text can be fully displayed.

Typically, any extra space needed is added to the line to prevent raised and lowered text from being truncated or hidden. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the height of the line shall be determined solely by the spacing settings on the parent paragraph, and any raised/lowered text shall just be clipped if it exceeds that space.

[*Example:* Consider a WordprocessingML document with both raised and lowered text. The default presentation would have that text visible:

This is text.

This is text – a lowered word, a raised word.

This is text.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:noSpaceRaiseLower />
</w:compat>
```

Then no additional space should be added to the line height, resulting in the following output:

This is text.

This is text – a lowered word, a raised word.

This is text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.31 `noTabHangInd` (Do Not Create Custom Tab Stop for Hanging Indent)

This element specifies whether applications should always create a hanging indent as a custom tab stop when handling tabs within the contents of a WordprocessingML paragraph. The `dontUseIndentAsNumberingTabStop` element (§14.7.3.16) specifies if this tab stop shall be used in the case of a tab added as the suffix to numbering in a numbered paragraph, while this element handles the same

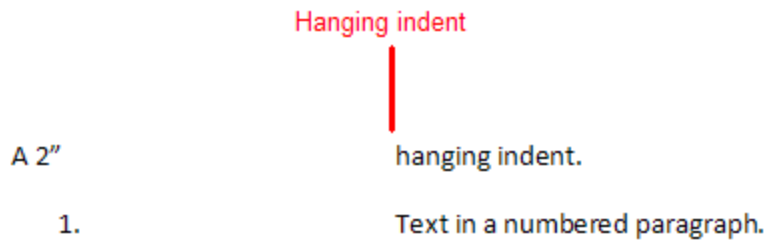
functionality in the generic case (i.e. this element, when set, renders that setting irrelevant as the tab stop is never used).

Typically, the hanging indent on a paragraph shall be treated as a custom tab stop location within that paragraph, allowing the first tab on the first line in the paragraph to advance to the location of the hanging indent. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that no custom tab stop shall be created for a hanging indent on a line under any circumstances.

[*Example:* Consider a WordprocessingML document with two paragraphs (the second numbered, the first not), each with a 2" hanging indent defined as follows (assume the numbering suffix - not shown - is a tab character):

```
<w:p>
  <w:pPr>
    <w:ind w:left="2880" w:hanging="2880" />
  </w:pPr>
  <w:r>
    <w:t>A 2"</w:t>
    <w:tab/>
    <w:t>hanging indent</w:t>
  </w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:numPr>
      <w:ilvl w:val="0" />
      <w:numId w:val="1" />
    </w:numPr>
    <w:ind w:left="2880" w:hanging="2880" />
  </w:pPr>
  <w:r>
    <w:t>Text in a numbered paragraph.</w:t>
  </w:r>
</w:p>
```

The default presentation would have both the numbering and the tab in the regular paragraph advancing to the 2" custom tab stop generated by the hanging indent:



A 2"

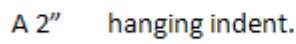
1. Text in a numbered paragraph.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:noTabHangInd />
</w:compat>
```

Then no tab stop exists at 2", and therefore the tab stops must advance to the location of the next automatic tab stop for this document (which is set to occur every 0.5"), resulting in the following output:

Hanging indent



1. Text in a numbered paragraph.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.32 `printBodyTextBeforeHeader` (Print Body Text before Header/Footer Contents)

This element specifies the order in which the contents of the main document story and any headers and/or footers shall be sent to the printer.

Typically, the contents of a document are sent to the printer as follows:

- First, the contents of headers/footers are sent to the printer
- Finally, the contents of the main document story are sent to the printer

This element, when present with a `val` attribute value of `true` (or equivalent), specifies that this order shall be reversed, and that the body text shall be sent to the printer before any header/footer text. This reversal allows for the processing of PostScript codes in the text layer in the same order as afforded by some legacy word processing applications.

[*Example:* Consider a WordprocessingML document which is printed. The default resulting print order is the headers and footers for each page, followed by the page contents.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:printBodyTextBeforeHeader />
</w:compat>
```

Then this order must be reversed, and the page contents must be printed before the corresponding header and/or footer for each page. *end example]*

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.33 `printColBlack` (Print Colors as Black And White without Dithering)

This element specifies the way in which colored text and/or objects shall be handled when printed to a printer whose printer settings indicate that it can only handle black and white text.

Typically, the contents of a colored document are sent to a black and white printer using grayscale (different shades of gray) to represent each of the possible colors. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that colors are not printed as mapped shades of grey, but rather exclusively in solid black and white. This setting prevents the fuzzy look that can occur when gray or blue content is dithered. *Dithering* is the process by which colors are simulated using various patterns of black dots on a white background

[*Example:* Consider a WordprocessingML document which is printed to a black and white printer. The default resulting printed content is typically dithered to appear in the appropriate shade of grayscale text.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:printColBlack />
</w:compat>
```

Then the page contents must be printed as exclusively black or exclusively white text as needed, and no grayscale output must occur. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.34 `selectFldWithFirstOrLastChar` (Select Field When First or Last Character Is Selected)

This element specifies whether applications should automatically select the entire contents of a field in a WordprocessingML document when the first or last character is selected.

Typically, users can select any character individually within the result of a field in the document. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that selecting the first or last character of that field result shall automatically result in the selection of the entire field.

[*Example:* Consider a WordprocessingML document which contains the following (with a field marked in gray shading):

Author Tristan Davis would like to welcome you.

The default presentation would allow the first character of that field to be selected:

Author **Tristan Davis** would like to welcome you.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:selectFldWithFirstOrLastChar />
</w:compat>
```

Then that selection would automatically result in the entire field being selected, resulting in the following:

Author **Tristan Davis** would like to welcome you.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.35 `shapeLayoutLikeWW8` (Ignore Text Wrapping around Objects at Bottom of Page)

This element specifies that applications should ignore the line wrapping setting specified by a floating object, instead allowing text to be displayed beneath it under the specific set of conditions identified below.

Typically, text wrapping around a floating object is dictated by the presence of one of the following as a child element of the object's anchor element (Part 1, §20.4.2.3):

- `wrapNone` (Part 1, §20.4.2.15) element, which specifies no text wrapping
- `wrapSquare` (Part 1, §20.4.2.17) element, which specifies square text wrapping
- `wrapThrough` (Part 1, §20.4.2.18) element, which specifies through text wrapping
- `wrapTight` (Part 1, §20.4.2.19) element, which specifies tight text wrapping
- `wrapTopAndBottom` (Part 1, §20.4.2.19) element, which specifies top and bottom text wrapping

This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall allow text to wrap beneath a floating object, ignoring the object's true wrapping setting, when the following conditions are met:

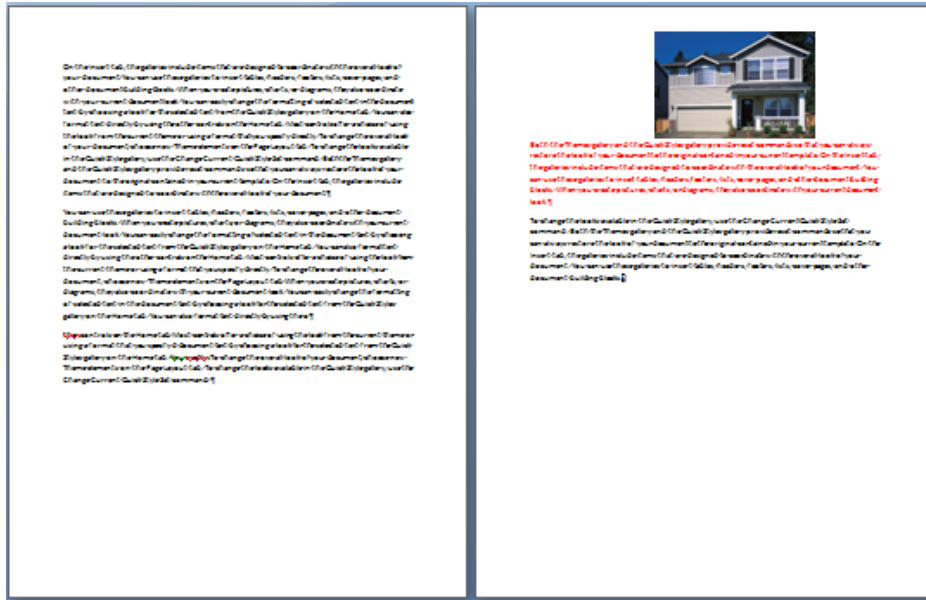
- The floating object has any of the following elements present as a child of the object's anchor element: `wrapSquare`, `wrapTight`, or `wrapTopAndBottom`.
- The floating object has a `positionV` element (Part 1, §20.4.2.11) with a `relativeFrom` attribute value of `line`.
- The floating object has a negative value for the child `posOffset` element (Part 1, §20.4.2.12) of the `positionV` element.
- The paragraph containing the anchor element would appear directly after the previous paragraph if the wrapping settings were ignored.

- The paragraph containing the anchor element would be pushed to the next page if the wrapping settings were respected.

[*Example:* Consider a WordprocessingML document containing a DrawingML object which meets the conditions outlined above:

```
<w:p>
  <w:r>
    <w:t>Sample text. Sample text. Sample text. Sample text. Sample text. Sample
text.</w:t>
  </w:r>
  <w:r>
    <w:drawing>
      <wp:anchor ... >
        <wp:positionV relativeFrom="line">
          <wp:posOffset>-428914</wp:posOffset>
        </wp:positionV>
        <wp:wrapTopAndBottom />
        ...
      </wp:anchor>
    </w:drawing>
  </w:r>
  <w:r>
    <w:t> Sample text. Sample text. Sample text. Sample text. Sample text.
Sample text.</w:t>
  </w:r>
  ...
</w:p>
```

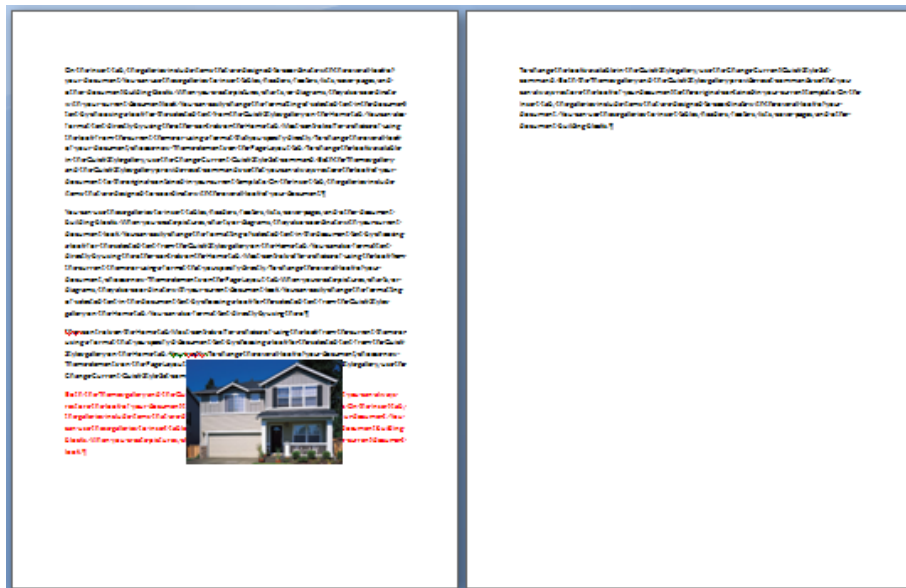
When the wrapping settings are respected, the shape and its paragraph do not fit on the page, so they are moved to the next page (the paragraph containing the anchor has been highlighted for illustrative purposes):



If this compatibility setting is turned on:

```
<w:compat>
  <w:shapeLayoutLikeWW8 />
</w:compat>
```

Then applications should ignore the wrapping setting and allow text to wrap below the object. This behaviour results in the following (again, the paragraph containing the anchor has been highlighted for illustrative purposes):



end example]

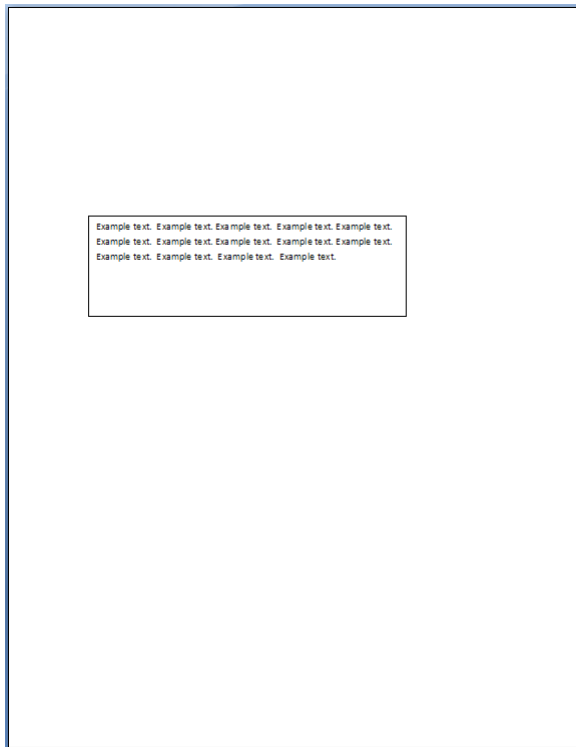
This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.36 showBreaksInFrames (Display Page/Column Breaks Present in Frames)

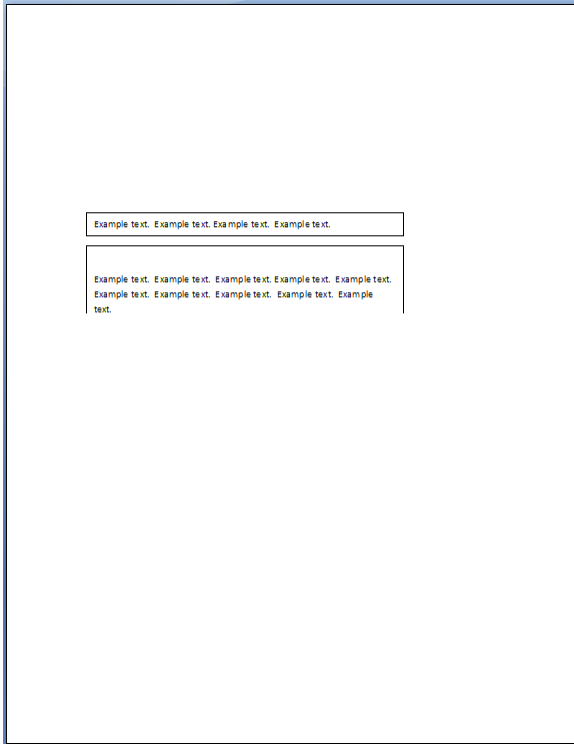
This element specifies whether applications should honor the presence of page and/or column breaks which are present within the contents of paragraphs which have been defined as frames using the framePr element (Part 1, §17.3.1.11).

Typically, breaks within frames shall be ignored and shall have no effect on the display of the paragraph in which they are contained. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that rather than completely ignoring these breaks, applications should display the break and move the remaining frame content, and all subsequent text, to the next page and/or column, as needed.

[Example: Consider a WordprocessingML document with a paragraph contained within a text frame:



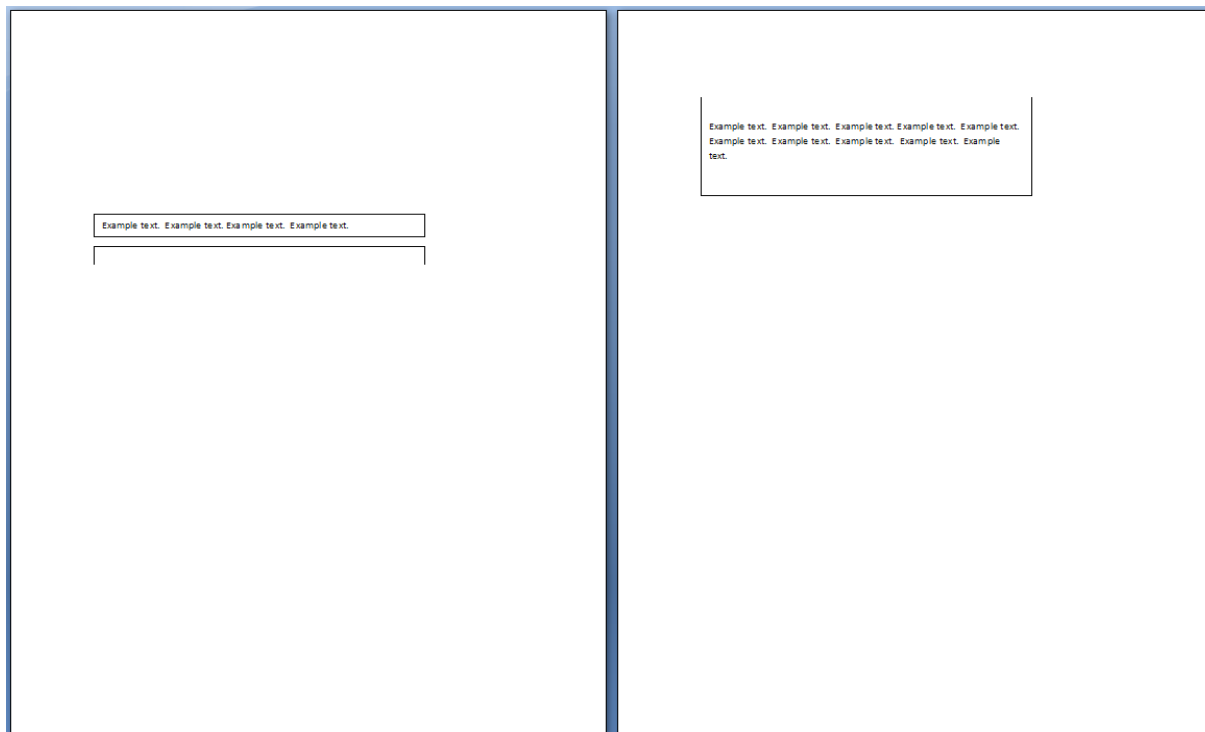
The default presentation would display the page break inline in the frame (breaking the frame into two) but would not actually break the page:



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:showBreaksInFrames />
</w:compat>
```

Then the page breaks is used even though they are present in the frame, breaking the page and resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.37 `spacingInWholePoints` (Only Expand/Condense Text By Whole Points)

This element specifies how applications should apply text expansion/compression defined using the `spacing` element (Part 1, §17.3.2.35) within a set of run properties.

Typically, as defined in the `spacing` element, text within runs in a WordprocessingML document can be expanded or compressed in increments of twentieths of a point. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the expansion and compression of text shall only be performed in increments of points. Any value which is not equal to an expansion or compression of a whole point shall be rounded down to the nearest whole point when the text is expanded/compressed within the WordprocessingML document.

[*Example:* Consider a WordprocessingML document with three paragraphs of text, each expanded by a varying amount, as follows:

```
<w:p>
...
<w:r>
  <w:t>This is text.</w:t>
</w:r>
</w:p>
```

```

<w:p>
...
<w:r>
  <w:rPr>
    <w:spacing w:val="20" />
  </w:rPr>
  <w:t>This is text.</w:t>
</w:r>
</w:p>
<w:p>
...
<w:r>
  <w:rPr>
    <w:spacing w:val="36" />
  </w:rPr>
  <w:t>This is text.</w:t>
</w:r>
</w:p>

```

The default presentation would have each run of text expanded exactly as requested:

Regular Text:	This is text.
Text expanded by 1 point:	This is text.
Text expanded by 1.8 points:	This is text.

However, if this compatibility setting is turned on:

```

<w:compat>
  <w:spacingInWholePoints />
</w:compat>

```

Then the third line - with an expansion of 1.8 points - would instead be rounded down to the nearest whole number of points when expanded, resulting in the following output:

Regular Text:	This is text.
Text expanded by 1 point:	This is text.
Text expanded by 1.8 points:	This is text.

In the resulting output, the second and third lines are identical, as the third line has a next expansion of exactly one point. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.38 `splitPgBreakAndParaMark` (Always Move Paragraph Mark to Page after a Page Break)

This element specifies whether a page break shall automatically complete the line on which it appears, moving the end of the paragraph to a new line on the next page, or if it shall behave as true run-level content within its current paragraph.

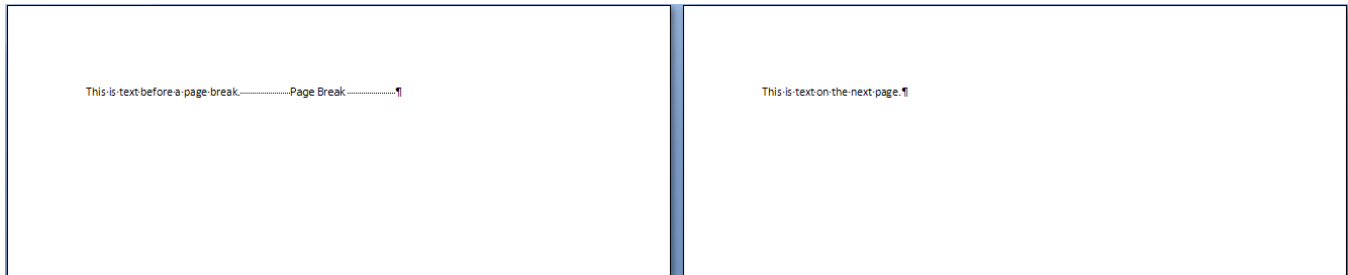
Typically, a page break defined using the `br` element (Part 1, §17.3.3.1) is treated as run-level content, which means that although it delimits the end of the page, if there is no content after it within the current paragraph, that the paragraph shall also end on that page. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that a page break shall always immediately end the current page, moving the paragraph mark which delimits the end of its parent paragraph to a new line on the next page.

Note that this setting only affects the case where there is no run-level content after the page break within the paragraph - if any further run content appears in the paragraph it shall appear on subsequent lines on the next page.

[*Example:* Consider a WordprocessingML document with two paragraphs of content - the first ending with a page break:

```
<w:p>
  <w:r>
    <w:t>This is text before a page break.</w:t>
    <w:br w:type="page" />
  </w:r>
</w:p>
<w:p>
  <w:r>
    <w:t>This is text on the next page.</w:t>
  </w:r>
</w:p>
```

The default presentation would have the text content `This is text on the next page.` as the first line of the second page, as there is no run content after the page break in paragraph one, and therefore no need for a new line on page two (in this image, a graphical illustration of the pilcrow and the page break have been added for clarity):



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:splitPgBreakAndParaMark />
</w:compat>
```

Then even though it is followed by no additional content, the page break must immediately end the first page, pushing the end of the first paragraph onto the first line of the second page, resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.39 `subFontBySize` (Require Exact Size During Font Substitution)

This element specifies whether applications shall accept fonts which cannot be rendered at the size specified by the `sz` (Part 1, §17.3.2.38) and/or `szCs` (Part 1, §17.3.2.39) elements on the parent run when performing font substitution.

Typically, applications can perform font substitution as defined in Part 1, §17.8.2, with no additional restrictions. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that when a potential substitute font has been located, an application shall check whether that font is capable of displaying characters at the specified point size. If it is not, that font is not considered as a substitute font (i.e. it is rejected, and the next closest match is considered).

[*Example:* Consider a WordprocessingML document with a series of characters in an unavailable font. The default presentation would use any method used by the application to perform that font substitution.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:subFontBySize />
</w:compat>
```

For each run, the application determines if the substitute font produced by its font substitution algorithm can be displayed at the size specified by the run's `sz` and/or `szCs` elements. If it cannot, that font is not used and the next closest match as substitute font is considered. *end example*

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.40 `suppressBottomSpacing` (Ignore Exact Line Height for Last Line on Page)

This element specifies whether an exact line height specified using the `spacing` element (Part 1, §17.3.1.33) with a `lineRule` attribute value of `exact` shall be ignored for the last line on each page.

Typically, if an exact line height has been specified using the `spacing` element, then all lines within that paragraph have the necessary line spacing added to them in order to meet this constraint. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that no additional spacing shall be added below the last line on each page as a result of these line spacing requirements - a line shall be placed on the bottom of the page if its characters fit on that page ignoring the necessary space after.

[*Example:* Consider a WordprocessingML document whose first paragraph has a line spacing setting requiring exactly 48 points of space per line:

```
<w:p>
  <w:pPr>
    <w:spacing w:line="960" w:lineRule="exact" />
  </w:pPr>
  ...
</w:p>
```

The default presentation would have the necessary amount of space added between each line such that all lines in the paragraph are centered within 48 points of spacing:

[illegible]

The first line from the following page was moved on the first page, as without being subjected to the line height constraint, it is possible to fit it at the bottom of the first page. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.41 suppressSpacingAtTopOfPage (Ignore Minimum Line Height for First Line on Page)

This element specifies whether the minimum line height specified using the spacing element (Part 1, §17.3.1.33) with a `lineRule` attribute value of `atLeast` shall be ignored for the first line on each page.

Typically, if a minimum line height has been specified using the spacing element, then all lines within that paragraph have the necessary line spacing added to them in order to meet this constraint. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that no additional spacing shall be added above the first line on each page as a result of this line spacing requirements - the top of the text characters on the first line shall be at the top edge of the page.

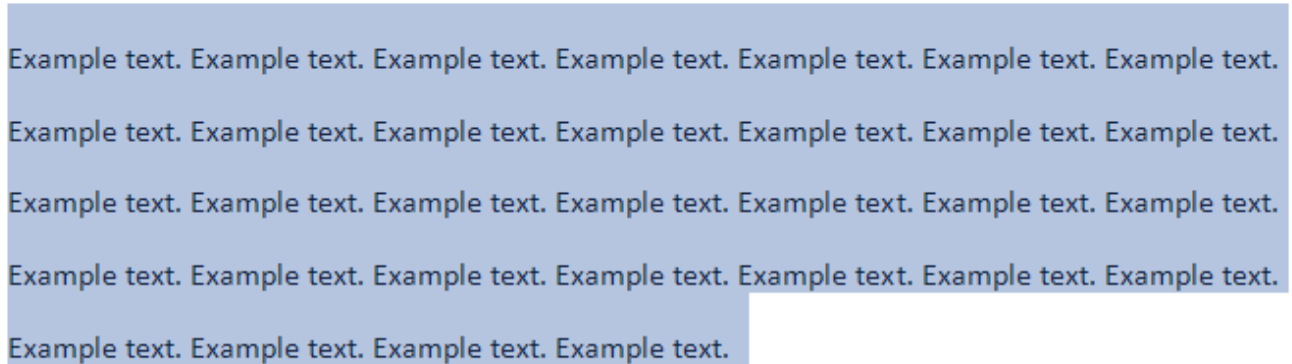
[*Example:* Consider a WordprocessingML document whose first paragraph has a line spacing setting requiring at least 25 points of space per line:

```

<w:p>
  <w:pPr>
    <w:spacing w:line="500" w:lineRule="atLeast" />
  </w:pPr>
  ...
</w:p>

```

The default presentation would have the necessary amount of space added between each line such that all lines in the paragraph are centered within 25 points of spacing (highlighting has been added to the image below in order to illustrate the additional spacing above the first line):



Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text.

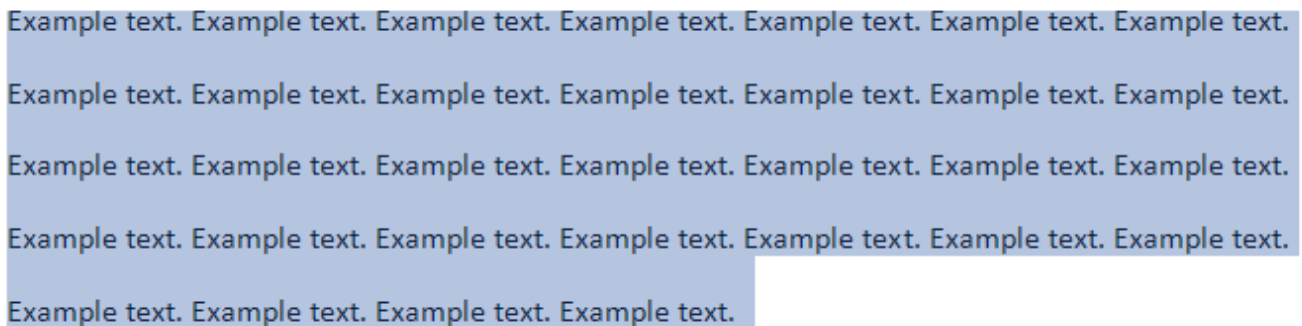
However, if this compatibility setting is turned on:

```

<w:compat>
  <w:suppressSpacingAtTopOfPage />
</w:compat>

```

Then no additional line spacing must be added above the first line on the page (although all other lines are unaffected), resulting in the following output:



Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text.

However, if this line spacing constraint was exactly 25 points, then this setting would have no effect:

Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.42 `suppressSpBfAfterPgBrk` (Do Not Use Space Before On First Line After a Page Break)

This element specifies that applications should not postpone any before paragraph spacing to the first line containing content after a page break.

Typically, a page break defined using the `br` element (Part 1, §17.3.3.1) is treated as run-level content, which means that although it delimits the end of the page, if there is no content after it within the current paragraph, that the paragraph shall also end on that page. However, in the case where there is additional run-level content within the same paragraph, that content, although part of the same paragraph as the page break, is displayed on the following page.

This leads to a situation where the only run content on the page with the page break is the break itself, with all subsequent content on the following page. In this case, applications shall apply the value specified by the spacing element's `before` attribute to the first line on the new page (since it is ostensibly the only page with content in that paragraph).

This element, when present with a `val` attribute value of `true` (or equivalent), specifies the paragraph before spacing shall not be 'postponed' in this way - if the line with the page break has no content, then the spacing element's `before` attribute is simply ignored.

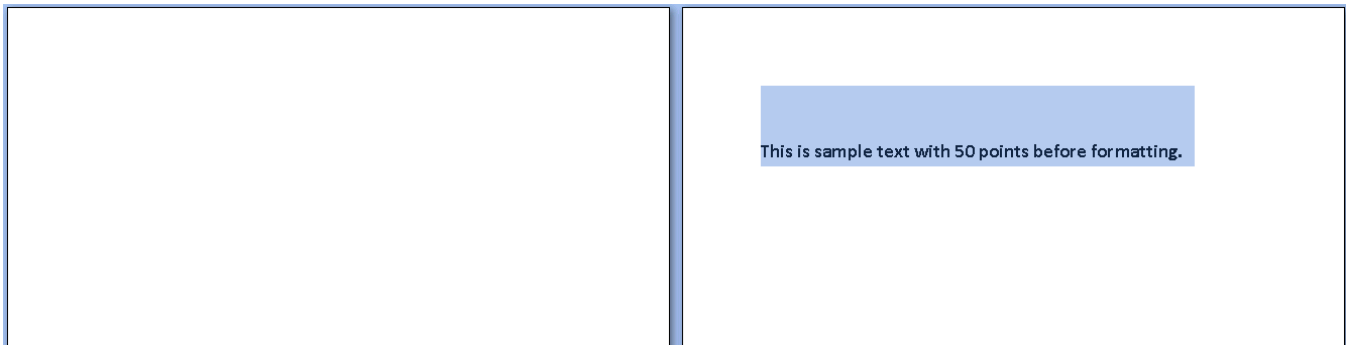
[*Example:* Consider a WordprocessingML document whose first paragraph specifies that it must be preceded by 50 points of additional spacing:

```

<w:p>
  <w:pPr>
    <w:spacing w:before="1000" />
  </w:pPr>
  <w:r>
    <w:br w:type="page" />
    <w:t>This is sample text with 50 points before formatting.</w:t>
  </w:r>
</w:p>

```

The default presentation would have the necessary amount of space added to the first line on the second page, as the page break was not preceded by any run content (highlighting has been added to the image below in order to illustrate the additional spacing above the first line):



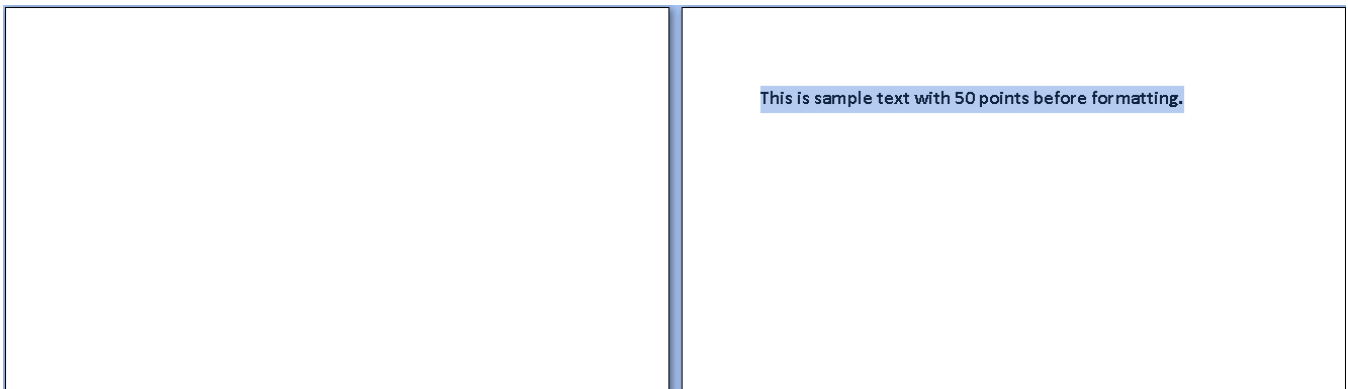
However, if this compatibility setting is turned on:

```

<w:compat>
  <w:suppressSpBfAfterPgBrk />
</w:compat>

```

Then the spacing must not be added above the first line on the page (it is essentially ignored), resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.43 `suppressTopSpacing` (Ignore Minimum and Exact Line Height for First Line on Page)

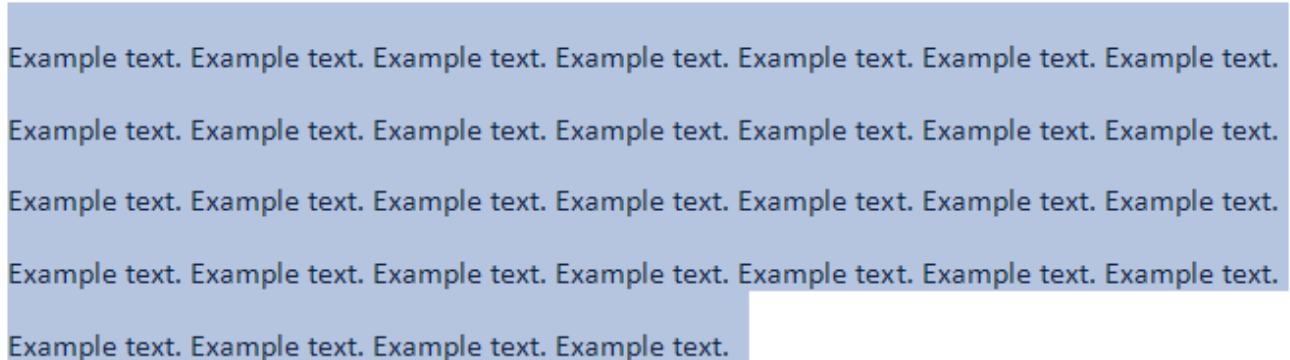
This element specifies whether the minimum line height specified using the spacing element (Part 1, §17.3.1.33) with a `lineRule` attribute value of `atLeast` or `exact` shall be ignored for the first line on each page.

Typically, if a minimum or exact line height has been specified using the spacing element, then all lines within that paragraph have the necessary line spacing added to them in order to meet this constraint. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that no additional spacing shall be added above the first line on each page as a result of these line spacing requirements - the top of the text characters on the first line shall be at the top edge of the page.

[*Example:* Consider a WordprocessingML document whose first paragraph has a line spacing setting requiring exactly 25 points of space per line:

```
<w:p>
  <w:pPr>
    <w:spacing w:line="500" w:lineRule="exact" />
  </w:pPr>
  ...
</w:p>
```

The default presentation would have the necessary amount of space added between each line such that all lines in the paragraph are centered within 25 points of spacing (highlighting has been added to the image below in order to illustrate the additional spacing above the first line):



Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:suppressTopSpacing />
</w:compat>
```


Then no additional line spacing must be added above the first line on the page (although all other lines are unaffected), resulting in the following output:

Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text. Example text. Example text. Example text.
 Example text. Example text. Example text. Example text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.44 `suppressTopSpacingWP` (Use Static Text Leading)

(The terms *baseline to baseline distance* and *unitsPerEm*, used below, are defined in ISO/IEC 14496-22.)

This element specifies that applications should use the values defined below to calculate the baseline to baseline distance (BTBD) in this document. This can result in lines appearing slightly condensed vertically.

Without this setting, applications calculate baseline to baseline distance using the metrics defined by ISO/IEC 14496-22. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications should calculate this as follows:

$$BTBD = \text{unitsPerEm} + 2\text{pt}$$

[*Example:* If this compatibility setting is turned on:

```
<w:compat>
  <w:suppressTopSpacingWP />
</w:compat>
```

Then applications use a baseline to baseline distance as calculated before. With a 16 point font, this would result in a baseline to baseline distance of 18 points. *end example]*

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.45 `swapBordersFacingPages` (Swap Paragraph Borders on Odd Numbered Pages)

This element specifies whether left and right paragraph borders defined under the `pBdr` element (Part 1, §17.3.1.24) shall be swapped under conditions where it is possible that the those pages are intended to be used to create a book-like publication.

Typically, no changes shall be made to the positions of paragraph borders defined under the pBdr element - a right border is always on the right, and a left border is always on the left. This element, when present with a val attribute value of true (or equivalent), specifies that under the two following conditions:

- The margins in this document are mirrored using the mirrorMargins element (Part 1, §17.15.1.57)
- The header/footers in this document are different on even and odd numbered pages using the evenAndOddHeaders element (Part 1, §17.10.1)

That paragraph borders on odd-numbered pages are swapped - that is, left borders shall be displayed on the right and right borders shall be displayed on the left.

[*Example:* Consider a WordprocessingML document for which the mirrorMargins element is present, and whose default paragraph style includes a paragraph border to be displayed on the right side of each paragraph:

```
<w:style w:type="paragraph" w:default="1" w:styleId="Normal" >
...
<w:pPr>
  <w:pBdr>
    <w:right w:val="single" w:color="auto" />
  </w:pBdr>
  ...
</w:pPr>
</w:style>
```

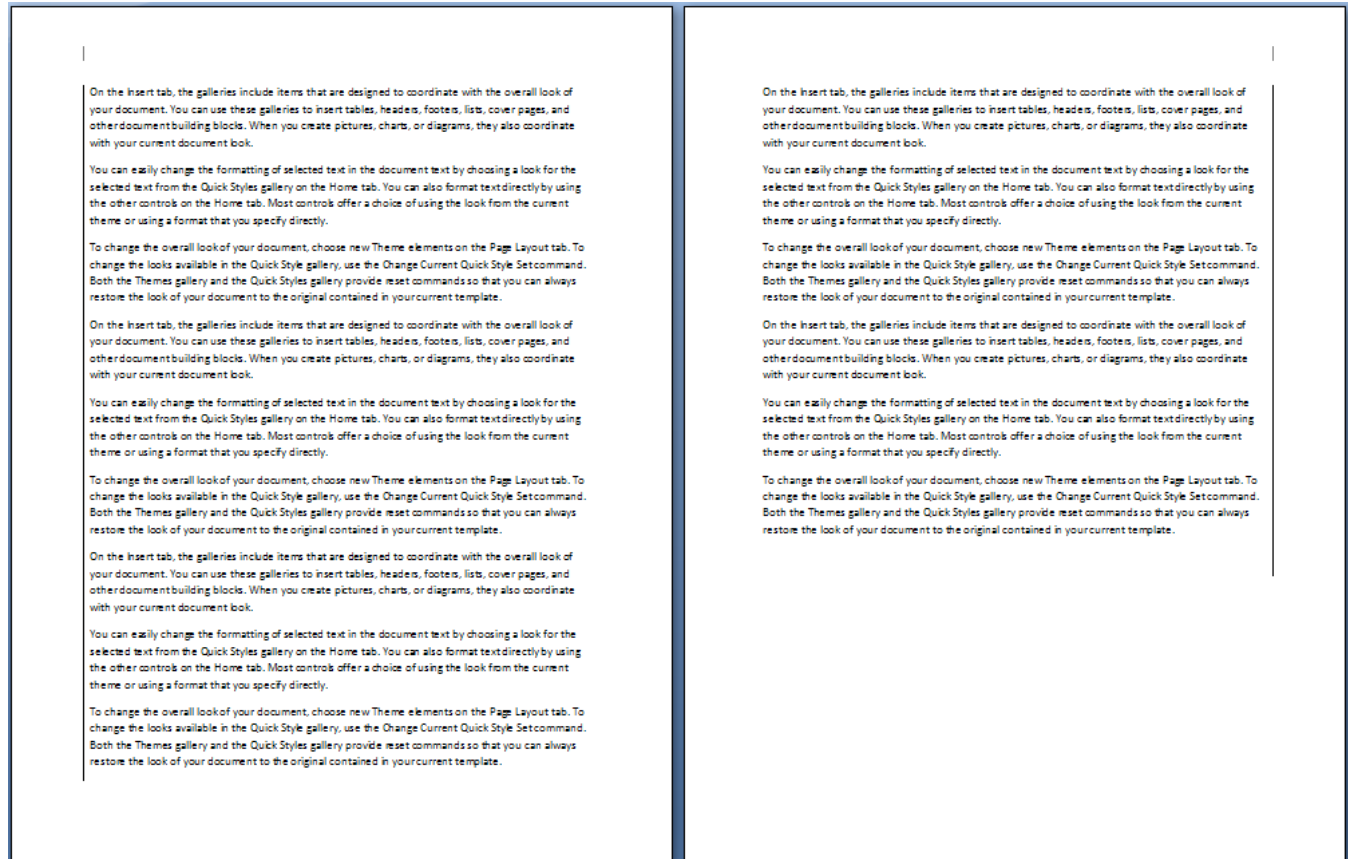
If a two-page document is created using this default paragraph style, then all paragraphs has a border on the right side, as follows:

<p>On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.</p> <p>You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.</p> <p>To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.</p> <p>On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.</p> <p>You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.</p> <p>To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.</p> <p>On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.</p> <p>You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.</p> <p>To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.</p>	<p>On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.</p> <p>You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.</p> <p>To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.</p> <p>On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert tables, headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your current document look.</p> <p>You can easily change the formatting of selected text in the document text by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or using a format that you specify directly.</p> <p>To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.</p>
---	---

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:swapBordersFacingPages />
</w:compat>
```

Then the borders on the first page (being an odd-numbered page) must be swapped, resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.46 `truncateFontHeightsLikeWP6` (Use Truncated Integer Division For Font Calculation)

This element specifies that applications should perform a specific method of calculation when converting font heights, specified in points using the `sz` (Part 1, §17.3.2.38) and `szCs` (Part 1, §17.3.2.39) elements, into pixels. This algorithm often results in a smaller than typical visual appearance of text for a given point size.

Typically, applications convert points to pixels using any approximate mathematical conversion mechanism (often, rounded integer division). This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications should use truncated integer division when performing this conversion (any non-integer value is truncated to determine the integer value resulting from the conversions).

[Example: If this compatibility setting is turned on:

```
<w:compat>
  <w:truncateFontHeightsLikeWP6 />
</w:compat>
```

Then applications shall use truncated integer division when calculating the height of characters.

For example, if the conversion is done as follows:

$$sz_{px} = sz_{pt} * N \frac{px}{inch} * \frac{1 inch}{72 pt}$$

where:

- sz_{pt} = size in points
- sz_{px} = size in pixels
- N = resolution in pixels per inch

Converting a 14 point font on a 96 dpi device results in $sz_{px} = 14 * 96 * \frac{1}{72} = 18\frac{2}{3}px$. If this setting is on, the result is truncated and the font is displayed using 18 pixels, even though 19 would be closer to the actual value. *end example]*

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.47 underlineTabInNumList (Underline Following Character Following Numbering)

This element specifies whether applications shall underline the character following the numbering defined using the `suff` element (Part 1, §17.9.29) when both the numbering itself and the first letter of the corresponding numbered paragraph is underlined.

Typically, the tab or space character generated between numbering and the corresponding paragraph of text is never formatted, since it is automatically generated by the `suff` element. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the tab or space shall be underlined the same way as the numbering symbol itself in the following conditions:

- The numbering is underlined
- The first character of the paragraph is underlined

[*Example:* Consider a WordprocessingML document with two numbered paragraphs: one with underlined text and the other without. The default presentation would have the tab characters free of underlining in both cases:

1. Example Text

2. Example Text

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:underlineTabInNumList />
</w:compat>
```

Then the second paragraph meets the criteria defined above for having the suffix character underlined, resulting in the following output:

1. Example Text

2. Example Text

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.48 `useAltKinsokuLineBreakRules` (Use Alternate Set of East Asian Line Breaking Rules)

This element specifies an alternate set of characters which can be used to determine which characters can begin and/or end a line when kinsoku line breaking rules are enabled using the kinsoku element (Part 1, §17.3.1.16).

Typically, the characters used to determine which characters shall not end a line are those listed by the kinsoku element in the paragraph properties subclause of this document. This element, when present with a val attribute value of true (or equivalent), specifies that the following settings shall be used instead (for brevity, only those settings which are different are listed below):

Chinese (Simplified)

- Cannot start a line:

!) , , ; ?] " ') , . : ; ?] ` | } ~ ¢

- Cannot end a line:

([{.““ < ≪ ∟ ∏ 【 ∫ ∫ (. [{ £ ¥

Chinese (Traditional)

- Cannot start a line:

!) , ; ?] } { — ” • / 、 。 > » 」 『 】) “ : || { ~~~~~ !) , . : ; ? | } 、

Korean

- Cannot end a line:

([{ £ ¥ “ ” ‹ › ≪ ≫ ∟ ∩ 【 】 ⌈ ⋄ \$ ([{ ≠

[*Example:* Consider a line of text in a WordprocessingML document within a paragraph marked as Chinese (Simplified) which begins with a % symbol, as follows:

%...

Typically, the kinsoku settings for Chinese (Simplified) do not allow this character to begin a line, so the character before that symbol would be moved down onto this line:

⌈%...

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:useAltKinsokuLineBreakRules />
</w:compat>
```

Then the alternate kinsoku rules are in place, which do not prevent the % character from beginning the new line, resulting in the following output:

%...

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.49 useAnsiKerningPairs (Use ANSI Kerning Pairs from Fonts)

This element specifies whether applications shall use the ANSI or Unicode kerning pair information from fonts stored in the document when displaying those characters within the document's contents.

Typically, applications shall use the Unicode kerning pair information in order to determine all possible kerning pairs in the fonts in use. This element, when present with a val attribute value of true (or equivalent), specifies that the ANSI kerning information shall be used instead.

[*Example:* Consider a WordprocessingML document with text that contains one or more kerning pairs.

If this compatibility setting is turned on:

```
<w:compat>
  <w:useAnsiKerningPairs />
</w:compat>
```

Then the ANSI kerning pairs are used in place of the Unicode kerning pairs, potentially resulting in different line breaks.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.50 useFELayout (Do Not Bypass East Asian/Complex Script Layout Code)

This element specifies that applications shall not bypass code relating to the layout of East Asian and/or Complex Script characters when presenting this document.

[*Guidance:* Previous word processing applications relied on this flag to determine whether to perform functions which allow for the correct layout of East Asian and Complex Script text. Although current applications no longer rely on this flag (as they should correctly use the Unicode subranges and code pages of the text in use), this flag

should be output in order to ensure that files with this content can be viewed correctly in previous word processors. *end guidance*]

[*Example:* Consider a WordprocessingML document with East Asian text.

If this compatibility setting is turned on:

```
<w:compat>
  <w:useFELayout />
</w:compat>
```

Then the flag is set telling previous applications that East Asian content is present, and they should display the document accordingly. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.51 `useNormalStyleForList` (Do Not Automatically Apply List Paragraph Style To Bulleted/Numbered Text)

This element specifies whether applications shall automatically apply the paragraph style with the `styleId` attribute `ListParagraph` when numbering is applied to a paragraph currently formatted using the default paragraph style.

Typically, when a paragraph is formatted using the default paragraph style, and numbering is subsequently applied, the paragraph style with the `styleId` attribute `ListParagraph` when numbering is applied to ensure that paragraph properties are appropriate for a numbered paragraph. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that no alternate paragraph style shall ever be applied

[*Example:* Consider a WordprocessingML document with five unnumbered paragraphs:

Example text.

Example text.

Example text.

Example text.

Example text.

If numbering is applied to the three center paragraphs, the default presentation would have the `ListParagraph` style applied as well:

Example text.

- Example text.
- Example text.
- Example text.

Example text.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:useNormalStyleForList />
</w:compat>
```

Then the new paragraph style must not be applied, resulting in the following output:

Example text.

- Example text.
- Example text.
- Example text.

Example text.

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.52 `usePrinterMetrics` (Use Printer Metrics To Display Documents)

This element specifies whether applications shall use the printer metrics of the currently active printer when determining how to display the contents of a WordprocessingML document. *Printer metrics* are printer-specific settings which can be queried to tell an application how and where text shall be displayed on a printed page.

Typically, applications display the content of a document in a device independent manner - the application is therefore not changing the layout of a document based on the currently attached printer, and instead shall dictate to the printer where characters shall be presented on the page when printed. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the metrics of the current printer shall be used to display the document instead.

Specifically, when this setting is enabled, the printer metrics are used to determine the number of pixels per logical inch along the screen width and height. This should then be used to compute the pixel height of the fonts requested when displaying the document, as well as to scale between any logical units within the document

(e.g. drawing object sizes) to the appropriate device units. Those units would then need to be scaled back into screen units for final display to a screen, but not scaled again when displayed to a printer.

[*Note:* On the Windows platform, you can use the `GetDeviceCaps` function to retrieve device-specific information for the specified printer. For this specific setting, you can use `GetDeviceCaps(hdc, LOGPIXELSX)` and `GetDeviceCaps(hdc, LOGPIXELSY)` with a printer DC to retrieve the number of pixels per logical inch along the screen width and height. With this, you can then use those DPI metrics to compute a pixel value for the font request in the LOGFONT structure (the LOGFONT structure defines the attributes of a font). A common formula to do this is $S_{px} = S_{pts} * \frac{LOGPIXELSY}{72}$. *end note*]

[*Example:* Consider a WordprocessingML document. The default must use device-independent layout to present the contents of the page.

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:usePrinterMetrics />
</w:compat>
```

Then the printer metrics of the current active printer must be used to determine the display of the contents of the document instead, as needed. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

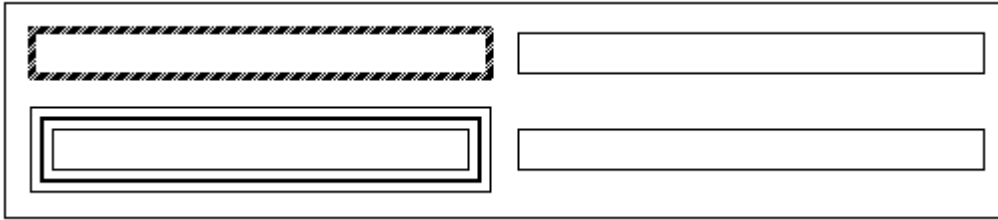
14.7.3.53 `useSingleBorderforContiguousCells` (Use Simplified Rules For Table Border Conflicts)

This element specifies whether applications should use an alternate simplified algorithm when handling conflicts between adjacent table borders within a table.

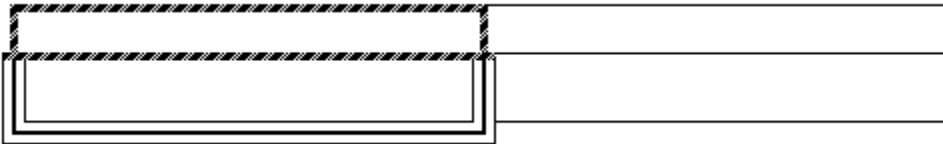
Typically, the conflicts between two adjacent table borders are handled using the conflict resolution algorithm defined in Part 1, §17.4.39 of ECMA-376. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that rather than using that algorithm to determine the outcome of the conflict to two adjacent borders, that the following logic shall be used instead:

- Cell borders shall supersede table borders
- Cell borders to the right shall supersede cell borders to the left (i.e. the rightmost border wins in conflicts between vertical borders)
- Cell borders below shall supersede cell borders above (i.e. the bottommost border wins in conflicts between horizontal borders)

[*Example:* Consider a WordprocessingML document with cell and table borders defined as follows. In the image below, 0.1" of padding has been added between each cell temporarily to clearly illustrate the borders on each cell and on the table:



The default presentation would have the border conflicts resolved using the algorithm defined by ECMA-376, resulting in the following table:



However, if this compatibility setting is turned on:

```
<w:compat>
  <w:useSingleBorderforContiguousCells />
</w:compat>
```

Then the simplified table algorithm above shall be used instead (bottom and right cell borders always win), resulting in the following output:



end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.54 useWord2002TableStyleRules (Incorrectly Display Top Border of Conditional Columns)

This element specifies whether applications should incorrectly calculate the top border of conditional columns (as specified by a `tblStylePr` element (Part 1, §17.7.6.6) with a type attribute value of `firstCol`, `lastCol`, `band1Vert`, or `band2Vert`) under the following conditions:

- A conditional formatting has also been defined for the first row (a `tblStylePr` element with a type attribute of `firstRow`)
- That conditional formatting has been applied to the table using the `tblLook` element (Part 1, §17.4.56)

Typically, table styles are applied according to the logic defined in Part 1, §17.7.2. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that the top border of those conditionally formatted columns should instead be displayed as the top border of the following row.

[Example: Consider a WordprocessingML document with table style that defines two conditional formats:

The first column has a one point border

The first row has red shading

That style would be defined as follows:

```
<w:style w:type="table" w:customStyle="1" w:styleId="TableTest">
  <w:name w:val="CompatibilitySetting"/>
  <w:tblStylePr w:type="firstRow">
    <w:tcPr>
      <w:shd w:val="clear" w:color="auto" w:fill="FF0000"/>
    </w:tcPr>
  </w:tblStylePr>
  <w:tblStylePr w:type="firstCol">
    <w:tcPr>
      <w:tcBorders>
        <w:top w:val="single" w:sz="4" w:space="0" w:color="auto"/>
        <w:left w:val="single" w:sz="4" w:space="0" w:color="auto"/>
        <w:bottom w:val="single" w:sz="4" w:space="0" w:color="auto"/>
        <w:right w:val="single" w:sz="4" w:space="0" w:color="auto"/>
      </w:tcBorders>
    </w:tcPr>
  </w:tblStylePr>
</w:style>
```

If the first column and first row formatting is applied, the table would appear as follows:

1,1	1,2
2,1	2,2

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:useWord2002TableStyleRules />
</w:compat>
```

Then the condition described by this element causes the top border defined by the conditional format for the first column to be displayed as the top border for the second column, resulting in the following output:

1,1	1,2
2,1	2,2

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.55 useWord97LineBreakRules (Use Incorrect Inter-Character Spacing Rules)

This element specifies that applications should perform specific calculations (detailed below) when determining inter-character spacing under certain conditions. These calculations would not normally be considered correct.

Typically, the behaviors specified by the following elements are applied unconditionally:

- The autoSpaceDE (Part 1, §17.3.1.2) and autoSpaceDN (Part 1, §17.3.1.3) elements
- The topLinePunct (Part 1, §17.3.1.43) element

The compatibility element described in this subclause, when present with a val attribute value of true (or equivalent), specifies that applications should ignore the settings listed above in the following scenarios:

1. If an ideographic character and a non-ideographic/numeric character are logically adjacent (ignoring all content which is not within a t element), but separated by a field boundary, i.e.:
 - The first character is within a fldSimple element, but the second is not.
 - The characters are separated by a fldChar element with a fldCharType attribute value of end

Then any appropriate inter-character spacing should be omitted. [*Note*: Inter-character spacing should still be calculated correctly within the field result. *end note*]

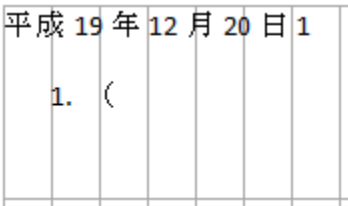
2. If a full-width punctuation character appears at the start of a paragraph which also specifies numbering via the numPr element (Part 1, §17.3.1.19), the compression specified by the topLinePunct element is ignored.

[*Example*: Consider a paragraph which contains a field ending in an ideograph and another paragraph, with numbering, which contains a full-width punctuation character in the first character position:

```
<w:p>
  <w:r>
    <w:fldChar w:fldCharType="begin" />
  </w:r>
  ...
  <w:r>
    <w:t>日</w:t>
  </w:r>
  <w:r>
    <w:fldChar w:fldCharType="end" />
```

```
</w:r>
<w:r>
  <w:t>1</w:t>
</w:r>
</w:p>
<w:p>
  <w:pPr>
    <w:numPr>
      ...
    </w:numPr>
  </w:pPr>
  <w:r>
    <w:t> (</w:t>
  </w:r>
</w:p>
```

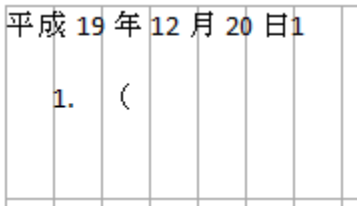
Typically, if both the autoSpaceDN and topLinePunct are true, additional spacing is added after the ideograph in the first paragraph and punctuation kerning is applied in the second paragraph (with gridlines added for visual reference):



If this compatibility setting is turned on:

```
<w:compat>
  <w:useWord97LineBreakRules />
</w:compat>
```

Then applications should not add any inter-character spacing at th end of the field and should turn off punctuation kerning in the second paragraph:



end example]

This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.56 wpJustification (Fit To Expanded Width When Performing Full Justification)

This element specifies that applications should perform a specific algorithm when determining the contents of each line in a fully justified paragraph (resulting from the use of the `jc` element (Part 1, §17.3.1.13)). This setting typically results in more words being fitted into lines (by reducing inter-word spacing as necessary).

Typically, applying full justification to a paragraph does not change the placement of line breaks, as inter-word spacing is expanded to ensure the resulting text is fully justified. This element, when present with a `val` attribute value of `true` (or equivalent), specifies that applications shall determine the contents of each line in a fully justified paragraph using the following algorithm:

For each line in the fully justified paragraph,

- Determine the actual line width, w , in pixels
- Calculate the “effective” line width by the following factor:

$$w_{\text{effective}} = w_{\text{actual}} + \left(w_{\text{actual}} * \frac{281}{7200} \right)$$

- Determine the text which can be displayed in a line of the “effective” line width
- Decrease the inter-word spacing as necessary to fit that text in the actual line width

[*Example:* Consider a WordprocessingML document with one or more paragraphs using full paragraph justification:

```
<w:p>
  <w:pPr>
    <w:jc w:val="both" />
  </w:pPr>
  ...
</w:p>
```

If this compatibility setting is turned on:

```
<w:compat>
  <w:wpJustification />
</w:compat>
```

Then, for a line 1000 pixels wide, an application would calculate the effective width as follows:

$$w_{\text{effective}} = 1000 + \left(1000 * \frac{281}{7200} \right) = 1039 \text{ pixels}$$

This effective width is then used to determine how much text can be displayed on line. After calculating the text, the application can display the text on the actual line, fully justified. *end example*]

This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.57 wpSpaceWidth (Use Specific Space Width)

(The terms *ascent* and *descent* are used as defined in ISO/IEC 14496-22.)

This element specifies that applications should perform determine the width of the space character for all proportional fonts used in this document using the calculation specified below.

Typically, applications calculate the width of a whitespace character dynamically to optimize for the output device. This element, when present with a *val* attribute value of *true* (or equivalent), specifies that applications should instead use the following algorithm to determine the width of a whitespace character:

$$w_{\text{space}} = \left(\frac{\text{ascent} + \text{descent}}{3} \right)$$

where

- w_{space} is the width of a space character
- is the ascent for the font
- is the descent for the font

[*Example*: Consider a WordprocessingML document with this compatibility setting turned on:

```
<w:compat>
  <w:suppressTopSpacingWP />
</w:compat>
```

If the font applied to a run specified an ascent value of 8 points and a descent value of 2 points, each space in that run would have a width of three and one-third points. *end example*]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.3.58 wrapTrailSpaces (Line Wrap Trailing Spaces)

This element specifies whether applications shall perform line wrapping on trailing spaces in the contents of a line when displaying in it a paragraph. *Trailing spaces* are all space characters which are not followed by non-space characters on the same line.

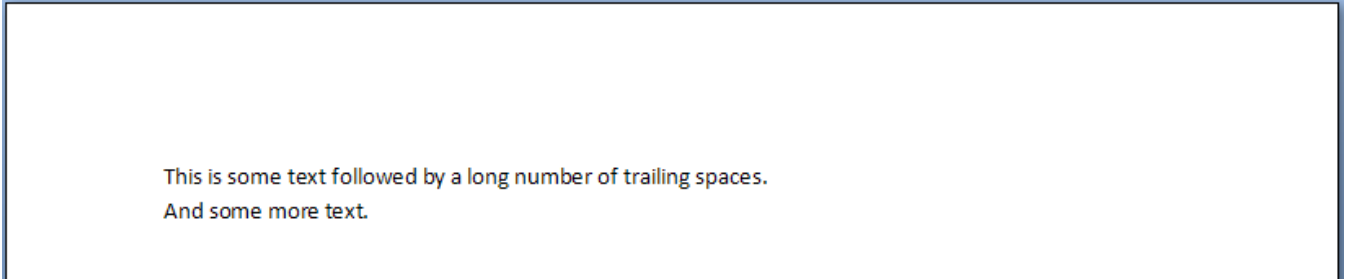
Typically, applications do not line wrap trailing spaces, instead allowing an unbounded number of trailing spaces on a line, with the next non-space character starting at the first character position on the next line. This element, when present with a *val* attribute value of *true* (or equivalent), specifies that all characters, including trailing spaces, shall be line wrapped normally.

[*Example*: Consider a WordprocessingML document with the following paragraph of text, including a long interstitial of spaces which become trailing spaces when the paragraph is displayed:


```
<w:r>
  <w:t> This is some text followed by a long number of trailing spaces.
```

```
    And some more text.</w:t>
</w:r>
```

The default presentation would not wrap those trailing spaces, so the text at the end of the run would begin at the first character position on the second line:

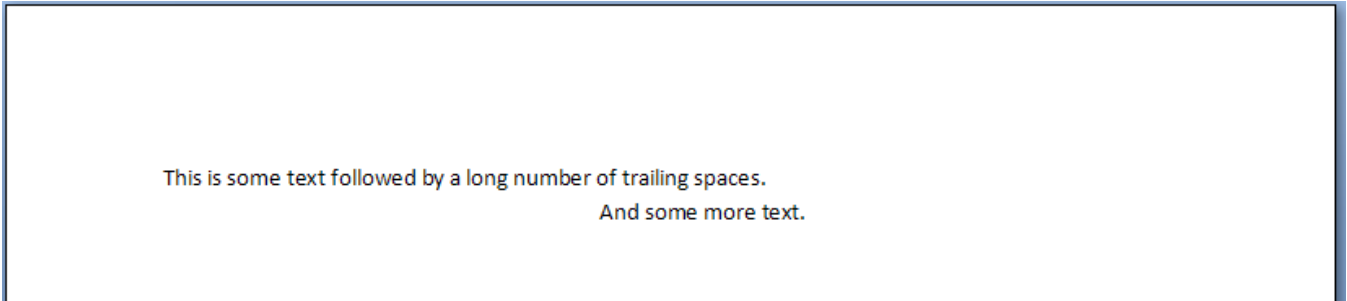


```
This is some text followed by a long number of trailing spaces.
And some more text.
```

However, if this compatibility setting is turned on:

```
<w:compat>
  <w:wrapTrailSpaces />
</w:compat>
```

Then all trailing spaces would be handled as regular characters when line wrapping, resulting in the following output:



```
This is some text followed by a long number of trailing spaces.
    And some more text.
```

end example]

This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

14.7.4 Web Page Settings

14.7.4.1 [relyOnVML \(Utilize VML When Saving as Web Page\)](#)

This element specifies whether applications can utilize the Vector Markup Language format when saving the content of this WordprocessingML document as a web page, when graphical elements that can leverage this format are present in the document.

If this element is omitted, then a graphic image format should be used either in place of or in concert with the Vector Markup Language output in order to specify the formatting and positioning for objects that are part of the resulting web page.

[*Note:* This setting is intended for applications to save web pages that can be supported by legacy web browsers that do not support Vector Markup Language when attempting to read and display the resulting web page. *end note*]

[*Example:* Consider a WordprocessingML document that contains the following content within the web settings part:

```
<w:webSettings>
  <w:relyOnVML w:val="false" />
</w:webSettings>
```

The relyOnVML element has a val attribute value of false, which specifies that applications should utilize a graphical image version of all objects that could utilize Vector Markup Language output. This does not preclude the use of the VML output, but does specify that a graphical element must be included as well. *end example*]

This element's content model is defined by the common boolean property definition in Part 1 §17.17.4.

14.8 Miscellaneous Topics

14.8.1 Text Box Content

All VML-based drawing objects (except for connectors) support the addition of rich WordprocessingML content within their extents. When WordprocessingML contents have been added to a VML drawing object, the resulting text is contained within a *text box*.

When WordprocessingML content is contained within a text box, it is allowed within the object by specifying the VML textbox element (§19.1.2.22), which contains within it a single txbxContent element that contains all of the desired WordprocessingML content.

14.8.1.1 txbxContent (Rich Text Box Content Container)

This element specifies that its contents shall be any rich WordprocessingML content, and that this content is the rich contents of a drawing object defined using the Vector Markup Language (VML) syntax (§19.1).

If this element contains within any of its contents any of the following content, then the document shall be considered non-conformant:

- References to other WordprocessingML document stories (comments, footnotes, endnotes)
- Additional txbxContent elements (as part of nested VML objects)

[*Example:* Consider a WordprocessingML document consisting of a single VML shape element (§19.1.2.19) (in this case, a star) that contains within it some WordprocessingML content:



That drawing object now contains a text box, and so uses the syntax for that text box:

```
<v:shape id="_x0000_s1026" type="#_x0000_t12" style="...">
  <v:textbox>
    <w:txbxContent>
      <w:p>
        <w:pPr>
          <w:jc w:val="center"/>
        </w:pPr>
        <w:r>
          <w:t>Rich WordprocessingML content!</w:t>
        </w:r>
      </w:p>
    </w:txbxContent>
  </v:textbox>
</v:shape>
```

end example]

[*Note:* The txbxContent element is the container for the WordprocessingML contained within the text box inside that shape - once inside this element any content (subject to the restrictions defined above) can be used. For compatibility with existing implementations, unqualified elements are used inside the txbxContent element to represent HTML fragments. *end note]*

[Note: The W3C XML Schema definition of this element's content model ([CT_TxbxContent](#)) is located in §A.1.
end note]

14.9 Fields and Hyperlinks

14.9.1 Syntax

This subclause modifies the fields grammar defined in Part 1, §17.16.1 as necessary to support transitional use of fields.

The syntax rules in this subclause follow the system shown in ISO/IEC 14977: literal text is surrounded by double-quotes (or by apostrophes); the left-square-bracket and right-square-bracket designate the start and end of an option; the left-curly-bracket and right-curly-bracket designate the start and end of a sequence of one-or-more items; the vertical-line indicates an alternative; and each rule ends with a semicolon. Whenever hyphen is used as the exception-symbol (as per ISO/IEC 14977), it is surrounded by white space, and further clarified by a comment.

```
field-type=
    date-and-time |
    document-automation |
    document-information |
    document-property |
    equations-and-formulas |
    index-and-tables |
    links-and-references |
    mail-merge |
    numbering |
    user-information |
    form-field |
    user-defined-field |
    transitional-fields ;
transitional-fields=
    "AUTONUM" |
    "AUTONUMLGL" |
    "AUTONUMOUT" |
    "BARCODE", field-argument |
    "BIDIOUTLINE" |
    "EQ", [switches], "(", [eq-argument-list], ")" (* and see §14.9.4.6 *) |
    "INFO", info-category, [field-argument] ;
eq-argument-list=
    expression, {eq-list-separator, expression} ;
eq-list-separator=
    comma | semicolon ;          (* depending on the rules in §14.9.4.6 *)
```

info-category:

```
"AUTHOR" | "COMMENTS" | "CREATEDATE" | "EDITTIME" |
"FILENAME" | "FILESIZE" | "KEYWORDS" | "LASTSAVEDBY" |
"NUMCHARS" | "NUMPAGES" | "NUMWORDS" | "PRINTDATE" |
"REVNUM" | "SAVEDATE" | "SUBJECT" | "TEMPLATE" | "TITLE" ;
```

14.9.2 Legacy language references

Whenever a field requires a language identifier as one of its *field-switches*, that language identifier should be provided using the syntax defined by the ST_Lang simple type (Part 1, §22.9.2.6). However, there exists a legacy mechanism by which language identifiers can be stored. For that mechanism, the following table lists those language codes and their corresponding languages.

This mechanism can be used within the following *field-switches*:

- ADDRESSBLOCK (Part 1, §17.16.5.1), \l switch
- BIBLIOGRAPHY (Part 1, §17.16.5.7), \l and \f switches
- CITATION (Part 1, §17.16.5.8), \l switch
- GREETINGLINE (Part 1, §17.16.5.24), \l switch
- INDEX (Part 1, §17.16.5.29), \z switch

[*Rationale*: This list is maintained for compatibility with documents containing these values. The use of these identifiers is discouraged. *end rationale*]

[*Note*: The second column "Description" is informative only, and is provided as an aid to implementers. Note also that the inclusion of country subtags in the BCP 47 codes makes no assertion about the relationship between nations and languages. Rather, it reflects the historical commercial process by which office software products were localized for some particular market. For example, the Swahili language is spoken in several Eastern African nations. However, the localization identified by the legacy language code 1089 reflected work done in Kenya to address the needs of Swahili users there and thus is mapped to the modern BCP 47 code sw-KE. *end note*]

Language Code	Description (informative)	BCP 47 Code
1025	Arabic - Saudi Arabia	ar-SA
1026	Bulgarian	bg-BG
1027	Catalan	ca-ES
1028	Chinese - Taiwan	zh-TW
1029	Czech	cs-CZ
1030	Danish	da-DK
1031	German - Germany	de-DE
1032	Greek	el-GR
1033	English - United States	en-US

Language Code	Description (informative)	BCP 47 Code
1034	Spanish - Spain (Traditional Sort)	es-ES
1035	Finnish	fi-FI
1036	French - France	fr-FR
1037	Hebrew	he-IL
1038	Hungarian	hu-HU
1039	Icelandic	is-IS
1040	Italian - Italy	it-IT
1041	Japanese	ja-JP
1042	Korean	ko-KR
1043	Dutch - Netherlands	nl-NL
1044	Norwegian (Bokmål)	nb-NO
1045	Polish	pl-PL
1046	Portuguese - Brazil	pt-BR
1047	Rhaeto-Romanic	rm-CH
1048	Romanian	ro-RO
1049	Russian	ru-RU
1050	Croatian	hr-HR
1051	Slovak	sk-SK
1052	Albanian - Albania	sq-AL
1053	Swedish	sv-SE
1054	Thai	th-TH
1055	Turkish	tr-TR
1056	Urdu - Pakistan	ur-PK
1057	Indonesian	id-ID
1058	Ukrainian	uk-UA
1059	Belarusian	be-BY
1060	Slovenian	sl-SI
1061	Estonian	et-EE
1062	Latvian	lv-LV
1063	Lithuanian	lt-LT
1064	Tajik	tg-Cyrl-TJ
1065	Farsi	fa-IR
1066	Vietnamese	vi-VN
1067	Armenian - Armenia	hy-AM

Language Code	Description (informative)	BCP 47 Code
1068	Azeri (Latin)	az-Latn-AZ
1069	Basque	eu-ES
1070	Sorbian	wen-DE
1071	FYRO Macedonian	mk-MK
1072	Sutu	st-ZA
1073	Tsonga	ts-ZA
1074	Tswana	tn-ZA
1075	Venda	ven-ZA
1076	Xhosa	xh-ZA
1077	Zulu	zu-ZA
1078	Afrikaans - South Africa	af-ZA
1079	Georgian	ka-GE
1080	Faroese	fo-FO
1081	Hindi	hi-IN
1082	Maltese	mt-MT
1083	Sami	se-NO
1084	Gaelic (Scotland)	gd-GB
1085	Yiddish	yi
1086	Malay - Malaysia	ms-MY
1087	Kazakh	kk-KZ
1088	Kyrgyz (Cyrillic)	ky-KG
1089	Swahili	sw-KE
1090	Turkmen	tk-TM
1091	Uzbek (Latin)	uz-Latn-UZ
1092	Tatar	tt-RU
1093	Bengali (India)	bn-IN
1094	Punjabi	pa-IN
1095	Gujarati	gu-IN
1096	Oriya	or-IN
1097	Tamil	ta-IN
1098	Telugu	te-IN
1099	Kannada	kn-IN
1100	Malayalam	ml-IN
1101	Assamese	as-IN

Language Code	Description (informative)	BCP 47 Code
1102	Marathi	mr-IN
1103	Sanskrit	sa-IN
1104	Mongolian (Cyrillic)	mn-MN
1105	Tibetan - People's Republic of China	bo-CN
1106	Welsh	cy-GB
1107	Khmer	km-KH
1108	Lao	lo-LA
1109	Burmese	my-MM
1110	Galician	gl-ES
1111	Konkani	kok-IN
1112	Manipuri	mni
1113	Sindhi - India	sd-IN
1114	Syriac	syr-SY
1115	Sinhalese - Sri Lanka	si-LK
1116	Cherokee - United States	chr-US
1117	Inuktitut	iu-Cans-CA
1118	Amharic - Ethiopia	am-ET
1119	Tamazight (Arabic)	tmz
1120	Kashmiri (Arabic)	ks-Arab-IN
1121	Nepali	ne-NP
1122	Frisian - Netherlands	fy-NL
1123	Pashto	ps-AF
1124	Filipino	fil-PH
1125	Divehi	dv-MV
1126	Edo	bin-NG
1127	Fulfulde - Nigeria	fuv-NG
1128	Hausa - Nigeria	ha-Latn-NG
1129	Ibibio - Nigeria	ibb-NG
1130	Yoruba	yo-NG
1131	Quecha - Bolivia	quz-BO
1132	Sepedi	nso-ZA
1136	Igbo - Nigeria	ig-NG
1137	Kanuri - Nigeria	kr-NG
1138	Oromo	gaz-ET

Language Code	Description (informative)	BCP 47 Code
1139	Tigrigna - Ethiopia	ti-ER
1140	Guarani - Paraguay	gn-PY
1141	Hawaiian - United States	haw-US
1142	Latin	la
1143	Somali	so-SO
1144	Yi	ii-CN
1145	Papiamentu	pap-AN
1152	Uighur - China	ug-Arab-CN
1153	Maori - New Zealand	mi-NZ
2049	Arabic - Iraq	ar-IQ
2052	Chinese - People's Republic of China	zh-CN
2055	German - Switzerland	de-CH
2057	English - United Kingdom	en-GB
2058	Spanish - Mexico	es-MX
2060	French - Belgium	fr-BE
2064	Italian - Switzerland	it-CH
2067	Dutch - Belgium	nl-BE
2068	Norwegian (Nynorsk)	nn-NO
2070	Portuguese - Portugal	pt-PT
2072	Romanian - Moldava	ro-MO
2073	Russian - Moldava	ru-MO
2074	Serbian (Latin)	sr-Latn-CS
2077	Swedish - Finland	sv-FI
2080	Urdu - India	ur-IN
2092	Azeri (Cyrillic)	az-Cyrl-AZ
2108	Gaelic (Ireland)	ga-IE
2110	Malay - Brunei Darussalam	ms-BN
2115	Uzbek (Cyrillic)	uz-Cyrl-UZ
2117	Bengali (Bangladesh)	bn-BD
2118	Punjabi (Pakistan)	pa-PK
2128	Mongolian (Mongolian)	mn-Mong-CN
2129	Tibetan - Bhutan	bo-BT
2137	Sindhi - Pakistan	sd-PK
2143	Tamazight (Latin)	tzm-Latn-DZ

Language Code	Description (informative)	BCP 47 Code
2144	Kashmiri (Devanagari)	ks-Deva-IN
2145	Nepali - India	ne-IN
2155	Quecha - Ecuador	quz-EC
2163	Tigrigna - Eritrea	ti-ET
3073	Arabic - Egypt	ar-EG
3076	Chinese - Hong Kong SAR	zh-HK
3079	German - Austria	de-AT
3081	English - Australia	en-AU
3082	Spanish - Spain (Modern Sort)	es-ES
3084	French - Canada	fr-CA
3098	Serbian (Cyrillic)	sr-Cyrl-CS
3179	Quecha - Peru	quz-PE
4097	Arabic - Libya	ar-LY
4100	Chinese - Singapore	zh-SG
4103	German - Luxembourg	de-LU
4105	English - Canada	en-CA
4106	Spanish - Guatemala	es-GT
4108	French - Switzerland	fr-CH
4122	Croatian (Bosnia/Herzegovina)	hr-BA
5121	Arabic - Algeria	ar-DZ
5124	Chinese - Macao SAR	zh-MO
5127	German - Liechtenstein	de-LI
5129	English - New Zealand	en-NZ
5130	Spanish - Costa Rica	es-CR
5132	French - Luxembourg	fr-LU
5146	Bosnian (Bosnia/Herzegovina)	bs-Latn-BA
6145	Arabic - Morocco	ar-MO
6153	English - Ireland	en-IE
6154	Spanish - Panama	es-PA
6156	French - Monaco	fr-MC
7169	Arabic - Tunisia	ar-TN
7177	English - South Africa	en-ZA
7178	Spanish - Dominican Republic	es-DO
7180	French - West Indies	fr-029

Language Code	Description (informative)	BCP 47 Code
8193	Arabic - Oman	ar-OM
8201	English - Jamaica	en-JM
8202	Spanish - Venezuela	es-VE
8204	French - Reunion	fr-RE
9217	Arabic - Yemen	ar-YE
9225	English - Caribbean	en-029
9226	Spanish - Colombia	es-CO
9228	French - Democratic Rep. of Congo	fr-CG
10241	Arabic - Syria	ar-SY
10249	English - Belize	en-BZ
10250	Spanish - Peru	es-PE
10252	French - Senegal	fr-SN
11265	Arabic - Jordan	ar-JO
11273	English - Trinidad	en-TT
11274	Spanish - Argentina	es-AR
11276	French - Cameroon	fr-CM
12289	Arabic - Lebanon	ar-LB
12297	English - Zimbabwe	en-ZW
12298	Spanish - Ecuador	es-EC
12300	French - Cote d'Ivoire	fr-CI
13313	Arabic - Kuwait	ar-KW
13321	English - Philippines	en-PH
13322	Spanish - Chile	es-CL
13324	French - Mali	fr-ML
14337	Arabic - U.A.E.	ar-AE
14345	English - Indonesia	en-ID
14346	Spanish - Uruguay	es-UY
14348	French - Morocco	fr-MA
15361	Arabic - Bahrain	ar-BH
15369	English - Hong Kong SAR	en-HK
15370	Spanish - Paraguay	es-PY
15372	French - Haiti	fr-HT
16385	Arabic - Qatar	ar-QA
16393	English - India	en-IN

Language Code	Description (informative)	BCP 47 Code
16394	Spanish - Bolivia	es-BO
17417	English - Malaysia	en-MY
17418	Spanish - El Salvador	es-SV
18441	English - Singapore	en-SG
18442	Spanish - Honduras	es-HN
19466	Spanish - Nicaragua	es-NI
20490	Spanish - Puerto Rico	es-PR
21514	Spanish - United States	es-US
58378	Spanish - Latin America	es-419
58380	French - North Africa	fr-015
Any other value	Undefined. Shall not be used.	

14.9.3 Use of DOS File Paths

The following fields allow the use of a DOS file path in place of the (preferred) IRI syntax:

- INCLUDEPICTURE (Part 1, §17.16.5.27)
- INCLUDETTEXT (Part 1, §17.16.5.28)

When a DOS file path is specified in a *field-argument*, each backslash character shall be preceded directly by another backslash character [*Example*: E:\\example.docx *end example*] If *field-argument* contains white space, it shall be enclosed in double quotes.

14.9.4 Field definitions

14.9.4.1 AUTONUM

Syntax:

AUTONUM [*switches*]

Description: In paragraphs formatted with one of the nine built-in heading styles, paragraph numbering restarts at 1 in each successive heading level. If headings that contain AUTONUM fields are followed by body text paragraphs that also contain AUTONUM fields, the paragraph numbering of the body text is restarted at 1 after each heading. If the headings don't contain AUTONUM fields, body text paragraphs that contain AUTONUM fields are numbered in a continuous, sequential series throughout the document. [*Note*: This field is supported for legacy reasons, It is recommended that LISTNUM (Part 1, §17.16.5.33) be used instead. *end note*]

The XML generated for a complex field implementation shall not have the optional field value stored.

Field Value: A new paragraph number in ascending sequential order.

Switches: Zero or one of the *general-formatting-switches*, or zero or more of the following *field-specific-switches*.

<code>\s <i>field-argument</i></code>	<i>text</i> in this switch's <i>field-argument</i> specifies the separator character to be used. If \s is omitted, a period (.) is used.
---------------------------------------	--

[*Example:* When the following fields are updated:

```
AUTONUM
AUTONUM \* Arabic \s :
AUTONUM \* alphabetic \s " "xxx
AUTONUM \* ROMAN
AUTONUM \* OrdText
```

The results are:

```
1.
2:
c xxx
IV.
fifth.
```

end example]

14.9.4.2 AUTONUMLGL

Syntax:

AUTONUMLGL [*switches*]

Description: For legal and technical publications, use the nine built-in heading styles to format headings in the document, and then insert an AUTONUMLGL field at the beginning of each heading paragraph. The numbers reflect the heading levels that correspond to the heading styles. If an AUTONUMLGL field is inserted in paragraphs of body text paragraphs not formatted with built-in heading styles, the number of the preceding heading is included in the paragraph number. [*Note:* This field is supported for legacy reasons, It is recommended that LISTNUM (Part 1, §17.16.5.33) be used instead. *end note*]

This field only makes sense in terms of multi-level headings. Given the following headings:

```
Heading 1
Heading 2
Heading 2
Heading 1
```

this field allows

- 1. Heading 1
- 1.1. Heading 2
- 1.2. Heading 2
- 2. Heading 1

At each level, the numbering sequence does two things—it increments specific to that level, and it includes the value from the previous level.

The XML generated for a complex field implementation shall not have the optional field value stored.

Field Value: A new paragraph number in ascending sequential order.

Switches: Zero or one of the *general-formatting-switches*, or zero or more of the following *field-specific-switches*.

<code>\e</code>	Removes the trailing separator (period).
<code>\s <i>field-argument</i></code>	<i>text</i> in this switch's <i>field-argument</i> specifies the separator character to be used. If <code>\s</code> is omitted, a period (.) is used.

[*Example:* When the following fields are updated:

```
AUTONUMLGL
AUTONUMLGL \* Arabic \s :
AUTONUMLGL \* alphabetic \s " "xxx
AUTONUMLGL \* ROMAN
AUTONUMLGL \e xxx
```

The results are:

```
1.
2:
c xxx
IV.
5xxx
```

end example]

14.9.4.3 AUTONUMOUT

Syntax:

AUTONUMOUT

Description: Use the nine built-in heading styles to format headings in the document, and then insert an AUTONUMOUT field at the beginning of each heading paragraph. The numbers reflect the heading levels that

correspond to the heading styles. [*Note: This field is supported for legacy reasons, It is recommended that LISTNUM (Part 1, §17.16.5.33) be used instead. end note*]

The XML generated for a complex field implementation shall not have the optional field value stored.

This field allows the numbering to be incremented based on the heading level. Given the following:

```
{AutoNumOut} Heading 1
{AutoNumOut} Heading 2
{AutoNumOut} Heading 2
{AutoNumOut} Heading 1
```

results in

```
I. Heading 1
A. Heading 2
B. Heading 2
II. Heading 1
```

Field Value: A paragraph number.

Switches: None.

[*Example: When the following fields are updated:*

```
AUTONUMOUT
AUTONUMOUT
```

The results are:

```
1.
2.
```

end example]

14.9.4.4 BARCODE

Syntax:

```
BARCODE field-argument [ switches ]
```

Description: Produces a postal bar code in a machine-readable form of address used by the U.S. Postal Service. The barcode is in the form of either a POSTNET delivery-point bar code or a Facing Identification Mark (FIM). *text* in *field-argument* can be either a postal address or a bookmark name. In the case of a postal address, all that is needed is a 5-digit or 9-digit ZIP code; the rest of the address is superfluous.

Field Value: A postal bar code.

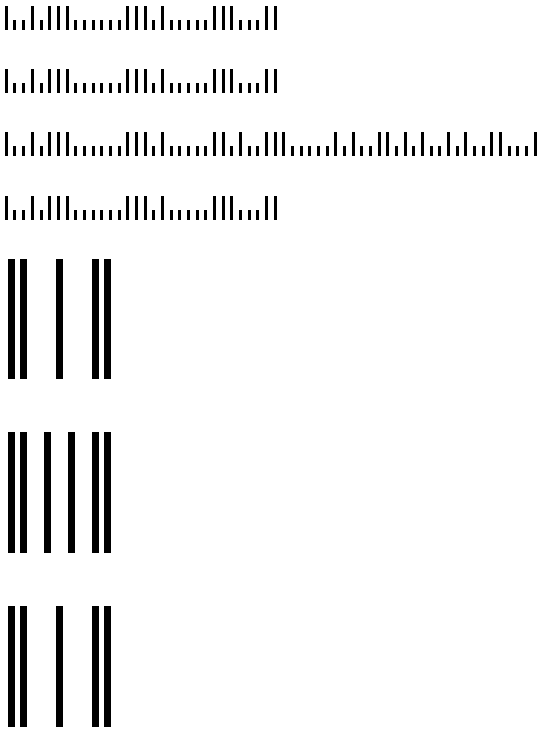
Switches: Zero or more of the following *field-specific-switches*.

<code>\b</code>	Indicates that <i>text</i> in <i>field-argument</i> is the name of a bookmark.
<code>\f <i>field-argument</i></code>	Inserts a Facing Identification Mark (FIM). <i>text</i> in this switch's <i>field-argument</i> shall be either "A" (courtesy reply mark) or "C" (business reply mark).
<code>\u</code>	Indicates that <i>text</i> in <i>field-argument</i> is a U.S. postal address.

[*Example:* Consider the case in which PostalAddress is the name of a bookmark for the text "2051 Swans Neck Way, Reston VA 20191". When the following fields are updated:

```
BARCODE 20191
BARCODE 20191 \u
BARCODE 20191-4023 \u
BARCODE "2051 Swans Neck Way, Reston VA 20191" \u
BARCODE "2051 Swans Neck Way, Reston VA 20191" \f A
BARCODE 20191 \f C
BARCODE PostalAddress \b \f A
```

The results are:



end example]

14.9.4.5 BIDIOUTLINE

Syntax:

BIDIOUTLINE

Description: This field is identical to the AUTONUMGL field (§14.9.4.3), except for the separator that delimits each level of the paragraph numbering (this field uses a hyphen-minus (U+002D) instead of a full stop (U+002E) character as the default separator character).

Field Value: A new paragraph number in ascending sequential order, as defined by the description in §14.9.4.3.

Switches: None.

14.9.4.6 EQ

Syntax:EQ [*switches*] (*eq-argument-list*) [*switches*]

eq-argument-list is a list of arguments separated using a separator character. For implementations using a period (.) as the radix point, the separator character is a comma (,). For implementations using a comma (,) as the radix point, the separator character is a semicolon (;).

Description: Computes the specified mathematical equation.

Field Value: The result of the specified mathematical equation. [*Note:* The result of an EQ field can be used as an argument in another EQ field's *eq-argument-list*. *end note*]

Switches: The left-hand *switches* can only be one of the following: \a, \b, \d, \f, \i, \l, \o, \r, \s, and \x. Each of these switches has one or more subswitches, as shown below.

\a produces an array using the argument values in *eq-argument-list* (which are in row-major order) and the *field-specific-switches* below:

\ac	Alignment is centered in each array column.
\al	Alignment is left in each array column.
\ar	Alignment is right in each array column.
\co <i>field-argument</i>	The number of columns in the array is specified by <i>text</i> in this switch's <i>field-argument</i> . In the absence of this switch, the number is 1.
\hs <i>field-argument</i>	Adds the integral number of points of horizontal spacing specified by <i>text</i> in this switch's <i>field-argument</i> between columns.
\vs <i>field-argument</i>	Adds the integral number of points of vertical spacing specified by <i>text</i> in this switch's <i>field-argument</i> between lines.

`\b` brackets the single element in *eq-argument-list* in a size appropriate for that element. The default form of brackets is parentheses. The *field-specific-switches* below can be used:

<code>\bc \char</code>	Uses the character designated by <i>char</i> as both the left and right bracket character. However, if <i>char</i> is {, [, (, or <, that character is used for the left bracket, and },],), or >, respectively, is used for right bracket.
<code>\lc \char</code>	Uses the character designated by <i>char</i> as the left bracket character.
<code>\rc \char</code>	Uses the character designated by <i>char</i> as the right bracket character.

`\d` Controls where the next character following the EQ field is drawn (that is, the displacement). *eq-argument-list* shall have no arguments. The *field-specific-switches* below can be used:

<code>\ba field-argument</code>	Draws to the left (backward) the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .
<code>\fo field-argument</code>	Draws to the right (forward) the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .
<code>\li</code>	Underlines the space up to the next character.

`\f` Creates a fraction with the first argument as numerator and the second argument as denominator, centered above and below the division line, respectively. *eq-argument-list* shall have exactly two arguments. There are no *field-specific-switches* for this switch.

`\i` Creates an integral using the specified or default symbol and three elements. The first argument is the lower limit, the second is the upper limit, and the third is the integrand. *eq-argument-list* shall have exactly three arguments. The *field-specific-switches* below can be used:

<code>\fc \char</code>	Uses the character designated by <i>char</i> as the fixed-height character for the symbol.
<code>\in</code>	Uses an inline format with the limits displayed to the right of the symbol instead of above and below it.
<code>\pr</code>	Uses the symbol Capital pi and creates a product.
<code>\su</code>	Uses the symbol Capital sigma and creates a summation.
<code>\vc \char</code>	Uses the character designated by <i>char</i> as the variable-height character for the symbol. The symbol matches the height of the third argument.

`\lf` Creates a list from an arbitrary number of arguments. There are no *field-specific-switches* for this switch.

`\o` Using an arbitrary number of arguments, displays each successive argument on top of the previous one. Each character is displayed within an invisible character box, with the switches being available to align the boxes on top of one another. The *field-specific-switches* below can be used:

<code>\ac</code>	Alignment character box center (the default).
<code>\al</code>	Alignment character box left.
<code>\ar</code>	Alignment character box right.

`\r` Creates a radical. *eq-argument-list* shall have either one or two arguments. If it has one argument, the result is the square root of that argument. If it has two arguments, the result is the *n*th root of the second argument, where *n* is the first argument. There are no *field-specific-switches* for this switch.

`\s` Creates a subscript or superscript. One or more arguments are permitted. If more than one element is specified, the elements are stacked and left-aligned. The *field-specific-switches* below can be used:

<code>\ai <i>field-argument</i></code>	Adds space above a line in a paragraph by the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> . The default is 2 points.
<code>\di <i>field-argument</i></code>	Adds space below a line in a paragraph by the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .
<code>\do <i>field-argument</i></code>	Moves a single argument below the adjacent text by the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> . The default is 2 points.
<code>\up <i>field-argument</i></code>	Moves a single argument above the adjacent text by the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .

`\x` Creates one or more border segments around a single argument. By default, all four borders are added. *eq-argument-list* shall have no arguments. The *field-specific-switches* below can be used:

<code>\bo</code>	Draws a horizontal border below the argument.
<code>\le</code>	Draws a vertical border to the left of the argument.
<code>\ri</code>	Draws a vertical border to the right of the argument.
<code>\to</code>	Draws a horizontal border above the argument.

[Example: When the following fields are updated:

```
EQ \a \co 2 \ac \hs 10 ( 1000, 20, A, Sunday )
EQ \b \bc \l ( -100 ) EQ \b \bc \l ( \r(3, a + b)
xx EQ \d \fo 20 ( ) xx EQ \d \fo 30 \li ( )xx
EQ \f ( 1, 32 ) EQ \f ( 7, 64 )
EQ \i ( 0, ∞, x ) EQ \i \su \in ( 0, 10, x ) EQ \i \pr \in ( 0, 5, x )
EQ \i \fc \{ ( 0, 5, \f (x, 0.34) ) EQ \i \vc \{ ( 0, 5, \f (x, 0.34) )

EQ \l ( 0, 10 )
EQ \b \lc \l ( \rc \l ( \l ( 0, 10) )
```

$\text{EQ} \setminus \text{o} (0, 0, 0)$ $\text{EQ} \setminus \text{o} (0, +)$ $\text{EQ} \setminus \text{o} \setminus \text{ar} (0, |, _)$
 $\text{EQ} \setminus \text{r} (2)$ $\text{EQ} \setminus \text{r} (2, \text{x})$
 $\text{a EQ} \setminus \text{s} \setminus \text{up} (2) + \text{b EQ} \setminus \text{s} \setminus \text{up} (2)$
 $\text{a EQ} \setminus \text{x} (+) \text{b}$ $\text{a EQ} \setminus \text{x} \setminus \text{to} \setminus \text{le} (+) \text{b}$ $\text{a EQ} \setminus \text{x} \setminus \text{bo} \setminus \text{ri} (+) \text{b}$

The results are:

1000 20
A Sunday
| -100 | $\left| \sqrt[3]{a+b} \right|$

xx xx_____xx

$\frac{1}{32}$ $\frac{7}{64}$

$\int\limits_0^{\infty} \text{x} \sum\limits_0^{10 \text{x}} \prod\limits_0^{5 \text{x}}$

$\frac{5}{0} \frac{\text{x}}{0.34}$ $\left\{ \frac{\text{x}}{0.34} \right.$
0

0, 10
[0, 10)

0 0 0

$\sqrt{2}$ $\sqrt[2]{\text{x}}$

$\text{a}^2 + \text{b}^2$

$\text{a} \boxed{+} \text{b}$ $\text{a} \overline{+} \text{b}$ $\text{a} \underline{+} \text{b}$

end example]

14.9.4.7 INFO

Syntax:

INFO info-category [*field-argument*] [*switches*]

This field is documented for purposes of backwards compatibility. Each permitted value for *info-category* is also permitted as a *field-type*. Instances of the INFO field shall be treated as an instance of the *field-type* with the same value as *info-category*; that is, as if the INFO token was not present.

14.9.4.8 QUOTE

This field retrieves the text specified by *text* in *field-argument*. In strict conformance mode, this text may include any other fields except SYMBOL. However, in transitional conformance mode, this text may include any other fields except AUTONUM, AUTONUMLGL, AUTONUMOUT, and SYMBOL.

14.9.5 fldData (Custom Field Data)

This element specifies custom field data which shall be associated with the parent field. No information or semantics are applied to the contents of this data by ECMA-376, and therefore this field can be used as desired to store additional application-defined data with the field. However, applications should not lose the contents of this custom data if they do not understand or utilize it (i.e. the information should continue to be saved with the file).

If this element is omitted, then no custom field data is stored with the parent field. If the type attribute of the current field character is not start, then his setting can be ignored.

[*Example*: Consider the following WordprocessingML fragment for a complex field:

```
<w:r>
  <w:fldChar w:fldCharType="start">
    <w:fldData xml:space="preserve">///3645ERKJHE</w:fldData>
  </w:fldChar>
</w:r>
<w:r>
  <w:instrText>PRIVATE</w:instrText>
</w:r>
<w:r>
  <w:fldChar w:fldCharType="separate" />
</w:r>
...
```

The fldData element contains custom data stored with this PRIVATE field (Part 1, §17.16.5.48), the contents of which are determined by a hosting application. *end example*]

Attributes	Description
xml:space (Content Contains Significant Whitespace)	Specifies how white space should be handled for the contents of this element using the W3C space preservation rules.
Namespace: http://www.w3.or	[<i>Example</i> : Consider the following run contained within a WordprocessingML document: <w:r>

Attributes	Description
g/XML/1998/namespace	<p><code><w:t></code> significant whitespace <code></w:t></code> <code></w:r></code></p> <p>Although there are three spaces on each side of the text content in the run, that whitespace has not been specifically marked as significant, therefore it is subject to the space preservation rules currently specified in that run's scope. <i>end example</i>]</p> <p>The possible values for this attribute are defined by §2.10 of the XML 1.0 specification.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Text](#)) is located in §A.1. *end note*]

14.9.6 fldData (Custom Field Data)

This element specifies custom field data which shall be associated with the parent field. No information or semantics are applied to the contents of this data by ECMA-376, and therefore this field can be used as desired to store additional application-defined data with the field. However, applications should not lose the contents of this custom data if they do not understand or utilize it (i.e. the information should continue to be saved with the file).

If this element is omitted, then no custom field data is stored with the parent field.

[Example: Consider the following WordprocessingML fragment for a simple field:

```
<w:fldSimple w:instr="PRIVATE">
  <w:fldData xml:space="preserve">///3645ERKJHE</w:fldData>
</w:fldSimple>
```

The fldData element contains custom data stored with this PRIVATE field (Part 1, §17.16.5.48), the contents of which are determined by a hosting application. *end example*]

Attributes	Description
xml:space (Content Contains Significant Whitespace) Namespace: http://www.w3.org/XML/1998/namespaces	<p>Specifies how white space should be handled for the contents of this element using the W3C space preservation rules.</p> <p>[Example: Consider the following run contained within a WordprocessingML document:</p> <p><code><w:r></code> <code><w:t></code> significant whitespace <code></w:t></code> <code></w:r></code></p> <p>Although there are three spaces on each side of the text content in the run, that whitespace has not been specifically marked as significant, therefore it is subject to the space preservation rules currently specified in that run's scope. <i>end example</i>]</p> <p>The possible values for this attribute are defined by §2.10 of the XML 1.0 specification.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Text](#)) is located in §A.1. *end note*]

14.9.7 hyperlink (Hyperlink) (Part 1, §17.16.22)

Attributes	Description
id (Hyperlink Target) Namespace: .../officeDocument /2006/relationships	The same as the id attribute in Part 1, §17.16.22.

14.10 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/wordprocessingml/2006/main> namespace is used for documents of a transitional conformance class.

14.10.1 Additional member types for the union in ST_DecimalNumberOrPercent (Part 1, §17.18.11)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_UnqualifiedPercentage simple type (§14.10.10).

14.10.2 Additional enumeration values for ST_Jc (Part 1, §17.18.44)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
left (Align to Leading Edge)	Semantically equivalent to start.
right (Align to Trailing Edge)	Semantically equivalent to end.

14.10.3 Additional enumeration values for ST_JcTable (Part 1, §17.18.45)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
left (Align to Starting Edge)	Specifies that the table shall be aligned to the leading edge of the text flow – the left text margin (for a left-to-right table); or the right text margin (for a right-to-left table) in the document. (See Part 1, §17.4.1)

Enumeration Value	Description
right (Align to Trailing Edge)	Specifies that the table shall be aligned to the trailing edge of the text flow – the right text margin (for a left-to-right table); or the left text margin (for a right-to-left table) in the document. (See Part 1, §17.4.1)

14.10.4 Additional enumeration values for ST_NumberFormat (Part 1, §17.18.59)

Enumeration Value	Description
decimalFullWidth2 (Full Width Arabic Numerals Alternate)	<p>Specifies that the sequence shall consist of a set of full-width Arabic numbering.</p> <p>To determine the text that is displayed for any value, this sequence specifies a set of characters that represent positions 1–9 and then those same characters are combined with each other and 0 (represents the number zero) to construct the remaining values.</p> <p>The set of characters used by this numbering format for values 0–9 is U+FF10–U+FF19, respectively.</p> <p>For values greater than the size of the set, the number is constructed by following these steps:</p> <ol style="list-style-type: none"> 1. Divide the value by 10 and write the symbol which represents the remainder. 2. Divide the quotient of the previous division by 10 and write the symbol, which represents the remainder, to the left of the existing position. 3. Repeat step 2 until the remaining value is equal to zero. <p>[<i>Example:</i> The numbering for the items should be represented by the following pattern: 1, 2, 3, ..., 8, 9, 10, 11, 12, ..., 18, 19, 20, 21, ... <i>end example</i>]</p>

14.10.5 Additional enumeration values for ST_StyleSort (Part 1, §17.18.82)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
0000 (Sort by Style Name)	Specifies that styles which are visible should be sorted by their names.
0001 (Sort by Style Priority)	Specifies that styles which are visible should be sorted by their UI priority using the uiPriority element (Part 1, §17.7.4.19).
0002 (Sort by Default Method)	Specifies that styles which are visible should be sorted

Enumeration Value	Description
	by the default sorting of the host application.
0003 (Sort by Font)	Specifies that styles which are visible should be sorted by the font which they apply.
0004 (Sort by Based On Style)	Specifies that styles which are visible should be sorted by the style on which they are based using the basedOn element (Part 1, §17.7.4.3).
0005 (Sort by Style Type)	Specifies that styles which are visible should be sorted by their style types (i.e. character, linked, paragraph).

14.10.6 Additional enumeration values for ST_TabJc (Part 1, §17.18.84)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
left (Leading Tab)	Semantically equivalent to start.
right (Trailing Tab)	Semantically equivalent to end.

14.10.7 Additional enumeration values for ST_TextDirection (Part 1, §17.18.93)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
btLr (Lines Flow From Left to Right)	Semantically equivalent to lr.
lrTb (Lines Flow From Top To Bottom)	Semantically equivalent to tb.
lrTbV (Lines Flow From Top to Bottom, Rotated)	Semantically equivalent to tbV.
tbLrV (Lines Flow From Left to Right, Rotated)	Semantically equivalent to lrV.
tbRl (Lines Flow From Right to Left)	Semantically equivalent to rl.
tbRlV (Lines Flow From Right to Left, Rotated)	Semantically equivalent to rlV.

14.10.8 Additional member types for the union in ST_TextScale (Part 1, §17.18.95)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

The ST_TextScaleDecimal simple type (§9.10.11).

14.10.9 ST_Cnf (Conditional Formatting Bitmask)

This simple type specifies the format for the set of conditional formatting properties that have been applied to this object.

These properties are expressed using a string serialization of a binary bitmask for each of the following properties (reading from the first character position right):

- First Row - Is this the first row of the table?
- Last Row - Is this the last row of the table?
- First Column - Does this belong to the first column of the table?
- Last Column - Does this belong to the last column of the table?
- Band 1 Vertical - Does this belong to a column which should receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered columns (e.g. 1,3,5,...)
- Band 2 Vertical - Does this belong to a column which should receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered columns (e.g. 2,4,6...)
- Band 1 Horizontal - Does this receive band 1 formatting? This property specifies whether the cell should receive the formatting specified for odd-numbered rows (e.g. 1,3,5,...)
- Band 2 Horizontal - Does this receive band 2 formatting? This property specifies whether the cell should receive the formatting specified for even-numbered rows (e.g. 2,4,6...)
- NE Cell - Is this part of the top-right corner of the table?
- NW Cell - Is this part of the top-left corner of the table?
- SE Cell - Is this part of the bottom-right corner of the table?
- SW Cell - Is this part of the bottom-left corner of the table?

For each of these properties, a value of 1 in the specified character position in the string means that the value is true, a value of 0 means false. All values shall be specified.

[*Example:* Consider a paragraph in the top right corner of a table with a table style applied. This paragraph would need to specify the following WordprocessingML:

```
<w:p>
  <w:pPr>
    <w:cnfStyle w:val="101000000100" />
    ...
  </w:pPr>
  ...
</w:p>
```

This paragraph specifies that it has the conditional properties from the table style for the first column, first row, and the NW corner of the parent table by setting the appropriate bits in the val attribute. *end example*]

This simple type's contents are a restriction of the W3C XML Schema string datatype.

This simple type also specifies the following restrictions:

- This simple type's contents have a length of exactly 12 characters.

- This simple type's contents shall match the following regular expression pattern: `[01]*`.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_Cnf](#)) is located in §A.1. *end note*]

14.10.10 ST_UnqualifiedPercentage (Percentage Value Without Percent Sign)

This simple type specifies additional formats for percentage-based values which can only be used within the transitional conformance class.

Specifically, this value allows percentage-based values to be specified as follows:

- For the `w` attribute in `CT_TblWidth` (Part 1, §17.4.88), the value is stored in 50ths of a percent.
- For all other uses, the value is stored in whole percentage points.

[*Example:* Consider the following WordprocessingML fragment:

```
<w:tblW w:w="1000" w:type="pct" />
```

The `tblW` element is based on the `CT_TblWidth` complex type, and the type attribute's value is `pct`, which means that this value is measured in 50ths of a percent (i.e. 1000 is equal to 20%). *end example*]

This simple type's contents are a restriction of the W3C XML Schema integer datatype.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_UnqualifiedPercentage](#)) is located in §A.1. *end note*]

14.10.11 ST_TextScaleDecimal (Text Expansion/Compression Percentage)

This simple type specifies that the percentage by which the contents of a run shall be expanded or compressed with respect to its normal (100%) character width, with a minimum width of 1% and maximum width of 600%.

[*Example:* Consider a run of text which must be expanded to 300% when displaying each character within the contents of the run. This constraint is specified using the following WordprocessingML:

```
<w:rPr>
  <w:w w:val="300"/>
</w:rPr>
```

This run explicitly declares that the `w` value is 300, so the contents of this run appear at 300% of their normal character width by expanding the width of each character. *end example*]

This simple type's contents are a restriction of the W3C XML Schema integer datatype.

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to 0.
- This simple type has a maximum value of less than or equal to 600.

14.11 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §17, “WordprocessingML”, have different source relationships when used in documents of the Transitional conformance class:

14.11.1 Changed attribute for contentPart element (Part 1, §17.3.3.2)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.2 Changed attribute for control element (Part 1, §17.3.3.3)

Attributes	Description
id (Embedded Control Properties Relationship Reference) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID for the relationship that contains the properties for this embedded control. This property bag is contained in a separate part within the Office Open XML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/control</code> or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then the embedded control shall be given no property bag when instantiated.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for an embedded control in a document:</p> <pre><w:control r:id="rId5" w:name="CheckBox1" w:shapeid="_x0000_s1027" /></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rId5 must contain the property data for this embedded control. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.3 Changed attribute for movie element (Part 1, §17.3.3.17)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> <code>http://schemas.openxmlformats.org/officeDocument/2006/customXml</code> for the contentPart element <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer</code> for the footerReference element <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/header</code> for the headerReference element <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font</code> for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings</code> for the printerSettings element <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink</code> for the longDesc or hyperlink element <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p>

Attributes	Description
	<p data-bbox="451 285 727 315"><... r:id="rId1" /></p> <p data-bbox="414 354 1463 422">The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p data-bbox="414 462 1451 529">The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.4 Changed attribute for objectEmbed element (Part 1, §17.3.3.20)

Attributes	Description
<p data-bbox="139 674 363 774">id (Relationship to Embedded Object Data)</p> <p data-bbox="139 814 375 951">Namespace: .../officeDocument /2006/relationships</p>	<p data-bbox="414 674 1451 741">Specifies the relationship ID for the relationship that targets the Embedded Object Part containing the embedded object data.</p> <p data-bbox="414 781 1372 882">The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/oleObject or the document shall be considered non-conformant.</p> <p data-bbox="414 921 1284 951">[Example: Consider an XML element which has the following id attribute:</p> <p data-bbox="451 991 727 1020"><... r:id="rId1" /></p> <p data-bbox="414 1060 1446 1127">The markup specifies the associated relationship part with relationship ID rId1 targets the part containing the corresponding embedded object information. <i>end example</i>]</p> <p data-bbox="414 1167 1451 1234">The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.5 Changed attribute for objectLink element (Part 1, §17.3.3.21)

Attributes	Description
id (Relationship to Embedded Object Data) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID for the relationship that targets the Embedded Object Part containing the embedded object data.</p> <p>The specified relationship shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/oleObject</code> or the document shall be considered non-conformant.</p> <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 targets the part containing the corresponding embedded object information. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.6 Changed attribute for bottom element (Part 1, §17.6.2)

Attributes	Description
bottomLeft (Custom Defined Bottom Left Border Relationship Reference) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID for the relationship that contains the custom bottom left border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/image</code> or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom bottom left border shall be used.</p> <p>[<i>Example</i>: Consider the following WordprocessingML markup for a custom bottom left border in a document:</p> <pre><w:bottom w:val="custom" r:bottomLeft="rIdCustomBottomLeftBorder" .../></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomBottomLeftBorder must contain the custom bottom left border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
bottomRight (Custom Defined	<p>Specifies the relationship ID for the relationship that contains the custom bottom right border image for the parent element. This custom border image is contained in a</p>

Attributes	Description
<p>Bottom Right Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom bottom right border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom bottom right border in a document:</p> <pre><w:bottom w:val="custom" r:bottomRight="rIdCustomBottomRightBorder" .../></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomBottomRightBorder must contain the custom bottom right border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
<p>id (Custom Defined Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID for the relationship that contains the custom border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom bottom border in a document:</p> <pre><w:bottom w:val="custom" r:id="rIdCustomBottomBorder" .../></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomBottomBorder must contain the custom bottom border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.7 Changed attribute for left element (Part 1, §17.6.7)

Attributes	Description
id (Custom Defined	Specifies the relationship ID for the relationship that contains the custom border image

Attributes	Description
<p>Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom bottom border in a document:</p> <pre><w:bottom w:val="custom" r:id="rIdCustomBottomBorder" .../></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomBottomBorder must contain the custom bottom border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.8 Changed attribute for printerSettings element (Part 1, §17.6.14)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains</p>

Attributes	Description
	<p>the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.9 Changed attribute for right element (Part 1, §17.6.15)

Attributes	Description
<p>id (Custom Defined Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID for the relationship that contains the custom border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom bottom border in a document:</p> <pre><w:bottom w:val="custom" r:id="rIdCustomBottomBorder" .../></pre> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomBottomBorder must contain the custom bottom border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.10 Changed attribute for top element (Part 1, §17.6.21)

Attributes	Description
<p>id (Custom Defined Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID for the relationship that contains the custom border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom bottom border in a document:</p>

Attributes	Description
	<p><code><w:bottom w:val="custom" r:id="rIdCustomBottomBorder" .../></code></p> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID <code>rIdCustomBottomBorder</code> must contain the custom bottom border image for the document. <i>end example</i></p> <p>The possible values for this attribute are defined by the <code>ST_RelationshipId</code> simple type (Part 1, §22.8.2.1).</p>
<p><code>topLeft</code> (Custom Defined Top Left Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID for the relationship that contains the custom top left border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/image</code> or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom top left border shall be used.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom top left border in a document:</p> <p><code><w:top w:val="custom" r:topLeft="rIdCustomTopLeftBorder" .../></code></p> <p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID <code>rIdCustomTopLeftBorder</code> must contain the custom top left border image for the document. <i>end example</i></p> <p>The possible values for this attribute are defined by the <code>ST_RelationshipId</code> simple type (Part 1, §22.8.2.1).</p>
<p><code>topRight</code> (Custom Defined Top Right Border Relationship Reference)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID for the relationship that contains the custom top right border image for the parent element. This custom border image is contained in a separate part within the WordprocessingML package.</p> <p>The relationship explicitly targeted by this attribute shall be of type <code>http://schemas.openxmlformats.org/officeDocument/2006/relationships/image</code> or the document shall be considered non-conformant.</p> <p>If this attribute is omitted, then no custom top right border shall be used when the parent element is instantiated.</p> <p>[<i>Example:</i> Consider the following WordprocessingML markup for a custom top right border in a document:</p> <p><code><w:top w:val="custom" r:topRight="rIdCustomTopRightBorder" ... /></code></p>

Attributes	Description
	<p>The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rIdCustomTopRightBorder must contain the custom top right border image for the document. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.11 Changed attribute for embedBold element (Part 1, §17.8.3.3)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.12 Changed attribute for embedBoldItalic element (Part 1, §17.8.3.4)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element

Attributes	Description
ps	<p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.13 Changed attribute for embedItalic element (Part 1, §17.8.3.5)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships ps	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

14.11.14 Changed attribute for embedRegular element (Part 1, §17.8.3.6)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.15 Changed attribute for footerReference element (Part 1, §17.10.2)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header

Attributes	Description
	<p>r for the headerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.16 Changed attribute for headerReference element (Part 1, §17.10.5)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type</p>

Attributes	Description
	(Part 1, §22.8.2.1).

14.11.17 Changed attribute for dataSource element (Part 1, §17.14.9)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.18 Changed attribute for headerSource element (Part 1, §17.14.16)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font

Attributes	Description
	<p>for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.19 Changed attribute for recipientData element (Part 1, §17.14.28)

Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.20 Changed attribute for src element (Part 1, §17.14.30)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[Example: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.21 Changed attribute for attachedTemplate element (Part 1, §17.15.1.6)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink

Attributes	Description
	<p>link for the longDesc or hyperlink element</p> <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.22 Changed attribute for saveThroughXslt element (Part 1, §17.15.1.76)

Attributes	Description
id (XSL Transformation Location) Namespace: .../officeDocument/2006/relationships	<p>Specifies an explicit relationship to the location of the XSL Transformation which shall be applied.</p> <p>The relationship targeted by this element shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/transform, or this document shall be declared non-conformant.</p> <p>[<i>Example:</i> Consider a XML document that must have the XSL transform located at c:\Example Transform.xslt applied when the document is saved as a single XML file. This requirement would be specified using the following WordprocessingML in the document settings:</p> <pre><w:saveThroughXslt r:id="rId5" /></pre> <p>The saveThroughXslt element specifies that the relationship located at rId5 must be used when saving as a single XML file in this case, that relationship must target c:\Example Transform.xslt. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.23 Changed attribute for longDesc element (Part 1, §17.15.2.23)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer</p>

Attributes	Description
	<p>for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.24 Changed attribute for sourceFileName element (Part 1, §17.15.2.39)

Attributes	Description
id (Relationship to Part)	Specifies the relationship ID to a specified part.
Namespace: .../officeDocument/2006/relationships	<p>The specified relationship shall match the relationship type required by the parent element:</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/footer for the footerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element</p> <p>http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element</p> <p>[<i>Example:</i> Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

14.11.25 Changed attribute for subDoc element (Part 1, §17.17.1.1)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part.</p> <p>The specified relationship shall match the relationship type required by the parent element:</p> <ul style="list-style-type: none"> http://schemas.openxmlformats.org/officeDocument/2006/customXml for the contentPart element http://schemas.openxmlformats.org/officeDocument/2006/relationships/ for the footerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/header for the headerReference element http://schemas.openxmlformats.org/officeDocument/2006/relationships/font for the embedBold, embedBoldItalic, embedItalic, or embedRegular elements http://schemas.openxmlformats.org/officeDocument/2006/relationships/printerSettings for the printerSettings element http://schemas.openxmlformats.org/officeDocument/2006/relationships/hyperlink for the longDesc or hyperlink element <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

14.11.26 Changed attribute for altChunk element (Part 1, §17.17.2.1)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a specified part containing alternate content for import.</p> <p>If the specified relationship does not match the relationship type required by the parent element, then this document shall be considered to be non-conformant.</p> <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre>

Attributes	Description
	<p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (§Part 1, §22.8.2.1).</p>

15. SpreadsheetML Reference Material

[Note: For further information on the mapping of elements and attributes to OPC parts, see the Bibliography entry, “Information on elements, attributes, and OPC parts in ECMA-376 (OOXML)”. *end note*]

15.1 Table of Contents

This subclause is informative.

15.2	Workbook	206
15.2.1	Additional attribute for fileSharing element (Part 1, §18.2.12)	206
15.2.2	Additional attribute for webPublishing element (Part 1, §18.2.24)	206
15.2.3	Additional attributes for workbookProtection element (Part 1, §18.2.29)	206
15.2.4	Modified content for Date Conversion for Serial Date-Times (Part 1, §18.17.4.1)	212
15.3	Worksheets	213
15.3.1	Worksheets	213
15.3.1.1	legacyDrawing (Legacy Drawing Reference)	213
15.3.1.2	legacyDrawingHF (Legacy Drawing Reference in Header Footer)	214
15.3.1.3	Additional attribute for dataConsolidate element (Part 1, §18.3.1.29)	214
15.3.1.4	Additional attributes for protectedRange element (Part 1, §18.3.1.71)	214
15.3.1.5	Additional attribute for sheetProtection element (Part 1, §18.3.1.84)	215
15.3.1.6	Additional attribute for sheetProtection element (Part 1, §18.3.1.85)	215
15.3.2	AutoFilter Settings	216
15.3.2.1	Attributes with modified descriptions for dynamicFilter element (Part 1, §18.3.2.5)	216
15.4	Styles	217
15.4.1	left (Leading Edge Border)	217
15.4.2	right (Trailing Edge Border)	217
15.5	Pivot Tables	217
15.5.1	Pivot Tables	217
15.5.1.1	Additional attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)	217
15.6	External Data Connections	218
15.6.1	Additional attribute for textPr element (Part 1, §18.13.12)	218
15.7	Simple Types	218
15.7.1	Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)	218
15.7.2	ST_UnsignedShortHex (Unsigned Short Hex)	218

End of informative text.

15.2 Workbook

15.2.1 Additional attribute for fileSharing element (Part 1, §18.2.12)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
reservationPassword (Write Reservation Password)	<p>Specifies the legacy hash of the password required for editing this workbook.</p> <p>The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2).</p>

15.2.2 Additional attribute for webPublishing element (Part 1, §18.2.24)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
codePage (Code Page)	<p>This attribute is used only for compatibility with the existing corpus of binary documents, and is ignored if the characterSet attribute is present. Specifies the encoding the application uses when a Web page is saved. A code page is a table that relates the binary character codes used by a program to keys on the keyboard or to the appearance of characters on the display. Code pages are a means of providing support for the languages used in different countries.</p> <p>[Note: There are a number of code page technologies. One example of potential values can be found at: http://www.unicode.org/Public/MAPPINGS/ end note]</p> <p>The default value for this attribute is the workbook's encoding.</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

15.2.3 Additional attributes for workbookProtection element (Part 1, §18.2.29)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
revisionsPassword (Legacy Revisions Password)	<p>Specifies the legacy hash of the password required for unlocking revisions in this workbook. The hash is generated from an 8-bit wide character. The input string shall be in UTF-16LE format (if there is a leading BOM character (U+FEFF) in the encoded password it is removed before hash calculation), and these 16-bit Unicode characters shall be converted down to 8 bits before the hash is computed, using the following logic:</p> <p>[Note: This legacy conversion attempts to fit UTF-16 encoded characters into a single-</p>

Attributes	Description
	<p>byte character set. As such, if the input string uses characters from multiple character sets, many characters are unmapped in the destination character set and take on the default value, 0x3F. For this reason, it is recommended that applications choose a character set which maps the maximum number of characters from the input string and explicitly declare the character set used in the revisionsCharacterSet attribute. Not doing so will inhibit interoperability. <i>end note</i>]</p> <p>For SpreadsheetML password hash purposes, Unicode UTF-16 input code points are converted to a single or double byte character set.</p> <p>Code points with no representation in the target character set are replaced with Unicode character 0x3f (?).</p> <p>The values permitted by this attribute are names and aliases listed in the IANA character set listing found at http://www.iana.org/assignments/character-sets. For single byte character sets, each Unicode code point is replaced by a single byte or 0x3f if an appropriate character doesn't exist in the character set.</p> <p>For double byte character sets, each Unicode code point is replaced by either a single byte, or a two byte sequence, depending on the input character, or 0x3f if an appropriate character doesn't exist in the character set. In our tables the target is a single byte sequence if the most significant byte is 0x00, otherwise it is a double byte sequence, with the lead byte being the most significant byte.</p> <p>To convert, first check if conversion is being done to a single or double byte code page and load the appropriate WCTABLE code page table.</p> <p>For each input character, look up the code point in the WCTABLE. There are 3 possibilities: Not found, single byte, or double byte.</p> <ul style="list-style-type: none"> • If the input character is not found, append 0x3f and continue to the next character. • If the result is a single byte, check to make sure the entry in the MBTABLE matches the input. If it matches, append the single byte to the output. If it does not match, append 0x3f to the output. • If the result is a double byte, check to make sure the entry in the DBCSEENTRY table for the appropriate lead byte matches the input character. If it matches, append the lead byte and trail byte to the output. If it does not match, append 0x3f to the output. <p>The following pseudocode describes how this conversion should be done:</p> <pre> int WideCharToMultiByte(wchar_t* wszInput, byte* szOutput) { // Remember output start so we can return length byte* szOutputStart = szOutput; </pre>

Attributes	Description
	<pre> // Load Character Set Tables and determine // double/single byte nature. // This will depend on how the character sets are represented on // the target machine. TABLECLASS represents some abstract // representation of this structure here. TABLECLASS pTables = LoadCharacterSetTables(); Bool bDoubleByte = IsCharacterSetDoubleByte(); while (*wszInput != 0) { if (bDoubleByte) szOutput = AppendDoubleByte(pTables, *wszInput, szOutput); else szOutput = AppendSingleByte(pTables, *wszInput, szOutput); // Read next input wchar_t wszInput++; } // Null terminate the output *szOutput = 0; // Return output length return szOutput - szOutputStart; } byte* AppendSingleByte(TABLECLASS pTables, wchar_t wcIn, byte* szOutput) { // Look up byte that we want to append. byte bOut = pTables->LookupSingleByte(wcIn); // Make sure that bOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookupWideChar(bOut)) bOut = 0x3f; *szOutput = bOut; szOutput++; return szOutput; } </pre>

Attributes	Description
	<pre> byte* AppendDoubleByte(TABLECLASS pTables, wchar_t wcIn, byte* szOutput) { // Look up bytes that we want to append. UINT16 bytesOut = pTables->LookUpDoubleByte(wcIn); // See if it is a single or double byte sequence if (bytesOut & 0xFF00) { // It is a double byte sequence // Make sure that bytesOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookUpWideChar(bytesOut)) { // Use ?, it will be added below bytesOut = 0x003f; } else { // It matched, use the lead byte we found // trail byte will be added below *szOutput = bytesOut >> 8; szOutput++; } } else { // It is a single byte sequence // Make sure that bytesOut matches the input, otherwise use ? // (ie: no best fit behavior allowed) if (wcIn != pTables->LookUpWideChar(bytesOut & 0xFF)) bytesOut = 0x003f; } // Add the single or trail byte *szOutput = bytesOut & 0xFF; szOutput++; return szOutput; } class pTables { // Construction depends on how you choose to store & load the // table files </pre>

Attributes	Description
	<pre> byte LookUpSingleByte(wchar_t wcIn) { // How you access the table depends on your storage mechanism. // Look up the line in WCTABLE where the first column matches wcIn, // and then return the byte value from the second column. if (exists WCTABLE{wcIn}) return WCTABLE{wcIn}.SecondColumn; // If it doesn't exist, return ? return 0x3f; } UINT16 LookUpDoubleByte(wchar_t wcIn) { // How you access the table depends on your storage mechanism. // Look up the line in WCTABLE where the first column matches wcIn, // and then return the double byte value from the second column. if (exists WCTABLE{wcIn}) return WCTABLE{wcIn}.SecondColumn; // If it doesn't exist, return ? return 0x003f; } // Overload that looks up wide chars from single byte code points. wchar_t LookUpWideChar(byte bIn) { // How you access the table depends on your storage mechanism. // Look up the line in MBTABLE where the first column matches bIn, // and then return the wchar_t value from the second column. if (exists MBTABLE{bIn}) return MBTABLE{bIn}.SecondColumn; // If it doesn't exist, return ? return 0x003f; } // Overload that looks up wide chars from double byte code </pre>

Attributes	Description
	<pre> points wchar_t LookUpWideChar(UINT16 bytesIn) { // How you access the table depends on your storage mechanism. // First find the DBCSTABLE where the LeadByte matches // the lead (most significant) input byte. if (exists DBCSTABLE{bytesIn >> 8}) { DbcTable = DBCSTABLE{bytesIn >> 8}; // Look up the line in DbcTable where the first column // matches the input trail (least significant) byte, // and then return the wchar_t value from the second column. if (exists DbcTable{bytesIn & 0xFF}) return DbcTable{bytesIn & 0xFF}.SecondColumn; } // Either the lead byte table or specific trail byte // doesn't exist in the table, return ? return 0x003f; } } </pre> <p>The resulting value is hashed using the low-order word algorithm defined in §14.7.1. This step assumes that all words are unsigned, the word size is two bytes, and that bit-level shift-left/shift-right operations shift in the direction of the highest-order and lowest-order bit, respectively. [Example: 0x61 SHR 1 is 0xC2, as 01100001 shifted one position in the direction of its highest-order bit is 11000010. end example]</p> <p>[Example: This algorithm can be represented by the following pseudocode:</p> <pre> // Function Input: // szPassword: NULL terminated C-Style string // cchPassword: The number of characters in szPassword (not // including the NULL terminator) unsigned_short GetPasswordHash(const char *szPassword, int cchPassword) { unsigned_short wPasswordHash; const char *pch; wPasswordHash = 0; if (cchPassword > 0) { </pre>

Attributes	Description
	<pre> pch = &szPassword[cchPassword]; while (pch-- != szPassword) { wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); wPasswordHash ^= *pch; } wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); wPasswordHash ^= cchPassword; wPasswordHash ^= (0x8000 ('N' << 8) 'K'); } return(wPasswordHash); } end example] The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2). </pre>
revisionsPassword CharacterSet (Revisions Password Character Set)	Name of the character set associated with the legacy revisionsPassword hash. The values permitted by this attribute are names and aliases listed in the IANA CHARACTER SETS listing found at http://www.iana.org/assignments/character-sets . <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
workbookPasswor d (Legacy Workbook Password)	Specifies the legacy hash of the password required for unlocking revisions in this workbook. <p>The hash is generated using the logic defined in the preceding revisionsPassword attribute.</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2).</p>
workbookPasswor dCharacterSet (Workbook Password Character Set)	Name of the character set associated with the workbookPassword hash. The values permitted by this attribute are the names and aliases listed in the IANA CHARACTER SETS listing found at http://www.iana.org/assignments/character-sets . <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

15.2.4 Modified content for Date Conversion for Serial Date-Times (Part 1, §18.17.4.1)

When interpreting a document of a transitional conformance class, Part 1, §18.17.4.1 is replaced by the following text:

A *serial date-time* is a number that represents a date and time. This signed value is in units of days relative to the base date for the selected date system. Serial date-times increase by 1 into each successive day, and decrease by 1 into each preceding day. Fractional portions of serial date-times represent fractions of a single day. [Example: When using the 1900 date system, which has a base date of 30th December 1899, a serial date-time of 1.5 represents midday on the 31st December 1899 (serial date-time day 1); that is, 1899-12-31T12:00. A serial date-time of -4.25 represents 6 pm on the 25th December 1899; that is, 1899-12-25T18:00. end example] The base dates and the related serial date-times represent local date and time.

Two different bases are used for converting dates to and from serial date-times:

- In the *1900 date system*, the lower limit is January 1, 1900, 00:00:00, which has a serial date-time of 1. The upper limit is December 31, 9999, 23:59:59, which has a serial date-time of 2,958,465.9999884. The base date for this date base system is December 31, 1899, which has a serial date-time of 0.
- In the *1904 date system*, the lower limit is January 1st, 0001, 00:00:00, which has a serial date-time of -695055. The upper limit is December 31st, 9999, 23:59:59.999, which has a serial date-time of 2,957,003.9999884. The base date for this system is midnight (00:00:00) on the morning of January 1st, 1904, which has a serial date-time of 0.

A serial date-time outside the temporal range for the selected date system is invalid.

The date system is specified by the value of the date1904 attribute of the workbookPr element. [Example:

1900 date system: <workbookPr showObjects="all"/>

1904 date system: <workbookPr date1904="1" showObjects="all"/>

end example]

15.3 Worksheets

15.3.1 Worksheets

15.3.1.1 legacyDrawing (Legacy Drawing Reference)

This element is present when the sheet contains drawing shapes defined by VML. In this case, the element contains an explicit relationship whose ID points to the part containing the VML definitions.

[Example:

<drawing r:id="rId1"/>

end example]

Attributes	Description
id (Relationship Id)	This value references a relationship Id for the sheet. The relationship shall point to the part containing the VML definition.
Namespace: .../officeDocument	The possible values for this attribute are defined by the ST_RelationshipId simple type

Attributes	Description
/2006/relationships	(Part 1, §22.8.2.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_LegacyDrawing](#)) is located in §A.2.
end note]

15.3.1.2 legacyDrawingHF (Legacy Drawing Reference in Header Footer)

This element specifies the explicit relationship to the part containing the VML defining pictures rendered in the header / footer of the sheet.

Attributes	Description
id (Relationship Id)	This value references a relationship Id for the sheet. The relationship shall point to the part containing the VML definition.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_LegacyDrawing](#)) is located in §A.2.
end note]

15.3.1.3 Additional attribute for dataConsolidate element (Part 1, §18.3.1.29)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
leftLabels (Starting Column Labels)	Semantically equivalent to startLabels. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

15.3.1.4 Additional attributes for protectedRange element (Part 1, §18.3.1.71)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Legacy Password)	Specifies the legacy hash of the password required for editing this range. The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29). The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2).

Attributes	Description
securityDescriptor (Security Descriptor)	<p>Optional setting to specify the relative security descriptor. The security descriptor defines user accounts who can edit this range without providing a password to access the range.</p> <p>The format of a securityDescriptor is application defined; however, it is recommended that the following format be used for interoperability between implementations:</p> <ul style="list-style-type: none"> <code>username@domain</code> <p>If multiple user accounts are specified in the securityDescriptor attribute, each account shall be delimited by parentheses.</p> <p>[Example: This example demonstrates two user accounts in the security descriptor attribute:</p> <pre><protectedRanges> <protectedRange sqref="A1:C5" name="Range1" securityDescriptor="(user1@iso.org)(user2@iso.org)"/> </protectedRanges></pre> <p>end example]</p> <p>If an application is unable to resolve the meaning of the securityDescriptor, it shall treat the attribute as if it had been removed.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

15.3.1.5 Additional attribute for sheetProtection element (Part 1, §18.3.1.84)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Password)	<p>Specifies the hash of the password required for editing this chart sheet.</p> <p>The hash is generated using the logic defined in the revisionPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2).</p>

15.3.1.6 Additional attribute for sheetProtection element (Part 1, §18.3.1.85)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
password (Legacy Password)	Specifies the legacy hash of the password required for editing this worksheet.

Attributes	Description
	<p>The hash is generated using the logic defined in the revisionsPassword attribute of the workbookProtection element (Part 1, §18.2.29).</p> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§15.7.2).</p>

15.3.2 AutoFilter Settings

15.3.2.1 Attributes with modified descriptions for dynamicFilter element (Part 1, §18.3.2.5)

The following attributes have modified descriptions when specified for a document of a transitional conformance class:

Attributes	Description
maxVal (Max Value)	<p>A maximum value for dynamic filter. maxVal/maxValIso shall be required for today, yesterday, tomorrow, nextWeek, thisWeek, lastWeek, nextMonth, thisMonth, lastMonth, nextQuarter, thisQuarter, lastQuarter, nextYear, thisYear, lastYear, and yearToDate.</p> <p>The above criteria are based on a value range. [<i>Example: If today's date is September 22nd, then the range for thisWeek is the values greater than or equal to September 17 and less than September 24. end example</i>] In the thisWeek range, the lower value is expressed using val or valIso. The higher value is expressed using maxVal or maxValIso.</p> <p>These dynamic filter shall not require val/valIso or maxVal/maxValIso: Q1, Q2, Q3, Q4, M1, M2, M3, M4, M5, M6, M7, M8, M9, M10, M11 and M12.</p> <p>The above criteria shall not specify the range using val/valIso and maxVal/maxValIso because Q1 always starts from M1 to M3, and M1 is always January.</p> <p>These types of dynamic filters shall use val and shall not use maxVal/maxValIso: aboveAverage and belowAverage.</p> <p>If maxValIso and maxVal are both present, maxValIso shall take precedence.</p> <p>The possible values for this attribute are defined by the W3C XML Schema double datatype.</p>
val (Value)	<p>A minimum numeric or serial date value for dynamic filter. (See description of ValIso to understand when val is required.)</p> <p>If valIso and val are both present, valIso shall take precedence.</p> <p>The possible values for this attribute are defined by the W3C XML Schema double datatype.</p>
valIso (ISO Value)	<p>A minimum date value for dynamic filter. (See description of maxVal/maxValIso to</p>

Attributes	Description
	understand when val/valIso is required.) The possible values for this attribute are defined by the W3C XML Schema dateTime datatype.

15.4 Styles

15.4.1 left (Leading Edge Border)

Semantically equivalent to start (Part 1, §18.8.37).

Attributes	Description
style (Line Style)	The line style for this border. The possible values for this attribute are defined by the ST_BorderStyle simple type (Part 1, §18.18.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_BorderPr](#)) is located in §A.2. *end note*]

15.4.2 right (Trailing Edge Border)

Semantically equivalent to end (Part 1, §18.8.16).

Attributes	Description
style (Line Style)	The line style for this border. The possible values for this attribute are defined by the ST_BorderStyle simple type (Part 1, §18.18.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_BorderPr](#)) is located in §A.2. *end note*]

15.5 Pivot Tables

15.5.1 Pivot Tables

15.5.1.1 Additional attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
refreshedDate (PivotCache Last	Specifies the date when the cache was last refreshed. This attribute depends on whether the application exposes mechanisms via the user interface whereby the end-user can

Attributes	Description
Refreshed Date)	<p>refresh the cache.</p> <p>If refreshedDateIso and refreshedDate are both present, refreshedDateIso shall take precedence.</p> <p>The possible values for this attribute are defined by the W3C XML Schema double datatype.</p>

15.6 External Data Connections

15.6.1 Additional attribute for textPr element (Part 1, §18.13.12)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
codePage (Code Page)	<p>Code page associated with the text file. This attribute is used only for backwards compatibility, and is ignored if the characterSet attribute is present.</p> <p>[Note: There are a number of code page technologies. One example of potential values can be found at: http://www.unicode.org/Public/MAPPINGS end note]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

15.7 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/spreadsheetml/2006/main> namespace is used for documents of a transitional conformance class.

15.7.1 Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)

The following additional enumeration values can be specified for a document of a transitional conformance class.

Enumeration Value	Description
topRight (Top Corner, Trailing Edge)	Semantically equivalent to topEnd.

15.7.2 ST_UnsignedShortHex (Unsigned Short Hex)

This simple type defines the Hex representation of an unsigned short.

This simple type's contents are a restriction of the W3C XML Schema hexBinary datatype.

This simple type also specifies the following restrictions:

- This simple type's contents have a length of exactly 4 hexadecimal digit(s).

[Note: The W3C XML Schema definition of this simple type's content model ([ST_UnsignedShortHex](#)) is located in §A.2. *end note*]

15.7.3 Removed enumeration values for ST_CellType (Part 1, §18.18.11)

For transitional documents, the restriction on the simple type ST_CellType having the value “d” (ISO 8601 format) is removed.

15.8 Formulas

15.8.1 Attribute synonym for c element (Part 1, §18.6.1)

The following additional attribute can be specified for a document of a transitional conformance class:

Attributes	Description
ref (Cell Reference)	An A-1 style reference to a cell. The possible values for this attribute are defined by the ST_CellRef simple type (Part 1, §18.18.7).

This attribute is semantically equivalent to r (Part 1, §18.6.1).

Only one or the other of r and ref can be defined in any given instance.

15.8.2 Additional representation for dates and times (Part 1, §18.17.4)

For a document of a transitional conformance class, each unique instant in SpreadsheetML time shall be stored as an ISO 8601-formatted string or as a serial value.

15.9 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §18, “SpreadsheetML”, have different source relationships when used in documents of the Transitional conformance class:

15.9.1 Changed attribute for externalReference element (Part 1, §18.2.8)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Specifies a unique identifier that is used to identify a relationship to another part in the file. Relationship identifiers link the element definition with the part where data for the element is stored. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.2 Changed attribute for pivotCache element (Part 1, §18.2.17)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument	Specifies the identifier to a pivot cache definition part where cached data is stored. This attribute is required.

Attributes	Description
/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.3 Changed attribute for sheet element (Part 1, §18.2.19)

Attributes	Description
id (Relationship Id)	Specifies the identifier of the sheet part where the definition for this sheet is stored.
Namespace: .../officeDocument	This attribute is required.
/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.4 Changed attribute for control element (Part 1, §18.3.1.19)

Attributes	Description
id (Relationship Id)	This relationship ID references an Embedded Control Data part that contains control-specific properties and state information about this particular embedded control.
Namespace: .../officeDocument	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).
/2006/relationships	

15.9.5 Changed attribute for controlPr element (Part 1, §18.3.1.20)

Attributes	Description
id (Relationship ID for Embedded Control Properties)	Specifies the relationship ID for the relationship which contains the properties for this embedded control. This property bag is contained in a separate part within the package.
Namespace: .../officeDocument	The relationship explicitly targeted by this attribute shall be of relationship type http://schemas.openxmlformats.org/officeDocument/2006/relationships/control or the document shall be considered non-conformant.
/2006/relationships	If this attribute is omitted, then the embedded control shall be given no property bag when instantiated.
	[Example: Consider the following WordprocessingML markup for an embedded control in a document:
	<pre><w:control r:id="rId5" w:id="CheckBox1" w:name="CheckBox1" w:shapeid="_x0000_s1027" w:class="shape" w:w="145" w:h="28" w:align="left" /></pre>
	The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rId5 must contain the property data for this embedded control. <i>end example</i>

Attributes	Description
	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.6 Changed attribute for customPr element (Part 1, §18.3.1.22)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	This relationship references the binary part containing the specified custom properties. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.7 Changed attribute for dataRef element (Part 1, §18.3.1.30)

Attributes	Description
id (relationship Id) Namespace: .../officeDocument /2006/relationships	Used only when the source range is external to this workbook. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.8 Changed attribute for drawing element (Part 1, §18.3.1.36)

Attributes	Description
id (Relationship id) Namespace: .../officeDocument /2006/relationships	Relationship Id referencing a part containing DrawingML definitions for this worksheet. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.9 Changed attribute for drawingHF element (Part 1, §18.3.1.37)

Attributes	Description
id (Relationship ID for Embedded Control Properties) Namespace: .../officeDocument /2006/relationships	Specifies the relationship ID for the relationship to the DrawingML part that contains the drawing objects used in the header and footer. This DrawingML part is a separate part within the package. [Example: <div style="text-align: center;"> <code><drawingHF r:id="rId2" lho="7" lhf="6"/></code> </div> The id attribute in the relationship reference namespace specifies that the relationship with relationship ID rId5 must contain the drawing objects used in the header and

Attributes	Description
	<p>footer. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.9.10 Changed attribute for hyperlink element (Part 1, §18.3.1.47)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	<p>Relationship Id in this sheet's relationships part, expressing the target location of the resource.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.9.11 Changed attribute for objectPr element (Part 1, §18.3.1.56)

Attributes	Description
id (Relationship ID to Embedded Object Data) Namespace: .../officeDocument /2006/relationships	<p>Specifies the relationship ID for the relationship that targets the Embedded Object Part containing the embedded object data.</p> <p>The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/oleObject or the document shall be considered non-conformant.</p> <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 targets the part containing the corresponding embedded object information. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.9.12 Changed attribute for oleObject element (Part 1, §18.3.1.59)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	<p>Relationship Id of the relationship pointing to the object persistence part.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

15.9.13 Changed attribute for pageSetup element (Part 1, §18.3.1.63)

Attributes	Description
id (Id) Namespace: .../officeDocument /2006/relationships	Relationship Id of the devMode printer settings part. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.14 Changed attribute for pageSetup element (Part 1, §18.3.1.64)

Attributes	Description
id (Id) Namespace: .../officeDocument /2006/relationships	Relationship Id of the devMode printer settings part. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.15 Changed attribute for picture element (Part 1, §18.3.1.67)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Relationship Id pointing to the image part. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.16 Changed attribute for pivotSelection element (Part 1, §18.3.1.69)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	Relationship Id pointing to the particular PivotTable Part corresponding to this selection. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.17 Changed attribute for tablePart element (Part 1, §18.3.1.94)

Attributes	Description
id (Relationship Id) Namespace: .../officeDocument /2006/relationships	This relationship Id is used to locate a particular table definition part. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

Attributes	Description
ps	

15.9.18 Changed attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)

Attributes	Description
id (Relationship Identifier)	Specifies the unique identifier that corresponds to the related pivotCacheRecords part. See (Part 1, §18.10.1.68) for more information.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.19 Changed attribute for rangeSet element (Part 1, §18.10.1.79)

Attributes	Description
id (Relationship Id)	Specifies the unique identifier of the Workbook part where the range set is stored. See Workbook (Part 1, §18.2) for more information.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.20 Changed attribute for worksheetSource element (Part 1, §18.10.1.95)

Attributes	Description
id (Relationship Id)	Specifies the identifier to the Sheet part whose data is stored in the cache. See the Sheet section (Part 1, §18.2) for more information.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.21 Changed attribute for header element (Part 1, §18.11.1.1)

Attributes	Description
id (Relationship ID)	This is the ID that is used to find the corresponding log record of the changes made for this header.
Namespace: .../officeDocument /2006/relationships	Use the corresponding relationship expressed in the revisionHeaders part to locate the log record that lists the specific changes. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.22 Changed attribute for externalBook element (Part 1, §18.14.7)

Attributes	Description
id (Relationship to supporting book file path)	Relationship ID that references a link in the relationships collection. The target attribute in the associated relationship will specify the worksheet XML file in the current SpreadsheetML document ZIP archive that makes use of this externalbook.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

15.9.23 Changed attribute for oleLink element (Part 1, §18.14.11)

Attributes	Description
id (Object Link Relationship)	Relationship ID that references a link in the relationships collection. The target attribute in the associated relationship will specify the external file name used for this oleLink.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16. PresentationML Reference Material

[Note: For further information on the mapping of elements and attributes to OPC parts, see the Bibliography entry, “Information on elements, attributes, and OPC parts in ECMA-376 (OOXML)” . *end note*]

16.1 Table of Contents

This subclause is informative.

16.2	Presentation.....	227
16.2.1	Presentation Properties.....	227
16.2.1.1	htmlPubPr (HTML Publishing Properties)	227
16.2.1.2	webPr (Web Properties)	228
16.2.1.3	Additional attributes for modifyVerifier element (Part 1, §19.2.1.19).....	229
16.3	Slides	235
16.3.1	Embedded Objects.....	235
16.3.1.1	Additional attribute for control element (Part 1, §19.3.2.1)	235
16.3.1.2	Additional attribute for oleObj element (Part 1, §19.3.2.4)	235
16.4	Simple Types	236
16.4.1	ST_WebColorType (HTML Slide Navigation Control Colors).....	236
16.4.2	ST_WebEncoding (Web Encoding)	236
16.4.3	ST_WebScreenSize (HTML/Web Screen Size Target)	236
16.5	Changed attributes.....	237
16.5.1	Changed attribute for bold element (Part 1, §19.2.1.1).....	237
16.5.2	Changed attribute for boldItalic element (Part 1, §19.2.1.2)	238
16.5.3	Changed attribute for font element (Part 1, §19.2.1.13)	238
16.5.4	Changed attribute for handoutMasterId element (Part 1, §19.2.1.14).....	240
16.5.5	Changed attribute for italic element (Part 1, §19.2.1.16)	240
16.5.6	Changed attribute for notesMasterId element (Part 1, §19.2.1.20)	241
16.5.7	Changed attribute for notesSz element (Part 1, §19.2.1.22).....	241
16.5.8	Changed attribute for regular element (Part 1, §19.2.1.29)	242
16.5.9	Changed attribute for sld element (Part 1, §19.2.1.31)	242
16.5.10	Changed attribute for sldId element (Part 1, §19.2.1.33)	242
16.5.11	Changed attribute for sldMasterId element (Part 1, §19.2.1.36)	242
16.5.12	Changed attribute for SmartTags element (Part 1, §19.2.1.40).....	243
16.5.13	Changed attribute for gridSpacing element (Part 1, §19.2.2.3)	243
16.5.14	Changed attribute for origin element (Part 1, §19.2.2.9)	244
16.5.15	Changed attribute for sld element (Part 1, §19.2.2.14)	244
16.5.16	Changed attribute for bgRef element (Part 1, §19.3.1.3)	245
16.5.17	Changed attribute for blipFill element (Part 1, §19.3.1.4)	245
16.5.18	Changed attribute for clrMap element (Part 1, §19.3.1.6)	245
16.5.19	Changed attribute for cNvPicPr element (Part 1, §19.3.1.11).....	247

16.5.20	Changed attribute for cNvPr element (Part 1, §19.3.1.12)	247
16.5.21	Changed attribute for cNvSpPr element (Part 1, §19.3.1.13)	249
16.5.22	Changed attribute for contentPart element (Part 1, §19.3.1.14)	249
16.5.23	Changed attribute for custData element (Part 1, §19.3.1.17)	250
16.5.24	Changed attribute for grpSpPr element (Part 1, §19.3.1.23)	250
16.5.25	Changed attribute for sldLayoutId element (Part 1, §19.3.1.40)	250
16.5.26	Changed attribute for spPr element (Part 1, §19.3.1.44)	250
16.5.27	Changed attribute for tags element (Part 1, §19.3.1.47)	251
16.5.28	Changed attribute for xfrm element (Part 1, §19.3.1.53)	251
16.5.29	Changed attribute for control element (Part 1, §19.3.2.1)	252
16.5.30	Changed attribute for oleObj element (Part 1, §19.3.2.4)	252
16.5.31	Changed attribute for pos element (Part 1, §19.4.5)	252
16.5.32	Changed attribute for snd element (Part 1, §19.5.68)	253
16.5.33	Changed attribute for sndTgt element (Part 1, §19.5.70)	253

End of informative text.

16.2 Presentation

16.2.1 Presentation Properties

16.2.1.1 htmlPubPr (HTML Publishing Properties)

This element specifies the publishing properties to be used when publishing this presentation document to the HTML file format. The target output profile is identified by the contents of the target attribute.

Attributes	Description		
id (Publish Path)	Specifies the path that should be used when publishing.		
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).		
showSpeakerNotes (Show Speaker Notes)	Specifies whether to show speaker notes when publishing. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.		
target (Target Output Profile)	Specifies the version of HTML output targeted by the output of any web page produced by this document. This attribute shall only contain a string that represents an output profile defined by published standards and W3C recommendations. Product names shall not be used to define a profile. The following reserved values and their targets are listed below: <table> <tr> <th>Value</th><th>Target</th></tr> </table>	Value	Target
Value	Target		

Attributes	Description	
	W3C XHTML+CSS1	W3C XHTML 1.0 + CSS 1
	W3C HTML4+CSS1	W3C HTML 4.01 + CSS 1
	W3C XHTML+CSS2	W3C XHTML 1.0 + CSS 2
	W3C HTML4+CSS2	W3C HTML 4.01 + CSS 2
	<i>[Example:</i> For example, consider the following set of HTML publishing settings:	
	<pre><p:htmlPubPr ... target="W3C HTML4+CSS2"> ... </p:htmlPubPr></pre>	
	The target attribute explicitly declares that any web page generated from this document should target the W3C HTML4+CSS2 profile. <i>end example]</i>	
	The possible values for this attribute are defined by the W3C XML Schema string datatype.	
title (HTML Output Title)	Specifies a title for the HTML output file.	
	The possible values for this attribute are defined by the W3C XML Schema string datatype.	

[*Note:* The W3C XML Schema definition of this element's content model ([CT_HtmlPublishProperties](#)) is located in §A.3. *end note*]

16.2.1.2 webPr (Web Properties)

This element specifies all general output properties that pertain to generating a web format version of the presentation document.

Attributes	Description
allowPng (Allow PNG in HTML output)	<p>Specifies whether to allow the output of PNG format pictures in the HTML document.</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
clr (Slide Navigation Colors for HTML output)	<p>Specifies the color constraints that are to be used when generating HTML output.</p> <p>The possible values for this attribute are defined by the ST_WebColorType simple type (§16.4.1).</p>
encoding (Encoding for HTML output)	<p>Specifies the particular HTML character set encoding that should be used when generating output.</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_WebEncoding simple type (§16.4.2).
imgSz (Image size for HTML output)	Specifies the screen size for which the images in the HTML output should be optimized. The possible values for this attribute are defined by the ST_WebScreenSize simple type (§16.4.3).
organizeInFolders (Organize HTML output in folders)	Specifies whether the supporting output files should be automatically organized into a folder. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.
relyOnVml (Rely on VML for HTML output)	Specifies whether graphics should be output in VML within the HTML. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.
resizeGraphics (Resize graphics in HTML output)	Specifies whether to resize graphics to fit within the browser window when generating the HTML output. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.
showAnimation (Show animation in HTML output)	Specifies whether to show presentation animation in the HTML output file. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.
useLongFilenames (Use long file names in HTML output)	Specifies whether to allow the use of long file names when generating the HTML output. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

[Note: The W3C XML Schema definition of this element's content model ([CT_WebProperties](#)) is located in §A.3. end note]

16.2.1.3 Additional attributes for modifyVerifier element (Part 1, §19.2.1.19)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
algIdExt (Cryptographic Algorithm Extensibility)	Specifies that a cryptographic algorithm which was not defined by ECMA-376 has been used to generate the hash value stored with this document. This value, when present, shall be interpreted based on the value of the algIdExtSource attribute in order to determine the algorithm used, which shall be application-defined. [Rationale: This extensibility affords the fact that with exponentially increasing

Attributes	Description
	<p>computing power, documents created in the future might need to utilize as yet undefined hashing algorithms in order to remain secure. <i>end rationale</i>]</p> <p>If this value is present, the cryptAlgorithmClass, cryptAlgorithmType, and cryptAlgorithmSid attribute values shall be ignored in favor of the algorithm defined by this attribute.</p> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:algIdExt="0000000A" p:algIdExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The algIdExt attribute value of 0000000A specifies that the algorithm with hex code A shall be used as defined by the futureCryptography application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
algIdExtSource (Algorithm Extensibility Source)	<p>Specifies the application which defined the algorithm value specified by the algIdExt attribute.</p> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in one its protection element:</p> <pre><... p:algIdExt="0000000A" p:algIdExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The algIdExtSource attribute value of futureCryptography specifies that the algorithm used here was published by the futureCryptography application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
cryptAlgorithmClass (Cryptographic Algorithm Class)	<p>Specifies the class of cryptographic algorithm used by this protection. [<i>Note:</i> The initial version of ECMA-376 only supports a single version - hash - but future versions can expand this as necessary. <i>end note</i>]</p> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptAlgorithmClass="hash" p:cryptAlgorithmType="typeAny" p:cryptAlgorithmSid="1" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre>

Attributes	Description																																
	<p>The cryptAlgorithmClass attribute value of hash specifies that the algorithm used for the password is a hashing algorithm. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_AlgClass simple type (§20.1.2.1).</p>																																
cryptAlgorithmSid (Cryptographic Hashing Algorithm)	<p>Specifies the specific cryptographic hashing algorithm which shall be used along with the saltData attribute and user-supplied password in order to compute a hash value for comparison.</p> <p>The possible values for this attribute shall be interpreted as follows:</p> <table data-bbox="415 615 1351 1423"> <tr> <th>Value</th><th>Algorithm</th></tr> <tr><td>1</td><td>MD2</td></tr> <tr><td>2</td><td>MD4</td></tr> <tr><td>3</td><td>MD5</td></tr> <tr><td>4</td><td>SHA-1</td></tr> <tr><td>5</td><td>MAC</td></tr> <tr><td>6</td><td>RIPEMD</td></tr> <tr><td>7</td><td>RIPEMD-160</td></tr> <tr><td>8</td><td>Undefined. Shall not be used.</td></tr> <tr><td>9</td><td>HMAC</td></tr> <tr><td>10</td><td>Undefined. Shall not be used.</td></tr> <tr><td>11</td><td>Undefined. Shall not be used.</td></tr> <tr><td>12</td><td>SHA-256</td></tr> <tr><td>13</td><td>SHA-384</td></tr> <tr><td>14</td><td>SHA-512</td></tr> <tr><td>Any other value</td><td>Undefined. Shall not be used.</td></tr> </table> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in its protection element:</p> <pre data-bbox="451 1570 1192 1705"><... p:cryptAlgorithmClass="hash" p:cryptAlgorithmType="typeAny" p:cryptAlgorithmSid="1" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmSid attribute value of 1 specifies that the SHA-1 hashing algorithm shall be used to generate a hash from the user-defined password. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt</p>	Value	Algorithm	1	MD2	2	MD4	3	MD5	4	SHA-1	5	MAC	6	RIPEMD	7	RIPEMD-160	8	Undefined. Shall not be used.	9	HMAC	10	Undefined. Shall not be used.	11	Undefined. Shall not be used.	12	SHA-256	13	SHA-384	14	SHA-512	Any other value	Undefined. Shall not be used.
Value	Algorithm																																
1	MD2																																
2	MD4																																
3	MD5																																
4	SHA-1																																
5	MAC																																
6	RIPEMD																																
7	RIPEMD-160																																
8	Undefined. Shall not be used.																																
9	HMAC																																
10	Undefined. Shall not be used.																																
11	Undefined. Shall not be used.																																
12	SHA-256																																
13	SHA-384																																
14	SHA-512																																
Any other value	Undefined. Shall not be used.																																

Attributes	Description
	datatype.
cryptAlgorithmType (Cryptographic Algorithm Type)	<p>Specifies the kind of cryptographic algorithm used by this protection. [<i>Note</i>: The initial version of ECMA-376 only supports a single type - typeAny - but future versions can expand this as necessary. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptAlgorithmClass="hash" p:cryptAlgorithmType="typeAny" p:cryptAlgorithmSid="1" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptAlgorithmType attribute value of typeAny specifies that any algorithm type might have been used for the password. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_AlgType simple type (§20.1.2.2).</p>
cryptProvider (Cryptographic Provider)	<p>Specifies the cryptographic provider which was used to generate the hash value stored in this document. If the user provided a cryptographic provider which was not the system's built-in provider, then that provider shall be stored here so it can subsequently be used if available.</p> <p>If this attribute is omitted, then the built-in cryptographic provider on the system shall be used.</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptProvider="Krista'sProvider" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProvider attribute value of Krista'sProvider specifies that the cryptographic provider with name "Krista's Provider" shall be used if available. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
cryptProviderType (Cryptographic Provider Type)	<p>Specifies the kind of cryptographic provider to be used.</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptProviderType="rsaAES" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre>

Attributes	Description
	<p>The cryptProviderType attribute value of rsaAES specifies that the cryptographic provider type shall be an Advanced Encryption Standard provider. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_CryptProv simple type (§20.1.2.4).</p>
<p>cryptProviderTypeExt (Cryptographic Provider Type Extensibility)</p>	<p>Specifies that a cryptographic provider type which was not defined by ECMA-376 has been used to generate the hash value stored with this document.</p> <p>This value, when present, shall be interpreted based on the value of the cryptProviderTypeExtSource attribute in order to determine the provider type used, which shall be application-defined. [<i>Rationale</i>: This extensibility affords the fact that with exponentially increasing computing power, documents created in the future might need to utilize as yet undefined cryptographic provider types in order to remain secure. <i>end rationale</i>]</p> <p>If this value is present, the cryptProviderType attribute value shall be ignored in favor of the provider type defined by this attribute.</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptProviderTypeExt="00A5691D" p:cryptProvideTypeExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProviderTypeExt attribute value of 00A5691D specifies that the provider type associated with hex code A5691D shall be used as defined by the futureCryptography application. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
<p>cryptProviderTypeExtSource (Provider Type Extensibility Source)</p>	<p>Specifies the application which defined the provider type value specified by the cryptProviderTypeExt attribute.</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptProviderTypeExt="00A5691D" p:cryptProvideTypeExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The cryptProvideTypeExtSource attribute value of futureCryptography specifies that the provider type used here was published by the futureCryptography application. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
hashData (Password Hash)	<p>Specifies the hash value for the password stored with this document. This value shall be compared with the resulting hash value after hashing the user-supplied password using the algorithm specified by the preceding attributes and parent XML element, and if the two values match, the protection shall no longer be enforced.</p> <p>If this value is omitted, then no password shall be associated with the protection, and it can be turned off without supplying any password.</p> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:cryptAlgorithmClass="hash" p:cryptAlgorithmType="typeAny" p:cryptAlgorithmSid="1" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The hashData attribute value of 9oN7nWkCAyEZib1RomSJTjmPpCY= specifies that the user-supplied password shall be hashed using the pre-processing defined by the parent element (if any) followed by the SHA-1 algorithm (specified via the cryptAlgorithmSid attribute value of 1) and that the resulting has value must be 9oN7nWkCAyEZib1RomSJTjmPpCY= for the protection to be disabled. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>
saltData (Salt for Password Verifier)	<p>Specifies the salt which was prepended to the user-supplied password before it was hashed using the hashing algorithm defined by the preceding attribute values to generate the hashData attribute, and which shall also be prepended to the user-supplied password before attempting to generate a hash value for comparison. A <i>salt</i> is a random string which is added to a user-supplied password before it is hashed in order to prevent a malicious party from pre-calculating all possible password/hash combinations and simply using those precalculated values (often referred to as a "dictionary attack").</p> <p>If this attribute is omitted, then no salt shall be prepended to the user-supplied password before it is hashed for comparison with the stored hash value.</p> <p>[<i>Example:</i> Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:saltData="ZUdHa+D8F/OAKP3I7ssUnQ==" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The saltData attribute value of ZUdHa+D8F/OAKP3I7ssUnQ== specifies that the user-supplied password shall have this value prepended before it is run through the specified hashing algorithm to generate a resulting hash value for comparison. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.
spinCount (Iterations to Run Hashing Algorithm)	<p>Specifies the number of times the hashing function shall be iteratively run (runs using each iteration's result plus a 4 byte value (0-based, little endian) containing the number of the iteration as the input for the next iteration) when attempting to compare a user-supplied password with the value stored in the hashData attribute. [<i>Rationale</i>: Running the algorithm many times increases the cost of exhaustive search attacks correspondingly. Storing this value allows for the number of iterations to be increased over time to accommodate faster hardware (and hence the ability to run more iterations in less time). <i>end rationale</i>]</p> <p>[<i>Example</i>: Consider a PresentationML document with the following information stored in its protection element:</p> <pre><... p:spinCount="100000" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The spinCount attribute value of 100000 specifies that the hashing function shall be run one hundred thousand times to generate a hash value for comparison with the hash attribute. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

16.3 Slides

16.3.1 Embedded Objects

16.3.1.1 Additional attribute for control element (Part 1, §19.3.2.1)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
spid (Embedded object Shape ID)	<p>Specifies the identifier of the shape associated with this Embedded object. The shape contains all coordinate anchoring information.</p> <p>This optional attribute shall be present if the parent element does not contain a child pic element.</p> <p>The possible values for this attribute are defined by the ST_ShapeID simple type (Part 1, §20.1.10.55).</p>

16.3.1.2 Additional attribute for oleObj element (Part 1, §19.3.2.4)

The following additional attributes can be specified for a document of a transitional conformance class:

Attributes	Description
spid (Embedded object Shape ID)	<p>Specifies the identifier of the shape associated with this Embedded object. The shape contains all coordinate anchoring information.</p> <p>This optional attribute shall be present if the parent element does not contain a child pic element.</p> <p>The possible values for this attribute are defined by the ST_ShapeID simple type (Part 1, §20.1.10.55).</p>

16.4 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/presentationml/2006/main> namespace is used for documents of a transitional conformance class.

16.4.1 ST_WebColorType (HTML Slide Navigation Control Colors)

This simple type specifies the coloring that should be used when outputting to web formats.

This simple type's contents are a restriction of the W3C XML Schema token datatype.

Enumeration Value	Description
blackTextOnWhite (Black Text on White Colors)	Black Text on White coloring should be used.
browser (Browser Colors)	Browser coloring should be used.
none (Non-specific Colors)	No specific coloring has been specified.
presentationAccent (Presentation Accent Colors)	Presentation accent coloring should be used.
presentationText (Presentation Text Colors)	Presentation text coloring should be used.
whiteTextOnBlack (White Text on Black Colors)	White text on black coloring should be used.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WebColorType](#)) is located in §A.3. *end note*]

16.4.2 ST_WebEncoding (Web Encoding)

This simple type specifies a string representing the HTML character set used when outputting to web formats.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WebEncoding](#)) is located in §A.3. *end note*]

16.4.3 ST_WebScreenSize (HTML/Web Screen Size Target)

This simple type specifies the intended screen resolution for output to web formats.

This simple type's contents are a restriction of the W3C XML Schema token datatype.

Enumeration Value	Description
1024x768 (HTML/Web Size Enumeration 1024x768)	Screen size is 1024x768 pixels
1152x882 (HTML/Web Size Enumeration 1152x882)	Screen size is 1152x882 pixels
1152x900 (HTML/Web Size Enumeration 1152x900)	Screen size is 1152x900 pixels
1280x1024 (HTML/Web Size Enumeration 1280x1024)	Screen size is 1280x1024 pixels
1600x1200 (HTML/Web Size Enumeration 1600x1200)	Screen size is 1600x1200 pixels
1800x1400 (HTML/Web Size Enumeration 1800x1400)	Screen size is 1800x1400 pixels
1920x1200 (HTML/Web Size Enumeration 1920x1200)	Screen size is 1920x1200 pixels
544x376 (HTML/Web Size Enumeration 544x376)	Screen size is 544x376 pixels
640x480 (HTML/Web Size Enumeration 640x480)	Screen size is 640x480 pixels
720x512 (HTML/Web Size Enumeration 720x512)	Screen size is 720x512 pixels
800x600 (HTML/Web Size Enumeration 800x600)	Screen size is 800x600 pixels

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WebScreenSize](#)) is located in §A.3. *end note*]

16.5 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §19, “PresentationML”, have different source relationships when used in documents of the Transitional conformance class:

16.5.1 Changed attribute for bold element (Part 1, §19.2.1.1)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this embedded font that is referenced in a presentation.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.2 Changed attribute for boldItalic element (Part 1, §19.2.1.2)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this embedded font that is referenced in a presentation.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.3 Changed attribute for font element (Part 1, §19.2.1.13)

Attributes	Description																																
charset (Similar Character Set)	Specifies the character set that is supported by the parent font. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available. This information is determined by querying the font when present and shall not be modified when the font is not available.																																
Namespace: .../drawingml/2006/main	<p>The value of this attribute shall be interpreted as follows:</p> <table> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>0x00</td><td>Specifies the ANSI character set. (IANA name iso-8859-1)</td></tr> <tr> <td>0x01</td><td>Specifies the default character set.</td></tr> <tr> <td>0x02</td><td>Specifies the Symbol character set. This value specifies that the characters in the Unicode private use area (U+FF00 to U+FFFF) of the font should be used to display characters in the range U+0000 to U+00FF.</td></tr> <tr> <td>0x4D</td><td>Specifies a Macintosh (Standard Roman) character set. (IANA name macintosh)</td></tr> <tr> <td>0x80</td><td>Specifies the JIS character set. (IANA name shift_jis)</td></tr> <tr> <td>0x81</td><td>Specifies the Hangul character set. (IANA name ks_c_5601-1987)</td></tr> <tr> <td>0x82</td><td>Specifies a Johab character set. (IANA name KS C-5601-1992)</td></tr> <tr> <td>0x86</td><td>Specifies the GB-2312 character set. (IANA name GBK)</td></tr> <tr> <td>0x88</td><td>Specifies the Chinese Big Five character set. (IANA name Big5)</td></tr> <tr> <td>0xA1</td><td>Specifies a Greek character set. (IANA name windows-1253)</td></tr> <tr> <td>0xA2</td><td>Specifies a Turkish character set. (IANA name iso-8859-9)</td></tr> <tr> <td>0xA3</td><td>Specifies a Vietnamese character set. (IANA name windows-1258)</td></tr> <tr> <td>0xB1</td><td>Specifies a Hebrew character set. (IANA name windows-1255)</td></tr> <tr> <td>0xB2</td><td>Specifies an Arabic character set. (IANA name windows-1256)</td></tr> <tr> <td>0xBA</td><td>Specifies a Baltic character set. (IANA name windows-1257)</td></tr> </table>	Value	Description	0x00	Specifies the ANSI character set. (IANA name iso-8859-1)	0x01	Specifies the default character set.	0x02	Specifies the Symbol character set. This value specifies that the characters in the Unicode private use area (U+FF00 to U+FFFF) of the font should be used to display characters in the range U+0000 to U+00FF.	0x4D	Specifies a Macintosh (Standard Roman) character set. (IANA name macintosh)	0x80	Specifies the JIS character set. (IANA name shift_jis)	0x81	Specifies the Hangul character set. (IANA name ks_c_5601-1987)	0x82	Specifies a Johab character set. (IANA name KS C-5601-1992)	0x86	Specifies the GB-2312 character set. (IANA name GBK)	0x88	Specifies the Chinese Big Five character set. (IANA name Big5)	0xA1	Specifies a Greek character set. (IANA name windows-1253)	0xA2	Specifies a Turkish character set. (IANA name iso-8859-9)	0xA3	Specifies a Vietnamese character set. (IANA name windows-1258)	0xB1	Specifies a Hebrew character set. (IANA name windows-1255)	0xB2	Specifies an Arabic character set. (IANA name windows-1256)	0xBA	Specifies a Baltic character set. (IANA name windows-1257)
Value	Description																																
0x00	Specifies the ANSI character set. (IANA name iso-8859-1)																																
0x01	Specifies the default character set.																																
0x02	Specifies the Symbol character set. This value specifies that the characters in the Unicode private use area (U+FF00 to U+FFFF) of the font should be used to display characters in the range U+0000 to U+00FF.																																
0x4D	Specifies a Macintosh (Standard Roman) character set. (IANA name macintosh)																																
0x80	Specifies the JIS character set. (IANA name shift_jis)																																
0x81	Specifies the Hangul character set. (IANA name ks_c_5601-1987)																																
0x82	Specifies a Johab character set. (IANA name KS C-5601-1992)																																
0x86	Specifies the GB-2312 character set. (IANA name GBK)																																
0x88	Specifies the Chinese Big Five character set. (IANA name Big5)																																
0xA1	Specifies a Greek character set. (IANA name windows-1253)																																
0xA2	Specifies a Turkish character set. (IANA name iso-8859-9)																																
0xA3	Specifies a Vietnamese character set. (IANA name windows-1258)																																
0xB1	Specifies a Hebrew character set. (IANA name windows-1255)																																
0xB2	Specifies an Arabic character set. (IANA name windows-1256)																																
0xBA	Specifies a Baltic character set. (IANA name windows-1257)																																

Attributes	Description																																					
	0xCC	Specifies a Russian character set. (IANA name windows-1251)																																				
	0xDE	Specifies a Thai character set. (IANA name windows-874)																																				
	0xEE	Specifies an Eastern European character set. (IANA name windows-1250)																																				
	0xFF	Specifies an OEM character set not defined by ECMA-376.																																				
	Any other value	Application-defined, can be ignored.																																				
	The possible values for this attribute are defined by the W3C XML Schema byte datatype.																																					
panose (Panose Setting) Namespace: .../drawingml/2006/main	Specifies the Panose-1 classification number for the current font using the mechanism defined in §5.2.7.17 of ISO/IEC 14496-22. The possible values for this attribute are defined by the ST_Panose simple type (Part 1, §22.9.2.8).																																					
pitchFamily (Similar Font Family) Namespace: .../drawingml/2006/main	<div>Specifies the font pitch as well as the font family for the corresponding font. Because the value of this attribute is determined by a byte variable this value shall be interpreted as follows:</div> <table><tr><th>Value</th><th>Description</th></tr><tr><td>0x00</td><td>DEFAULT PITCH + UNKNOWN FONT FAMILY</td></tr><tr><td>0x01</td><td>FIXED PITCH + UNKNOWN FONT FAMILY</td></tr><tr><td>0x02</td><td>VARIABLE PITCH + UNKNOWN FONT FAMILY</td></tr><tr><td>0x10</td><td>DEFAULT PITCH + ROMAN FONT FAMILY</td></tr><tr><td>0x11</td><td>FIXED PITCH + ROMAN FONT FAMILY</td></tr><tr><td>0x12</td><td>VARIABLE PITCH + ROMAN FONT FAMILY</td></tr><tr><td>0x20</td><td>DEFAULT PITCH + SWISS FONT FAMILY</td></tr><tr><td>0x21</td><td>FIXED PITCH + SWISS FONT FAMILY</td></tr><tr><td>0x22</td><td>VARIABLE PITCH + SWISS FONT FAMILY</td></tr><tr><td>0x30</td><td>DEFAULT PITCH + MODERN FONT FAMILY</td></tr><tr><td>0x31</td><td>FIXED PITCH + MODERN FONT FAMILY</td></tr><tr><td>0x32</td><td>VARIABLE PITCH + MODERN FONT FAMILY</td></tr><tr><td>0x40</td><td>DEFAULT PITCH + SCRIPT FONT FAMILY</td></tr><tr><td>0x41</td><td>FIXED PITCH + SCRIPT FONT FAMILY</td></tr><tr><td>0x42</td><td>VARIABLE PITCH + SCRIPT FONT FAMILY</td></tr><tr><td>0x50</td><td>DEFAULT PITCH + DECORATIVE FONT FAMILY</td></tr><tr><td>0x51</td><td>FIXED PITCH + DECORATIVE FONT FAMILY</td></tr></table>		Value	Description	0x00	DEFAULT PITCH + UNKNOWN FONT FAMILY	0x01	FIXED PITCH + UNKNOWN FONT FAMILY	0x02	VARIABLE PITCH + UNKNOWN FONT FAMILY	0x10	DEFAULT PITCH + ROMAN FONT FAMILY	0x11	FIXED PITCH + ROMAN FONT FAMILY	0x12	VARIABLE PITCH + ROMAN FONT FAMILY	0x20	DEFAULT PITCH + SWISS FONT FAMILY	0x21	FIXED PITCH + SWISS FONT FAMILY	0x22	VARIABLE PITCH + SWISS FONT FAMILY	0x30	DEFAULT PITCH + MODERN FONT FAMILY	0x31	FIXED PITCH + MODERN FONT FAMILY	0x32	VARIABLE PITCH + MODERN FONT FAMILY	0x40	DEFAULT PITCH + SCRIPT FONT FAMILY	0x41	FIXED PITCH + SCRIPT FONT FAMILY	0x42	VARIABLE PITCH + SCRIPT FONT FAMILY	0x50	DEFAULT PITCH + DECORATIVE FONT FAMILY	0x51	FIXED PITCH + DECORATIVE FONT FAMILY
Value	Description																																					
0x00	DEFAULT PITCH + UNKNOWN FONT FAMILY																																					
0x01	FIXED PITCH + UNKNOWN FONT FAMILY																																					
0x02	VARIABLE PITCH + UNKNOWN FONT FAMILY																																					
0x10	DEFAULT PITCH + ROMAN FONT FAMILY																																					
0x11	FIXED PITCH + ROMAN FONT FAMILY																																					
0x12	VARIABLE PITCH + ROMAN FONT FAMILY																																					
0x20	DEFAULT PITCH + SWISS FONT FAMILY																																					
0x21	FIXED PITCH + SWISS FONT FAMILY																																					
0x22	VARIABLE PITCH + SWISS FONT FAMILY																																					
0x30	DEFAULT PITCH + MODERN FONT FAMILY																																					
0x31	FIXED PITCH + MODERN FONT FAMILY																																					
0x32	VARIABLE PITCH + MODERN FONT FAMILY																																					
0x40	DEFAULT PITCH + SCRIPT FONT FAMILY																																					
0x41	FIXED PITCH + SCRIPT FONT FAMILY																																					
0x42	VARIABLE PITCH + SCRIPT FONT FAMILY																																					
0x50	DEFAULT PITCH + DECORATIVE FONT FAMILY																																					
0x51	FIXED PITCH + DECORATIVE FONT FAMILY																																					

Attributes	Description		
	<table border="1" data-bbox="414 247 1474 294"> <tr> <td data-bbox="414 247 609 294">0x52</td><td data-bbox="609 247 1474 294">VARIABLE PITCH + DECORATIVE FONT FAMILY</td></tr> </table> <p>This information is determined by querying the font when present and shall not be modified when the font is not available. This information can be used in font substitution logic to locate an appropriate substitute font when this font is not available.</p> <p>The possible values for this attribute are defined by the W3C XML Schema byte datatype.</p>	0x52	VARIABLE PITCH + DECORATIVE FONT FAMILY
0x52	VARIABLE PITCH + DECORATIVE FONT FAMILY		
typeface (Text Typeface) Namespace: .../drawingml/2006/main	<p>Specifies the typeface, or name of the font that is to be used. The typeface is a string name of the specific font that should be used in rendering the presentation. If this font is not available within the font list of the generating application than font substitution logic should be utilized in order to select an alternate font.</p> <p>The possible values for this attribute are defined by the ST_TextTypeface simple type (Part 1, §20.1.10.81).</p>		

16.5.4 Changed attribute for handoutMasterId element (Part 1, §19.2.1.14)

Attributes	Description
id (Relationship Identifier) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location within a presentation of the handoutMaster element defining this handout master.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

16.5.5 Changed attribute for italic element (Part 1, §19.2.1.16)

Attributes	Description
id (Relationship Identifier) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this embedded font that is referenced in a presentation.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

16.5.6 Changed attribute for notesMasterId element (Part 1, §19.2.1.20)

Attributes	Description
id (Relationship Identifier) Namespace: .../officeDocument/2006/relationships	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location within a presentation of the notesMaster element defining this notes master. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.7 Changed attribute for notesSz element (Part 1, §19.2.1.22)

Attributes	Description
cx (Extent Length) Namespace: .../drawingml/2006/main	Specifies the length of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object). [Example: Consider a DrawingML object specified as follows: <code><... cx="1828800" cy="200000"/></code> The cx attributes specifies that this object has a height of 1828800 EMUs (English Metric Units). <i>end example</i>] The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).
cy (Extent Width) Namespace: .../drawingml/2006/main	Specifies the width of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object). [Example: Consider a DrawingML object specified as follows: <code>< ... cx="1828800" cy="200000"/></code> The cy attribute specifies that this object has a width of 200000 EMUs (English Metric Units). <i>end example</i>] The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).

16.5.8 Changed attribute for regular element (Part 1, §19.2.1.29)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this embedded font that is referenced in a presentation.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.9 Changed attribute for sld element (Part 1, §19.2.1.31)

Attributes	Description
id (Relationship ID)	This attribute specifies the relationship id that is used to reference to the actual slide XML file that contains all the information to the slide listed within the slide list.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.10 Changed attribute for sldId element (Part 1, §19.2.1.33)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location within a presentation of the sld element defining this slide.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.11 Changed attribute for sldMasterId element (Part 1, §19.2.1.36)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location within a presentation of the sldMaster element defining this slide master.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.12 Changed attribute for SmartTags element (Part 1, §19.2.1.40)

Attributes	Description
id (Relationship Identifier)	Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this smart tag.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.13 Changed attribute for gridSpacing element (Part 1, §19.2.2.3)

Attributes	Description
cx (Extent Length) Namespace: .../drawingml/2006/main	<p>Specifies the length of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example:</i> Consider a DrawingML object specified as follows:</p> <pre><... cx="1828800" cy="200000"/></pre> <p>The cx attributes specifies that this object has a height of 1828800 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>
cy (Extent Width) Namespace: .../drawingml/2006/main	<p>Specifies the width of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example:</i> Consider a DrawingML object specified as follows:</p> <pre>< ... cx="1828800" cy="200000"/></pre> <p>The cy attribute specifies that this object has a width of 200000 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

16.5.14 Changed attribute for origin element (Part 1, §19.2.2.9)

Attributes	Description
<p>x (X-Axis Coordinate)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The x attribute defines an x-coordinate of 0. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>
<p>y (Y-Axis Coordinate)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The y attribute defines a y-coordinate of 100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

16.5.15 Changed attribute for sld element (Part 1, §19.2.2.14)

Attributes	Description
<p>id (Relationship Identifier)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship identifier that is used in conjunction with a corresponding relationship file to resolve the location of this presentation slide within a presentation.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

16.5.16 Changed attribute for bgRef element (Part 1, §19.3.1.3)

Attributes	Description
idx (Style Matrix Index) Namespace: .../drawingml/2006/main	Specifies the style matrix index of the style referred to. The possible values for this attribute are defined by the ST_StyleMatrixColumnIndex simple type (Part 1, §20.1.10.57).

16.5.17 Changed attribute for blipFill element (Part 1, §19.3.1.4)

Attributes	Description
dpi (DPI Setting) Namespace: .../drawingml/2006/main	Specifies the DPI (dots per inch) used to calculate the size of the blip. If not present or zero, the DPI in the blip is used. [Note: This attribute is primarily used to keep track of the picture quality within a document. There are different levels of quality needed for print than on-screen viewing and thus a need to track this information. <i>end note</i>] The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.
rotWithShape (Rotate With Shape) Namespace: .../drawingml/2006/main	Specifies that the fill should rotate with the shape. That is, when the shape that has been filled with a picture and the containing shape (say a rectangle) is transformed with a rotation then the fill is transformed with the same rotation. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

16.5.18 Changed attribute for clrMap element (Part 1, §19.3.1.6)

Attributes	Description
accent1 (Accent 1) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 1 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent2 (Accent 2) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 2 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent3 (Accent 3) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 3 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

Attributes	Description
accent4 (Accent 4) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 4 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent5 (Accent 5) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 5 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent6 (Accent 6) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the accent 6 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
bg1 (Background 1) Namespace: .../drawingml/2006/main	A color defined which is associated as the first background color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
bg2 (Background 2) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the second background color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
folHlink (Followed Hyperlink) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the color for a followed hyperlink. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
hlink (Hyperlink) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the color for a hyperlink. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
tx1 (Text 1) Namespace: .../drawingml/2006/main	Specifies a color defined which is associated as the first text color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

Attributes	Description
tx2 (Text 2)	Specifies a color defined which is associated as the second text color.
Namespace: .../drawingml/2006/main	The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

16.5.19 Changed attribute for cNvPicPr element (Part 1, §19.3.1.11)

Attributes	Description
preferRelativeResize (Relative Resize Preferred)	Specifies if the user interface should show the resizing of the picture based on the picture's current size or its original size. If this attribute is set to true, then scaling is relative to the original picture size as opposed to the current picture size.
Namespace: .../drawingml/2006/main	<p>[Example: Consider the case where a picture has been resized within a document and is now 50% of the originally inserted picture size. Now if the user chooses to make a later adjustment to the size of this picture within the generating application, then the value of this attribute should be checked.</p> <p>If this attribute is set to true then a value of 50% is shown. Similarly, if this attribute is set to false, then a value of 100% should be shown because the picture has not yet been resized from its current (smaller) size. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

16.5.20 Changed attribute for cNvPr element (Part 1, §19.3.1.12)

Attributes	Description
descr (Alternative Text for Object)	Specifies alternative text for the current DrawingML object, for use by assistive technologies or applications that do not display the current object.
Namespace: .../drawingml/2006/main	<p>If this element is omitted, then no alternative text is present for the parent object.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... descr="A picture of a bowl of fruit"></pre> <p>The descr attribute contains alternative text that can be used in place of the actual DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
<p>hidden (Hidden)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies whether this DrawingML object is displayed. When a DrawingML object is displayed within a document, that object can be hidden (i.e., present, but not visible). This attribute determines whether the object is rendered or made hidden. [<i>Note</i>: An application can have settings which allow this object to be viewed. <i>end note</i>]</p> <p>If this attribute is omitted, then the parent DrawingML object shall be displayed (i.e., not hidden).</p> <p>[<i>Example</i>: Consider an inline DrawingML object that must be hidden within the document's content. This setting would be specified as follows:</p> <pre><... hidden="true" /></pre> <p>The hidden attribute has a value of true, which specifies that the DrawingML object is hidden and not displayed when the document is displayed. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>id (Unique Identifier)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a unique identifier for the current DrawingML object within the current document. This ID can be used to assist in uniquely identifying this object so that it can be referred to by other parts of the document.</p> <p>If multiple objects within the same document share the same id attribute value, then the document shall be considered non-conformant.</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre><... id="10" ... ></pre> <p>The id attribute has a value of 10, which is the unique identifier for this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DrawingElementId simple type (Part 1, §20.1.10.21).</p>
<p>name (Name)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the name of the object. [<i>Note</i>: Typically, this is used to store the original file name of a picture object. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre>< ... name="foo.jpg" ></pre> <p>The name attribute has a value of foo.jpg, which is the name of this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>

Attributes	Description
	datatype.
title (Title) Namespace: .../drawingml/2006/main	<p>Specifies the title (caption) of the current DrawingML object.</p> <p>If this attribute is omitted, then no title text is present for the parent object.</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre><... title="Process Flow Diagram"></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

16.5.21 Changed attribute for cNvSpPr element (Part 1, §19.3.1.13)

Attributes	Description
txBox (Text Box) Namespace: .../drawingml/2006/main	<p>Specifies that the corresponding shape is a text box and thus should be treated as such by the generating application. If this attribute is omitted then it is assumed that the corresponding shape is not specifically a text box.</p> <p>[<i>Note</i>: Because a shape is not specified to be a text box does not mean that it cannot have text attached to it. A text box is merely a specialized shape with specific properties. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

16.5.22 Changed attribute for contentPart element (Part 1, §19.3.1.14)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a content part.</p> <p>[<i>Example</i>: Consider an XML element which has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

16.5.23 Changed attribute for custData element (Part 1, §19.3.1.17)

Attributes	Description
id (Relationship ID) Namespace: .../officeDocument /2006/relationships	This attribute specifies the relationship id for referencing other resources outside the scope of the current PresentationML file. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.24 Changed attribute for grpSpPr element (Part 1, §19.3.1.23)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	Specifies that the group shape should be rendered using only black and white coloring. That is the coloring information for the group shape should be converted to either black or white when rendering the corresponding shapes. No gray is to be used in rendering this image, only stark black and stark white. [Note: This does not mean that the group shapes themselves are stored with only black and white color information. This attribute instead sets the rendering mode that the shapes use when rendering. <i>end note</i>] The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).

16.5.25 Changed attribute for sldLayoutId element (Part 1, §19.3.1.40)

Attributes	Description
id (ID Tag) Namespace: .../officeDocument /2006/relationships	Specifies the relationship id value that the generating application can use to resolve which slide layout is used in the creation of the slide. This relationship id is used within the relationship file for the master slide to expose the location of the corresponding layout file within the presentation. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.26 Changed attribute for spPr element (Part 1, §19.3.1.44)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture. No gray is to be used in rendering this image, only stark black and stark white. [Note: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that

Attributes	Description
	<p>the picture has applied to when rendering. <i>end note</i></p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

16.5.27 Changed attribute for tags element (Part 1, §19.3.1.47)

Attributes	Description
<p>id (Relationship ID)</p> <p>Namespace: .../officeDocument /2006/relationships</p>	<p>This attribute specifies the relationship identifier for the customer data tag. This allows for a link to a resource that is external from the current XML document but still contained within the presentation document.</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

16.5.28 Changed attribute for xfrm element (Part 1, §19.3.1.53)

Attributes	Description
<p>flipH (Horizontal Flip)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a horizontal flip. When true, this attribute defines that the shape is flipped horizontally about the center of its bounding box.</p> <p>[<i>Example:</i> The following illustrates the effect of a horizontal flip.</p> <div data-bbox="412 1058 1029 1226"> </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>flipV (Vertical Flip)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a vertical flip. When true, this attribute defines that the group is flipped vertically about the center of its bounding box.</p> <p>[<i>Example:</i> The following illustrates the effect of a vertical flip.</p> <div data-bbox="412 1562 1029 1730"> </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

Attributes	Description
rot (Rotation) Namespace: .../drawingml/2006/main	Specifies the rotation of the Graphic Frame. The units for which this attribute is specified in reside within the simple type definition referenced below. The possible values for this attribute are defined by the ST_Angle simple type (Part 1, §20.1.10.3).

16.5.29 Changed attribute for control element (Part 1, §19.3.2.1)

Attributes	Description
id (Relationship ID) Namespace: .../officeDocument/2006/relationships	Specifies the relationship id that is used to identify this Embedded object from within a slide. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.30 Changed attribute for oleObj element (Part 1, §19.3.2.4)

Attributes	Description
id (Relationship ID) Namespace: .../officeDocument/2006/relationships	Specifies the relationship id that is used to identify this Embedded object from within a slide. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

16.5.31 Changed attribute for pos element (Part 1, §19.4.5)

Attributes	Description
x (X-Axis Coordinate) Namespace: .../drawingml/2006/main	Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element. [Example: Consider the following point on a basic wrapping polygon for a DrawingML object: <pre><... x="0" y="100" /></pre> The x attribute defines an x-coordinate of 0. <i>end example</i>] The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).
y (Y-Axis Coordinate) Namespace: .../drawingml/2006/main	Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element. [Example: Consider the following point on a basic wrapping polygon for a DrawingML object:

Attributes	Description
6/main	<p><... x="0" y="100" /></p> <p>The y attribute defines a y-coordinate of 100. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

16.5.32 Changed attribute for snd element (Part 1, §19.5.68)

Attributes	Description
embed (Embedded Audio File Relationship ID) Namespace: .../officeDocument/2006/relationships	<p>Specifies the identification information for an embedded audio file. This attribute is used to specify the location of an object that resides locally within the file. [Note: A list of suggested audio types is provided in Part 1, §15.2.2. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
name (Sound Name) Namespace: .../drawingml/2006/main	<p>Specifies the original name or given short name for the corresponding sound. This is used to distinguish this sound from others by providing a human readable name for the attached sound should the user need to identify the sound among others within the UI.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

16.5.33 Changed attribute for sndTgt element (Part 1, §19.5.70)

Attributes	Description
embed (Embedded Audio File Relationship ID) Namespace: .../officeDocument/2006/relationships	<p>Specifies the identification information for an embedded audio file. This attribute is used to specify the location of an object that resides locally within the file. [Note: A list of suggested audio types is provided in Part 1, §15.2.2. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
name (Sound Name) Namespace: .../drawingml/2006/main	<p>Specifies the original name or given short name for the corresponding sound. This is used to distinguish this sound from others by providing a human readable name for the attached sound should the user need to identify the sound among others within the UI.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

17. DrawingML - Framework Reference Material

[Note: For further information on the mapping of elements and attributes to OPC parts, see the Bibliography entry, “Information on elements, attributes, and OPC parts in ECMA-376 (OOXML)”. *end note*]

17.1 DrawingML - Main

17.1.1 Table of Contents

This subclause is informative.

17.1.2 Simple Types	254
17.1.2.1 Additional member types for the union in ST_FixedPercentage (Part 1, §20.1.10.24)	254
17.1.2.2 Additional member types for the union in ST_Percentage (Part 1, §20.1.10.40)	255
17.1.2.3 Additional member types for the union in ST_PositiveFixedPercentage (Part 1, §20.1.10.45)	255
17.1.2.4 Additional member types for the union in ST_PositivePercentage (Part 1, §20.1.10.46)	255
17.1.2.5 Additional member types for the union in ST_TextFontScalePercentOrPercentString (Part 1, §20.1.10.67)	255
17.1.2.6 Additional member types for the union in ST_TextSpacingPercentOrPercentString (Part 1, §20.1.10.77)	255
17.1.2.7 ST_FixedPercentageDecimal (Fixed Percentage)	255
17.1.2.8 ST_PositiveFixedPercentageDecimal (Positive Fixed Percentage)	256
17.1.2.9 ST_PositivePercentageDecimal (Positive Percentage as Decimal Number)	256
17.1.2.10 ST_TextFontScalePercent (Text Font Scale Percentage)	256
17.1.2.11 ST_TextSpacingPercent (Text Spacing Percent)	256
17.1.2.12 ST_PercentageDecimal (Percentage as Decimal Number)	257
17.1.2.13 Additional member types for the union in ST_PrSetCustVal (Part 1, §21.4.7.66)	257
17.1.2.14 ST_TextBulletSizeDecimal (Bullet Size Percentage)	257
17.1.2.15 Additional member types for the union in ST_TextBulletSize (Part 1, §20.1.10.86)	257

End of informative text.

17.1.2 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/drawingml/2006/main> namespace is used for documents of a transitional conformance class.

17.1.2.1 Additional member types for the union in ST_FixedPercentage (Part 1, §20.1.10.24)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_FixedPercentageDecimal simple type (§17.1.2.7).

17.1.2.2 Additional member types for the union in ST_Percentage (Part 1, §20.1.10.40)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_PercentageDecimal simple type (Part 4, §12.1.2.12).

17.1.2.3 Additional member types for the union in ST_PositiveFixedPercentage (Part 1, §20.1.10.45)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_PositiveFixedPercentageDecimal simple type (§17.1.2.8).

17.1.2.4 Additional member types for the union in ST_PositivePercentage (Part 1, §20.1.10.46)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_PositivePercentageDecimal simple type (§17.1.2.9).

17.1.2.5 Additional member types for the union in ST_TextFontScalePercentOrPercentString (Part 1, §20.1.10.67)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_TextFontScalePercent simple type (§17.1.2.10).

17.1.2.6 Additional member types for the union in ST_TextSpacingPercentOrPercentString (Part 1, §20.1.10.77)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_TextSpacingPercent simple type (§17.1.2.11).

17.1.2.7 ST_FixedPercentageDecimal (Fixed Percentage)

This simple type represents a fixed percentage in 1000ths of a percent. Range from [-100%, 100%].

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 1, §20.1.10.41).

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to -100000.

- This simple type has a maximum value of less than or equal to 100000.

[*Note:* The W3C XML Schema definition of this simple type's content model (ST_FixedPercentageDecimal) is located in §A.4.1. *end note*]

17.1.2.8 ST_PositiveFixedPercentageDecimal (Positive Fixed Percentage)

This simple type represents a positive fixed percentage in 1000ths of a percent. Range from [0%, 100%].

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 1, §20.1.10.41).

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to 0.
- This simple type has a maximum value of less than or equal to 100000.

[*Note:* The W3C XML Schema definition of this simple type's content model (ST_PositiveFixedPercentageDecimal) is located in §A.4.1. *end note*]

17.1.2.9 ST_PositivePercentageDecimal (Positive Percentage as Decimal Number)

This simple type represents a positive percentage in 1000ths of a percent. Range from 0% up to and including infinity.

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 1, §20.1.10.41).

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to 0.

[*Note:* The W3C XML Schema definition of this simple type's content model (ST_PositivePercentageDecimal) is located in §A.4.1. *end note*]

17.1.2.10 ST_TextFontScalePercent (Text Font Scale Percentage)

This simple type specifies the percentage range text can be scaled to in order to fit, in 1000ths of a percent.

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 1, §20.1.10.41).

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to 1000.
- This simple type has a maximum value of less than or equal to 100000.

[*Note:* The W3C XML Schema definition of this simple type's content model (ST_TextFontScalePercent) is located in §A.4.1. *end note*]

17.1.2.11 ST_TextSpacingPercent (Text Spacing Percent)

This type specifies the range of text spacing in thousandths of a percent, in terms of a line.

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 1, §20.1.10.41).

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to 0.
- This simple type has a maximum value of less than or equal to 13200000.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_TextSpacingPercent](#)) is located in §A.4.1. *end note*]

17.1.2.12 ST_PercentageDecimal (Percentage as Decimal Number)

This simple type represents a percentage in 1000ths of a percent, e.g., a value of 1 represents 0.001% == 0.00001; a value of 100000 is equal to 100%. Percentages have no intrinsic units, but are used to scale other values with units.

This simple type's contents are a restriction of the W3C XML Schema int datatype.

[Note: The W3C XML Schema definition of this simple type's content model (ST_PercentageDecimal) is located in §A.4.1. *end note*]

17.1.2.13 Additional member types for the union in ST_PrSetCustVal (Part 1, §21.4.7.66)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

The W3C XML Schema int datatype.

17.1.2.14 ST_TextBulletSizeDecimal (Bullet Size Percentage)

This simple type specifies the range that the bullet percent can be. A bullet percent is the size of the bullet with respect to the text that should follow it. 25000 = 25%, 400000 = 400%

This simple type's contents are a restriction of the ST_PercentageDecimal datatype (Part 4, §12.1.2.12).

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 25000.

This simple type has a maximum value of less than or equal to 400000.

[Note: The W3C XML Schema definition of this simple type's content model (ST_TextBulletSizeDecimal) is located in §A.4.1. *end note*]

17.1.2.15 Additional member types for the union in ST_TextBulletSize (Part 1, §20.1.10.86)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

The ST_TextBulletSizeDecimal simple type (Part 4, §12.1.2.14).

17.2 DrawingML - Legacy Compatibility

Within the context of DrawingML, it shall be possible (for considerations to legacy compatibility) to be able to include explicit references to specific shapes within VML Drawing parts.

[*Example:* A VML Drawing part is used to define ink on a PresentationML slide, but the resulting ink is referenced from the slide by its shape ID using the elements of this namespace. *end example*]

17.2.1 Table of Contents

This subclause is informative.

17.2.2 Basics	259
17.2.2.1 legacyDrawing (Legacy Drawing Object)	259
17.3.1 Changed attribute for hlinkHover element (Part 1, §20.1.2.2.23)	260
17.3.2 Changed attribute for snd element (Part 1, §20.1.2.2.32)	260
17.3.3 Changed attribute for audioFile element (Part 1, §20.1.3.2)	260
17.3.4 Changed attribute for quickTimeFile element (Part 1, §20.1.3.4)	261
17.3.5 Changed attribute for videoFile element (Part 1, §20.1.3.6)	261
17.3.6 Changed attribute for wavAudioFile element (Part 1, §20.1.3.7)	261
17.3.7 Changed attribute for blip element (Part 1, §20.1.8.13)	261
17.3.8 Changed attribute for blipFill element (Part 1, §20.2.2.1)	262
17.3.9 Changed attribute for cNvPicPr element (Part 1, §20.2.2.2)	262
17.3.10 Changed attribute for cNvPr element (Part 1, §20.2.2.3)	263
17.3.11 Changed attribute for spPr element (Part 1, §20.2.2.6)	264
17.3.12 Changed attribute for docPr element (Part 1, §20.4.2.5)	265
17.3.13 Changed attribute for extent element (Part 1, §20.4.2.7)	266
17.3.14 Changed attribute for lineTo element (Part 1, §20.4.2.9)	267
17.3.15 Changed attribute for simplePos element (Part 1, §20.4.2.13)	267
17.3.16 Changed attribute for start element (Part 1, §20.4.2.14)	268
17.3.17 Changed attribute for blipFill element (Part 1, §20.5.2.2)	269
17.3.18 Changed attribute for cNvPicPr element (Part 1, §20.5.2.7)	269
17.3.19 Changed attribute for cNvPr element (Part 1, §20.5.2.8)	270
17.3.20 Changed attribute for cNvSpPr element (Part 1, §20.5.2.9)	271
17.3.21 Changed attribute for contentPart element (Part 1, §20.5.2.12)	272

17.3.22	Changed attribute for ext element (Part 1, §20.5.2.14)	272
17.3.23	Changed attribute for grpSpPr element (Part 1, §20.5.2.18)	273
17.3.24	Changed attribute for pos element (Part 1, §20.5.2.26)	273
17.3.25	Changed attribute for spPr element (Part 1, §20.5.2.30)	274
17.3.26	Changed attribute for xfrm element (Part 1, §20.5.2.36)	274

End of informative text.

17.2.2 Basics

Legacy Compatibility is part of the shape definitions and properties of the DrawingML framework.

17.2.2.1 legacyDrawing (Legacy Drawing Object)

This element specifies the shape ID for a legacy drawing object. These legacy drawing objects all have a shape ID associated with them that is unique across the entire document. In order to store these legacy shape IDs as well as new shape IDs this legacyDrawing element should be used.

Attributes	Description
spid (Shape ID)	<p>Legacy Shape ID that is unique throughout the entire document. Legacy shape IDs should be assigned based on which portion of the document the drawing resides on. The assignment of these ids is broken down into clusters of 1024 values. The first cluster is 1-1024, the second 1025-2048 and so on.</p> <p>This optional attribute shall be present if the parent element does not contain a child picture element.</p> <p><i>[Example: Within a word processing application the spid should be assigned based on the page that the drawing resides on. If the drawing resides on the second page then the assigned spid should be a value between 1025 and 2048. end example]</i></p> <p><i>[Example: Within a spreadsheet application the spid should be assigned based on the sheet that the drawing resides on. If the drawing resides on the second sheet then the assigned spid should be a value between 1025 and 2048. end example]</i></p> <p><i>[Example: Within a presentation application the spid should be assigned based on the slide that the drawing resides on. If the drawing resides on the second slide then the assigned spid should be a value between 1025 and 2048. end example]</i></p> <p>The possible values for this attribute are defined by the ST_ShapeID simple type (Part 1, §20.1.10.55).</p>

[*Note: The W3C XML Schema definition of this element’s content model ([CT_Compat](#)) is located in §Error! Reference source not found.. end note*]

17.3 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §20, “DrawingML - Framework Reference Material”, have different source relationships when used in documents of the Transitional conformance class:

17.3.1 Changed attribute for hlinkHover element (Part 1, §20.1.2.2.23)

Attributes	Description
id (Drawing Object Hyperlink Target)	Specifies the relationship id that when looked up in this slides relationship file contains the target of this hyperlink. This attribute cannot be omitted.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.2 Changed attribute for snd element (Part 1, §20.1.2.2.32)

Attributes	Description
embed (Embedded Audio File Relationship ID)	Specifies the identification information for an embedded audio file. This attribute is used to specify the location of an object that resides locally within the file. [<i>Note: A list of suggested audio types is provided in Part 1, §15.2.2. end note</i>]
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.3 Changed attribute for audioFile element (Part 1, §20.1.3.2)

Attributes	Description
link (Linked Relationship ID)	Specifies the identification information for a linked object. This attribute is used to specify the location of an object that does not reside within this file.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.4 Changed attribute for quickTimeFile element (Part 1, §20.1.3.4)

Attributes	Description
link (Linked Relationship ID)	Specifies the identification information for a linked object. This attribute is used to specify the location of an object that does not reside within this file.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.5 Changed attribute for videoFile element (Part 1, §20.1.3.6)

Attributes	Description
link (Linked Relationship ID)	Specifies the identification information for a linked object. This attribute is used to specify the location of an object that does not reside within this file.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.6 Changed attribute for wavAudioFile element (Part 1, §20.1.3.7)

Attributes	Description
embed (Embedded Audio File Relationship ID)	Specifies the identification information for an embedded audio file. This attribute is used to specify the location of an object that resides locally within the file. [Note: A list of suggested audio types is provided in Part 1, §15.2.2. <i>end note</i>]
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.7 Changed attribute for blip element (Part 1, §20.1.8.13)

Attributes	Description
embed (Embedded Picture Reference)	Specifies the identification information for an embedded picture. This attribute is used to specify an image that resides locally within the file.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

Attributes	Description
link (Linked Picture Reference)	Specifies the identification information for a linked picture. This attribute is used to specify an image that does not reside within this file.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

17.3.8 Changed attribute for blipFill element (Part 1, §20.2.2.1)

Attributes	Description
dpi (DPI Setting)	Specifies the DPI (dots per inch) used to calculate the size of the blip. If not present or zero, the DPI in the blip is used.
Namespace: .../drawingml/2006/main	<p>[Note: This attribute is primarily used to keep track of the picture quality within a document. There are different levels of quality needed for print than on-screen viewing and thus a need to track this information. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
rotWithShape (Rotate With Shape)	Specifies that the fill should rotate with the shape. That is, when the shape that has been filled with a picture and the containing shape (say a rectangle) is transformed with a rotation then the fill is transformed with the same rotation.
Namespace: .../drawingml/2006/main	The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

17.3.9 Changed attribute for cNvPicPr element (Part 1, §20.2.2.2)

Attributes	Description
preferRelativeResize (Relative Resize Preferred)	Specifies if the user interface should show the resizing of the picture based on the picture's current size or its original size. If this attribute is set to true, then scaling is relative to the original picture size as opposed to the current picture size.
Namespace: .../drawingml/2006/main	<p>[Example: Consider the case where a picture has been resized within a document and is now 50% of the originally inserted picture size. Now if the user chooses to make a later adjustment to the size of this picture within the generating application, then the value of this attribute should be checked.</p> <p>If this attribute is set to true then a value of 50% is shown. Similarly, if this attribute is set to false, then a value of 100% should be shown because the picture has not yet been resized from its current (smaller) size. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

17.3.10 Changed attribute for cNvPr element (Part 1, §20.2.2.3)

Attributes	Description
<p>descr (Alternative Text for Object)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies alternative text for the current DrawingML object, for use by assistive technologies or applications that do not display the current object.</p> <p>If this element is omitted, then no alternative text is present for the parent object.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... descr="A picture of a bowl of fruit"></pre> <p>The descr attribute contains alternative text that can be used in place of the actual DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hidden (Hidden)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies whether this DrawingML object is displayed. When a DrawingML object is displayed within a document, that object can be hidden (i.e., present, but not visible). This attribute determines whether the object is rendered or made hidden. [Note: An application can have settings that allow this object to be viewed. <i>end note</i>]</p> <p>If this attribute is omitted, then the parent DrawingML object shall be displayed (i.e., not hidden).</p> <p>[Example: Consider an inline DrawingML object that must be hidden within the document's content. This setting would be specified as follows:</p> <pre><... hidden="true" /></pre> <p>The hidden attribute has a value of true, which specifies that the DrawingML object is hidden and not displayed when the document is displayed. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>id (Unique Identifier)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a unique identifier for the current DrawingML object within the current document. This ID can be used to assist in uniquely identifying this object so that it can be referred to by other parts of the document.</p> <p>If multiple objects within the same document share the same id attribute value, then the document shall be considered non-conformant.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... id="10" ... ></pre> <p>The id attribute has a value of 10, which is the unique identifier for this DrawingML</p>

Attributes	Description
	<p>object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DrawingElementId simple type (Part 1, §20.1.10.21).</p>
<p>name (Name)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the name of the object. [<i>Note</i>: Typically, this is used to store the original file name of a picture object. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre>< ... name="foo.jpg" ></pre> <p>The name attribute has a value of foo.jpg, which is the name of this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>title (Title)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the title (caption) of the current DrawingML object.</p> <p>If this attribute is omitted, then no title text is present for the parent object.</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre><... title="Process Flow Diagram"></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

17.3.11 Changed attribute for spPr element (Part 1, §20.2.2.6)

Attributes	Description
<p>bwMode (Black and White Mode)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note</i>: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that the picture has applied to when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

17.3.12 Changed attribute for docPr element (Part 1, §20.4.2.5)

Attributes	Description
<p>descr (Alternative Text for Object)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies alternative text for the current DrawingML object, for use by assistive technologies or applications that do not display the current object.</p> <p>If this element is omitted, then no alternative text is present for the parent object.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... descr="A picture of a bowl of fruit"></pre> <p>The descr attribute contains alternative text that can be used in place of the actual DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hidden (Hidden)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies whether this DrawingML object is displayed. When a DrawingML object is displayed within a document, that object can be hidden (i.e., present, but not visible). This attribute determines whether the object is rendered or made hidden. [Note: An application can have settings that allow this object to be viewed. <i>end note</i>]</p> <p>If this attribute is omitted, then the parent DrawingML object shall be displayed (i.e., not hidden).</p> <p>[Example: Consider an inline DrawingML object that must be hidden within the document's content. This setting would be specified as follows:</p> <pre><... hidden="true" /></pre> <p>The hidden attribute has a value of true, which specifies that the DrawingML object is hidden and not displayed when the document is displayed. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>id (Unique Identifier)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a unique identifier for the current DrawingML object within the current document. This ID can be used to assist in uniquely identifying this object so that it can be referred to by other parts of the document.</p> <p>If multiple objects within the same document share the same id attribute value, then the document shall be considered non-conformant.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... id="10" ... ></pre> <p>The id attribute has a value of 10, which is the unique identifier for this DrawingML</p>

Attributes	Description
	<p>object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DrawingElementId simple type (Part 1, §20.1.10.21).</p>
<p>name (Name)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the name of the object. [<i>Note</i>: Typically, this is used to store the original file name of a picture object. <i>end note</i>]</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre>< ... name="foo.jpg" ></pre> <p>The name attribute has a value of foo.jpg, which is the name of this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>title (Title)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the title (caption) of the current DrawingML object.</p> <p>If this attribute is omitted, then no title text is present for the parent object.</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre><... title="Process Flow Diagram"></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

17.3.13 Changed attribute for extent element (Part 1, §20.4.2.7)

Attributes	Description
<p>cx (Extent Width)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the width of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example</i>: Consider a DrawingML object specified as follows:</p> <pre><... cx="1828800" cy="200000"/></pre> <p>The cx attribute specifies that this object has a width of 1828800 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

Attributes	Description
cy (Extent Height) Namespace: .../drawingml/2006/main	<p>Specifies the height of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example:</i> Consider a DrawingML object specified as follows:</p> <pre>< ... cx="1828800" cy="200000" /></pre> <p>The cy attribute specifies that this object has a height of 200000 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

17.3.14 Changed attribute for lineTo element (Part 1, §20.4.2.9)

Attributes	Description
x (X-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The x attribute defines an x-coordinate of 0. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>
y (Y-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The y attribute defines a y-coordinate of 100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

17.3.15 Changed attribute for simplePos element (Part 1, §20.4.2.13)

Attributes	Description
x (X-Axis Coordinate)	<p>Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element.</p>

Attributes	Description
Namespace: .../drawingml/2006/main	<p>[<i>Example</i>: Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The x attribute defines an x-coordinate of 0. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>
y (Y-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example</i>: Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The y attribute defines a y-coordinate of 100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

17.3.16 Changed attribute for start element (Part 1, §20.4.2.14)

Attributes	Description
x (X-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example</i>: Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The x attribute defines an x-coordinate of 0. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

Attributes	Description
y (Y-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The y attribute defines a y-coordinate of 100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

17.3.17 Changed attribute for blipFill element (Part 1, §20.5.2.2)

Attributes	Description
dpi (DPI Setting) Namespace: .../drawingml/2006/main	<p>Specifies the DPI (dots per inch) used to calculate the size of the blip. If not present or zero, the DPI in the blip is used.</p> <p>[<i>Note:</i> This attribute is primarily used to keep track of the picture quality within a document. There are different levels of quality needed for print than on-screen viewing and thus a need to track this information. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
rotWithShape (Rotate With Shape) Namespace: .../drawingml/2006/main	<p>Specifies that the fill should rotate with the shape. That is, when the shape that has been filled with a picture and the containing shape (say a rectangle) is transformed with a rotation then the fill is transformed with the same rotation.</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

17.3.18 Changed attribute for cNvPicPr element (Part 1, §20.5.2.7)

Attributes	Description
preferRelativeResize (Relative Resize Preferred) Namespace: .../drawingml/2006/main	<p>Specifies if the user interface should show the resizing of the picture based on the picture's current size or its original size. If this attribute is set to true, then scaling is relative to the original picture size as opposed to the current picture size.</p> <p>[<i>Example:</i> Consider the case where a picture has been resized within a document and is now 50% of the originally inserted picture size. Now if the user chooses to make a later adjustment to the size of this picture within the generating application, then the value of this attribute should be checked.</p> <p>If this attribute is set to true then a value of 50% is shown. Similarly, if this attribute is set to false, then a value of 100% should be shown because the picture has not yet been</p>

Attributes	Description
	<p>resized from its current (smaller) size. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

17.3.19 Changed attribute for cNvPr element (Part 1, §20.5.2.8)

Attributes	Description
<p>descr (Alternative Text for Object)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies alternative text for the current DrawingML object, for use by assistive technologies or applications that do not display the current object.</p> <p>If this element is omitted, then no alternative text is present for the parent object.</p> <p>[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <pre><... descr="A picture of a bowl of fruit"></pre> <p>The descr attribute contains alternative text that can be used in place of the actual DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hidden (Hidden)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies whether this DrawingML object is displayed. When a DrawingML object is displayed within a document, that object can be hidden (i.e., present, but not visible). This attribute determines whether the object is rendered or made hidden. [<i>Note</i>: An application can have settings that allow this object to be viewed. <i>end note</i>]</p> <p>If this attribute is omitted, then the parent DrawingML object shall be displayed (i.e., not hidden).</p> <p>[<i>Example</i>: Consider an inline DrawingML object that must be hidden within the document's content. This setting would be specified as follows:</p> <pre><... hidden="true" /></pre> <p>The hidden attribute has a value of true, which specifies that the DrawingML object is hidden and not displayed when the document is displayed. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

Attributes	Description
id (Unique Identifier) Namespace: .../drawingml/2006/main	<p>Specifies a unique identifier for the current DrawingML object within the current document. This ID can be used to assist in uniquely identifying this object so that it can be referred to by other parts of the document.</p> <p>If multiple objects within the same document share the same id attribute value, then the document shall be considered non-conformant.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... id="10" ... ></pre> <p>The id attribute has a value of 10, which is the unique identifier for this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_DrawingElementId simple type (Part 1, §20.1.10.21).</p>
name (Name) Namespace: .../drawingml/2006/main	<p>Specifies the name of the object. [Note: Typically, this is used to store the original file name of a picture object. <i>end note</i>]</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre>< ... name="foo.jpg" ></pre> <p>The name attribute has a value of foo.jpg, which is the name of this DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the type W3C XML Schema string.</p>
title (Title) Namespace: .../drawingml/2006/main	<p>Specifies the title (caption) of the current DrawingML object.</p> <p>If this attribute is omitted, then no title text is present for the parent object.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... title="Process Flow Diagram"></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

17.3.20 Changed attribute for cNvSpPr element (Part 1, §20.5.2.9)

Attributes	Description
txBox (Text Box)	Specifies that the corresponding shape is a text box and thus should be treated as such by the generating application. If this attribute is omitted then it is assumed that the

Attributes	Description
Namespace: .../drawingml/2006/main	<p>corresponding shape is not specifically a text box.</p> <p>[<i>Note</i>: Because a shape is not specified to be a text box does not mean that it cannot have text attached to it. A text box is merely a specialized shape with specific properties. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

17.3.21 Changed attribute for contentPart element (Part 1, §20.5.2.12)

Attributes	Description
id (Relationship to Part) Namespace: .../officeDocument/2006/relationships	<p>Specifies the relationship ID to a content part.</p> <p>[<i>Example</i>: Consider an XML element that has the following id attribute:</p> <pre><... r:id="rId1" /></pre> <p>The markup specifies the associated relationship part with relationship ID rId1 contains the corresponding relationship information for the parent XML element. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>

17.3.22 Changed attribute for ext element (Part 1, §20.5.2.14)

Attributes	Description
cx (Extent Width) Namespace: .../drawingml/2006/main	<p>Specifies the width of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example</i>: Consider a DrawingML object specified as follows:</p> <pre><... cx="1828800" cy="200000"/></pre> <p>The cx attribute specifies that this object has a width of 1828800 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

Attributes	Description
cy (Extent Height) Namespace: .../drawingml/2006/main	<p>Specifies the height of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example:</i> Consider a DrawingML object specified as follows:</p> <pre>< ... cx="1828800" cy="200000" /></pre> <p>The cy attribute specifies that this object has a height of 200000 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

17.3.23 Changed attribute for grpSpPr element (Part 1, §20.5.2.18)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	<p>Specifies that the group shape should be rendered using only black and white coloring. That is the coloring information for the group shape should be converted to either black or white when rendering the corresponding shapes.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note:</i> This does not mean that the group shapes themselves are stored with only black and white color information. This attribute instead sets the rendering mode that the shapes use when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

17.3.24 Changed attribute for pos element (Part 1, §20.5.2.26)

Attributes	Description
x (X-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the x-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example:</i> Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The x attribute defines an x-coordinate of 0. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>


Attributes	Description
y (Y-Axis Coordinate) Namespace: .../drawingml/2006/main	<p>Specifies a coordinate on the y-axis. The origin point for this coordinate shall be specified by the parent XML element.</p> <p>[<i>Example</i>: Consider the following point on a basic wrapping polygon for a DrawingML object:</p> <pre><... x="0" y="100" /></pre> <p>The y attribute defines a y-coordinate of 100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).</p>

17.3.25 Changed attribute for spPr element (Part 1, §20.5.2.30)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	<p>Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note</i>: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that the picture has applied to when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

17.3.26 Changed attribute for xfrm element (Part 1, §20.5.2.36)

Attributes	Description
flipH (Horizontal Flip) Namespace: .../drawingml/2006/main	<p>Specifies a horizontal flip. When true, this attribute defines that the shape is flipped horizontally about the center of its bounding box.</p> <p>[<i>Example</i>: The following illustrates the effect of a horizontal flip.</p> <div data-bbox="410 1549 1023 1719" data-label="Image"> </div> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

Attributes	Description
<p>flipV (Vertical Flip)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a vertical flip. When true, this attribute defines that the group is flipped vertically about the center of its bounding box.</p> <p>[<i>Example</i>: The following illustrates the effect of a vertical flip.</p> <div data-bbox="412 424 1026 592">  </div> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>rot (Rotation)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies the rotation of the Graphic Frame. The units for that this attribute is specified in reside within the simple type definition referenced below.</p> <p>The possible values for this attribute are defined by the ST_Angle simple type (Part 1, §20.1.10.3).</p>

18. DrawingML - Components Reference Material

18.1 DrawingML - Charts

18.1.1 Table of Contents

This subclause is informative.

18.1.2 Elements	277
18.1.2.1 legacyDrawingHF (Legacy Drawing for Headers and Footers)	277
18.1.3 Simple Types	277
18.1.3.1 Additional member types for union in ST_DepthPercent	277
18.1.3.2 ST_DepthPercentUShort (Depth Percent UnsignedShort) (Part 1, §21.2.3.9)	278
18.1.3.3 Additional member types for union in ST_HPercent (Part 1, §21.2.3.19)	278
18.1.3.4 ST_HPercentUShort (Depth Percent UnsignedShort)	278
18.1.3.5 Additional member types for union in ST_GapAmount (Part 1, §21.2.3.16)	278
18.1.3.6 ST_GapAmountUShort (Gap Amount UnsignedShort)	278
18.1.3.7 Additional member types for union in ST_SecondPieSize (Part 1, §21.2.3.41)	279
18.1.3.8 ST_SecondPieSizeUShort (Second Pie Size UnsignedShort)	279
18.1.3.9 Additional member types for union in ST_HoleSize (Part 1, §21.2.3.18)	279
18.1.3.10 ST_HoleSizeUByte (Hole Size UnsignedByte)	279
18.1.3.11 Additional member types for union in ST_LblOffset (Part 1, §21.2.3.23)	279
18.1.3.12 ST_LblOffsetUShort (Label Offset UnsignedShort)	279
18.1.3.13 Additional member types for union in ST_Overlap (Part 1, §21.2.3.31)	280
18.1.3.14 ST_OverlapByte (Overlap Byte)	280
18.1.3.15 Additional member types for union in ST_BubbleScale (Part 1, §21.2.3.5)	280
18.1.3.16 ST_BubbleScaleUInt (Bubble Scale UnsignedInt)	280
18.1.3.17 Additional member types for union in ST_Thickness (Part 1, §21.2.3.206)	280
18.2.1 Changed attribute for hlinkClick element (Part 1, §21.1.2.3.5)	281
18.2.2 Changed attribute for hlinkMouseOver element (Part 1, §21.1.2.3.6)	281
18.2.3 Changed attribute for chart element (Part 1, §21.2.2.26)	281
18.2.4 Changed attribute for clrMapOvr element (Part 1, §21.2.2.30)	281
18.2.5 Changed attribute for externalData element (Part 1, §21.2.2.63)	283
18.2.6 Changed attribute for spPr element (Part 1, §21.2.2.197)	283
18.2.7 Changed attribute for userShapes element (Part 1, §21.2.2.221)	284
18.2.8 Changed attribute for blipFill element (Part 1, §21.3.2.2)	284
18.2.9 Changed attribute for cNvPicPr element (Part 1, §21.3.2.6)	284

18.2.10	Changed attribute for cNvPr element (Part 1, §21.3.2.7)	285
18.2.11	Changed attribute for cNvSpPr element (Part 1, §21.3.2.8)	287
18.2.12	Changed attribute for ext element (Part 1, §21.3.2.10)	287
18.2.13	Changed attribute for grpSpPr element (Part 1, §21.3.2.14)	288
18.2.14	Changed attribute for spPr element (Part 1, §21.3.2.23)	288
18.2.15	Changed attribute for xfrm element (Part 1, §21.3.2.28)	289
18.2.16	Changed attribute for rellds element (Part 1, §21.4.2.22)	289
18.2.17	Changed attribute for shape element (Part 1, §21.4.2.27)	290
18.2.18	Changed attribute for spPr element (Part 1, §21.4.3.7)	291
18.2.19	Changed attribute for sp3d element (Part 1, §21.4.5.6)	291

End of informative text.

18.1.2 Elements

18.1.2.1 legacyDrawingHF (Legacy Drawing for Headers and Footers)

This element specifies the VML Drawing part that contains any pictures used in the header or footer of the chart.

Attributes	Description
id (Relationship Reference)	Specifies the relationship ID for the relationship for this Chart or Chart Drawing part. The type of relationship needed is specified by the parent element.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_RelId](#)) is located in §A.5.1. *end note*]

18.1.3 Simple Types

18.1.3.1 Additional member types for union in ST_DepthPercent

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_DepthPercentUShort simple type (§13.1.3.2).

18.1.3.2 [ST_DepthPercentUShort \(Depth Percent UnsignedShort\) \(Part 1, §21.2.3.9\)](#)

This simple type specifies that its contents contain a whole number between 20 and 2000, whose contents are a percentage. This simple type's contents are a restriction of the W3C XML Schema unsignedShort datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 20.

This simple type has a maximum value of less than or equal to 2000.

18.1.3.3 [Additional member types for union in ST_HPercent \(Part 1, §21.2.3.19\)](#)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_HPercentUShort simple type (§13.1.3.4).

18.1.3.4 [ST_HPercentUShort \(Depth Percent UnsignedShort\)](#)

This simple type specifies that its contents contain a whole number between 5 and 500, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedShort datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 5.

This simple type has a maximum value of less than or equal to 500.

18.1.3.5 [Additional member types for union in ST_GapAmount \(Part 1, §21.2.3.16\)](#)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_GapAmountUShort simple type (§13.1.3.6).

18.1.3.6 [ST_GapAmountUShort \(Gap Amount UnsignedShort\)](#)

This simple type specifies that its contents contain a whole number between 0 and 500, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedShort datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 0.

This simple type has a maximum value of less than or equal to 500.

18.1.3.7 Additional member types for union in ST_SecondPieSize (Part 1, §21.2.3.41)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_SecondPieSizeUShort simple type (§13.1.3.10).

18.1.3.8 ST_SecondPieSizeUShort (Second Pie Size UnsignedShort)

This simple type specifies that its contents contain a whole number between 5 and 200, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedShort datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 5.

This simple type has a maximum value of less than or equal to 200.

18.1.3.9 Additional member types for union in ST_HoleSize (Part 1, §21.2.3.18)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_HoleSizeUByte simple type (§13.1.3.12).

18.1.3.10 ST_HoleSizeUByte (Hole Size UnsignedByte)

This simple type specifies that its contents contain a whole number between 10 and 90, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedByte datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 10.

This simple type has a maximum value of less than or equal to 90.

18.1.3.11 Additional member types for union in ST_LblOffset (Part 1, §21.2.3.23)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_LblOffsetUShort simple type (§13.1.3.14).

18.1.3.12 ST_LblOffsetUShort (Label Offset UnsignedShort)

This simple type specifies that its contents contain a whole number between 0 and 1000, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedShort datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 0.

This simple type has a maximum value of less than or equal to 1000.

18.1.3.13 Additional member types for union in ST_Overlap (Part 1, §21.2.3.31)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_OverlapByte simple type (§13.1.3.16).

18.1.3.14 ST_OverlapByte (Overlap Byte)

This simple type specifies that its contents contain a whole number between -100 and 100, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema byte datatype.

This simple type also specifies the following restrictions:

- This simple type has a minimum value of greater than or equal to -100.

This simple type has a maximum value of less than or equal to 100.

18.1.3.15 Additional member types for union in ST_BubbleScale (Part 1, §21.2.3.5)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The ST_BubbleScaleUInt simple type (§13.1.3.18).

18.1.3.16 ST_BubbleScaleUInt (Bubble Scale UnsignedInt)

This simple type specifies that its contents contain a whole number between 0 and 300, whose contents are a percentage.

This simple type's contents are a restriction of the W3C XML Schema unsignedInt datatype.

This simple type also specifies the following restrictions:

This simple type has a minimum value of greater than or equal to 0.

This simple type has a maximum value of less than or equal to 300.

18.1.3.17 Additional member types for union in ST_Thickness (Part 1, §21.2.3.206)

The value space of the following additional member types can be used within the context of this simple type for a document of a transitional conformance class.

- The W3C XML Schema unsignedInt datatype.

18.2 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §21, “DrawingML - Components Reference Material”, have different source relationships when used in documents of the Transitional conformance class:

18.2.1 Changed attribute for `hlinkClick` element (Part 1, §21.1.2.3.5)

Attributes	Description
id (Drawing Object Hyperlink Target)	Specifies the relationship id that when looked up in this slides relationship file contains the target of this hyperlink. This attribute cannot be omitted.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.2 Changed attribute for `hlinkMouseOver` element (Part 1, §21.1.2.3.6)

Attributes	Description
id (Drawing Object Hyperlink Target)	Specifies the relationship id that when looked up in this slides relationship file contains the target of this hyperlink. This attribute cannot be omitted.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.3 Changed attribute for `chart` element (Part 1, §21.2.2.26)

Attributes	Description
id (Relationship Reference)	Specifies the relationship ID for the relationship for this Chart or Chart Drawing part. The type of relationship needed is specified by the parent element.
Namespace: .../officeDocument /2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.4 Changed attribute for `clrMapOvr` element (Part 1, §21.2.2.30)

Attributes	Description
accent1 (Accent 1)	Specifies a color defined that is associated as the accent 1 color.
Namespace: .../drawingml/2006/main	The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

Attributes	Description
accent2 (Accent 2) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the accent 2 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent3 (Accent 3) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the accent 3 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent4 (Accent 4) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the accent 4 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent5 (Accent 5) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the accent 5 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
accent6 (Accent 6) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the accent 6 color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
bg1 (Background 1) Namespace: .../drawingml/2006/main	A color defined that is associated as the first background color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
bg2 (Background 2) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the second background color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
folHlink (Followed Hyperlink) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the color for a followed hyperlink. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

Attributes	Description
hlink (Hyperlink) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the color for a hyperlink. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
tx1 (Text 1) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the first text color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).
tx2 (Text 2) Namespace: .../drawingml/2006/main	Specifies a color defined that is associated as the second text color. The possible values for this attribute are defined by the ST_ColorSchemeIndex simple type (Part 1, §20.1.10.14).

18.2.5 Changed attribute for externalData element (Part 1, §21.2.2.63)

Attributes	Description
id (Relationship Reference) Namespace: .../officeDocument/2006/relationships	Specifies the relationship ID for the relationship for this chart. The relationship explicitly targeted by this attribute shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/package . The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.6 Changed attribute for spPr element (Part 1, §21.2.2.197)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture. No gray is to be used in rendering this image, only stark black and stark white. [Note: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that the picture has applied to when rendering. <i>end note</i>] The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).

18.2.7 Changed attribute for userShapes element (Part 1, §21.2.2.221)

Attributes	Description
id (Relationship Reference)	Specifies the relationship ID for the relationship for this Chart or Chart Drawing part. The type of relationship needed is specified by the parent element.
Namespace: .../officeDocument/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.8 Changed attribute for blipFill element (Part 1, §21.3.2.2)

Attributes	Description
dpi (DPI Setting)	Specifies the DPI (dots per inch) used to calculate the size of the blip. If not present or zero, the DPI in the blip is used.
Namespace: .../drawingml/2006/main	<p>[<i>Note:</i> This attribute is primarily used to keep track of the picture quality within a document. There are different levels of quality needed for print than on-screen viewing and thus a need to track this information. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
rotWithShape (Rotate With Shape)	Specifies that the fill should rotate with the shape. That is, when the shape that has been filled with a picture and the containing shape (say a rectangle) is transformed with a rotation then the fill is transformed with the same rotation.
Namespace: .../drawingml/2006/main	The possible values for this attribute are defined by the W3C XML Schema boolean datatype.

18.2.9 Changed attribute for cNvPicPr element (Part 1, §21.3.2.6)

Attributes	Description
preferRelativeResize (Relative Resize Preferred)	Specifies if the user interface should show the resizing of the picture based on the picture's current size or its original size. If this attribute is set to true, then scaling is relative to the original picture size as opposed to the current picture size.
Namespace: .../drawingml/2006/main	<p>[<i>Example:</i> Consider the case where a picture has been resized within a document and is now 50% of the originally inserted picture size. Now if the user chooses to make a later adjustment to the size of this picture within the generating application, then the value of this attribute should be checked.</p> <p>If this attribute is set to true then a value of 50% is shown. Similarly, if this attribute is set to false, then a value of 100% should be shown because the picture has not yet been resized from its current (smaller) size. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean</p>

Attributes	Description
	datatype.

18.2.10 Changed attribute for cNvPr element (Part 1, §21.3.2.7)

Attributes	Description
<p>descr (Alternative Text for Object)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies alternative text for the current DrawingML object, for use by assistive technologies or applications that do not display the current object.</p> <p>If this element is omitted, then no alternative text is present for the parent object.</p> <p>[Example: Consider a DrawingML object defined as follows:</p> <pre><... descr="A picture of a bowl of fruit"></pre> <p>The descr attribute contains alternative text that can be used in place of the actual DrawingML object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hidden (Hidden)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies whether this DrawingML object is displayed. When a DrawingML object is displayed within a document, that object can be hidden (i.e., present, but not visible). This attribute determines whether the object is rendered or made hidden. [Note: An application can have settings that allow this object to be viewed. <i>end note</i>]</p> <p>If this attribute is omitted, then the parent DrawingML object shall be displayed (i.e., not hidden).</p> <p>[Example: Consider an inline DrawingML object that must be hidden within the document's content. This setting would be specified as follows:</p> <pre><... hidden="true" /></pre> <p>The hidden attribute has a value of true, which specifies that the DrawingML object is hidden and not displayed when the document is displayed. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
<p>id (Unique Identifier)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Specifies a unique identifier for the current DrawingML object within the current document. This ID can be used to assist in uniquely identifying this object so that it can be referred to by other parts of the document.</p> <p>If multiple objects within the same document share the same id attribute value, then the document shall be considered non-conformant.</p> <p>[Example: Consider a DrawingML object defined as follows:</p>

Attributes	Description
	<p data-bbox="451 281 678 312"><... id="10" ... ></p> <p data-bbox="414 354 1403 420">The id attribute has a value of 10, which is the unique identifier for this DrawingML object. <i>end example</i>]</p> <p data-bbox="414 462 1445 527">The possible values for this attribute are defined by the ST_DrawingElementId simple type (Part 1, §20.1.10.21).</p>
<p data-bbox="139 543 306 575">name (Name)</p> <p data-bbox="139 617 375 714">Namespace: .../drawingml/2006/main</p>	<p data-bbox="414 543 1427 609">Specifies the name of the object. [<i>Note</i>: Typically, this is used to store the original file name of a picture object. <i>end note</i>]</p> <p data-bbox="414 651 1117 682">[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <p data-bbox="451 724 776 756">< ... name="foo.jpg" ></p> <p data-bbox="414 798 1468 863">The name attribute has a value of foo.jpg, which is the name of this DrawingML object. <i>end example</i>]</p> <p data-bbox="414 905 1377 970">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p data-bbox="139 978 269 1010">title (Title)</p> <p data-bbox="139 1052 375 1148">Namespace: .../drawingml/2006/main</p>	<p data-bbox="414 978 1135 1010">Specifies the title (caption) of the current DrawingML object.</p> <p data-bbox="414 1052 1317 1083">If this attribute is omitted, then no title text is present for the parent object.</p> <p data-bbox="414 1125 1117 1157">[<i>Example</i>: Consider a DrawingML object defined as follows:</p> <p data-bbox="451 1199 967 1230"><... title="Process Flow Diagram"></p> <p data-bbox="414 1272 574 1304"><i>end example</i>]</p> <p data-bbox="414 1346 1377 1411">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

18.2.11 Changed attribute for cNvSpPr element (Part 1, §21.3.2.8)

Attributes	Description
txBox (Text Box) Namespace: .../drawingml/2006/main	<p>Specifies that the corresponding shape is a text box and thus should be treated as such by the generating application. If this attribute is omitted then it is assumed that the corresponding shape is not specifically a text box.</p> <p>[<i>Note</i>: Because a shape is not specified to be a text box does not mean that it cannot have text attached to it. A text box is merely a specialized shape with specific properties. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>

18.2.12 Changed attribute for ext element (Part 1, §21.3.2.10)

Attributes	Description
cx (Extent Width) Namespace: .../drawingml/2006/main	<p>Specifies the width of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example</i>: Consider a DrawingML object specified as follows:</p> <pre><... cx="1828800" cy="200000"/></pre> <p>The cx attribute specifies that this object has a width of 1828800 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>
cy (Extent Height) Namespace: .../drawingml/2006/main	<p>Specifies the height of the extents rectangle in EMUs. This rectangle shall dictate the size of the object as displayed (the result of any scaling to the original object).</p> <p>[<i>Example</i>: Consider a DrawingML object specified as follows:</p> <pre>< ... cx="1828800" cy="200000"/></pre> <p>The cy attribute specifies that this object has a height of 200000 EMUs (English Metric Units). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>



18.2.13 Changed attribute for grpSpPr element (Part 1, §21.3.2.14)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	<p>Specifies that the group shape should be rendered using only black and white coloring. That is the coloring information for the group shape should be converted to either black or white when rendering the corresponding shapes.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note</i>: This does not mean that the group shapes themselves are stored with only black and white color information. This attribute instead sets the rendering mode that the shapes use when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

18.2.14 Changed attribute for spPr element (Part 1, §21.3.2.23)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	<p>Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note</i>: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that the picture has applied to when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

18.2.15 Changed attribute for xfrm element (Part 1, §21.3.2.28)

Attributes	Description
flipH (Horizontal Flip) Namespace: .../drawingml/2006/main	<p>Specifies a horizontal flip. When true, this attribute defines that the shape is flipped horizontally about the center of its bounding box.</p> <p>[Example: The following illustrates the effect of a horizontal flip.</p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
flipV (Vertical Flip) Namespace: .../drawingml/2006/main	<p>Specifies a vertical flip. When true, this attribute defines that the group is flipped vertically about the center of its bounding box.</p> <p>[Example: The following illustrates the effect of a vertical flip.</p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema boolean datatype.</p>
rot (Rotation) Namespace: .../drawingml/2006/main	<p>Specifies the rotation of the Graphic Frame. The units for which this attribute is specified in reside within the simple type definition referenced below.</p> <p>The possible values for this attribute are defined by the ST_Angle simple type (Part 1, §20.1.10.3).</p>

18.2.16 Changed attribute for relIds element (Part 1, §21.4.2.22)

Attributes	Description
cs (Explicit Relationship to Diagram Colors Part) Namespace: .../officeDocument	<p>Specifies the relationship ID for the explicit relationship to the Diagram Colors part used by this diagram.</p> <p>This relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramColors or the document shall be considered non-conformant.</p>

Attributes	Description
/2006/relationships	The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).
dm (Explicit Relationship to Diagram Data Part) Namespace: .../officeDocument/2006/relationships	Specifies the relationship ID for the explicit relationship to the Diagram Data part used by this diagram. This relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramData or the document shall be considered non-conformant. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).
lo (Explicit Relationship to Diagram Layout Definition Part) Namespace: .../officeDocument/2006/relationships	Specifies the relationship ID for the explicit relationship to the Diagram Layout Definition part used by this diagram. This relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramLayout or the document shall be considered non-conformant. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).
qs (Explicit Relationship to Style Definition Part) Namespace: .../officeDocument/2006/relationships	Specifies the relationship ID for the explicit relationship to the Diagram Style part used by this diagram. This relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/diagramQuickStyle or the document shall be considered non-conformant. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.17 Changed attribute for shape element (Part 1, §21.4.2.27)

Attributes	Description
blip (Relationship to Image Part) Namespace: .../officeDocument/2006/relationships	Specifies the relationship ID of the explicit relationship to an image that shall be used as the image for the contents of this shape. This relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant. The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).

18.2.18 Changed attribute for spPr element (Part 1, §21.4.3.7)

Attributes	Description
bwMode (Black and White Mode) Namespace: .../drawingml/2006/main	<p>Specifies that the picture should be rendered using only black and white coloring. That is the coloring information for the picture should be converted to either black or white when rendering the picture.</p> <p>No gray is to be used in rendering this image, only stark black and stark white.</p> <p>[<i>Note</i>: This does not mean that the picture itself that is stored within the file is necessarily a black and white picture. This attribute instead sets the rendering mode that the picture has applied to when rendering. <i>end note</i>]</p> <p>The possible values for this attribute are defined by the ST_BlackWhiteMode simple type (Part 1, §20.1.10.10).</p>

18.2.19 Changed attribute for sp3d element (Part 1, §21.4.5.6)

Attributes	Description
contourW (Contour Width) Namespace: .../drawingml/2006/main	<p>Defines the width of the contour on the shape.</p> <p>[<i>Example</i>: Consider the following example of a contourW in use within the sp3d element:</p> <pre><a:sp3d extrusionH="165100" contourW="50800" prstMaterial="plastic"> <a:bevelT w="254000" h="254000"/> <a:bevelB w="254000" h="254000"/> <a:extrusionClr> <a:srgbClr val="FF0000"/> </a:extrusionClr> <a:contourClr> <a:schemeClr val="accent3"/> </a:contourClr> </a:sp3d></pre> <p>In this example, we see a countourW defined as 50800. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>

Attributes	Description
<p>extrusionH (Extrusion Height)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Defines the height of the extrusion applied to the shape.</p> <p>[<i>Example:</i> Consider the following example of an extrusionH in use within the sp3d element:</p> <pre><a:sp3d extrusionH="165100" contourW="50800" prstMaterial="plastic"> <a:bevelT w="254000" h="254000"/> <a:bevelB w="254000" h="254000"/> <a:extrusionClr> <a:srgbClr val="FF0000"/> </a:extrusionClr> <a:contourClr> <a:schemeClr val="accent3"/> </a:contourClr> </a:sp3d></pre> <p>In this example, we see a extrusionH defined as 165100. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PositiveCoordinate simple type (Part 1, §20.1.10.42).</p>
<p>prstMaterial (Preset Material Type)</p> <p>Namespace: .../drawingml/2006/main</p>	<p>Defines the preset material that is combined with the lighting properties to give the final look and feel of a shape.</p> <p>[<i>Example:</i> Consider the following example of a prstMaterial in use within the sp3d element:</p> <pre><a:sp3d extrusionH="165100" contourW="50800" prstMaterial="plastic"> <a:bevelT w="254000" h="254000"/> <a:bevelB w="254000" h="254000"/> <a:extrusionClr> <a:srgbClr val="FF0000"/> </a:extrusionClr> <a:contourClr> <a:schemeClr val="accent3"/> </a:contourClr> </a:sp3d></pre> <p>In this example, we see a prstMaterial defined as plastic. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_PresetMaterialType simple type (Part 1, §20.1.10.50).</p>

Attributes	Description
z (Shape Depth) Namespace: .../drawingml/2006/main	Defines the z coordinate for the 3D shape. The possible values for this attribute are defined by the ST_Coordinate simple type (Part 1, §20.1.10.16).

19. VML Reference Material

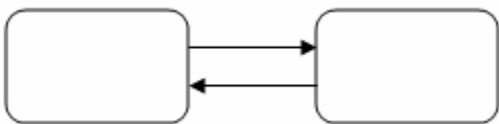
[No documentation has been entered for this section heading.]

19.1 VML

VML is a language for defining graphical objects in cases where DrawingML does not apply, such as text boxes and shapes in WordprocessingML documents and comments and controls in SpreadsheetML documents. The urn:schemas-microsoft-com:vml namespace provides the base elements and attributes for defining shape primitives. The urn:schemas-microsoft-com:office:office, urn:schemas-microsoft-com:office:word, urn:schemas-microsoft-com:office:excel and urn:schemas-microsoft-com:office:powerpoint namespaces define elements that layer on information beyond the baseline graphical definition. To maintain backward compatibility, all VML namespaces defined in ECMA-376 maintain the legacy namespace structure used by the existing corpus of binary documents.

[*Note:* The VML format is a legacy format used in an existing corpus of binary documents and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a transitional format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML .*end note*]

[*Example:* Assume the following shapes exist in a WordprocessingML document:



The drawing consists of four shapes. The arrows are specified by extending the shape type base definition in the shapetype element. Each shape representing an arrow references the shapetype it is extending via its type attribute.

```
<v:shapetype id="_x0000_t32" coordsize="21600,21600" o:spt="32" o:oned="t"
  path="m,l21600,21600e" filled="f">
  <v:path arrowok="t" fillok="f" o:connecttype="none"/>
  <o:lock v:ext="edit" shapetype="t"/>
</v:shapetype>
```

```
<v:shape id="_x0000_s1030" type="#_x0000_t32" style="position:absolute;left:0;
  text-align:left;margin-left:105pt;margin-top:36pt;width:48pt;height:0;flip:x;
  z-index:251661312" o:connectortype="straight">
```



```
<v:stroke endarrow="block"/>
</v:shape>
```

```
<v:shape id="_x0000_s1029" type="#_x0000_t32" style="position:absolute;left:0;
text-align:left;margin-left:105pt;margin-top:21.75pt;width:48pt;height:0;
z-index:251660288" o:connectortype="straight">
  <v:stroke endarrow="block"/>
</v:shape>
```

The rounded rectangles use the VML roundrect element.

```
<v:roundrect id="_x0000_s1028" style="position:absolute;left:0;
text-align:left;margin-left:153pt;margin-top:8.25pt;width:68.25pt;height:42pt;
z-index:251659264" arcsize="10923f"/>
```

```
<v:roundrect id="_x0000_s1027" style="position:absolute;left:0;
text-align:left;margin-left:36.75pt;margin-top:8.25pt;width:68.25pt;
height:42pt;z-index:251658240" arcsize="10923f"/>
```

end example]

Throughout VML, numeric values that are allowed to take units can be specified in: cm (centimeters), mm (millimeters), in (inches), pt (points), pc (picas), px (pixels).

19.1.1 Table of Contents

This subclause is informative.

19.1.2	Elements	296
19.1.2.1	arc (Arc Segment)	296
19.1.2.2	background (Document Background).....	323
19.1.2.3	curve (Bezier Curve)	326
19.1.2.4	f (Single Formula).....	353
19.1.2.5	fill (Shape Fill Properties).....	357
19.1.2.6	formulas (Set of Formulas)	368
19.1.2.7	group (Shape Group)	369
19.1.2.8	h (Shape Handle)	391
19.1.2.9	handles (Set of Handles).....	395
19.1.2.10	image (Image File)	395
19.1.2.11	imagedata (Image Data)	425
19.1.2.12	line (Line)	433
19.1.2.13	oval (Oval).....	460
19.1.2.14	path (Shape Path)	486
19.1.2.15	polyline (Multiple Path Line)	494
19.1.2.16	rect (Rectangle)	521
19.1.2.17	roundrect (Rounded Rectangle)	547
19.1.2.18	shadow (Shadow Effect)	573

19.1.2.19	shape (Shape Definition)	579
19.1.2.20	shapetype (Shape Template)	607
19.1.2.21	stroke (Line Stroke Settings)	634
19.1.2.22	textbox (Text Box)	647
19.1.2.23	textpath (Text Layout Path)	659
19.1.3	Simple Types	671
19.1.3.1	ST_EditAs (Shape Grouping Types)	671
19.1.3.2	ST_Ext (VML Extension Handling Behaviors)	672
19.1.3.3	ST_FillMethod (Gradient Fill Computation Type)	672
19.1.3.4	ST_FillType (Shape Fill Type)	673
19.1.3.5	ST_ImageAspect (Image Scaling Behavior)	674
19.1.3.6	ST_ShadowType (Shadow Type)	674
19.1.3.7	ST_StrokeArrowLength (Stroke Arrowhead Length)	675
19.1.3.8	ST_StrokeArrowType (Stroke Arrowhead Type)	675
19.1.3.9	ST_StrokeArrowWidth (Stroke Arrowhead Width)	676
19.1.3.10	ST_StrokeEndCap (Stroke End Cap Type)	676
19.1.3.11	ST_StrokeJoinStyle (Line Join Type)	677
19.1.3.12	ST_StrokeLineStyle (Stroke Line Style)	677

End of informative text.

19.1.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:vml namespace:

[*Note:* As the VML format is a format provided for backward compatibility, many VML elements are defined in the same urn:schemas-microsoft-com:vml namespace that is already used by millions of documents already using VML. *end note*]

19.1.2.1 arc (Arc Segment)

This element specifies an arc defined as a segment of an oval. The CSS2 style content width and height define the width and height of that oval. The arc is defined by the intersection of the oval with the start and end radius vectors given by the angles. The angles are calculated on the basis of a circle (width equal to height) which is then scaled anisotropically to the desired width and height.

[*Example:* The following specifies a simple half-circle arc open at the top:

```
<v:arc
  style="position:relative;top:120;left:20;width:200;height:200"
  startangle="90" endangle="270">
</v:arc>
```

The shape looks like this:



end example]


Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>alt (Alternate Text)</p>	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre>


Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderbottomcolor (Bottom Border Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderleftcolor (Border Left Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderrightcolor (Border Right Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bordertopcolor (Border Top Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bullet (Graphical Bullet)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is a graphical bullet. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>button (Button Behavior Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape exhibits button press behavior on click. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>bwmode (Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is <code>auto</code>, which uses <code>o:bwnormal</code> for normal black-and-white rendering and <code>o:bwpure</code> for pure black-and-white rendering.</p> <p><code>bwnormal</code> and <code>bwpure</code> are subordinate to <code>bwmode</code>. If <code>bwmode</code> is <code>"auto"</code> then the value for <code>bwnormal</code> or <code>bwpure</code> is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [Example: Normal B&W might allow greyscale and pure B&W might not. <i>end example]</i></p> <p>[Example: This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>


Attributes	Description
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p><i>[Example:</i></p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p><i>[Example: The snippets below are equivalent:</i></p> <pre>... .narrowstyle {width:50;height:100} ...</pre>

Attributes	Description
	<pre><v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>clip (Clipping Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[<i>Example:</i> The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[<i>Example:</i> The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"></pre>

Attributes	Description
	<p data-bbox="456 247 613 279"></v:shape></p>  <p data-bbox="415 457 574 489"><i>end example]</i></p> <p data-bbox="415 527 1377 590">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p data-bbox="142 611 350 716">dgmlayout (Diagram Node Layout Identifier)</p> <p data-bbox="142 751 350 884">Namespace: urn:schemas-microsoft-com:office:office</p>	<p data-bbox="415 611 1393 716">Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p data-bbox="415 720 532 751"><i>[Example:</i></p> <pre data-bbox="456 789 854 852"><v:shape ... dgmlayout="1"> </v:shape></pre> <p data-bbox="415 894 574 926"><i>end example]</i></p> <p data-bbox="415 963 1458 1026">The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p data-bbox="142 1047 331 1180">dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p data-bbox="142 1222 350 1354">Namespace: urn:schemas-microsoft-com:office:office</p>	<p data-bbox="415 1047 1438 1152">Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p data-bbox="415 1226 532 1257"><i>[Example:</i></p> <pre data-bbox="456 1295 854 1358"><v:shape ... dgmlayout="1"> </v:shape></pre> <p data-bbox="415 1400 574 1432"><i>end example]</i></p> <p data-bbox="415 1470 1458 1533">The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p data-bbox="142 1556 321 1661">dgmnodekind (Diagram Node Identifier)</p> <p data-bbox="142 1696 350 1829">Namespace: urn:schemas-microsoft-com:office:office</p>	<p data-bbox="415 1556 1446 1619">Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p data-bbox="415 1661 532 1692"><i>[Example:</i></p> <pre data-bbox="456 1730 886 1793"><v:shape ... dgmnodekind="1"> </v:shape></pre> <p data-bbox="415 1835 574 1866"><i>end example]</i></p>

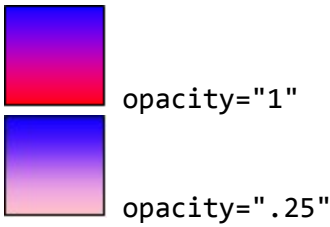
Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>endAngle (Ending Angle)</p>	<p>Specifies the angle that defines the endpoint of the arc. The angle is measured in degrees clockwise from the vertical. Default is 90.</p> <p>[Example: This arc ends at the bottom center of the shape's region:</p> <pre><v:arc ... endangle="180" ... > </v:arc></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type</p>

Attributes	Description
filled (Shape Fill Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>(§20.1.2.3).</p> <p>Specifies whether the closed path is filled. Default is <code>true</code>. This attribute is overridden by the <code>fill on</code> attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
forcedash (Force Dashed Outline) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that a shape is a horizontal rule. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
hralign (Horizontal	<p>Specifies the alignment of a horizontal rule. Default is <code>left</code>.</p>




Attributes	Description
<p>Rule Alignment)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
<p>href (Hyperlink Target)</p>	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hrnoshade (Horizontal Rule 3D Shading Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hrpct (Horizontal Rule Length Percentage)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>hrstd (Horizontal Rule Standard Display Toggle)</p>	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p>

Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p>end example]</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"></pre>

Attributes	Description
	<p><code></v:fill></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>spt (Optional Number)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>startAngle (Starting Angle)</p>	<p>Specifies an angle that defines the starting point of the arc. The angle is measured in degrees clockwise from the vertical.</p> <p>Default is 0.</p> <p>[Example: This arc begins in the upper-right quadrant:</p> <pre><v:arc ... startangle="45" ... > </v:arc></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
<p>strokecolor (Shape Stroke Color)</p>	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available</p>

Attributes	Description								
	<p>here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></p> <table> <tr> <th data-bbox="415 703 664 751">Property</th><th data-bbox="664 703 1482 751">Description</th></tr> <tr> <td data-bbox="415 751 664 1020">flip</td><td data-bbox="664 751 1482 1020"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> <tr> <td data-bbox="415 1020 664 1430">height</td><td data-bbox="664 1020 1482 1430"> <p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. </td></tr> <tr> <td data-bbox="415 1430 664 1869">left</td><td data-bbox="664 1430 1482 1869"> <p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the </td></tr> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the
Property	Description								
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 								
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 								
left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the 								

Attributes	Description	
		parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page.

Attributes	Description	
		<ul style="list-style-type: none"> • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page

Attributes	Description	
		<ul style="list-style-type: none"> • text • line
	mso-wrap-distance-bottom	Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-left	Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-right	Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the

Attributes	Description	
		normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none"> • hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • inherit - The visibility state is inherited from the parent of the shape.
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Uses the order that the shapes appear in the page,

Attributes	Description	
		<p>bottom to top.</p> <ul style="list-style-type: none">• <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.
	The following properties are only used by the textbox element (§19.1.2.22):	
	Property	Description
	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none">• horizontal - Text is displayed horizontally.• vertical - Text is displayed vertically.• vertical-ideographic - Ideographic text is displayed vertically.• horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.</p>
	mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p>
	mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is false.</p>
	mso-layout-flow-alt	<p>Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.</p>
mso-next-textbox	<p>Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.</p>	

Attributes	Description											
	mso-rotate	Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are: <ul style="list-style-type: none">• 0• 90• 180• -90										
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.										
	v-text-anchor	Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are: <ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline										
The following properties are only used by the textpath element (§19.1.2.23):												
<table><tr><th>Property</th><th>Description</th></tr><tr><td>font</td><td>Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr><tr><td>font-family</td><td>Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr><tr><td>font-size</td><td>Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr><tr><td>font-style</td><td>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</td></tr></table>			Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:
Property	Description											
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.											
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.											
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.											
font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:											

Attributes	Description																			
		<ul style="list-style-type: none">normalitalicoblique - Treated the same as italic.																		
	font-variant	<p>Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are:</p> <ul style="list-style-type: none">normalsmall-caps																		
	font-weight	<p>Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td rowspan="5">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr><tr><td>300</td></tr><tr><td>400</td><td rowspan="7">Treated as bold.</td></tr><tr><td>bold</td></tr><tr><td>bolder</td></tr><tr><td>500</td></tr><tr><td>600</td></tr><tr><td>700</td></tr><tr><td>800</td></tr><tr><td>900</td></tr></table>		Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	Treated as bold.	bold	bolder	500	600	700	800	900
		Value	Description																	
		normal	Treated as non-bold.																	
lighter																				
100																				
200																				
300																				
400	Treated as bold.																			
bold																				
bolder																				
500																				
600																				
700																				
800																				
900																				
mso-text-shadow	<p>Specifies whether a shadow is applied to the text on a text path. Default is false.</p>																			
text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none">none																			

Attributes	Description	
		<ul style="list-style-type: none"> • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are: <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height 	

Attributes	Description																
	<p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 1031 1479 1667"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </pre>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																

Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>title (Shape Title)</p>	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... title="tooltip" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is</p>

Attributes	Description
	<p>tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to <i>tight</i> or <i>through</i>.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Arc](#)) is located in §A.6.1. *end note*]

19.1.2.2 background (Document Background)

This element describes the fill of the background of a page using vector graphics fills. Fills consist of simple colors, more advanced effects defined through the fill element (§19.1.2.5), or images.

[Example: The following shades the page background a pale red:

```
<v:background fillcolor="#c0504d">
</v:background>
```


This uses the fill element (§19.1.2.5) to create a gradient background fill:

```
<v:background>
  <v:fill type="gradient" color="#c0504d" color2="#f0504d" angle="45"/>
</v:background>
```

end example]

Attributes	Description
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is <i>auto</i>, which uses <i>o:bwnormal</i> for normal black-and-white rendering and <i>o:bwpure</i> for pure black-and-white rendering</p> <p>[Example: This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwnormal (Normal Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in a pale grayscale in a normal black-and-white environment:</p> <pre><v:shape ... o:bwmode="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwpure (Pure Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in high contrast when in a pure black-and-white environment:</p> <pre><v:shape ... o:bwmode="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[<i>Example:</i> This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... ></pre>

Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>filled (Shape Fill Toggle)</p>	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>targetscreensize (Target Screen Size)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the target resolution used for WordprocessingML documents with a gradient or picture filled background. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • 544,376 • 640,480 • 720,512 • 800,600 • 1024,768 • 1152,862

Attributes	Description
	The possible values for this attribute are defined by the ST_ScreenSize simple type (§19.2.3.23).

[Note: The W3C XML Schema definition of this element’s content model ([CT_Background](#)) is located in §A.6.1.
end note]

19.1.2.3 **curve (Bezier Curve)**

This element is used to draw a cubic bézier curve.

The following properties of the style attribute are ignored:

- top
- margin-top
- center-y
- left
- margin-left
- center-x
- width
- height

[Example: The following specifies a simple curve that opens upward:

```
<v:curve id="mycurve"
from="10pt,10pt" to="100pt,10pt"
control1="40pt,30pt" control2="85pt,30pt">
</v:curve>
```

This shape is created:



end example]

Attributes	Description
allowincell (Allow in Table Cell) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether a shape can be placed in a table. Default is false. [Example: <v:shape ... o:allowincell="true" ... > </v:shape> end example]


Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
allowoverlap (Allow Shape Overlap) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
alt (Alternate Text)	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor	<p>Specifies the left border color of an inline shape. Default is no value.</p>


Attributes	Description
<p>(Border Left Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderrightcolor (Border Right Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bordertopcolor (Border Top Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bullet (Graphical Bullet)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is a graphical bullet. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>button (Button Behavior Toggle)</p>	<p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<p>[<i>Example:</i></p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [<i>Example:</i> Normal B&W might allow greyscale and pure B&W might not. <i>end example]</i></p> <p>[<i>Example:</i> This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in a pale grayscale in a normal black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode)	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in high contrast when in a pure black-and-white</p>


Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	environment: <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).
chromakey (Image Transparency Color)	Specifies a color value that is transparent and show anything behind the shape. Default is no value. <i>[Example:</i> <pre><v:image ... chromakey="white" ...> </v:image></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
class (CSS Reference)	Specifies a reference to the definition of a CSS style. Default is no value. <i>[Example:</i> The snippets below are equivalent: <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
clip (Clipping Toggle) Namespace: urn:schemas-	Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region. <i>[Example:</i>

Attributes	Description
microsoft-com:office:office	<pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
cliptowrap (Clip to Wrapping Polygon) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
connectortype (Shape Connector Type) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
control1 (First Curve Control Point)	<p>Specifies the first control point for the curve, given in the coordinate space of the parent element. Default is "10,10". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p>[Example:</p> <pre><v:curve ... control1="20,30" ... > </v:curve></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
control2 (Second Curve Control Point)	<p>Specifies the second control point for the curve, given in the coordinate space of the parent element. Default is "20,0". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p>[Example:</p> <pre><v:curve ... control2="50,20" ... > </v:curve></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which</p>

Attributes	Description
	<p>the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>dgmlayout (Diagram Node Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>

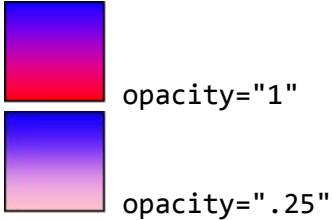
Attributes	Description
dgmnodekind (Diagram Node Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
doubleclicknotify (Double-click Notification Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
filled (Shape Fill Toggle)	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p>


Attributes	Description
	<p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>from (Curve Starting Point)</p>	<p>Specifies the starting point of the line in the coordinate space of the parent element. Default is "0,0". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p>[Example:</p> <pre><v:curve ... from="10,10" ... > </v:curve></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hr (Horizontal Rule Toggle)</p>	<p>Specifies that a shape is a horizontal rule. Default is false.</p>



Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the alignment of a horizontal rule. Default is left.</p> <p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
href (Hyperlink Target)	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hrpct (Horizontal Rule Length Percentage)	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
hrstd (Horizontal Rule Standard Display Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetmode (Text Inset Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
insetpen (Inset	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather</p>

Attributes	Description
Border From Path)	<p>than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
ole (Embedded Object Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
oleicon (Embedded Object Icon Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
oned (Shape Handle Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
opacity (Fill Color Opacity)	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. <i>[Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p><i>[Example: The red color is 25% opaque:</i></p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p><i>[Example:</i></p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
print (Print Toggle)	<p>Specifies whether the shape is printed. Default is true.</p> <p><i>[Example:</i></p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
regroupid (Regroup ID) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
spid (Optional String) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
spt (Optional Number) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p>

Attributes	Description									
	<p>[Example: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> end example]</p>									
	<table> <tr> <th data-bbox="415 495 664 537">Property</th><th data-bbox="664 495 1484 537">Description</th></tr> <tr> <td data-bbox="415 537 664 810">flip</td><td data-bbox="664 537 1484 810"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> <tr> <td data-bbox="415 810 664 1220">height</td><td data-bbox="664 810 1484 1220"> <p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. </td></tr> <tr> <td data-bbox="415 1220 664 1696">left</td><td data-bbox="664 1220 1484 1696"> <p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. </td></tr> <tr> <td data-bbox="415 1696 664 1885">margin-bottom</td><td data-bbox="664 1696 1484 1885"> <p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> </td></tr> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. 	margin-bottom
Property	Description									
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 									
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 									
left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. 									
margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>									

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-	Specifies the horizontal positioning data for objects in

Attributes	Description	
	horizontal	<p>WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change</p>

Attributes	Description	
		the origin.
	mso-wrap-distance-left	Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-right	Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.

Attributes	Description	
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only <code>inherit</code> and <code>hidden</code> are used; any other values are mapped to <code>inherit</code>. Default is <code>inherit</code>. Allowed values are:</p> <ul style="list-style-type: none"> • hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • inherit - The visibility state is inherited from the parent of the shape.
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Uses the order that the shapes appear in the page, bottom to top. • <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.

Attributes	Description																		
	<p>The following properties are only used by the textbox element (§19.1.2.22):</p> <table border="1"> <thead> <tr> <th data-bbox="418 352 662 401">Property</th><th data-bbox="662 352 1481 401">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="418 401 662 667">direction</td><td data-bbox="662 401 1481 667"> <p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> ltr - Text is displayed left-to-right. rtl - Text is displayed right-to-left. </td></tr> <tr> <td data-bbox="418 667 662 1045">layout-flow</td><td data-bbox="662 667 1481 1045"> <p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> horizontal - Text is displayed horizontally. vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally. </td></tr> <tr> <td data-bbox="418 1045 662 1157">mso-direction-alt</td><td data-bbox="662 1045 1481 1157"> <p>Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.</p> </td></tr> <tr> <td data-bbox="418 1157 662 1268">mso-fit-shape-to-text</td><td data-bbox="662 1157 1481 1268"> <p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p> </td></tr> <tr> <td data-bbox="418 1268 662 1379">mso-fit-text-to-shape</td><td data-bbox="662 1268 1481 1379"> <p>Specifies whether the text stretches to fit the textbox. Default is false.</p> </td></tr> <tr> <td data-bbox="418 1379 662 1535">mso-layout-flow-alt</td><td data-bbox="662 1379 1481 1535"> <p>Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.</p> </td></tr> <tr> <td data-bbox="418 1535 662 1646">mso-next-textbox</td><td data-bbox="662 1535 1481 1646"> <p>Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.</p> </td></tr> <tr> <td data-bbox="418 1646 662 1864">mso-rotate</td><td data-bbox="662 1646 1481 1864"> <p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180 </td></tr> </tbody> </table>	Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> ltr - Text is displayed left-to-right. rtl - Text is displayed right-to-left. 	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> horizontal - Text is displayed horizontally. vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally. 	mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.</p>	mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p>	mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is false.</p>	mso-layout-flow-alt	<p>Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.</p>	mso-next-textbox	<p>Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.</p>	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180
Property	Description																		
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> ltr - Text is displayed left-to-right. rtl - Text is displayed right-to-left. 																		
layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> horizontal - Text is displayed horizontally. vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally. 																		
mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.</p>																		
mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p>																		
mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is false.</p>																		
mso-layout-flow-alt	<p>Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.</p>																		
mso-next-textbox	<p>Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.</p>																		
mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180 																		

Attributes	Description	
		<ul style="list-style-type: none"> • -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> • top • middle • bottom • top-center • middle-center • bottom-center • top-baseline • bottom-baseline • top-center-baseline • bottom-center-baseline
The following properties are only used by the textpath element (§19.1.2.23):		
Property		Description
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	
font-style	<p>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</p> <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. 	
font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are:	

Attributes	Description																		
		<ul style="list-style-type: none">normalsmall-caps																	
	font-weight	<p>Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td rowspan="6">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr><tr><td>300</td></tr><tr><td>400</td></tr><tr><td>bold</td><td rowspan="7">Treated as bold.</td></tr><tr><td>bolder</td></tr><tr><td>500</td></tr><tr><td>600</td></tr><tr><td>700</td></tr><tr><td>800</td></tr><tr><td>900</td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.	bolder	500	600	700	800	900
	Value	Description																	
	normal	Treated as non-bold.																	
	lighter																		
	100																		
	200																		
300																			
400																			
bold	Treated as bold.																		
bolder																			
500																			
600																			
700																			
800																			
900																			
mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.																		
text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none">noneunderlineoverlineline-throughblink																		
v-rotate-	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.																		

Attributes	Description	
	letters	
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip 		

Attributes	Description																
	<ul style="list-style-type: none"> • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 848 1479 1486"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																

Attributes	Description
	datatype.
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... title="tooltip" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
to (Curve Ending Point)	<p>Specifies the ending point of the line in the coordinate space of the parent element. Default is "30,20". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p>[Example:</p> <pre><v:curve ... to="40,40" ... > </v:curve></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
userdrawn (Exists In Master Slide) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
userhidden (Hide Script Anchors) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... ></pre>

Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
wrapcoords (Shape Bounding Polygon)	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre> <v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Curve](#)) is located in §A.6.1. *end note]*

19.1.2.4 f (Single Formula)

This element defines a single value as the result of the evaluation of an expression. The expression is defined by the eqn attribute and has the general form of an operation followed by up to three arguments, which consist of adjustment values (see the adj attribute of the shape element (§19.1.2.19)), the results of earlier formulas, fixed numbers or pre-defined values. Each f value is referenced using "@" followed by a number corresponding to the zero-based index for that value in the list of f elements. [Example: For example, the value of the second f element is referenced as "@2". *end example]*

[Example: The following defines a blue arrow pointing to the right:

```

<v:shape coordsize="21600,21600" adj="18000,5400,10800"
  path="m @0,0 l @0,@1 0,@1 0,@3 @0,@3 @0,21600 21600,10800 x e"
  style='left:50pt;top:50pt;width:90pt;height:30pt'
  fillcolor="#4f81bd" strokecolor="#4f81bd" strokeweight="2pt">
<v:formulas>
  <v:f eqn="val #0"/>
  <v:f eqn="val #1"/>
  <v:f eqn="val #2"/>

```

```
<v:f eqn="sum height 0 #1"/>
<v:f eqn="sum #2 0 #1"/>
<v:f eqn="sum width 0 #0"/>
<v:f eqn="prod @5 @4 #2"/>
<v:f eqn="sum width 0 @6"/>
</v:formulas>
</v:shape>
```

The shape looks like this:



end example]

Attributes	Description																						
eqn (Equation)	<p>Specifies a single formula, which consists of a named operation followed by up to three parameters, typically described as v, P1 and P2. Up to 128 formulas can be specified. These operations are defined (calculation accuracy is discussed below):</p> <table> <tr> <th>Operation</th><th>Description</th></tr> <tr> <td>val</td><td>v Returns the supplied value. Exact.</td></tr> <tr> <td>sum</td><td>$v + P1 - P2$ Addition and subtraction. Exact.</td></tr> <tr> <td>product</td><td>$v \times P1 / P2$ Multiplication and division. Rounds up.</td></tr> <tr> <td>mid</td><td>$(v + P1) / 2$ Simple average. Rounds toward zero.</td></tr> <tr> <td>abs</td><td> v Absolute value. Exact.</td></tr> <tr> <td>min</td><td>min(v, P1) The lesser of two values. Exact.</td></tr> <tr> <td>max</td><td>max(v, P1) The greater of two values. Exact.</td></tr> <tr> <td>if</td><td>$v > 0 ? P1 : P2$ Conditional selection. Exact.</td></tr> <tr> <td>mod</td><td>$\sqrt{v^2 + P1^2 + P1^2}$ Modulus. Inexact.</td></tr> <tr> <td>atan2</td><td>atan2(P1, v) Trigonometric arc tangent of a quotient. Result is in "fd" units or fractional degrees - degrees $\times 2^{16}$.</td></tr> </table>	Operation	Description	val	v Returns the supplied value. Exact.	sum	$v + P1 - P2$ Addition and subtraction. Exact.	product	$v \times P1 / P2$ Multiplication and division. Rounds up.	mid	$(v + P1) / 2$ Simple average. Rounds toward zero.	abs	v Absolute value. Exact.	min	min(v, P1) The lesser of two values. Exact.	max	max(v, P1) The greater of two values. Exact.	if	$v > 0 ? P1 : P2$ Conditional selection. Exact.	mod	$\sqrt{v^2 + P1^2 + P1^2}$ Modulus. Inexact.	atan2	atan2(P1, v) Trigonometric arc tangent of a quotient. Result is in "fd" units or fractional degrees - degrees $\times 2^{16}$.
Operation	Description																						
val	v Returns the supplied value. Exact.																						
sum	$v + P1 - P2$ Addition and subtraction. Exact.																						
product	$v \times P1 / P2$ Multiplication and division. Rounds up.																						
mid	$(v + P1) / 2$ Simple average. Rounds toward zero.																						
abs	v Absolute value. Exact.																						
min	min(v, P1) The lesser of two values. Exact.																						
max	max(v, P1) The greater of two values. Exact.																						
if	$v > 0 ? P1 : P2$ Conditional selection. Exact.																						
mod	$\sqrt{v^2 + P1^2 + P1^2}$ Modulus. Inexact.																						
atan2	atan2(P1, v) Trigonometric arc tangent of a quotient. Result is in "fd" units or fractional degrees - degrees $\times 2^{16}$.																						

Attributes	Description	
		Inexact.
sin	$v \times \sin(P1)$	Sine. Argument is in "fd" units or fractional degrees - degrees $\times 2^{16}$. Inexact.
cos	$v \times \cos(P1)$	Cosine. Argument is in "fd" units or fractional degrees - degrees $\times 2^{16}$. Inexact.
cosatan2	$v \times \cos(\text{atan2}(P2, P1))$	Preserves full accuracy in the intermediate calculation. Inexact.
sinatan2	$v \times \sin(\text{atan2}(P2, P1))$	Preserves full accuracy in the intermediate calculation. Inexact.
sqrt	\sqrt{v}	Square root. Result is positive and rounds down. Inexact.
sumangle	$v + P1 \times 2^{16} - P2 \times 2^{16}$	Adds an existing angle in fd units (v) to two other angles specified in degrees. P1 and P2 are scaled by 2^{16} . Exact.
ellipse	$P2 \sqrt{1 - \left(\frac{v}{P1}\right)^2}$	The eccentricity formula for an ellipse, where v is length of the semiminor axis and P1 is the length of the semimajor axis. Inexact.
tan	$v \times \tan(P1)$	Tangent. Argument is in "fd" units or fractional degrees - degrees $\times 2^{16}$. Inexact.
<p>Formulas are evaluated to full precision, but the result is always a 32-bit integer. Formula authors should avoid formulas which are discontinuous - not only are many of the trigonometric operations inexact, the transformations within the coordinate spaces are also inexact. This can mean that a set of formulas which is discontinuous evaluates to give very different path values with the same input on two different systems.</p> <p>When an operation is marked as exact then a conforming implementation shall always generate the correct arithmetic answer (unless the calculations overflow internally). The product operation is required to round to the nearest integer. If the result is exactly 0.5 then it shall be rounded up to the next numerically greater integer. The mid operation is required to round towards 0.</p> <p>All other operations are inexact, but the implementation shall round non-integral values down (towards -infinity) and should perform internal calculations with this form of</p>		

Attributes	Description																										
	<p>rounding.</p> <p>The arguments used in the evaluation of a formula are normally either fixed numbers, the result of the evaluation of a previous formula or an adjust value - the value of the corresponding entry in the shape adj attribute. Fixed numbers shall be positive integral values in the range 0 to 65535 (unsigned 16-bit numbers). The following named values are defined:</p> <table border="1" data-bbox="415 531 1206 1898"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>@n</td><td>The value of formula n, where n is the zero-based index of the formula in the list of formulas. n shall be less than the current formula index.</td></tr> <tr> <td>#n</td><td>Adjustment (adj) value n. n shall be in the range 0 to 7.</td></tr> <tr> <td>width</td><td>The width defined by the coordsize attribute.</td></tr> <tr> <td>height</td><td>The height defined by the coordsize attribute.</td></tr> <tr> <td>xcenter</td><td>The x ordinate of the center of the coordinate space defined by coordorigin and coordsize.</td></tr> <tr> <td>ycenter</td><td>The y ordinate of the center of the coordinate space defined by coordorigin and coordsize.</td></tr> <tr> <td>xlimo</td><td>The x value of the limo attribute (see also the path element (§19.1.2.14)).</td></tr> <tr> <td>ylimo</td><td>The y value of the limo attribute (see also the path element (§19.1.2.14)).</td></tr> <tr> <td>hasstroke</td><td>1 if the shape has a stroke operation, 0 if it does not, as determined by the on attribute of the stroke element (§19.1.2.21).</td></tr> <tr> <td>hasfill</td><td>1 if the shape has a fill operation, 0 if it does not, as determined by the on attribute of the fill element (§19.1.2.5).</td></tr> <tr> <td>pixellinewidth</td><td>The line width in output device pixels. This is used to outset lines from the edge of a rectangle on the assumption that the implementation draws to lower right pixel in preference to the upper left pixel when a line is on a pixel boundary.</td></tr> <tr> <td>pixelwidth</td><td>The width of the shape in device pixels (i.e., the coordsize width transformed into device space).</td></tr> </tbody> </table>	Value	Description	@n	The value of formula n, where n is the zero-based index of the formula in the list of formulas. n shall be less than the current formula index.	#n	Adjustment (adj) value n. n shall be in the range 0 to 7.	width	The width defined by the coordsize attribute.	height	The height defined by the coordsize attribute.	xcenter	The x ordinate of the center of the coordinate space defined by coordorigin and coordsize.	ycenter	The y ordinate of the center of the coordinate space defined by coordorigin and coordsize.	xlimo	The x value of the limo attribute (see also the path element (§19.1.2.14)).	ylimo	The y value of the limo attribute (see also the path element (§19.1.2.14)).	hasstroke	1 if the shape has a stroke operation, 0 if it does not, as determined by the on attribute of the stroke element (§19.1.2.21).	hasfill	1 if the shape has a fill operation, 0 if it does not, as determined by the on attribute of the fill element (§19.1.2.5).	pixellinewidth	The line width in output device pixels. This is used to outset lines from the edge of a rectangle on the assumption that the implementation draws to lower right pixel in preference to the upper left pixel when a line is on a pixel boundary.	pixelwidth	The width of the shape in device pixels (i.e., the coordsize width transformed into device space).
Value	Description																										
@n	The value of formula n, where n is the zero-based index of the formula in the list of formulas. n shall be less than the current formula index.																										
#n	Adjustment (adj) value n. n shall be in the range 0 to 7.																										
width	The width defined by the coordsize attribute.																										
height	The height defined by the coordsize attribute.																										
xcenter	The x ordinate of the center of the coordinate space defined by coordorigin and coordsize.																										
ycenter	The y ordinate of the center of the coordinate space defined by coordorigin and coordsize.																										
xlimo	The x value of the limo attribute (see also the path element (§19.1.2.14)).																										
ylimo	The y value of the limo attribute (see also the path element (§19.1.2.14)).																										
hasstroke	1 if the shape has a stroke operation, 0 if it does not, as determined by the on attribute of the stroke element (§19.1.2.21).																										
hasfill	1 if the shape has a fill operation, 0 if it does not, as determined by the on attribute of the fill element (§19.1.2.5).																										
pixellinewidth	The line width in output device pixels. This is used to outset lines from the edge of a rectangle on the assumption that the implementation draws to lower right pixel in preference to the upper left pixel when a line is on a pixel boundary.																										
pixelwidth	The width of the shape in device pixels (i.e., the coordsize width transformed into device space).																										



Attributes	Description	
	pixelheight	The height of the coordsize in device pixels.
	emuwidth	The width of the coordsize in EMUs.
	emuheight	The height of the coordsize in EMUs.
	emuwidth2	Half the width of the coordsize in EMUs.
	emuheight2	Half the height of the coordsize in EMUs.
<p>The EMU, or English Metric Unit, is the smallest unit of measure in VML and corresponds to 914400 EMU per inch or 12700 EMU per point.</p> <p>See above for an example.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		


[Note: The W3C XML Schema definition of this element's content model ([CT_F](#)) is located in §A.6.1. *end note*]


19.1.2.5 fill (Shape Fill Properties)


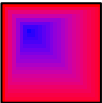
This element specifies how the path should be filled if something beyond a solid color fill is desired. The attributes of the fill element can be used to describe a powerful set of image- or gradient-based fill patterns. Extensions to the VML fill definition are encoded as sub-elements of fill.

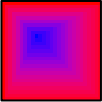
Attributes	Description
alignshape (Align Image With Shape)	<p>Specifies whether an image aligns with the shape. Default is true.</p> <p>[Example: The image displayed in the shape is not rotated even though the shape is rotated 30 degrees:</p> <pre><v:shape coordorigin="0,0" coordsize="200,200" style="top:1;left:1;width:50; height:50;rotation:30" path="m 1,1 l 1,200, 200,200, 200,1 x e"> <v:fill alignshape="false" type="frame" src="myimage.gif"> </v:fill> </v:shape></pre> <p>Applied to a simple square the fill looks like this:</p>


Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>althref (Alternate Image Reference Location)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Defines an alternate reference for an image in Macintosh PICT format.</p> <p>[Example:</p> <pre><v:fill ... althref="myimage.pcz" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>angle (Gradient Angle)</p>	<p>Specifies the direction of a gradient. The vector of a gradient is perpendicular to the vector of the blend direction from one color to another. The default value is zero degrees, which is a horizontal vector from left to right. Positive angles rotate the gradient in a counter-clockwise direction.</p> <p>[Example: The fill is composed of a 45-degree gradient of two colors. Blue is in the top left corner and red is in the bottom right corner.</p> <pre><v:fill type="gradient" color="red" color2="blue" angle="45"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
<p>aspect (Image Aspect Ratio)</p>	<p>Specifies how the fill image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <ul style="list-style-type: none"> • ignore - Ignore aspect ratio. • atleast - At least as large as defined by the size attribute.


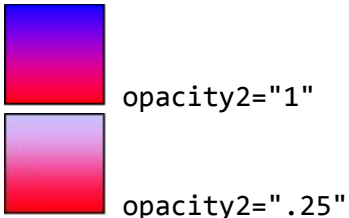


Attributes	Description
	<ul style="list-style-type: none"> • atmost - No larger than that defined by the size attribute. <p>In each case, the size attribute is adjusted to preserve the aspect ratio of the image.</p> <p>[Example: The image that makes up the fill is no larger than 20 points by 20 points, limiting the size of the tiles inside the shape.</p> <pre><v:fill type="tile" aspect="atmost" size="20pt,20pt" src="myimage.gif"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>
color (Primary Color)	<p>Specifies the main fill color; functions the same as the fillcolor attribute of the shape element (§19.1.2.19). This attribute overrides the shape's fillcolor. Default is white. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: The shape is blue:</p> <pre><v:shape ... fillcolor="red" ... > <v:fill color="blue"/> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
color2 (Secondary Color)	<p>Specifies the secondary fill color, used when a fill type is a pattern or a gradient. Default is white. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: The shape is filled with a horizontal gradient with red at the bottom and blue on top:</p> <pre><v:fill type="gradient" color="red" color2="blue"> </v:fill></pre>



Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>colors (Intermediate Colors)</p>	<p>Specifies an array of comma-separated percentage-color pairs that define intermediate colors and their positions in a gradient fill. The primary color, specified either by the fillcolor attribute of the shape element (§19.1.2.19) or the color attribute of the fill element (§19.1.2.5), is used at the 0% endpoint. The secondary color, specified by the color2 attribute of the fill element (§19.1.2.5), is used at the 100% endpoint. The numeric values can also be specified in 1/65536ths if a trailing "f" is supplied. <i>[Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p><i>[Example: The shape is filled with a horizontal gradient colored, from bottom to top, red, yellow, green, blue:</i></p> <pre><v:fill type="gradient" color="red" color2="blue" colors="30% yellow,70% green"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>detectmouseclick (Detect Mouse Click)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a mouse click is detected on the fill of a shape.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>focus (Gradient Center)</p>	<p>Specifies the center starting position of a gradient. Values are in the range 100% to -100%. Default is 0.</p> <p>A value of 100% or -100% reverses the direction of the gradient (in effect swapping color and color2). A value of 50% changes the gradient so that color is at both ends and color2 is in the middle. A value of -50% changes the gradient so that color2 is at both ends and color is in the middle.</p>

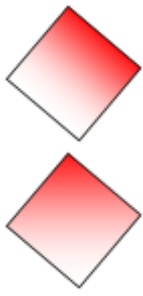
Attributes	Description
	<p>[<i>Example:</i> The shape is filled with a horizontal gradient with red at both ends and blue in the middle:</p> <pre><v:fill type="gradient" color="red" color2="blue" focus="50%"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>focusposition (Radial Gradient Center)</p>	<p>Specifies the position of the center rectangle of a radial gradient. The vector is a fraction of the width and height of the shape. The first is a percentage of the fill to the left edge; the second is a percentage of the fill to the top. Default is 0,0. To position a radial fill at the center of a shape, use a value of 50%,50%.</p> <p>[<i>Example:</i> The shape is filled with a rectangular gradient positioned in the top-left quadrant of the shape. The interior of the gradient is blue and the exterior is red:</p> <pre><v:fill type="gradientradial" color="red" color2="blue" focusposition="25%,25%"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>focussize (Radial Gradient Size)</p>	<p>Specifies the size of the center rectangle of a radial gradient. The vector is a fraction of the width and height of the shape. The first is a percentage of the fill to the right edge; the second is a percentage of the fill to the bottom. Default is 0,0.</p> <p>A focussize value of 100%,100% and a focusposition of 0,0 makes color2 dominate the gradient completely. Small values of around 10%,10% are recommended for balanced gradients.</p> <p>[<i>Example:</i> The shape is filled with a rectangular gradient positioned in the top-left</p>


Attributes	Description
	<p>quadrant of the shape. The interior of the gradient is blue and the exterior is red. The red portion is wider on the bottom and right sides of the blue region. The pure blue region is 25% the width and 25% the height of the shape:</p> <pre><v:fill type="gradientradial" color="red" color2="blue" focussize="25%,25%" focusposition="25%,25%"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>href (Hyperlink Target)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID of the relationship to the image used for this fill. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information for the fill:</p> <pre>< ... r:id="rId10" /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
<p>id (Unique</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p>

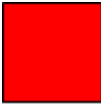











Attributes	Description
Identifier)	<p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
method (Gradient Fill Method)	<p>Specifies the method used to generate the transition from color to color2 in a gradient fill. Default is <i>sigma</i>.</p> <p>[Example:</p> <pre><v:fill type="gradient" color="red" color2="blue" method="any"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillMethod simple type (§19.1.3.3).</p>
on (Fill Toggle)	<p>Specifies whether to fill the shape. Default is <i>true</i>. This attribute overrides the shape's fill attribute.</p> <p>[Example: The shape has a transparent fill:</p> <pre><v:shape ... fill="true" ... > <v:fill color="red" on="false"> </v:fill> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Primary Color Opacity)	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f"</p>

Attributes	Description
	<p>represents 52429/65536 or 0.8.</p> <p>[<i>Example</i>: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre>  <p>opacity="1"</p>  <p>opacity=".25"</p> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>opacity2 (Secondary Color Opacity)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies the opacity of the secondary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [<i>Example</i>: For example, a value of "52429f" represents 52429/65536 or 0.8. <i>end example</i>]</p> <p>[<i>Example</i>: The blue color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" o:opacity2=".25"> </v:fill></pre>  <p>opacity2="1"</p>  <p>opacity2=".25"</p> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>origin (Fill Image Origin)</p>	<p>Specifies the position of the origin of a fill image as a point relative to the top left corner of the image. The vector is a fraction of the width and height of the image. Default is the center of the image. These numeric values can also be specified in 1/65536-ths if a trailing "f" is supplied. [<i>Example</i>: For example, a value of "52429f" represents 52429/65536 or 0.8. <i>end example</i>]</p>

Attributes	Description
	<p>[Example: The origin of the image is 25% to the right and 25% above the image's top left corner:</p> <pre><v:fill type="tile" src="myimage.gif" origin="0.25,-0.25"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
position (Fill Image Position)	<p>Specifies the position of the origin of a fill image as a point within its containing shape. The vector is a fraction of the width and height of the shape. These numeric values can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f" represents 52429/65536 or 0.8.</p> <p>[Example: The origin of the image is positioned 25% to the right of the left edge of the shape and 25% down from the shape's top:</p> <pre><v:fill type="tile" src="myimage.gif" position="0.25,0.25"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
recolor (Recolor Fill as Picture)	<p>Specifies that the fill uses an image. Default is false.</p> <p>[Example:</p> <pre><v:fill r:id="rId4" o:title="MyPic" recolor="true" type="frame"> </v:fill></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
relid (Relationship to Part) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the relationship ID of the relationship to the image. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information:</p> <pre><v:fill ... o:relid="rId10" ...> </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
rotate (Rotate Fill with Shape)	<p>Specifies whether the fill is rotated with the shape. Default is false.</p> <p>[Example: The gradient is rotated with the shape:</p> <pre><v:fill color2="white" focus="100%" rotate="true" type="gradient"> </v:fill></pre> <div data-bbox="451 1150 831 1444">  <div data-bbox="609 1266 816 1297" style="position: absolute; left: 375px; top: 603px;">rotate="true"</div> <div data-bbox="609 1413 831 1444" style="position: absolute; left: 375px; top: 673px;">rotate="false"</div> </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
size (Fill Image Size)	<p>Specifies the size of the fill image. Default is the native image pixel size.</p> <p>[Example: The image is reduced in size disproportionately:</p> <pre><v:fill type="tile" src="myimage.gif" size="25pt,15pt"> </v:fill></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Fill Image Source)	<p>Specifies the URL specifying the fill image to use.</p> <p>[Example:</p> <pre><v:fill ... src="myimage.gif" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
title (Title) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the title of an embedded fill image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
type (Fill Type)	<p>Specifies the kind of fill. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • solid • gradient • gradientradial • tile • pattern • frame <p>[Example: Applied to a simple square using the following fill element, the three gradient types look like this:</p>

Attributes	Description						
	<div><div><div><div><v:fill color="red" color2="blue" type="solid"> </v:fill></div><div><div>type="solid"</div></div><div><div>type="gradient"</div></div><div><div>type="gradientradial"</div></div></div></div><div>Applied to a simple square using the following fill elements, the three image types look like this:</div><table><tr><td></td><td><div><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></div></td></tr><tr><td></td><td><div><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></div></td></tr><tr><td></td><td><div><v:fill src="myimage.gif" color="red" color2="blue" type="pattern"> </v:fill></div></td></tr></table><div>end example]</div><div>The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).</div></div>		<div><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></div>		<div><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></div>		<div><v:fill src="myimage.gif" color="red" color2="blue" type="pattern"> </v:fill></div>
	<div><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></div>						
	<div><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></div>						
	<div><v:fill src="myimage.gif" color="red" color2="blue" type="pattern"> </v:fill></div>						

[Note: The W3C XML Schema definition of this element’s content model ([CT_Fill](#)) is located in §A.6.1. end note]

19.1.2.6 formulas (Set of Formulas)

This element defines a set of formulas whose calculated values are referenced by other attributes. Each formula is contained in a child f element (§19.1.2.4).

[Note: The W3C XML Schema definition of this element’s content model ([CT_Formulas](#)) is located in §A.6.1. end note]

19.1.2.7 group (Shape Group)

This element is used to collect shapes and groups so they can be positioned and transformed as a single unit. A group contains group, shapetype, shape, pre-defined shape - arc, curve, image, line, oval, polyline, rect, roundrect - and lock elements.

[Example: The following example defines a few basic parts of a flying saucer graphic. The group consists of five shapes. Each shape's position is determined within the coordinate space of the group, which is defined by the group's attributes.

```
<v:group id="saucer"
  style='position:relative;left:200;top:200;width:50;height:50'
  coordorigin="0,0" coordsize="6000,6000">
  <v:shape id="body"
    style='position:relative;left:234.75pt;top:208.875pt;
    width:235.25pt;height:128.875pt' coordsize="3765,2060"
    path="m1285,2511126,469,580,1009,,1285,25,1412,93,1547,194,1673,
    1017,2026,2312,2060,3209,1756,3765,1388,3278,680,3059,319,2976,,
    1285,251,1285,251xe"
    fillcolor="#bcbcd6" stroked="f">
    <v:path arrowok="t"/>
  </v:shape>
  <v:shape id="canopy"
    style='position:relative;left:314.625pt;top:140.5pt;
    width:104pt;height:102pt' coordsize="1663,1633"
    path="m0,13551177,1498,353,1582,840,1633,1378,1498,1663,1295,
    1545,456,1260,10,1025,,656,260,253,874,,1355,,1355xe"
    fillcolor="#99ebff" stroked="f">
    <v:path arrowok="t"/>
  </v:shape>
  <v:shape id="light1"
    style='position:relative;left:408.625pt;top:268.75pt;
    width:24.25pt;height:27.375pt' coordsize="388,437"
    path="m209,0134,101,,302,125,437,329,327,388,152,209,,209,0xe"
    fillcolor="#fff27f" stroked="f">
    <v:path arrowok="t"/>
  </v:shape>
  <v:shape id="light2"
    style='position:relative;left:356.625pt;top:279.25pt;
    width:28.875pt;height:30pt' coordsize="462,479"
    path="m135,010,186,59,422,344,479,462,228,135,,135,0xe"
    fillcolor="#fff27f" stroked="f">
    <v:path arrowok="t"/>
  </v:shape>
```

```
<v:shape id="light3"
  style='position:relative;left:302.625pt;top:274pt;
  width:23pt;height:23.625pt' coordsize="369,378"
  path="m0,591226,,369,186,243,378,32,363,,59,,59xe"
  fillcolor="#fff27f" stroked="f">
  <v:path arrowok="t"/>
</v:shape>
</v:group>
```





end example]

Attributes	Description
allowincell (Allow in Table Cell) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
allowoverlap (Allow Shape Overlap) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
alt (Alternate Text)	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red"</pre>


Attributes	Description
	<pre>alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor (Border Left Color) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[Example: The snippets below are equivalent:</p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre>

Attributes	Description
	<pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[<i>Example:</i> The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[<i>Example:</i> The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p>

Attributes	Description
	<pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the <code>orgchart</code> value of the <code>editas</code> attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_DiagramLayout</code> simple type (§19.2.3.10).</p>
dgmlayoutmru (Diagram Node Recent Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the <code>orgchart</code> value of the <code>editas</code> attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_DiagramLayout</code> simple type (§19.2.3.10).</p>
dgmnodekind (Diagram Node Identifier) Namespace:	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p>

Attributes	Description
urn:schemas-microsoft-com:office:office	<pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
doubleclicknotify (Double-click Notification Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
editas (Group Diagram Type)	<p>Specifies which diagram type the contained shapes represent. This is used in conjunction with the diagram element (§19.2.2.8). A value of canvas indicates that the group is a regular group of shapes and does not represent a diagram. Other values indicate that the diagram element and its children contain semantic information relevant to that type of diagram, which is represented by the shapes in the group.</p> <p>[Example:</p> <pre><v:group ... editas="orgchart"> </v:group></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_EditAs simple type (§19.1.3.1).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p>

Attributes	Description
	<pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
filled (Shape Fill Toggle)	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the alignment of a horizontal rule. Default is left.</p> <p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).
href (Hyperlink Target) Namespace: urn:schemas-microsoft-com:office:office	Specifies a hyperlink URL target for the shape. Default is no value. <i>[Example:</i> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that the horizontal rule does not have 3-D shading. Default is false. <i>[Example:</i> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	Specifies the length of a horizontal rule as a percentage of page width. Default is 0. <i>[Example:</i> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema float datatype.
hrstd (Horizontal Rule Standard Display Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false. <i>[Example:</i> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type

Attributes	Description
<p>id (Unique Identifier)</p>	<p>(§20.1.2.5).</p> <p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre>

Attributes	Description				
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>				
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p>style (Shape Styling Properties)</p>	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table border="1" data-bbox="415 1661 1479 1854"> <thead> <tr> <th data-bbox="415 1661 664 1707">Property</th><th data-bbox="664 1661 1479 1707">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1707 664 1854">flip</td><td data-bbox="664 1707 1479 1854"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. </td></tr> </tbody> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates.
Property	Description				
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. 				

Attributes	Description	
		<ul style="list-style-type: none"> • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-</p>

Attributes	Description	
	relative	<p>position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies the distance from the top of the shape to the text that</p>

Attributes	Description	
	distance-top	wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are: <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are: <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are: <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.

Attributes	Description							
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none">hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.inherit - The visibility state is inherited from the parent of the shape.						
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">auto - Default position of an element in the flow of the page.<units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.<percentage>- Value expressed as a percentage of the parent object's width.						
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">auto - Uses the order that the shapes appear in the page, bottom to top.<order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.						
<p>The following properties are only used by the textbox element (§19.1.2.22):</p>								
<table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p><ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.</td></tr><tr><td>layout-flow</td><td><p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p></td></tr></table>			Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p>
Property	Description							
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.							
layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p>							

Attributes	Description	
		<ul style="list-style-type: none"> • <code>horizontal</code> - Text is displayed horizontally. • <code>vertical</code> - Text is displayed vertically. • <code>vertical-ideographic</code> - Ideographic text is displayed vertically. • <code>horizontal-ideographic</code> - Ideographic text is displayed horizontally.
	<code>mso-direction-alt</code>	Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code> .
	<code>mso-fit-shape-to-text</code>	Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code> .
	<code>mso-fit-text-to-shape</code>	Specifies whether the text stretches to fit the textbox. Default is <code>false</code> .
	<code>mso-layout-flow-alt</code>	Specifies the alternate layout flow for text in textboxes. This property is used instead of <code>layout-flow</code> when the layout flow is from bottom to top for non-ideographic languages. Its only value is <code>bottom-to-top</code> .
	<code>mso-next-textbox</code>	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	<code>mso-rotate</code>	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	<code>mso-text-scale</code>	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if <code>mso-fit-text-to-shape</code> is <code>true</code> .
	<code>v-text-anchor</code>	<p>Specifies the vertical anchoring of text in a textbox. Default is <code>top</code>. The alignment of a text anchor only becomes evident if <code>mso-fit-text-to-shape</code> is <code>false</code>. This property is different from the <code>vertical-align</code> CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> • <code>top</code> • <code>middle</code> • <code>bottom</code> • <code>top-center</code>

Attributes	Description							
		<ul style="list-style-type: none">• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline						
	The following properties are only used by the textpath element (§19.1.2.23):							
	Property	Description						
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.						
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.						
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.						
	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.						
	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">• normal• small-caps						
	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td>Treated as non-bold.</td></tr><tr><td>lighter</td><td></td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	
	Value	Description						
normal	Treated as non-bold.							
lighter								

Attributes	Description	
		100 200 300 400
		bold bolder 500 600 700 800 900 Treated as bold.
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.
	text-decoration	Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are: <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between

Attributes	Description	
		<p>the letters.</p> <ul style="list-style-type: none"> stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> tightening tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> top left width height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> flip height left margin-left margin-top position rotation top visibility width z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		

Attributes	Description								
<p>tablelimits (Table Row Height Limits)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a list of minimum height values for each row in a table. Default is no value.</p> <p>Used by PresentationML for native tables. This attribute is only useful when the table is made up of shapes that are grouped. When text is added to table cells, the row height can increase. The tablelimits attribute stores the original row height so that if text is deleted, the row height does not fall below the original value.</p> <p>[Example:</p> <pre><v:shape ... o:tablelimits="30pt 20pt" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>								
<p>tableproperties (Table Properties)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a bitmask, represented as an integer, that determines table properties. Only the first three bits of this integer are used. Default is 0.</p> <p>Used by PresentationML for native tables. This attribute is only useful when the table is made up of shapes that are grouped. Allowed values are:</p> <table border="1" data-bbox="415 1037 1260 1234"> <thead> <tr> <th>Bit</th><th>Description</th></tr> </thead> <tbody> <tr> <td>1</td><td>Set if the group of shapes is a table.</td></tr> <tr> <td>2</td><td>Set if the shape is a placeholder.</td></tr> <tr> <td>3</td><td>Set if the table text is bi-directional.</td></tr> </tbody> </table> <p>[Example: Decimal 3 means that bits 1 and 2 are set.</p> <pre><v:shape ... o:tableproperties="3" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Bit	Description	1	Set if the group of shapes is a table.	2	Set if the shape is a placeholder.	3	Set if the table text is bi-directional.
Bit	Description								
1	Set if the group of shapes is a table.								
2	Set if the shape is a placeholder.								
3	Set if the table text is bi-directional.								
<p>target (Hyperlink Display Target)</p>	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 1705 1479 1835"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> </tbody> </table>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.				
Value	Description								
<targetname>	String containing the name of the frame or window in which to load the document.								

Attributes	Description												
	<table border="1"> <tr> <td data-bbox="410 247 625 331"><code>_blank</code></td><td data-bbox="625 247 1485 331">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="410 331 625 415"><code>_media</code></td><td data-bbox="625 331 1485 415">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="410 415 625 499"><code>_parent</code></td><td data-bbox="625 415 1485 499">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="410 499 625 583"><code>_search</code></td><td data-bbox="625 499 1485 583">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="410 583 625 667"><code>_self</code></td><td data-bbox="625 583 1485 667">Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td data-bbox="410 667 625 751"><code>_top</code></td><td data-bbox="625 667 1485 751">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<code>_blank</code>	Specifies that the linked document is loaded into a new blank window. This window is not named.	<code>_media</code>	Specifies that the linked document is loaded into the browser's multimedia pane.	<code>_parent</code>	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	<code>_search</code>	Specifies that the linked document is loaded into the browser's search pane.	<code>_self</code>	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	<code>_top</code>	Specifies that the linked document is loaded into the topmost window.
<code>_blank</code>	Specifies that the linked document is loaded into a new blank window. This window is not named.												
<code>_media</code>	Specifies that the linked document is loaded into the browser's multimedia pane.												
<code>_parent</code>	Specifies that the linked document is loaded into the immediate parent of the document containing the link.												
<code>_search</code>	Specifies that the linked document is loaded into the browser's search pane.												
<code>_self</code>	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).												
<code>_top</code>	Specifies that the linked document is loaded into the topmost window.												
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>												
userdrawn (Exists In Master Slide) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre> <v:shape ... o:userdrawn="true" ... > </v:shape> </pre>												

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Group](#)) is located in §A.6.1. *end note]*

19.1.2.8 h (Shape Handle)

This element defines a single handle, which is a user interface element tied to one or two adj values. Moving the handle changes its linked adj values, which in turn changes formulas and attributes that depend on them. The handle is optionally constrained vertically or horizontally. The linked adj values store the position of the handle in the shape's coordinate space.

[Example: The example below defines a simple kite shape with a resizable width:

```

<v:shape coordsize="200,200" coordorigin="-100,-100" adj="100"
style="width:50;height:50;position:relative"
path="m @1,-50 l 0,-200 @0,-50 0,200 x e">
  <v:formulas>
    <v:f eqn="val #0"/>
    <v:f eqn="sum 0 0 @0"/>
  </v:formulas>
  <v:handles>
    <v:h position="#0,0"/>
  </v:handles>
</v:shape>

```

end example]

Attributes	Description
invx (Invert Handle's X Position)	<p>Specifies whether the x position of the handle should be inverted according to:</p> $x_{\text{new}} = \text{coordorigin}_x + \text{coordsize}_x - x_{\text{old}}$ <p>Default is false.</p> <p>[Example:</p> <pre> <v:handles> <v:h ... invx="true" ... /> </v:handles> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
invy (Invert Handle's Y Position)	<p>Specifies whether the y position of the handle should be inverted according to:</p> $y_{\text{new}} = \text{coordorigin}_y + \text{coordsize}_y - y_{\text{old}}$ <p>Default is false.</p> <p>[Example:</p> <pre> <v:handles> <v:h ... invy="true" ... /> </v:handles> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
map (Handle	<p>Specifies how the x and y positions of the handle are mapped from the coordsize range</p>

Attributes	Description
Coordinate Mapping)	<p>into the specified range. Default is "0,1000".</p> <p>[Example:</p> <pre><v:handles> <v:h ... map="-1000,1000" ... /> </v:handles></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
polar (Handle Polar Center)	<p>Specifies the center position of a handle that uses polar coordinates. If specified, the position attribute is assumed to contain radius and angle values. If omitted, the position attribute is assumed to contain x and y positions. Default is no value.</p> <p>[Example:</p> <pre><v:handles> <v:h ... polar="0,0" ... /> </v:handles></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
position (Handle Position)	<p>Specifies the x and y position of the handle. If the polar attribute is present, defines the handle position using radius and angle values. Default is "0,0".</p> <p>Each value in the vector is one of the following:</p> <ul style="list-style-type: none"> • constant • formula (e.g., @2) • adj value (e.g., #2) • center • topleft • bottomright <p>Each of the above except for an adj value reference fixes the handle position for that dimension. Specifying an adj value allows the handle to move in that dimension and the handle position for that dimension is stored in the adj value.</p> <p>[Example: The handle's x position is fixed but it is free to move in the y dimension:</p> <pre><v:handles> <v:h ... position="topleft,#2" ... /></pre>

Attributes	Description
	<p><code></v:handles></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>radiusrange (Handle Polar Radius Range)</p>	<p>Specifies a range of minimum and maximum values that constrain the radius of a handle using polar coordinates. Default is "0,0". Each value is either a constant or a formula reference. Omitting a value leaves that bound unconstrained.</p> <p>[Example: The polar handle can only be moved within a radius range of 25 to 50.</p> <pre> <v:handles> <v:h ... radiusrange="25,50" ... /> </v:handles> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>switch (Handle Inversion Toggle)</p>	<p>Specifies whether the x and y dimensions of the handle are switched when the shape is taller than it is wide. Default is false. This is useful for shapes with limo stretch behavior.</p> <p>[Example:</p> <pre> <v:handles> <v:h ... switch="true" ... /> </v:handles> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>xrange (Handle X Position Range)</p>	<p>Specifies a range of minimum and maximum values that constrain the x position of a handle. Default is "0,0". Each value is either a constant or a formula reference. Omitting a value leaves that bound unconstrained.</p> <p>[Example: The handle's x position has a maximum bound of 500 and no minimum bound:</p> <pre> <v:handles> <v:h ... xrange=",500" ... /> </v:handles> </pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
yrange (Handle Y Position Range)	<p>Specifies a range of minimum and maximum values that constrain the y position of a handle. Default is "0,0". Each value is either a constant or a formula reference. Omitting a value leaves that bound unconstrained.</p> <p>[<i>Example:</i> The handle's y position has a minimum bound of -500 and no maximum bound:</p> <pre><v:handles> <v:h ... yrange="-500," ... /> </v:handles></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_H](#)) is located in §A.6.1. *end note*]

19.1.2.9 [handles \(Set of Handles\)](#)

This element defines a set of user interface elements which can vary a shape's adj values. All dependent formulas and attributes are recalculated. Each handle is defined by a child h element.

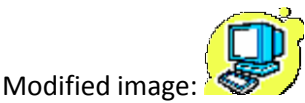
[*Note:* The W3C XML Schema definition of this element's content model ([CT_Handles](#)) is located in §A.6.1. *end note*]

19.1.2.10 [image \(Image File\)](#)

This element is used to draw an image that has been loaded from an external source. There is an implied rectangle that is the same size as the image. Any stroke or fill is applied to this implied rectangle. The stroke is drawn on top of the image. The fill is behind the image and therefore only visible through transparent areas of the image. Image transparency is either encoded in the file or defined via a color value using the chromakey attribute. Unlike the imagedata element (§19.1.2.11), the image element does not have a parent element.



[*Example:*

```
<v:image src="myimage.gif"
  style="position:relative;top:1;left:1;width:50;height:45"
  cropbottom="10%" gamma="0.5" gain="2">
</v:image>
```



end example]

Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>alt (Alternate Text)</p>	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"></pre>



Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bilevel (Image Bilevel Toggle)</p>	<p>Specifies that all colors in the picture shall be converted to either 0 or full intensity component values. This converts a color bitmap to 8 colors and a grayscale bitmap to black and white. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:image ... bilevel="true" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>blacklevel (Image Brightness)</p>	<p>Specifies the image brightness. Default is 0.</p> <p>[Example:</p> <pre><v:image ... blacklevel="0.1" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderbottomcolor (Bottom Border Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p>




Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
borderleftcolor (Border Left Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the left border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
borderrightcolor (Border Right Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the right border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the top border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether the shape is a graphical bullet. Default is false. <i>[Example:</i> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <i>end example]</i>


Attributes	Description
<p>button (Button Behavior Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p> <p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>bwmode (Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [Example: Normal B&W might allow greyscale and pure B&W might not. end example]</p> <p>[Example: This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwnormal (Normal Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>

Attributes	Description
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p><i>[Example:</i></p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p><i>[Example: The snippets below are equivalent:</i></p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>



Attributes	Description
<p>clip (Clipping Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
<p>coordorigin (Coordinate Space Origin)</p>	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p>



Attributes	Description
	<p>[<i>Example</i>: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[<i>Example</i>: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
cropbottom (Image Bottom Crop)	<p>Specifies the how much to crop the image from the bottom up as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f" represents 52429/65536 or 0.8.</p>

Attributes	Description
	<p>[Example:</p> <pre><v:image ... cropbottom="10%" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>cropleft (Image Left Crop)</p>	<p>Specifies how much to crop the image from the left in as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f" represents 52429/65536 or 0.8.</p> <p>[Example:</p> <pre><v:image ... cropleft="10%" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>cropright (Image Right Crop)</p>	<p>Specifies how much to crop the image from the right in as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example:</p> <pre><v:image ... cropright="10%" ...> </v:image></pre>  <p>end example]</p>

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>croptop (Image Top Crop)</p>	<p>Specifies how much to crop the image from the top down as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example:</p> <pre data-bbox="451 617 886 680"><v:image ... croptop="10%" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>dgmlayout (Diagram Node Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre data-bbox="451 1188 854 1251"><v:shape ... dgmlayout="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre data-bbox="451 1696 854 1759"><v:shape ... dgmlayout="1"> </v:shape></pre> <p>end example]</p>

Attributes	Description
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p> <p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>

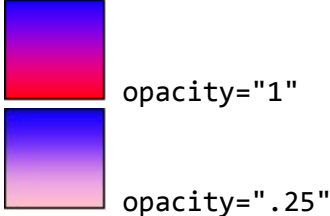
Attributes	Description
filled (Shape Fill Toggle)	<p>Specifies whether the closed path is filled. Default is <code>true</code>. This attribute is overridden by the <code>fill</code> on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
forcedash (Force Dashed Outline) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
gain (Image Intensity)	<p>Specifies an adjustment for the intensity of all colors. Essentially sets how bright white is. Default is 1.</p> <p>[Example:</p> <pre><v:image ... gain="0.5" ...> </v:image></pre>  <p>end example]</p>


Attributes	Description
gamma (Image Gamma Correction)	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p> <p>Specifies the gamma correction. Default is 1.</p> <p>Gamma correction is a factor by which the intended target display gamma differs from the sRGB profile. It can be used to correct for images not prepared for sRGB displays and to adjust overall image contrast. Decreasing it below 1 gives a higher contrast image.</p> <p>[Example:</p> <pre><v:image ... gamma="0.5" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
grayscale (Image Grayscale Toggle)	<p>Specifies to display the image in grayscale. Default is false.</p> <p>[Example:</p> <pre><v:image ... gamma="0.5" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type</p>




Attributes	Description
	(§20.1.2.5).
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	Specifies the alignment of a horizontal rule. Default is left. <i>[Example:</i> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).
href (Hyperlink Target)	Specifies a hyperlink URL target for the shape. Default is no value. <i>[Example:</i> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that the horizontal rule does not have 3-D shading. Default is false. <i>[Example:</i> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	Specifies the length of a horizontal rule as a percentage of page width. Default is 0. <i>[Example:</i> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema float datatype.
hrstd (Horizontal	Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is

Attributes	Description
<p>Rule Standard Display Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>[Example: The red color is 25% opaque:</p>

Attributes	Description
	<pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas- microsoft-</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... ></pre>

Attributes	Description
com:office:office	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
spid (Optional String) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional string that an application can use to Identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
spt (Optional Number) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
src (Image Source)	<p>Specifies the URL of the image to use.</p> <p>[Example:</p> <pre><v:image ... src="myimage.gif" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre data-bbox="451 789 1062 890"><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre data-bbox="451 1434 984 1497"><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
style (Shape Styling	Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level

Attributes	Description								
Properties)	<p>2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> end example]</p> <table border="1"> <thead> <tr> <th>Property</th><th>Description</th></tr> </thead> <tbody> <tr> <td>flip</td><td> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> <tr> <td>height</td><td> <p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. </td></tr> <tr> <td>left</td><td> <p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. </td></tr> </tbody> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.
Property	Description								
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 								
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 								
left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. 								

Attributes	Description	
		<ul style="list-style-type: none"> • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the

Attributes	Description	
		<p>page.</p> <ul style="list-style-type: none"> • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin

Attributes	Description	
		<ul style="list-style-type: none"> • page • text • line
	mso-wrap-distance-bottom	Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-left	Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-right	Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties.

Attributes	Description	
		<ul style="list-style-type: none"> • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none"> • hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • inherit - The visibility state is inherited from the parent of the shape.
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Uses the order that the shapes appear in the page, bottom to top. • <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.
	The following properties are only used by the textbox element (§19.1.2.22):	
	Property	Description
	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> • ltr - Text is displayed left-to-right. • rtl - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> • horizontal - Text is displayed horizontally. • vertical - Text is displayed vertically. • vertical-ideographic - Ideographic text is displayed vertically. • horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.

Attributes	Description											
	mso-rotate	Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are: <ul style="list-style-type: none">• 0• 90• 180• -90										
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.										
	v-text-anchor	Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are: <ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline										
The following properties are only used by the textpath element (§19.1.2.23):												
<table><tr><th>Property</th><th>Description</th></tr><tr><td>font</td><td>Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr><tr><td>font-family</td><td>Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr><tr><td>font-size</td><td>Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr><tr><td>font-style</td><td>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</td></tr></table>			Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:
Property	Description											
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.											
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.											
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.											
font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:											

Attributes	Description																		
		<ul style="list-style-type: none">normalitalicoblique - Treated the same as italic.																	
	font-variant	<p>Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are:</p> <ul style="list-style-type: none">normalsmall-caps																	
	font-weight	<p>Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td rowspan="5">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr><tr><td>300</td></tr><tr><td>400</td><td rowspan="6">Treated as bold.</td></tr><tr><td>bold</td></tr><tr><td>bolder</td></tr><tr><td>500</td></tr><tr><td>600</td></tr><tr><td>700</td></tr><tr><td>800</td></tr><tr><td>900</td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	Treated as bold.	bold	bolder	500	600	700	800	900
		Value	Description																
		normal	Treated as non-bold.																
lighter																			
100																			
200																			
300																			
400	Treated as bold.																		
bold																			
bolder																			
500																			
600																			
700																			
800																			
900																			
mso-text-shadow	<p>Specifies whether a shadow is applied to the text on a text path. Default is false.</p>																		
text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none">none																		

Attributes	Description	
		<ul style="list-style-type: none"> • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are: <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height 		

Attributes	Description																
	<p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 1031 1479 1667"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </pre>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																

Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>title (Shape Title)</p>	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... title="tooltip" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is</p>

Attributes	Description
	<p>tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to <i>tight</i> or <i>through</i>.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Image](#)) is located in §A.6.1. *end note]*

19.1.2.11 imagedata (Image Data)

This element is used to draw an image that has been loaded from an external source. There is an implied rectangle that is the same size as the image. Any stroke or fill is applied to this implied rectangle. The stroke is drawn on top of the image. The fill is behind the image and therefore only visible through transparent areas of the image. Image transparency is either encoded in the file or defined via a color value using the chromakey attribute. Unlike the image element (§19.1.2.10), the imagedata element shall have a parent element.



[Example:



```
<v:shape style="position:relative;top:1;left:1;width:50;height:50"
  path="m 0,0 l 1000,0 1000,1000 0,1000 x e" fillcolor="blue">
  <v:imagedata src="myimage.gif"/>
</v:shape>
```






end example]

Attributes	Description
<p>althref (Alternate Image Reference)</p> <p>Namespace: urn:schemas-microsoft-</p>	<p>Defines an alternate reference for an image in Macintosh PICT format.</p> <p>[Example:</p> <pre><v:imagedata ... althref="myimage.pcz" ... > </v:imagedata></pre>

Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bilevel (Image Bilevel Toggle)	<p>Specifies that all colors in the picture shall be converted to either 0 or full intensity component values. This converts a color bitmap to 8 colors and a grayscale bitmap to black and white. Default is false.</p> <p>[Example:</p> <pre data-bbox="451 653 902 720"><v:image ... bilevel="true" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
blacklevel (Image Brightness)	<p>Specifies the image brightness. Default is 0.</p> <p>[Example:</p> <pre data-bbox="451 1188 935 1255"><v:image ... blacklevel="0.1" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p>[Example:</p> <pre data-bbox="451 1759 951 1827"><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p>




Attributes	Description
<p>cropbottom (Image Bottom Crop)</p>	<p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p> <p>Specifies the how much to crop the image from the bottom up as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f" represents 52429/65536 or 0.8.</p> <p>[Example:</p> <pre><v:image ... cropbottom="10%" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>cropleft (Image Left Crop)</p>	<p>Specifies how much to crop the image from the left in as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. For example, a value of "52429f" represents 52429/65536 or 0.8.</p> <p>[Example:</p> <pre><v:image ... cropleft="10%" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>cropright (Image Right Crop)</p>	<p>Specifies how much to crop the image from the right in as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example:</p> <pre><v:image ... cropright="10%" ...> </v:image></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
croptop (Image Top Crop)	<p>Specifies how much to crop the image from the top down as a fraction of picture size. Default is 0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. <i>[Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p><i>[Example:</i></p> <pre data-bbox="451 821 886 884"><v:image ... croptop="10%" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
detectmouseclick (Detect Mouse Click) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a mouse click is detected on the fill of a shape.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
embosscolor (Embossed Color)	<p>Specifies the color to use to create an embossed effect in the image. Default is no value. This can be set to a percentage of the shadow color to create an embossed picture effect.</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
gain (Image Intensity)	<p>Specifies an adjustment for the intensity of all colors. Essentially sets how bright white is. Default is 1.</p> <p><i>[Example:</i></p>

Attributes	Description
	<pre><v:image ... gain="0.5" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
gamma (Image Gamma Correction)	<p>Specifies the gamma correction. Default is 1.</p> <p>Gamma correction is a factor by which the intended target display gamma differs from the sRGB profile. It can be used to correct for images not prepared for sRGB displays and to adjust overall image contrast. Decreasing it below 1 gives a higher contrast image.</p> <p>[Example:</p> <pre><v:image ... gamma="0.5" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
grayscale (Image Grayscale Toggle)	<p>Specifies to display the image in grayscale. Default is false.</p> <p>[Example:</p> <pre><v:image ... gamma="0.5" ...> </v:image></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
href (Explicit	<p>Specifies the relationship ID of the relationship to the hyperlink used for this VML object.</p>

Attributes	Description
<p>Relationship to Hyperlink Target)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information for the image data:</p> <pre>< ... r:href="rId5" /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
<p>href (Original Image Reference)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>id (Explicit Relationship to Image Data)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID of the relationship to the image used for this VML object. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information for the image data:</p> <pre>< ... r:id="rId10" /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p>

Attributes	Description
	<p><code><v:shape ... id="myShape" ... ></code> <code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>movie (Movie Reference)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a pointer to a movie image. This is a data block that contains a pointer to a pointer to movie data.</p> <p>[Example:</p> <p><code><v:imagedata ... o:movie="1434" ...></code> <code></v:imagedata></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>oleid (Image Embedded Object ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the embedded object ID of an image.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>pict (Explicit Relationship to Alternate Image Data)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID of the relationship to an alternate format image used for this VML object. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>If this attribute is specified, the application should attempt to display the image defined by the relationship. If the application cannot display the format of that image, the r:id attribute is used.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId7 contains the corresponding relationship information for the image data. The relationship part with relationship ID rId10 is used if the application cannot display the image referenced by rId7.:</p> <p><code>< ... r:id="rId10" r:pict="rId7"/></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type</p>

Attributes	Description
recolortarget (Black Recoloring Color)	<p>(Part 1, §22.8.2.1).</p> <p>Specifies the color to which black should be recolored.</p> <p>[Example:</p> <pre><v:imagedata r:id="rId4" recolortarget="red"> </v:imagedata></pre> <div data-bbox="451 541 555 646">  </div> <p>no recolor</p> <div data-bbox="451 655 555 760">  </div> <p>recolortarget="red"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
relid (Relationship to Part) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the relationship ID of the relationship to the image. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information:</p> <pre><v:imagedata ... o:relid="rId10" ...> </v:imagedata></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
src (Image Source)	<p>Specifies the URL of the image to use.</p> <p>[Example:</p> <pre><v:image ... src="myimage.gif" ...> </v:image></pre> <div data-bbox="412 1738 516 1843">  </div>

Attributes	Description
	<i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
title (Image Data Title) Namespace: urn:schemas-microsoft-com:office:office	Specifies the title of an embedded image. This is typically set to the comment property of the image, which is often blank. [Example: <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.

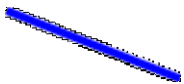
[Note: The W3C XML Schema definition of this element's content model ([CT_ImageData](#)) is located in §A.6.1.
end note]

19.1.2.12 line (Line)

This element draws a straight line.

[Example:

```
<v:line from="10pt,10pt" to="75pt,35pt"
strokecolor="blue" strokeweight="3pt">
</v:line>
```



end example]

Attributes	Description
allowincell (Allow in Table Cell) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether a shape can be placed in a table. Default is false. [Example: <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <i>end example]</i>



Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>alt (Alternate Text)</p>	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderbottomcolor (Bottom Border Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderleftcolor</p>	<p>Specifies the left border color of an inline shape. Default is no value.</p>

Attributes	Description
<p>(Border Left Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderrightcolor (Border Right Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bordertopcolor (Border Top Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>bullet (Graphical Bullet)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is a graphical bullet. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>button (Button Behavior Toggle)</p>	<p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p>


Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<p>[<i>Example:</i></p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [<i>Example:</i> Normal B&W might allow greyscale and pure B&W might not. <i>end example]</i></p> <p>[<i>Example:</i> This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in a pale grayscale in a normal black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode)	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in high contrast when in a pure black-and-white</p>


Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	environment: <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).
chromakey (Image Transparency Color)	Specifies a color value that is transparent and show anything behind the shape. Default is no value. <i>[Example:</i> <pre><v:image ... chromakey="white" ...> </v:image></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
class (CSS Reference)	Specifies a reference to the definition of a CSS style. Default is no value. <i>[Example:</i> The snippets below are equivalent: <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
clip (Clipping Toggle) Namespace: urn:schemas-	Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region. <i>[Example:</i>

Attributes	Description
microsoft-com:office:office	<pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
cliptowrap (Clip to Wrapping Polygon) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
connectortype (Shape Connector Type) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p>

Attributes	Description
	<pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[<i>Example:</i> The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas-microsoft-	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[<i>Example:</i></p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre>



Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
dgmlayoutmru (Diagram Node Recent Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
dgmnodekind (Diagram Node Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
doubleclicknotify (Double-click Notification Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue</p>

Attributes	Description
	<p>values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>filled (Shape Fill Toggle)</p>	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p>



Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>from (Line Start)</p>	<p>Specifies the starting point of the line in the coordinate space of the parent element. Default is "0,0". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p>[Example:</p> <pre><v:line from="10pt,10pt" to="50pt,50pt"> </v:line></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hr (Horizontal Rule Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hralign (Horizontal Rule Alignment)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the alignment of a horizontal rule. Default is left.</p> <p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
<p>href (Hyperlink Target)</p>	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p>


Attributes	Description
	<pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the horizontal rule does not have 3-D shading. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
hrstd (Horizontal Rule Standard Display Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if <code>hr</code> is <code>true</code>. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p>

Attributes	Description
	<p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>

Attributes	Description
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If <code>true</code>, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre> <div data-bbox="451 1467 779 1688">  opacity="1"  opacity=".25" </div> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>spt (Optional Number)</p>	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	The possible values for this attribute are defined by the W3C XML Schema float datatype.
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p>

Attributes	Description						
	<p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>						
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p>end example]</p> <table><tr><th>Property</th><th>Description</th></tr><tr><td>flip</td><td><p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p><ul style="list-style-type: none">• x - Flip along the y-axis, reversing the x-coordinates.• y - Flip along the x-axis, reversing the y-coordinates.• xy - Flip along both the y- and x-axis.• yx - Flip along both the x- and y-axis.</td></tr><tr><td>height</td><td><p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p><ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or</td></tr></table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none">• x - Flip along the y-axis, reversing the x-coordinates.• y - Flip along the x-axis, reversing the y-coordinates.• xy - Flip along both the y- and x-axis.• yx - Flip along both the x- and y-axis.	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or
Property	Description						
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none">• x - Flip along the y-axis, reversing the x-coordinates.• y - Flip along the x-axis, reversing the y-coordinates.• xy - Flip along both the y- and x-axis.• yx - Flip along both the x- and y-axis.						
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or						

Attributes	Description	
		<p>ex). If no units are given, pixels (px) is assumed.</p> <ul style="list-style-type: none"> • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-top	<p>Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-edited	<p>Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.</p>
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed</p>

Attributes	Description	
		<p>values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is <code>static</code>. When the element is contained inside a group, this property shall be <code>absolute</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>static</code> - The element is positioned according to the normal flow of the page. The <code>top</code> and <code>left</code> properties are ignored. If the object is anchored inline, this value is used. • <code>absolute</code> - The element is positioned relative to the parent, using the <code>top</code> and <code>left</code> properties. • <code>relative</code> - The element is positioned according to the normal flow of the page, but the <code>top</code> and <code>left</code> properties are used. The overlap of overlapping elements is governed by the <code>z-index</code> property.
	rotation	<p>Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.</p>
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • <code>auto</code> - Default position of an element in the flow of the page. • <code><units></code> - A number with an absolute units designator (<code>cm</code>, <code>mm</code>, <code>in</code>, <code>pt</code>, <code>pc</code>, or <code>px</code>) or a relative units designator (<code>em</code> or <code>ex</code>). If no units are given, pixels (<code>px</code>) is assumed. • <code><percentage></code> - Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only <code>inherit</code> and <code>hidden</code> are used; any other values are mapped to <code>inherit</code>. Default is <code>inherit</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>hidden</code> - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • <code>inherit</code> - The visibility state is inherited from the parent of the shape.

Attributes	Description								
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's width.							
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">• auto - Uses the order that the shapes appear in the page, bottom to top.• <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.							
	<p>The following properties are only used by the textbox element (§19.1.2.22):</p>								
	<table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p><ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.</td></tr><tr><td>layout-flow</td><td><p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p><ul style="list-style-type: none">• horizontal - Text is displayed horizontally.• vertical - Text is displayed vertically.• vertical-ideographic - Ideographic text is displayed vertically.• horizontal-ideographic - Ideographic text is displayed horizontally.</td></tr><tr><td>mso-direction-</td><td><p>Specifies an alternate direction for text in textboxes. Overrides</p></td></tr></table>		Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none">• horizontal - Text is displayed horizontally.• vertical - Text is displayed vertically.• vertical-ideographic - Ideographic text is displayed vertically.• horizontal-ideographic - Ideographic text is displayed horizontally.	mso-direction-
Property	Description								
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.								
layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none">• horizontal - Text is displayed horizontally.• vertical - Text is displayed vertically.• vertical-ideographic - Ideographic text is displayed vertically.• horizontal-ideographic - Ideographic text is displayed horizontally.								
mso-direction-	<p>Specifies an alternate direction for text in textboxes. Overrides</p>								


Attributes	Description	
	alt	the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> • top • middle • bottom • top-center • middle-center • bottom-center • top-baseline • bottom-baseline • top-center-baseline • bottom-center-baseline
The following properties are only used by the textpath element (§19.1.2.23):		

Attributes	Description												
	Property	Description											
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.											
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.											
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.											
	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">normalitalicoblique - Treated the same as italic.											
	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">normalsmall-caps											
	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><thead><tr><th>Value</th><th>Description</th></tr></thead><tbody><tr><td>normal</td><td rowspan="6">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr><tr><td>300</td></tr><tr><td>400</td></tr><tr><td>bold</td><td rowspan="2">Treated as bold.</td></tr><tr><td>bolder</td></tr></tbody></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.
Value	Description												
normal	Treated as non-bold.												
lighter													
100													
200													
300													
400													
bold	Treated as bold.												
bolder													

Attributes	Description	
		<div>500</div> <div>600</div> <div>700</div> <div>800</div> <div>900</div>
mso-text-shadow		Specifies whether a shadow is applied to the text on a text path. Default is false.
text-decoration		<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
v-rotate-letters		Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
v-same-letter-heights		Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
v-text-align		<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
v-text-kern		Specifies whether kerning is turned on. Default is false.
v-text-reverse		Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
v-text-		Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between

Attributes	Description							
	spacing-mode	each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are: <ul style="list-style-type: none">tighteningtracking						
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.						
	The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties: <ul style="list-style-type: none">topleftwidthheight							
	The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute: <ul style="list-style-type: none">flipheightleftmargin-leftmargin-toppositionrotationtopvisibilitywidthz-index							
	The possible values for this attribute are defined by the W3C XML Schema string datatype.							
target (Hyperlink Display Target)	Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr><tr><td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr></table>		Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.
Value	Description							
<targetname>	String containing the name of the frame or window in which to load the document.							
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.							

Attributes	Description										
	<table border="1" data-bbox="415 245 1482 667"> <tr> <td data-bbox="415 245 626 331"><code>_media</code></td><td data-bbox="626 245 1482 331">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="415 331 626 417"><code>_parent</code></td><td data-bbox="626 331 1482 417">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="415 417 626 504"><code>_search</code></td><td data-bbox="626 417 1482 504">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="415 504 626 590"><code>_self</code></td><td data-bbox="626 504 1482 590">Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td data-bbox="415 590 626 667"><code>_top</code></td><td data-bbox="626 590 1482 667">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 709 535 737"><i>[Example:</i></p> <pre data-bbox="453 779 1062 909"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 951 574 978"><i>end example]</i></p> <p data-bbox="415 1020 1377 1087">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<code>_media</code>	Specifies that the linked document is loaded into the browser's multimedia pane.	<code>_parent</code>	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	<code>_search</code>	Specifies that the linked document is loaded into the browser's search pane.	<code>_self</code>	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	<code>_top</code>	Specifies that the linked document is loaded into the topmost window.
<code>_media</code>	Specifies that the linked document is loaded into the browser's multimedia pane.										
<code>_parent</code>	Specifies that the linked document is loaded into the immediate parent of the document containing the link.										
<code>_search</code>	Specifies that the linked document is loaded into the browser's search pane.										
<code>_self</code>	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).										
<code>_top</code>	Specifies that the linked document is loaded into the topmost window.										
title (Shape Title)	<p data-bbox="415 1100 1482 1167">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 1209 535 1236"><i>[Example:</i></p> <pre data-bbox="453 1278 935 1346"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1388 574 1415"><i>end example]</i></p> <p data-bbox="415 1457 1377 1524">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>										
to (Line End Point)	<p data-bbox="415 1535 1482 1644">Specifies the ending point of the line in the coordinate space of the parent element. Default is "10,10". If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px.</p> <p data-bbox="415 1686 535 1713"><i>[Example:</i></p> <pre data-bbox="453 1755 1094 1822"> <v:line from="10pt,10pt" to="50pt,50pt"> </v:line> </pre>										

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is <code>false</code>. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is <code>false</code>. If <code>true</code>, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the <code>mso-wrap-mode</code> style attribute is set to <code>tight</code> or <code>through</code>.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.

[Note: The W3C XML Schema definition of this element’s content model ([CT_Line](#)) is located in §A.6.1. *end note*]

19.1.2.13 oval (Oval)

This element draws an oval sized according to the CSS2 style content width and height.

[Example:

```
<v:oval fillcolor="blue"
  style="position:relative;top:1;left:1;width:150;height:50">
</v:oval>
```



end example]


Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example</i>]</p>


Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
alt (Alternate Text)	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor (Border Left Color) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color)	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is <i>false</i>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is <i>false</i>.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace:	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is <i>auto</i>, which uses <i>o:bwnormal</i> for normal black-and-white rendering and <i>o:bwpure</i> for pure black-and-white rendering.</p>


Attributes	Description
urn:schemas-microsoft-com:office:office	<p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. <i>[Example: Normal B&W might allow greyscale and pure B&W might not. end example]</i></p> <p><i>[Example: This shape renders in grayscale in a black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p><i>[Example:</i></p>

Attributes	Description
	<pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[Example: The snippets below are equivalent:</p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
clip (Clipping Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
cliptowrap (Clip to Wrapping Polygon) Namespace: urn:schemas-microsoft-	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p>

Attributes	Description
com:office:office	<pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
connectortype (Shape Connector Type) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier)	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
dgmlayoutmru (Diagram Node Recent Layout Identifier)	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre>

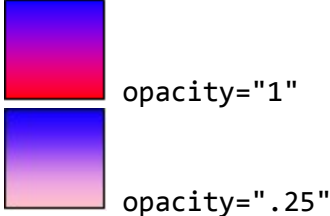
Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p><i>end example]</i></p>


Attributes	Description
<p>filled (Shape Fill Toggle)</p>	<p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p> <p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hr (Horizontal Rule Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type</p>

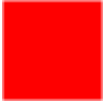

Attributes	Description
	(§20.1.2.5).
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	Specifies the alignment of a horizontal rule. Default is left. <i>[Example:</i> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).
href (Hyperlink Target)	Specifies a hyperlink URL target for the shape. Default is no value. <i>[Example:</i> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that the horizontal rule does not have 3-D shading. Default is false. <i>[Example:</i> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	Specifies the length of a horizontal rule as a percentage of page width. Default is 0. <i>[Example:</i> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema float datatype.
hrstd (Horizontal	Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is

Attributes	Description
<p>Rule Standard Display Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>[Example: The red color is 25% opaque:</p>

Attributes	Description
	<pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas- microsoft-</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... ></pre>

Attributes	Description
com:office:office	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
spid (Optional String) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional string that an application can use to Identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
spt (Optional Number) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p>

Attributes	Description		
	<pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>		
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table><tr><th>Property</th><th>Description</th></tr></table>	Property	Description
Property	Description		

Attributes	Description	
	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.

Attributes	Description	
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right

Attributes	Description	
		<ul style="list-style-type: none"> • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies the distance from the right side of the shape to the text</p>

Attributes	Description	
	distance-right	that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page.

Attributes	Description				
		<ul style="list-style-type: none">• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's height.			
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none">• hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.• inherit - The visibility state is inherited from the parent of the shape.			
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's width.			
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">• auto - Uses the order that the shapes appear in the page, bottom to top.• <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.			
	The following properties are only used by the textbox element (§19.1.2.22):				
	<table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</td></tr></table>	Property	Description	direction	Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:
Property	Description				
direction	Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:				

Attributes	Description	
		<ul style="list-style-type: none"> • <code>ltr</code> - Text is displayed left-to-right. • <code>rtl</code> - Text is displayed right-to-left.
	<code>layout-flow</code>	<p>Determines the flow of the text layout in a textbox. Default is <code>horizontal</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>horizontal</code> - Text is displayed horizontally. • <code>vertical</code> - Text is displayed vertically. • <code>vertical-ideographic</code> - Ideographic text is displayed vertically. • <code>horizontal-ideographic</code> - Ideographic text is displayed horizontally.
	<code>mso-direction-alt</code>	Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code> .
	<code>mso-fit-shape-to-text</code>	Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code> .
	<code>mso-fit-text-to-shape</code>	Specifies whether the text stretches to fit the textbox. Default is <code>false</code> .
	<code>mso-layout-flow-alt</code>	Specifies the alternate layout flow for text in textboxes. This property is used instead of <code>layout-flow</code> when the layout flow is from bottom to top for non-ideographic languages. Its only value is <code>bottom-to-top</code> .
	<code>mso-next-textbox</code>	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	<code>mso-rotate</code>	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	<code>mso-text-scale</code>	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if <code>mso-fit-text-to-shape</code> is <code>true</code> .
	<code>v-text-anchor</code>	Specifies the vertical anchoring of text in a textbox. Default is <code>top</code> . The alignment of a text anchor only becomes evident if <code>mso-fit-text-to-shape</code> is <code>false</code> . This property is different from the <code>vertical-align</code> CSS property, which is used for ideographic languages. Allowed values are:

Attributes	Description	
		<ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline
	The following properties are only used by the textpath element (§19.1.2.23):	

Attributes	Description					
		<table><tr><td>normal lighter 100 200 300 400</td><td>Treated as non-bold.</td></tr><tr><td>bold bolder 500 600 700 800 900</td><td>Treated as bold.</td></tr></table>	normal lighter 100 200 300 400	Treated as non-bold.	bold bolder 500 600 700 800 900	Treated as bold.
	normal lighter 100 200 300 400	Treated as non-bold.				
	bold bolder 500 600 700 800 900	Treated as bold.				
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.				
	text-decoration	Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are: <ul style="list-style-type: none">• none• underline• overline• line-through• blink				
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.				
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.				
v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none">• left					

Attributes	Description	
		<ul style="list-style-type: none"> • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width 		


Attributes	Description																
	<ul style="list-style-type: none"> • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 508 1479 1144"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p>end example]</p>																

Attributes	Description
<p></p>	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>


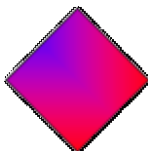
[Note: The W3C XML Schema definition of this element's content model ([CT_Oval](#)) is located in §A.6.1. *end note*]

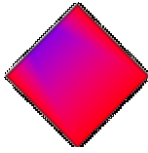
19.1.2.14 path (Shape Path)


This element defines the path that makes up the shape. This is done through a string that contains a rich set of pen movement commands. This element also describes the limo-stretch point, inscribed textbox rectangle locations and connection site locations. The limo-stretch definition and the formulas element (§19.1.2.6) allow greater designer control of how the path scales. [Example: They allow, for example, definition of a true rounded corner rectangle where the corners remain circular even though the rectangle is scaled anisotropically. *end example*]



Attributes	Description
<p>arrowok (Arrowhead Display Toggle)</p>	<p>Specifies whether arrowheads are allowed to be displayed. This attribute overrides all other arrowhead attributes in the parent or the stroke element (§19.1.2.21). Default is false.</p> <p>[Example:</p> <pre><v:shape style="width:50;height:50"> <v:stroke endarrow="block"/> <v:path arrowok="true" v="m 0,0 l 1000,0 1000,1000 e"/> </v:shape></pre>  <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectangles (Connection Point Connect Angles)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the angle at which curves connect to a shape's connection points. The connection angles are defined by a string consisting of angle values delimited by commas. Default is no value.</p> <p>[Example: Connections are made along the horizontal and vertical axes:</p> <pre><v:path ... o:connectangles="0,90,180,270" ... > </v:path></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>connectlocs (Connection Points)</p>	<p>Specifies the location of connection points on a path. The connection points are defined by a string consisting of pairs of x and y values, delimited by commas. This is used if</p>

Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>connecttype is custom. Default is no value.</p> <p>[Example: Connection points exist at the midpoints of the sides of the square:</p> <pre><v:path ... v="m 0,0 1 100,0 100,100 0,100 x e" o:connectlocs="50,0;100,50;50,100;0,50" ... > </v:path></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>connecttype (Connection Point Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connection points used for attaching shapes to other shapes. Default is none. If set to custom, connectlocs is used. Allowed values are:</p> <p>[Example:</p> <pre><v:path ... o:connecttype="custom" o:connectlocs="50,0;100,50;50,100;0,50" ... > </v:path></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectType simple type (§19.2.3.8).</p>
<p>extrusionok (Extrusion Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an extrusion is allowed to be displayed. This attribute overrides all other extrusion attributes in the parent or the extrusion element (§19.2.2.11). Default is true.</p> <p>[Example:</p> <pre><v:rect fillcolor="lime" style="width:50;height:50"> <v:extrusion on="true"/> <v:path o:extrusionok="false"/> </v:rect></pre> <div data-bbox="451 1570 553 1675" data-label="Image"> </div> <pre><v:path o:extrusionok="false"/></pre> <div data-bbox="451 1682 586 1818" data-label="Image"> </div> <pre><v:path o:extrusionok="true"/></pre> <p><i>end example]</i></p>

Attributes	Description
fillok (Shape Fill Toggle)	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p> <p>Specifies whether a fill is allowed to be displayed. This attribute overrides all other fill attributes in the parent or fill element (§19.1.2.5). Default is true.</p> <p>[Example:</p> <pre> <v:shape style="width:50;height:50" fillcolor="red"> <v:path fillok="false" v="m 0,0 l 0,1000, 1000,1000, 1000,0 x e"/> </v:shape> </pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
gradientshapeok (Gradient Shape Toggle)	<p>Specifies whether a gradient path is made up of repeated concentric paths. Default is false.</p> <p>If true, a gradient fill can be produced by repeated drawing of scaled versions of the path - this shall only be set if it is possible to scale the path in such a way that a fill is always contained in the original path. This controls the interpretation of the type="gradientradial" attribute of the fill element (§19.1.2.5).</p> <p>[Example: In the first case, the radial gradient is aligned irrespective of the shape's path:</p> <pre> <v:shape style="width:50;height:50;rotation:45" path="m 0,0 l 0,1000, 1000,1000, 1000,0 x e"> <v:path gradientshapeok="false"/> <v:fill type="gradientradial" color="red" color2="blue"/> </v:shape> </pre>  <p>gradientshapeok="false"</p>

Attributes	Description
	 <p><code>gradientshapeok="true"</code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetpenok (Inset Stroke From Path Flag)	<p>Specifies whether the stroke can be inset from the path. If this is <code>false</code>, it overrides the <code>insetpen</code> attribute and prevents the stroke from being inset.</p> <p>[Example: The stroke is not inset:</p> <pre><v:shape ... insetpen="true"> <v:path ... insetpenok="false"/> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
limo (Limo Stretch Point)	<p>Specifies a stretch point on the shape's edge that defines where and how a shape is allowed to be stretched by a user in a graphical editor. Default is "0,0".</p> <p>[Example:</p> <pre><v:line from="20pt,20pt" to="100pt,20pt"> <v:path limo="60pt,20pt"/> </v:line></pre> <p><i>end example]</i></p>

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>shadowok (Shadow Toggle)</p>	<p>Specifies whether a shadow is allowed to be displayed. This attribute overrides all other shadow attributes in the parent or the shadow element (§19.1.2.18). Default is true.</p> <p>[Example: The shape has no shadow:</p> <pre> <v:shape style="width:50;height:50"> <v:path v="m 0,0 l 0,1000, 1000,1000, 1000,0 x e" shadowok="false"/> <v:shadow on="true"/> </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>strokeok (Stroke Toggle)</p>	<p>Specifies whether a stroke is displayed. This attribute overrides all other stroke attributes in the parent or the stroke element (§19.1.2.21). Default is true.</p> <p>[Example: The shape's red stroke is not shown:</p> <pre> <v:shape style="width:50;height:50" fillcolor="blue" strokecolor="red"> <v:path v="m 0,0 l 0,1000, 1000,1000, 1000,0 x e" strokeok="false"/> </v:shape> </pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>textboxrect (Text Box Bounding Box)</p>	<p>Specifies one or more text boxes inside a shape. Default is the same as the geometry's bounding box.</p> <p>A textbox is defined by one or more sets of numbers specifying (in order) the left, top, right, and bottom points of the rectangle. Multiple sets are delimited by a semicolon. The default value is the same dimension value as the containing rectangle. If more than one textbox is defined, the comma-delimited quadruple sets that define each textbox are separated by semicolons. Normally textboxes come in sets of 1, 2, 3, or 6 rectangles on a</p>

Attributes	Description
	<p>shape. The textboxrect dimensions clip any text that extends beyond its region.</p> <p>[Example: The textbox is 25% down from the top and the exclamation point is clipped:</p> <pre><v:shape style="width:60;height:50"> <v:path v="m 0,0 l 0,1000, 1000,1000, 1000,0 x e" textboxrect="0,250,850,1000"/> <v:textbox>VML!</v:textbox> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
textpathok (Text Path Toggle)	<p>Specifies whether a text path is displayed. Default is false.</p> <p>If true, this indicates that the path is an appropriate warping path for the textpath element (§19.1.2.23). Otherwise, the textpath element shall be ignored.</p> <p>[Example: The defined textpath is ignored:</p> <pre><v:curve from="50,100" to="400,100" control1="200,200" control2="300,200"> <v:path textpathok="false"/> <v:textpath on="false" style="font:normal normal normal 36pt Arial" string="textpath"/> </v:curve></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
v (Path Definition)	<p>Specifies a string containing the commands that define the shape's path. This value consists of commands followed by zero or more parameters. Default is no value.</p> <p>The following rules apply to path strings:</p>

Attributes	Description																																
	<ul style="list-style-type: none">Commas or spaces delimit parameters for each command. Both "m 0,0" and "m0 0" are acceptable.A parameter that is omitted using commas is treated as having a value of zero. Thus, "c 10,10,0,0,25,13" and "c 10,10,,,25,13" are equivalent.Parameterized paths are also allowed. In this case, the shape shall also have a formulas element (§19.1.2.6) with a list of formulas that are substituted into the path using the @ symbol followed by the number of the formula. The adj property of the shape contains the input parameters for these formulas. [Example: For example, "moveto @1@4". end example] The evaluations of the formulas are substituted into the appropriate positions. The @ character also serves as a delimiter. <p>The allowed commands are given below. An asterisk (*) indicates that the command is allowed to be repeated. For the qb command, the controlpoint parameter is also allowed to be repeated.</p> <table><tr><th>Command</th><th>Name</th><th>Parameters</th><th>Description</th></tr><tr><td>m</td><td>moveto</td><td>2</td><td>Start a new sub-path at the given (x,y) coordinate.</td></tr><tr><td>l</td><td>lineto</td><td>2*</td><td>Draw a line from the current point to the given (x,y) coordinate which becomes the new current point. Specifying a number of coordinate pairs forms a polyline.</td></tr><tr><td>c</td><td>curveto</td><td>6*</td><td>Draw a cubic bézier curve from the current point to the coordinate given by the final two parameters. The control points are given by the first four parameters.</td></tr><tr><td>x</td><td>close</td><td>0</td><td>Close the current sub-path by drawing a straight line from the current point to the original moveto point.</td></tr><tr><td>e</td><td>end</td><td>0</td><td>End the current set of sub-paths. A given set of sub-paths (as delimited by end) is filled. Subsequent sets of sub-paths are filled independently and superimposed on existing ones.</td></tr><tr><td>t</td><td>rmoveto</td><td>2*</td><td>Start a new sub-path at a coordinate relative to the current point, cp (cpx+x, cpy+y).</td></tr><tr><td>r</td><td>rlineto</td><td>2*</td><td>Draw a line from the current point to the given relative coordinate (cpx+x,</td></tr></table>	Command	Name	Parameters	Description	m	moveto	2	Start a new sub-path at the given (x,y) coordinate.	l	lineto	2*	Draw a line from the current point to the given (x,y) coordinate which becomes the new current point. Specifying a number of coordinate pairs forms a polyline.	c	curveto	6*	Draw a cubic bézier curve from the current point to the coordinate given by the final two parameters. The control points are given by the first four parameters.	x	close	0	Close the current sub-path by drawing a straight line from the current point to the original moveto point.	e	end	0	End the current set of sub-paths. A given set of sub-paths (as delimited by end) is filled. Subsequent sets of sub-paths are filled independently and superimposed on existing ones.	t	rmoveto	2*	Start a new sub-path at a coordinate relative to the current point, cp (cpx+x, cpy+y).	r	rlineto	2*	Draw a line from the current point to the given relative coordinate (cpx+x,
Command	Name	Parameters	Description																														
m	moveto	2	Start a new sub-path at the given (x,y) coordinate.																														
l	lineto	2*	Draw a line from the current point to the given (x,y) coordinate which becomes the new current point. Specifying a number of coordinate pairs forms a polyline.																														
c	curveto	6*	Draw a cubic bézier curve from the current point to the coordinate given by the final two parameters. The control points are given by the first four parameters.																														
x	close	0	Close the current sub-path by drawing a straight line from the current point to the original moveto point.																														
e	end	0	End the current set of sub-paths. A given set of sub-paths (as delimited by end) is filled. Subsequent sets of sub-paths are filled independently and superimposed on existing ones.																														
t	rmoveto	2*	Start a new sub-path at a coordinate relative to the current point, cp (cpx+x, cpy+y).																														
r	rlineto	2*	Draw a line from the current point to the given relative coordinate (cpx+x,																														

Attributes	Description			
				cpy+y).
	v	rcurveto	6*	Cubic bézier curve using the given coordinate relative to the current point.
	nf	nofill	0	The current set of sub-paths (delimited by e) is not filled.
	ns	nostroke	0	The current set of sub-paths (delimited by e) is not stroked.
	ae	angleellipseto	6*	Draws a segment of an ellipse as described using these parameters. A straight line is drawn from the current point to the start point of the segment. The parameters are: center (x,y), size(w,h), start angle, end angle.
	al	angleellipse	6*	Same as angleellipseto except that there is an implied moveto the starting point of the segment.
	at	arcto	8*	A segment of the ellipse is drawn which starts at the angle defined by the start radius vector and ends at the angle defined by the end vector. A straight line is drawn from the current point to the start of the arc. The arc is always drawn in a counterclockwise direction. The parameters are: left, top, right, bottom, start(x,y), end(x,y). The first four values define the bounding box of an ellipse. The last four define two radial vectors.
	ar	arc	8*	Same as arcto except there is an implied moveto the start point of the arc.
	wa	clockwisearco	8*	Same as arcto but the arc is drawn in a clockwise direction.
	wr	clockwisearc	8*	Same as arc but the arc is drawn in a clockwise direction

Attributes	Description			
	qx	ellipticalquadrant x	2*	A quarter ellipse is drawn from the current point to the given end point. The elliptical segment is initially tangential to a line parallel to the x-axis. (i.e. the segment starts out horizontal). The parameters are: end(x,y).
	qy	ellipticalquadrant y	2*	Same as ellipticalquadrantx except that the elliptical segment is initially tangential to a line parallel to the y-axis (i.e. the segment starts out vertical).
	qb	quadraticbezier	2+2*	Defines one or more quadratic bézier curves by means of control points and an end point. Intermediate (on-curve) points are obtained by interpolation between successive control points as in the OpenType font specification. The sub-path need not be started in which case the sub-path is closed. In this case the last point of the sub-path defines the start point of the quadratic bézier. The parameters are: controlpoint(x,y)*, end(x,y).
	The possible values for this attribute are defined by the W3C XML Schema string datatype.			

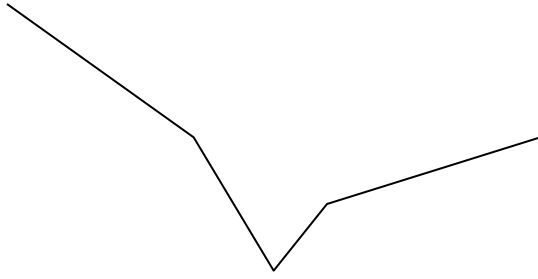
[Note: The W3C XML Schema definition of this element's content model ([CT_Path](#)) is located in §A.6.1. *end note*]

19.1.2.15 **polyline** (Multiple Path Line)

This element defines shapes made up of connected line segments.

[Example:

```
<v:polyline
  points="50pt,0pt 120pt,50pt 150pt,100pt 170pt,75pt 250pt,50pt">
</v:polyline>
```

end example]


Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>alt (Alternate Text)</p>	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p>


Attributes	Description
	<p><v:shape ... alt="Picture of a sunset"> </v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <p><v:shape ... o:borderbottomcolor="red" ... > </v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor (Border Left Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <p><v:shape ... o:borderleftcolor="red" ... > </v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <p><v:shape ... o:borderrightcolor="red" ... > </v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bordertopcolor (Border Top Color) Namespace: urn:schemas-	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <p><v:shape ... o:bordertopcolor="red" ... ></p>

Attributes	Description
microsoft-com:office:office	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is <code>auto</code>, which uses <code>o:bwnormal</code> for normal black-and-white rendering and <code>o:bwpure</code> for pure black-and-white rendering.</p> <p><code>bwnormal</code> and <code>bwpure</code> are subordinate to <code>bwmode</code>. If <code>bwmode</code> is <code>"auto"</code> then the value for <code>bwnormal</code> or <code>bwpure</code> is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [Example: Normal B&W might allow greyscale and pure B&W might not. <i>end example]</i></p> <p>[Example: This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p>


Attributes	Description
<p>bwnormal (Normal Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p> <p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwpure (Pure Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p>[Example: This shape renders in high contrast when in a pure black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>chromakey (Image Transparency Color)</p>	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p>[Example:</p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>class (CSS Reference)</p>	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[Example: The snippets below are equivalent:</p> <p>...</p>

Attributes	Description
	<pre>.narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape> <v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape> end example]</pre> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>clip (Clipping Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre>

Attributes	Description
microsoft-com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
coordorigin (Coordinate Space Origin)	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100"</pre>



Attributes	Description
	<pre>path="m 0,0 1 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>dgmlayout (Diagram Node Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>filled (Shape Fill Toggle)</p>	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>


Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hr (Horizontal Rule Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that a shape is a horizontal rule. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hralign (Horizontal Rule Alignment)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the alignment of a horizontal rule. Default is <code>left</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>

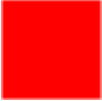

Attributes	Description
href (Hyperlink Target)	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
hrstd (Horizontal Rule Standard Display Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique	<p>Specifies a unique identifier that can be used to reference a VML object.</p>

Attributes	Description
Identifier)	<p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetmode (Text Inset Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
ole (Embedded Object Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre> <div data-bbox="451 1587 779 1808">  <p>opacity="1"</p>  <p>opacity=".25"</p> </div> <p>end example]</p>

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>points (Points for Compound Line)</p>	<p>Specifies a set of straight line segments that are composed of a series of pairs of points. Default is "0,0 10,10".</p> <p>Points are specified in the coordinate system of the parent element. If the parent is not a VML element, the default unit is a pixel. Allowed units are in, cm, mm, pt, pc and px. While commas are not required, they should be used for easier readability.</p> <p>See above for an example.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>spt (Optional Number)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>strokecolor (Shape Stroke Color)</p>	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>stroked (Shape Stroke Toggle)</p>	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red"</pre>

Attributes	Description		
	<pre>stroked="false" strokecolor="blue" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>		
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table border="1" data-bbox="415 1787 1479 1833"> <thead> <tr> <th data-bbox="415 1787 664 1833">Property</th><th data-bbox="664 1787 1479 1833">Description</th></tr> </thead> </table>	Property	Description
Property	Description		

Attributes	Description	
	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.

Attributes	Description	
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right

Attributes	Description	
		<ul style="list-style-type: none"> • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies the distance from the right side of the shape to the text</p>

Attributes	Description	
	distance-right	that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page.

Attributes	Description				
		<ul style="list-style-type: none">• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's height.			
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none">• hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.• inherit - The visibility state is inherited from the parent of the shape.			
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's width.			
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">• auto - Uses the order that the shapes appear in the page, bottom to top.• <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.			
	<p>The following properties are only used by the textbox element (§19.1.2.22):</p> <table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p></td></tr></table>		Property	Description	direction
Property	Description				
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p>				

Attributes	Description	
		<ul style="list-style-type: none"> • ltr - Text is displayed left-to-right. • rtl - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> • horizontal - Text is displayed horizontally. • vertical - Text is displayed vertically. • vertical-ideographic - Ideographic text is displayed vertically. • horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:

Attributes	Description																			
		<ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline																		
	The following properties are only used by the textpath element (§19.1.2.23):																			
		<table><tr><th>Property</th><th>Description</th></tr><tr><td>font</td><td>Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr><tr><td>font-family</td><td>Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr><tr><td>font-size</td><td>Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr><tr><td>font-style</td><td>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:<ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.</td></tr><tr><td>font-variant</td><td>Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are:<ul style="list-style-type: none">• normal• small-caps</td></tr><tr><td>font-weight</td><td>Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are:<table><tr><th>Value</th><th>Description</th></tr><tr><td></td><td></td></tr></table></td></tr></table>	Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">• normal• small-caps	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td></td><td></td></tr></table>	Value	Description		
	Property	Description																		
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.																		
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.																		
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.																		
font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.																			
font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">• normal• small-caps																			
font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td></td><td></td></tr></table>	Value	Description																	
Value	Description																			

Attributes	Description	
		<p>normal</p> <p>lighter</p> <p>100</p> <p>200</p> <p>300</p> <p>400</p> <p>Treated as non-bold.</p>
		<p>bold</p> <p>bolder</p> <p>500</p> <p>600</p> <p>700</p> <p>800</p> <p>900</p> <p>Treated as bold.</p>
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.
	text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left

Attributes	Description	
		<ul style="list-style-type: none"> • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width 		

Attributes	Description																
	<ul style="list-style-type: none"> • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 506 1479 1142"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p>end example]</p>																

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

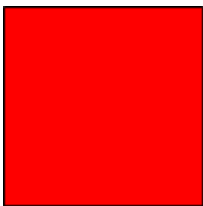
[*Note:* The W3C XML Schema definition of this element's content model ([CT_PolyLine](#)) is located in §A.6.1. *end note*]

19.1.2.16 `rect` (Rectangle)

This element is used to draw a simple rectangle. The CSS2 style content width and height define the width and height of the rectangle.

[*Example:*

```
<v:rect fillcolor="red"
  style="position:relative;top:0;left:0;width:100;height:100">
</v:rect>
```



end example]


Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[<i>Example:</i></p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[<i>Example:</i></p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>


Attributes	Description
alt (Alternate Text)	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor (Border Left Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color) Namespace: urn:schemas-	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... ></pre>

Attributes	Description
microsoft-com:office:office	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is <code>auto</code>, which uses <code>o:bwnormal</code> for normal black-and-white rendering and <code>o:bwpure</code> for pure black-and-white rendering.</p> <p><code>bwnormal</code> and <code>bwpure</code> are subordinate to <code>bwmode</code>. If <code>bwmode</code> is "auto" then the value</p>


Attributes	Description
com:office:office	<p>for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. <i>[Example: Normal B&W might allow greyscale and pure B&W might not. end example]</i></p> <p><i>[Example: This shape renders in grayscale in a black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p><i>[Example:</i></p> <pre><v:image ... chromakey="white" ...></pre>

Attributes	Description
	<p><code></v:image></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>class (CSS Reference)</p>	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[<i>Example:</i> The snippets below are equivalent:</p> <pre>narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape> <v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>clip (Clipping Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[<i>Example:</i></p> <pre> <v:shape ... o:clip="true"> </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[<i>Example:</i></p> <pre> <v:shape ... o:cliptowrap="true"> </v:shape> </pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
<p>coordorigin (Coordinate Space Origin)</p>	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>coordsize (Coordinate Space)</p>	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p>

Attributes	Description
Size)	<p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
dgmlayoutmru (Diagram Node Recent Layout Identifier) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p>

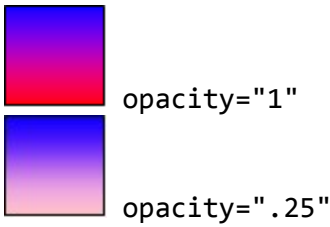
Attributes	Description
	<p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type</p>


Attributes	Description
filled (Shape Fill Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>(§20.1.2.3).</p> <p>Specifies whether the closed path is filled. Default is <code>true</code>. This attribute is overridden by the <code>fill on</code> attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
forcedash (Force Dashed Outline) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that a shape is a horizontal rule. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
hralign (Horizontal	<p>Specifies the alignment of a horizontal rule. Default is <code>left</code>.</p>

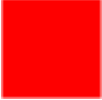

Attributes	Description
<p>Rule Alignment)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
<p>href (Hyperlink Target)</p>	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>hrnoshade (Horizontal Rule 3D Shading Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hrpct (Horizontal Rule Length Percentage)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>hrstd (Horizontal Rule Standard Display Toggle)</p>	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p>

Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"></pre>

Attributes	Description
	<p><code></v:fill></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to Identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>spt (Optional Number)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>strokecolor (Shape Stroke Color)</p>	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>stroked (Shape Stroke Toggle)</p>	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...></pre>

Attributes	Description				
	<p data-bbox="456 247 613 279"></v:shape></p>  <p data-bbox="415 453 574 485"><i>end example]</i></p> <p data-bbox="415 525 1390 590">The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
<p data-bbox="142 609 310 709">strokeweight (Shape Stroke Weight)</p>	<p data-bbox="415 609 1479 709">Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p data-bbox="415 751 532 783"><i>[Example:</i></p> <pre data-bbox="456 823 980 888"><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p data-bbox="415 1071 574 1102"><i>end example]</i></p> <p data-bbox="415 1142 1373 1207">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p data-bbox="142 1230 378 1295">style (Shape Styling Properties)</p>	<p data-bbox="415 1230 1479 1331">Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p data-bbox="415 1371 1451 1507">This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p data-bbox="415 1547 532 1579"><i>[Example:</i></p> <pre data-bbox="456 1583 1451 1682"><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p data-bbox="415 1686 574 1717"><i>end example]</i></p> <table data-bbox="415 1751 1479 1879"> <tr> <th data-bbox="415 1751 664 1801">Property</th><th data-bbox="664 1751 1479 1801">Description</th></tr> <tr> <td data-bbox="415 1801 664 1879">flip</td><td data-bbox="664 1801 1479 1879">Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</td></tr> </table>	Property	Description	flip	Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:
Property	Description				
flip	Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:				

Attributes	Description	
		<ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of</p>

Attributes	Description	
		<p>the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside

Attributes	Description	
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change</p>

Attributes	Description	
		the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.

Attributes	Description				
		<ul style="list-style-type: none">• <percentage>- Value expressed as a percentage of the parent object's height.			
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none">• hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.• inherit - The visibility state is inherited from the parent of the shape.			
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• auto - Default position of an element in the flow of the page.• <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <percentage>- Value expressed as a percentage of the parent object's width.			
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">• auto - Uses the order that the shapes appear in the page, bottom to top.• <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.			
The following properties are only used by the textbox element (§19.1.2.22):					
	<table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p><ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.</td></tr></table>	Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.
Property	Description				
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">• ltr - Text is displayed left-to-right.• rtl - Text is displayed right-to-left.				

Attributes	Description	
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> horizontal - Text is displayed horizontally. vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180 -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> top middle

Attributes	Description				
		<ul style="list-style-type: none">• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline			
	The following properties are only used by the textpath element (§19.1.2.23):				
	Property	Description			
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.			
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.			
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.			
	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.			
	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">• normal• small-caps			
	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td>Treated as non-bold.</td></tr></table>	Value	Description	normal
Value	Description				
normal	Treated as non-bold.				

Attributes	Description	
		lighter 100 200 300 400
		bold bolder 500 600 700 800 900 Treated as bold.
mso-text-shadow		Specifies whether a shadow is applied to the text on a text path. Default is false.
text-decoration		Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are: <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
v-rotate-letters		Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
v-same-letter-heights		Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
v-text-align		Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center

Attributes	Description	
		<ul style="list-style-type: none"> • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index 		

Attributes	Description																
	The possible values for this attribute are defined by the W3C XML Schema string datatype.																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 436 1482 1073"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre data-bbox="451 1182 1062 1314"><v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre data-bbox="451 1686 935 1751"><v:shape ... title="tooltip" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>																

Attributes	Description
	datatype.
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is <code>false</code>. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is <code>false</code>. If <code>true</code>, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the <code>mso-wrap-mode</code> style attribute is set to <code>tight</code> or <code>through</code>.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

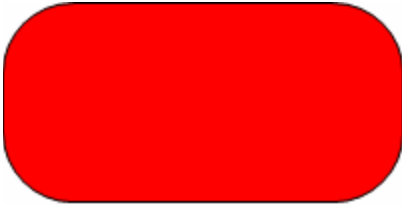
[Note: The W3C XML Schema definition of this element's content model ([CT_Rect](#)) is located in §A.6.1. *end note*]

19.1.2.17 `roundrect` (Rounded Rectangle)

This element is used to draw a rectangle with rounded corners. The CSS2 style content width and height define the width and height of the rectangle.

[Example:

```
<v:roundrect fillcolor="red" arcsize="35%"
  style="position:relative;top:0;left:0;width:200px;height:100px">
</v:roundrect>
```



end example]

Attributes	Description
allowincell (Allow in Table Cell) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether a shape can be placed in a table. Default is false. [Example: <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
allowoverlap (Allow Shape Overlap) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute. [Example: <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
alt (Alternate Text)	Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.



Attributes	Description
	<p>[<i>Example</i>: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>arcsize (Rounded Corner Arc Size)</p>	<p>Specifies the amount of roundness for a rounded rectangle as a percentage of half the smaller dimension of the length and width of the rectangle. Default is 20%. An arc size of 0% yields square corners and 100% forms circular corners. A square with an arc size value of 100% is a circle. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [<i>Example</i>: For example, a value of "52429f" represents 52429/65536 or 0.8. <i>end example</i>]</p> <p>[<i>Example</i>:</p> <pre><v:roundrect ... arcsize="35%"> </v:roundrect></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderbottomcolor (Bottom Border Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[<i>Example</i>:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>borderleftcolor (Border Left Color)</p>	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[<i>Example</i>:</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is <i>false</i>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace:	<p>Specifies whether a shape exhibits button press behavior on click. Default is <i>false</i>.</p> <p>[Example:</p>


Attributes	Description
urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:button="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. <i>[Example: Normal B&W might allow greyscale and pure B&W might not. end example]</i></p> <p><i>[Example: This shape renders in grayscale in a black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p>

Attributes	Description
microsoft-com:office:office	<pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p>[Example:</p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[Example: The snippets below are equivalent:</p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
clip (Clipping Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
<p>coordorigin (Coordinate Space Origin)</p>	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100"</pre>

Attributes	Description
	<pre>path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p>



Attributes	Description
	The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).
dgmlayoutmru (Diagram Node Recent Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element. <i>[Example:</i> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).
dgmnodekind (Diagram Node Identifier) Namespace: urn:schemas-microsoft-com:office:office	Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram. <i>[Example:</i> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema integer datatype.
doubleclicknotify (Double-click Notification Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that an event message is sent when a shape is double-clicked. Default is false. <i>[Example:</i> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
fillcolor (Fill Color)	Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.


Attributes	Description
	<p>[<i>Example:</i> This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>filled (Shape Fill Toggle)</p>	<p>Specifies whether the closed path is filled. Default is <code>true</code>. This attribute is overridden by the fill on attribute.</p> <p>[<i>Example:</i></p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[<i>Example:</i></p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>



Attributes	Description
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that a shape is a horizontal rule. Default is <code>false</code> . <i>[Example:</i> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	Specifies the alignment of a horizontal rule. Default is <code>left</code> . <i>[Example:</i> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).
href (Hyperlink Target)	Specifies a hyperlink URL target for the shape. Default is no value. <i>[Example:</i> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies that the horizontal rule does not have 3-D shading. Default is <code>false</code> . <i>[Example:</i> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
hrpct (Horizontal	Specifies the length of a horizontal rule as a percentage of page width. Default is 0.

Attributes	Description
<p>Rule Length Percentage)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>hrstd (Horizontal Rule Standard Display Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type</p>

Attributes	Description
	(§19.2.3.17).
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
ole (Embedded Object Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
oleicon (Embedded Object Icon Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
oned (Shape Handle Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>[Example: The red color is 25% opaque:</p> <pre data-bbox="451 617 1015 716"><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre> <div data-bbox="451 751 779 972">  opacity="1"  opacity=".25" </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre data-bbox="451 1381 1063 1444"><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre data-bbox="451 1780 901 1843"><v:shape ... print="false" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[<i>Example:</i> The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>spid (Optional String)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>spt (Optional Number)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>strokecolor (Shape Stroke Color)</p>	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[<i>Example:</i></p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre> 

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>stroked (Shape Stroke Toggle)</p>	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>strokeweight (Shape Stroke Weight)</p>	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>style (Shape Styling Properties)</p>	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the</p>

Attributes	Description									
	<p>surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></p>									
	<table> <tr> <th data-bbox="415 562 662 611">Property</th><th data-bbox="662 562 1497 611">Description</th></tr> <tr> <td data-bbox="415 611 662 877">flip</td><td data-bbox="662 611 1497 877"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> <tr> <td data-bbox="415 877 662 1289">height</td><td data-bbox="662 877 1497 1289"> <p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. </td></tr> <tr> <td data-bbox="415 1289 662 1766">left</td><td data-bbox="662 1289 1497 1766"> <p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. </td></tr> <tr> <td data-bbox="415 1766 662 1885">margin-bottom</td><td data-bbox="662 1766 1497 1885"> <p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of</p> </td></tr> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. 	margin-bottom
Property	Description									
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 									
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height. 									
left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width. 									
margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of</p>									

Attributes	Description	
		<p>the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the

Attributes	Description	
		parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different</p>

Attributes	Description	
	bottom	from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-left	Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-right	Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.

Attributes	Description	
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none"> • hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • inherit - The visibility state is inherited from the parent of the shape.
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Uses the order that the shapes appear in the page, bottom to top. • <order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on

Attributes	Description	
		top of those with lower numbers. Negative numbers are allowed.
	The following properties are only used by the textbox element (§19.1.2.22):	
	Property	Description
	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> ltr - Text is displayed left-to-right. rtl - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> horizontal - Text is displayed horizontally. vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0

Attributes	Description											
		<ul style="list-style-type: none">• 90• 180• -90										
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.										
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline										
The following properties are only used by the textpath element (§19.1.2.23):												
<table><tr><th>Property</th><th>Description</th></tr><tr><td>font</td><td>Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr><tr><td>font-family</td><td>Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr><tr><td>font-size</td><td>Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr><tr><td>font-style</td><td><p>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</p><ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.</td></tr></table>			Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	<p>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</p> <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.
Property	Description											
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.											
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.											
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.											
font-style	<p>Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are:</p> <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.											

Attributes	Description																		
	font-variant	<p>Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are:</p> <ul style="list-style-type: none">normalsmall-caps																	
	font-weight	<p>Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td rowspan="6">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr><tr><td>300</td></tr><tr><td>400</td></tr><tr><td>bold</td><td rowspan="7">Treated as bold.</td></tr><tr><td>bolder</td></tr><tr><td>500</td></tr><tr><td>600</td></tr><tr><td>700</td></tr><tr><td>800</td></tr><tr><td>900</td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.	bolder	500	600	700	800	900
	Value	Description																	
	normal	Treated as non-bold.																	
	lighter																		
100																			
200																			
300																			
400																			
bold	Treated as bold.																		
bolder																			
500																			
600																			
700																			
800																			
900																			
mso-text-shadow	<p>Specifies whether a shadow is applied to the text on a text path. Default is false.</p>																		
text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none">noneunderlineoverlineline-through																		

Attributes	Description	
		<ul style="list-style-type: none"> • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are: <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p>	

Attributes	Description																
	<ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 924 1481 1560"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>title (Shape Title)</p>	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... title="tooltip" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>wrapcoords (Shape Bounding Polygon)</p>	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p>

Attributes	Description
	<p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_RoundRect](#)) is located in §A.6.1.
end note]

19.1.2.18 shadow (Shadow Effect)


This element adds shadow effects to a shape. The on attribute shall be true for the shadow to be displayed.

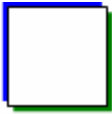
[Example:







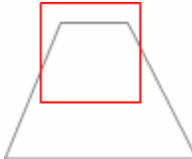
```
<v:shadow on="true" type="perspective"
  matrix="1.25,-2,,1.5,,.000001"
  offset="38pt,-6pt">
</v:shadow>
```


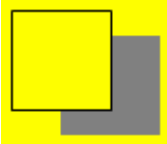



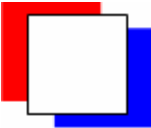
end example]


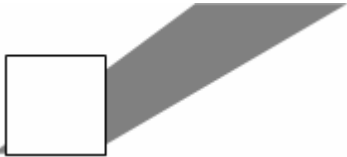
Attributes	Description
color (Shadow Primary Color)	<p>Specifies the color of the primary shadow. Default is gray (RGB 128,128,128).</p> <p>[Example:</p> <pre><v:shadow on="true" color="green"> </v:shadow></pre> <p>Applied to a simple square the shadow looks like this:</p> 

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>color2 (Shadow Secondary Color)</p>	<p>Specifies the color of the second shadow, or highlight in an embossed or engraved shadow. Default is light gray (RGB 203,203,203).</p> <p>[Example:</p> <pre><v:shadow on="true" type="double" color="green" color2="blue"> </v:shadow></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>matrix (Shadow Perspective Matrix)</p>	<p>Specifies a perspective transform for a shadow. Default is no value.</p> <p>The matrix is given in the form "s_{xx}, s_{xy}, s_{yx}, s_{yy}, p_x, p_y" where s = scale and p = perspective. If the offset attribute is in absolute units then p_x, p_y are in 1/EMU units; otherwise they are an inverse fraction of the shape size.</p> <p>[Example: The following snippets explain the matrix parameters. The shadow is applied to a simple square with no fill and a red stroke color (note there is a default shadow offset):</p>

Attributes	Description
	<div data-bbox="451 247 803 367">  <p>matrix=",,,,"</p> </div> <p data-bbox="414 403 1096 436">s_{xx}, s_{yy} specify scaling factors for the x and y dimensions:</p> <div data-bbox="451 472 917 592">  <p>matrix="2,,,,,"</p> </div> <div data-bbox="451 619 820 835">  <p>matrix=",,,2,,"</p> </div> <p data-bbox="414 871 987 907">s_{xy}, s_{yx} specify skews in the x and y dimensions:</p> <div data-bbox="451 940 1015 1060">  <p>matrix=",2,,,,,"</p> </div> <div data-bbox="451 1087 836 1402">  <p>matrix=",,-2,,,,"</p> </div> <p data-bbox="414 1438 1404 1474">p_x, p_y effectively set the perspective trapezoid skews along the x and y dimensions:</p> <div data-bbox="451 1507 917 1654">  <p>matrix=",,,,.000001,"</p> </div> <div data-bbox="451 1680 1015 1852">  <p>matrix=",,,,,-.000002"</p> </div>

Attributes	Description
	<p><i>end example]</i></p> <p>[Example:</p> <pre><v:shadow on="true" type="perspective" matrix="1.25,-2,,1.5,,.000001" offset="38pt,-6pt"> </v:shadow></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
obscured (Shadow Transparency)	<p>Specifies whether a shadow is transparent. Default is false. If true, the shadow is transparent if there is no fill on the shape.</p> <p>[Example:</p> <pre><v:background fillcolor="yellow"/> <v:shape style="width:50;height:50" filled="false" fillcolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:shadow on="true" offset="50%,25%" obscured="true"> </v:shadow> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
offset (Shadow Primary Offset)	<p>Specifies the primary shadow's x,y offset from the shape's location. Default is "2pt,2pt".</p> <p>Values are either an absolute measurement or a fractional value of the shape dimensions, from –50% to 50%.</p>

Attributes	Description
	<p>[Example:</p> <pre><v:shadow on="true" offset="50%,25%"> </v:shadow></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
offset2 (Shadow Secondary Offset)	<p>Specifies the secondary shadow's x,y offset from the shape's location. Default is "-2pt,-2pt".</p> <p>[Example:</p> <pre><v:shadow type="double" on="true" color="blue" offset="10pt,5pt" color2="red" offset2="-10pt,-5pt"> </v:shadow></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
on (Shadow Toggle)	<p>Specifies whether to show a shadow. Default is true.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Shadow Opacity)	<p>Specifies the opacity of the shadow. Default is 1. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example:</p> <pre><v:shadow type="double" on="true" opacity=".5" color="blue" offset="10pt,5pt"</pre>

Attributes	Description												
	<div><pre>color2="red" offset2="-10pt,-5pt"> </v:shadow></pre></div> <div></div> <div><p><i>end example]</i></p><p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p></div>												
origin (Shadow Origin)	<div><p>Specifies the center of the shadow relative to the shape's origin. Specified as a pair of fractional values of the shape dimensions, ranging from 50% to -50%. Default is "0,0".</p><p>[<i>Example:</i> This example is unchanged from above except for the addition of the origin attribute:</p><pre><v:shadow on="true" type="perspective" matrix="1.25,-2,,1.5,,.000001" offset="38pt,-6pt" origin="10%,-10%"> </v:shadow></pre></div> <div></div> <div><p><i>end example]</i></p><p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p></div>												
type (Shadow Type)	<div><p>Specifies the kind of shadow. Default is <code>single</code>. Allowed values are:</p><table><tr><th>Value</th><th>Description</th></tr><tr><td><code>single</code></td><td>Single shadow.</td></tr><tr><td><code>double</code></td><td>Double shadow. <code>color2</code> and <code>offset2</code> are used for the second shadow's color and offset.</td></tr><tr><td><code>perspective</code></td><td>Perspective shadow.</td></tr><tr><td><code>shaperelative</code></td><td>The shadow is created relative to the shape.</td></tr><tr><td><code>drawingrelative</code></td><td>The shadow is created relative to the drawing.</td></tr></table></div>	Value	Description	<code>single</code>	Single shadow.	<code>double</code>	Double shadow. <code>color2</code> and <code>offset2</code> are used for the second shadow's color and offset.	<code>perspective</code>	Perspective shadow.	<code>shaperelative</code>	The shadow is created relative to the shape.	<code>drawingrelative</code>	The shadow is created relative to the drawing.
Value	Description												
<code>single</code>	Single shadow.												
<code>double</code>	Double shadow. <code>color2</code> and <code>offset2</code> are used for the second shadow's color and offset.												
<code>perspective</code>	Perspective shadow.												
<code>shaperelative</code>	The shadow is created relative to the shape.												
<code>drawingrelative</code>	The shadow is created relative to the drawing.												

Attributes	Description	
	emboss	The shadow has an embossed look.
	The possible values for this attribute are defined by the ST_ShadowType simple type (§19.1.3.6).	

[Note: The W3C XML Schema definition of this element's content model ([CT_Shadow](#)) is located in §A.6.1. *end note*]

19.1.2.19 shape (Shape Definition)

This element is used to describe a shape, the core object in VML. This element can appear by itself or within a group element (§19.1.2.7). If a shapetype element (§19.1.2.20) is referenced using the type attribute, any attributes specified in the shape override those found in the shapetype.

[Example:

```
<v:shape style="position:absolute;top:50;left:20;width:50;height:50"
  path="m 0,0 l 0,1000 1000,1000 1000,0 x e">
  <v:shadow on="true" type="perspective"
    matrix="1.25,-2,,1.5,,.000001" offset="38pt,-6pt"/>
</v:shape>
```

```
<v:shape style="position:absolute;top:50;left:20;width:50;height:50"
  fillcolor="yellow" path="m 0,0 l 0,1000 1000,1000 1000,0 x e">
  <v:extrusion on="true" lightposition="0,-2000,10000"/>
</v:shape>
```



end example]

Attributes	Description
adj (Adjustment Parameters)	<p>Specifies a comma-delimited list of parameters, or adjustment values, used to define values for a parameterized formula. Values can be omitted. There can be up to 8 adjust values. Each value is referenced using # followed by a number corresponding to the zero-based index for that value in the list of adjustment values. [Example: For example, #2 references the second value in the adj list. <i>end example</i>]</p> <p>[Example: The following shape uses formulas to define a simple rectangle. The adj values</p>

Attributes	Description
	<p>are referenced by the eqn attribute of the f element (§19.1.2.4) and in turn referenced by the path element (§19.1.2.14).</p> <pre> <v:shape coordorigin="0 0" coordsize="200 200" style="position:relative;top:30;left:30;width:20;height:20" adj="1, 1, 1, 200, 200, 200, 200, 1"> <v:path v="m @0,@1 l @2,@3, @4,@5, @6,@7 x e"/> <v:formulas> <v:f eqn="val #0"/> <v:f eqn="val #1"/> <v:f eqn="val #2"/> <v:f eqn="val #3"/> <v:f eqn="val #4"/> <v:f eqn="val #5"/> <v:f eqn="val #6"/> <v:f eqn="val #7"/> </v:formulas> </v:shape> </pre> <p>This is the equivalent of:</p> <pre> <v:shape coordorigin="0 0" coordsize="200 200" style="position:relative;top:30;left:30;width:20;height:20" path="m 1,1 l 1,200, 200,200, 200,1 x e"> </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre> <v:shape ... o:allowincell="true" ... > </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p>



Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
alt (Alternate Text)	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the bottom border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderleftcolor (Border Left Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the left border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre>

Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
borderrightcolor (Border Right Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the right border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the top border color of an inline shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
bullet (Graphical Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the shape is a graphical bullet. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre>


Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>bwmode (Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. <i>[Example: Normal B&W might allow greyscale and pure B&W might not. end example]</i></p> <p><i>[Example: This shape renders in grayscale in a black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwnormal (Normal Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
<p>bwpure (Pure Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p><i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i></p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p>[Example:</p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[Example: The snippets below are equivalent:</p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
clip (Clipping Toggle) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p> <p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
<p>coordorigin (Coordinate Space Origin)</p>	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[Example: The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[Example: The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmlayout (Diagram Node Layout Identifier)	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>

Attributes	Description
<p>dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the orgchart value of the editas attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§19.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>equationxml (Storage for Alternate Math Content)</p>	<p>Specifies alternate XML markup which can be used to rehydrate an equation using the Office Open XML Math syntax. The actual format of the contents of this attribute is application-defined, but shall contain Office Open XML Math as well as any application-specific content. [Note: This form of storing alternate markup is inappropriate, and to be avoided in favor of the more flexible approach used by the child equationxml element (§19.2.2.10). end note]</p> <p>The XML markup stored in this attribute shall be escaped as needed to contain only those</p>

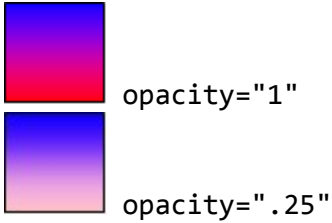
Attributes	Description
	<p>characters legal in an attribute value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
filled (Shape Fill Toggle)	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
forcedash (Force Dashed Outline)	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p>
Namespace:	Used by PresentationML placeholders to draw a dashed outline when there is no line and


Attributes	Description
urn:schemas-microsoft-com:office:office	<p>no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
gfxdata (Encoded Package) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies a base-64 encoded package as defined in ECMA-376-2 that contains DrawingML content as defined in ECMA-376-1. [<i>Rationale:</i> This attribute allows an application to use VML to represent graphical content for a legacy document while still persisting DrawingML for consuming applications that support DrawingML. For example, a diagram stored within this attribute would have the four parts defined for a DrawingML diagram, as well as any number of application-defined parts and relationships. end rationale]</p> <p>[Example: A DrawingML object is encoded in the gfxdata attribute, leaving VML to handle the visual display:</p> <pre><v:shape id="Diagram 1" o:spid="_x0000_i1025" type="#_x0000_t75" style="width:446.25pt;height:252pt; visibility:visible" o:gfxdata="UESDBBQABgAIAAAAIQDIu8KcTQE..."> <v:imagedata r:id="rId4" o:title=""/> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>
hr (Horizontal Rule Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hralign (Horizontal Rule Alignment)	<p>Specifies the alignment of a horizontal rule. Default is left.</p> <p>[Example:</p>

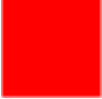

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
href (Hyperlink Target)	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
hrnoshade (Horizontal Rule 3D Shading Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
hrpct (Horizontal Rule Length Percentage) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
hrstd (Horizontal Rule Standard Display Toggle) Namespace:	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p> <p>[Example:</p>

Attributes	Description
urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetmode (Text Inset Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

Attributes	Description
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalseBlank</code> simple type (§20.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If <code>true</code>, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
path (Edge Path)	<p>Specifies the line that makes up the edges of a shape. See the v attribute of the path element (§19.1.2.14) for a full description.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
preferrelative (Relative Resize Toggle) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p><i>[Example:</i></p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
print (Print Toggle)	<p>Specifies whether the shape is printed. Default is true.</p> <p><i>[Example:</i></p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
regroupid (Regroup ID) Namespace: urn:schemas-	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p><i>[Example:</i> The shape was part of a group identified by the ID 040754:</p>

Attributes	Description
microsoft-com:office:office	<pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
spid (Optional String) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional string that an application can use to Identify the particular shape. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
spt (Optional Number) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p>

Attributes	Description
	<pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
strokeweight (Shape Stroke Weight)	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description	
	Property	Description
	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.

Attributes	Description	
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right

Attributes	Description	
		<ul style="list-style-type: none"> • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies the distance from the right side of the shape to the text</p>

Attributes	Description	
	distance-right	that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-distance-top	Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page.

Attributes	Description				
		<ul style="list-style-type: none">• <code><units></code>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <code><percentage></code>- Value expressed as a percentage of the parent object's height.			
	visibility	<p>Specifies whether a shape is displayed. Only <code>inherit</code> and <code>hidden</code> are used; any other values are mapped to <code>inherit</code>. Default is <code>inherit</code>. Allowed values are:</p> <ul style="list-style-type: none">• <code>hidden</code> - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.• <code>inherit</code> - The visibility state is inherited from the parent of the shape.			
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">• <code>auto</code> - Default position of an element in the flow of the page.• <code><units></code>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.• <code><percentage></code>- Value expressed as a percentage of the parent object's width.			
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">• <code>auto</code> - Uses the order that the shapes appear in the page, bottom to top.• <code><order></code>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.			
	<p>The following properties are only used by the <code>textbox</code> element (§19.1.2.22):</p> <table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the <code>textbox</code>. Default is <code>ltr</code>. This property is superceded by the <code>mso-direction-alt</code> property if that is specified. Allowed values are:</p></td></tr></table>		Property	Description	direction
Property	Description				
direction	<p>Specifies the direction of the text in the <code>textbox</code>. Default is <code>ltr</code>. This property is superceded by the <code>mso-direction-alt</code> property if that is specified. Allowed values are:</p>				

Attributes	Description	
		<ul style="list-style-type: none"> • ltr - Text is displayed left-to-right. • rtl - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none"> • horizontal - Text is displayed horizontally. • vertical - Text is displayed vertically. • vertical-ideographic - Ideographic text is displayed vertically. • horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:

Attributes	Description	
		<ul style="list-style-type: none">• top• middle• bottom• top-center• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline
	The following properties are only used by the textpath element (§19.1.2.23):	

Attributes	Description	
		<p>normal</p> <p>lighter</p> <p>100</p> <p>200</p> <p>300</p> <p>400</p> <p>Treated as non-bold.</p>
		<p>bold</p> <p>bolder</p> <p>500</p> <p>600</p> <p>700</p> <p>800</p> <p>900</p> <p>Treated as bold.</p>
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.
	text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left

Attributes	Description	
		<ul style="list-style-type: none"> • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width 		

Attributes	Description																
	<ul style="list-style-type: none"> • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
target (Hyperlink Display Target)	<p>Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:</p> <table border="1" data-bbox="415 508 1482 1144"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><targetname></td><td>String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td>_blank</td><td>Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td>_media</td><td>Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td>_parent</td><td>Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td>_search</td><td>Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td>_self</td><td>Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td>_top</td><td>Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p>end example]</p>																

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>type (Shape Type Reference)</p>	<p>Specifies a reference to a shapetype ID that describes the standard path, fill and stroke properties of a shape. Properties specified in the shape override the shapetype properties. Default is no value.</p> <p>[Example: The following example defines a shapetype that is a simple rectangle and an actual shape instance that uses it and overrides the fill color.</p> <pre> <v:shapetype id="mytype" fillcolor="red" strokecolor="blue" coordorigin="0 0" coordsize="200 200" path="m 0,0 l 0,200, 200,200, 200,0 x e"/> </v:shapetype> <v:shape id="shape02" type="#mytype" fillcolor="green" style="position:relative;top:1;left:1;width:20;height:20"> </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>userdrawn (Exists In Master Slide)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p> <p>[Example:</p> <pre> <v:shape ... o:userdrawn="true" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>userhidden (Hide Script Anchors)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre> <v:shape ... o:userhidden="true" ... > </v:shape> </pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
wrapcoords (Shape Bounding Polygon)	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Shape](#)) is located in §A.6.1. *end note]*

19.1.2.20 shapetype (Shape Template)

This element defines a shape template that can be used to create other shapes. Shapetype is identical to the shape element (§19.1.2.19) except it cannot reference another shapetype element. The type attribute shall not be used with shapetype. Attributes defined in the shape override any that appear in the shapetype. CSS positioning attributes (such as top, width, z-index, rotation, flip) are not passed to a shape from a shapetype. To use this element, create a shapetype with a specific id attribute. Then create a shape and reference the shapetype's id using the type attribute.

[Example:

```
<v:shapetype id="mytype" fillcolor="silver" strokecolor="blue">
  <v:path v="m 0,0 l 0,1000, 1000,1000, 1000,0 x e"/>
  <v:fill on="true" type="gradient" color2="navy" angle="-45"/>
</v:shapetype>
<v:shape type="#mytype"
  style="position:absolute;top:10;left:10;width:50;height:50"/>
<v:shape type="#mytype" fillcolor="teal"
  style="position:absolute;top:10;left:75;width:75;height:50"/>
<v:shape type="#mytype"
  style="position:absolute;top:10;left:165;width:50;height:50">
```

```
<v:fill type="solid"/>
</v:shape>
<v:shape type="#mytype" path="m 500,0 l 1000,1000 0,1000 x e"
style="position:absolute;top:10;left:230;width:50;height:50"/>
```



end example]

Attributes	Description
adj (Adjustment Parameters)	<p>Specifies a comma-delimited list of parameters, or adjustment values, used to define values for a parameterized formula. Values can be omitted. There can be up to 8 adjust values. Each value is referenced using # followed by a number corresponding to the zero-based index for that value in the list of adjustment values. [Example: For example, #2 references the second value in the adj list. end example]</p> <p>[Example: The following shape uses formulas to define a simple rectangle. The adj values are referenced by the eqn attribute of the f element (§19.1.2.4) and in turn referenced by the path element (§19.1.2.14).</p> <pre><v:shape coordorigin="0 0" coordsize="200 200" style="position:relative;top:30;left:30;width:20;height:20" adj="1, 1, 1, 200, 200, 200, 200, 1"> <v:path v="m @0,@1 l @2,@3, @4,@5, @6,@7 x e"/> <v:formulas> <v:f eqn="val #0"/> <v:f eqn="val #1"/> <v:f eqn="val #2"/> <v:f eqn="val #3"/> <v:f eqn="val #4"/> <v:f eqn="val #5"/> <v:f eqn="val #6"/> <v:f eqn="val #7"/> </v:formulas> </v:shape></pre> <p>This is the equivalent of:</p> <pre><v:shape coordorigin="0 0" coordsize="200 200" style="position:relative;top:30;left:30;width:20;height:20" path="m 1,1 l 1,200, 200,200, 200,1 x e"> </v:shape></pre> <p>end example]</p>



Attributes	Description
<p>allowincell (Allow in Table Cell)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p> <p>Specifies whether a shape can be placed in a table. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:allowincell="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape can overlap another shape. Default is true. If false, the shape shifts left or right so as not to overlap another shape, similar to the behavior of the HTML float attribute.</p> <p>[Example:</p> <pre><v:shape ... o:allowoverlap="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>alt (Alternate Text)</p>	<p>Specifies alternative text describing the graphical object. This text should provide a brief description of the shape for use by accessibility tools. Default is no value.</p> <p>[Example: The alt text describes the basic shape:</p> <pre><v:shape ... fillcolor="red" alt="Red rectangle"> </v:shape></pre> <p>The alt text describes the contents of a shape displaying an image:</p> <pre><v:shape ... alt="Picture of a sunset"> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
borderbottomcolor (Bottom Border Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the bottom border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:borderbottomcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
borderleftcolor (Border Left Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the left border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:borderleftcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
borderrightcolor (Border Right Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the right border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:borderrightcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
bordertopcolor (Border Top Color) Namespace: urn:schemas-microsoft-com:office:office	Specifies the top border color of an inline shape. Default is no value. <i>[Example:</i> <pre><v:shape ... o:bordertopcolor="red" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
bullet (Graphical	Specifies whether the shape is a graphical bullet. Default is <code>false</code> .


Attributes	Description
Bullet) Namespace: urn:schemas-microsoft-com:office:office	<p>[Example:</p> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
button (Button Behavior Toggle) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:button="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies how a shape renders for black-and-white output devices. When a shape is printed on a black-and-white printer or displayed in a black-and-white view in an application, several options are possible. Default is auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering.</p> <p>bwnormal and bwpure are subordinate to bwmode. If bwmode is "auto" then the value for bwnormal or bwpure is used depending on what the output format is. An application can define for itself what, if any, difference there is between normal B&W and pure B&W. [Example: Normal B&W might allow greyscale and pure B&W might not. end example]</p> <p>[Example: This shape renders in grayscale in a black-and-white environment:</p> <pre><v:shape ... o:bwmode="grayscale" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace:	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is auto.</p> <p>[Example: This shape renders in a pale grayscale in a normal black-and-white environment:</p>

Attributes	Description
urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p> <p>[<i>Example:</i> This shape renders in high contrast when in a pure black-and-white environment:</p> <pre><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§19.2.3.3).</p>
chromakey (Image Transparency Color)	<p>Specifies a color value that is transparent and show anything behind the shape. Default is no value.</p> <p>[<i>Example:</i></p> <pre><v:image ... chromakey="white" ...> </v:image></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
class (CSS Reference)	<p>Specifies a reference to the definition of a CSS style. Default is no value.</p> <p>[<i>Example:</i> The snippets below are equivalent:</p> <pre>... .narrowstyle {width:50;height:100} ... <v:shape ... class="narrowstyle" style="top:1;left:1"> </v:shape></pre>

Attributes	Description
	<pre><v:shape ... style="top:1;left:1; width:50;height:100"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>clip (Clipping Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§19.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>cliptowrap (Clip to Wrapping Polygon)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that the clipping region for the shape aligns with the wrapping polygon that tightly surrounds the entire shape (essentially, that the shape shall not be drawn beyond its wrapping polygon's extents – if it does, the shape shall be clipped). Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:cliptowrap="true"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre><v:shape ... o:connectortype="elbow" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ConnectorType simple type (§19.2.3.7).</p>
<p>coordorigin</p>	<p>Specifies the coordinate of the top left corner of the shape's coordinate space. This</p>

Attributes	Description
(Coordinate Space Origin)	<p>determines the position of the (0,0) coordinate space origin within the shape's bounding box. Default is "0,0", which places the (0,0) origin at the top left corner of the bounding box.</p> <p>This affects shape properties that specify coordinate positions, such as the path attribute. Thus a path can be defined against a generic (0,0) origin and the coordorigin value translates the entire path within the shape's bounding space.</p> <p>[<i>Example:</i> The horizontal and vertical coordinate space ranges from -100 to +100 because the coordinate space (coordsize) is 200 by 200 and the top left coordinate is (-100,-100). The (0,0) origin lies at the center of the shape's bounding box, as evidenced by the position of the shape's path within the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
coordsize (Coordinate Space Size)	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is "1000,1000".</p> <p>The physical size of a coordinate unit length is determined by both the size of the coordinate space (coordsize) and the size of the shape (style width and height). The coordsize attribute defines the number of horizontal and vertical subdivisions into which the shape's bounding box is divided. The combination of coordsize and style width/height effective scales the shape anisotropically.</p> <p>[<i>Example:</i> The path is 50 units wide and tall, which is 25% of the size of the coordinate space:</p> <pre><v:shape ... coordsize="200,200" coordorigin="-100,-100" path="m 0,0 l 0,50, 50,50, 50,0 x e"> </v:shape></pre> 


Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>dgmlayout (Diagram Node Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the <code>orgchart</code> value of the <code>editas</code> attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_DiagramLayout</code> simple type (§19.2.3.10).</p>
<p>dgmlayoutmru (Diagram Node Recent Layout Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the kind of automatic layout most recently used on the child elements of the diagram node. This is only meaningful if the shape is a node in an organization chart, which is denoted by the <code>orgchart</code> value of the <code>editas</code> attribute of the group element.</p> <p>[Example:</p> <pre><v:shape ... dgmlayout="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_DiagramLayout</code> simple type (§19.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined parameter that is intended to be used by the application to tag different types of nodes in a diagram.</p> <p>[Example:</p> <pre><v:shape ... dgmnodekind="1"> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>doubleclicknotify (Double-click</p>	<p>Specifies that an event message is sent when a shape is double-clicked. Default is <code>false</code>.</p>

Attributes	Description
<p>Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§19.1.2.5) is present, its color attribute takes precedence. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example: This shape is red if its fill is visible:</p> <pre><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre><v:shape ... fillcolor="#ff0000" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>filled (Shape Fill Toggle)</p>	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[Example:</p> <pre><v:shape ... filled="f" fillcolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>



Attributes	Description
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hr (Horizontal Rule Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies that a shape is a horizontal rule. Default is <code>false</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hralign (Horizontal Rule Alignment)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the alignment of a horizontal rule. Default is <code>left</code>.</p> <p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§19.2.3.16).</p>
<p>href (Hyperlink Target)</p>	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p> <pre><v:shape ... href="http://www.openxmlformats.org" ... > </v:shape></pre> <p><i>end example]</i></p>


Attributes	Description
<p>hrnoshade (Horizontal Rule 3D Shading Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p> <p>Specifies that the horizontal rule does not have 3-D shading. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrnoshade="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>hrpct (Horizontal Rule Length Percentage)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the length of a horizontal rule as a percentage of page width. Default is 0.</p> <p>[Example:</p> <pre><v:shape ... o:hrpct="85" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>hrstd (Horizontal Rule Standard Display Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a shape is displayed as a standard horizontal rule. Only applies if hr is true. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hrstd="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom. This attribute is only meaningful for text boxes.</p> <p>[Example:</p> <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>master (Master Element Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shapetype is a master element. If true, it is rendered by the rendering engine. Default is false.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the shape is an embedded object. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).
<p>oleicon (Embedded Object Icon Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>oned (Shape Handle Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the extra handles of a shape are hidden. If true, hides all shape handles except the top left and bottom right; that is, the same handles that are used for a straight line segment. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Fill Color Opacity)</p>	<p>Specifies the opacity of the primary fill color. Default is 1.0. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied. [Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</p> <p>[Example: The red color is 25% opaque:</p> <pre><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre> <div data-bbox="451 1551 779 1770">  <p>opacity="1"</p> <p>opacity=".25"</p> </div> <p>end example]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
path (Edge Path)	<p>Specifies the line that makes up the edges of a shape. See the v attribute of the path element (§19.1.2.14) for a full description.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>preferrelative (Relative Resize Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the original size of an object is saved after reformatting. Default is false. If true, the original size of the object is stored and all resizing is based on a percentage of that original size. Otherwise, each resizing resets the scale to 100%.</p> <p>[Example:</p> <pre><v:shape ... o:preferrelative="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
print (Print Toggle)	<p>Specifies whether the shape is printed. Default is true.</p> <p>[Example:</p> <pre><v:shape ... print="false" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>regroupid (Regroup ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies a previous group for a shape. An ID number is used to identify groups of shapes that are no longer grouped. This allows shapes to be regrouped programmatically.</p> <p>[Example: The shape was part of a group identified by the ID 040754:</p> <pre><v:shape ... o:regroupid="040754" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
spid (Optional	Specifies an optional string that an application can use to identify the particular shape.

Attributes	Description
String) Namespace: urn:schemas-microsoft-com:office:office	Default is no value. The possible values for this attribute are defined by the W3C XML Schema string datatype.
spt (Optional Number) Namespace: urn:schemas-microsoft-com:office:office	Specifies an optional number that an application can use to associate the particular shape with a defined shape type. Default is 0. The possible values for this attribute are defined by the W3C XML Schema float datatype.
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre><v:shape ... fillcolor="red" stroked="false" strokecolor="blue"...> </v:shape></pre> 

Attributes	Description				
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
<p>strokeweight (Shape Stroke Weight)</p>	<p>Specifies the width of the brush to use to stroke the path. Default is 1 point. If a number is given without units, the emu is used. The weight attribute of the stroke element (§19.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p>style (Shape Styling Properties)</p>	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table border="1" data-bbox="415 1549 1479 1860"> <thead> <tr> <th data-bbox="415 1549 664 1596">Property</th><th data-bbox="664 1549 1479 1596">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1596 664 1860">flip</td><td data-bbox="664 1596 1479 1860"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> </tbody> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
Property	Description				
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 				

Attributes	Description	
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm,

Attributes	Description	
		<p>mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.</p> <ul style="list-style-type: none"> • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-top	<p>Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the</p>

Attributes	Description	
		origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:

Attributes	Description	
		<ul style="list-style-type: none"> • <code>hidden</code> - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed. • <code>inherit</code> - The visibility state is inherited from the parent of the shape.
	<code>width</code>	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • <code>auto</code> - Default position of an element in the flow of the page. • <code><units></code>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <code><percentage></code>- Value expressed as a percentage of the parent object's width.
	<code>z-index</code>	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • <code>auto</code> - Uses the order that the shapes appear in the page, bottom to top. • <code><order></code>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.
The following properties are only used by the textbox element (§19.1.2.22):		
	Property	Description
	<code>direction</code>	<p>Specifies the direction of the text in the textbox. Default is <code>ltr</code>. This property is superseded by the <code>mso-direction-alt</code> property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> • <code>ltr</code> - Text is displayed left-to-right. • <code>rtl</code> - Text is displayed right-to-left.
	<code>layout-flow</code>	<p>Determines the flow of the text layout in a textbox. Default is <code>horizontal</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>horizontal</code> - Text is displayed horizontally. • <code>vertical</code> - Text is displayed vertically.

Attributes	Description	
		<ul style="list-style-type: none"> vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180 -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> top middle bottom top-center middle-center bottom-center top-baseline

Attributes	Description								
		<ul style="list-style-type: none">bottom-baselinetop-center-baselinebottom-center-baseline							
	The following properties are only used by the textpath element (§19.1.2.23):								
	Property	Description							
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.							
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.							
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.							
	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">normalitalicoblique - Treated the same as italic.							
	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">normalsmall-caps							
	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td rowspan="4">Treated as non-bold.</td></tr><tr><td>lighter</td></tr><tr><td>100</td></tr><tr><td>200</td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200
	Value	Description							
normal	Treated as non-bold.								
lighter									
100									
200									

Attributes	Description		
		300 400	
		bold bolder 500 600 700 800 900	Treated as bold.
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.	
	text-decoration	Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are: <ul style="list-style-type: none">• none• underline• overline• line-through• blink	
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.	
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.	
	v-text-align	Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none">• left• right• center• justify• letter-justify - Distributes the extra space between the letters.• stretch-justify - Stretches the letters to fill in the space.	

Attributes	Description	
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	
target (Hyperlink Display Target)	Specifies a frame or window that a URL is displayed in. Default is no value. Allowed values are:	

Attributes	Description																
	<table border="1"> <thead> <tr> <th data-bbox="414 247 625 296">Value</th><th data-bbox="625 247 1482 296">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="414 296 625 380"><targetname></td><td data-bbox="625 296 1482 380">String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td data-bbox="414 380 625 464">_blank</td><td data-bbox="625 380 1482 464">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="414 464 625 548">_media</td><td data-bbox="625 464 1482 548">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="414 548 625 632">_parent</td><td data-bbox="625 548 1482 632">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="414 632 625 716">_search</td><td data-bbox="625 632 1482 716">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="414 716 625 800">_self</td><td data-bbox="625 716 1482 800">Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> <tr> <td data-bbox="414 800 625 884">_top</td><td data-bbox="625 800 1482 884">Specifies that the linked document is loaded into the topmost window.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Description	<targetname>	String containing the name of the frame or window in which to load the document.	_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.	_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.	_search	Specifies that the linked document is loaded into the browser's search pane.	_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).	_top	Specifies that the linked document is loaded into the topmost window.
Value	Description																
<targetname>	String containing the name of the frame or window in which to load the document.																
_blank	Specifies that the linked document is loaded into a new blank window. This window is not named.																
_media	Specifies that the linked document is loaded into the browser's multimedia pane.																
_parent	Specifies that the linked document is loaded into the immediate parent of the document containing the link.																
_search	Specifies that the linked document is loaded into the browser's search pane.																
_self	Specifies that the linked document is loaded into the window in which the link was clicked (the active window).																
_top	Specifies that the linked document is loaded into the topmost window.																
title (Shape Title)	<p>Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>																
userdrawn (Exists In Master Slide)	<p>Specifies whether the user has added the shape to a master slide. Default is false. Used by PresentationML.</p>																
Namespace:	<p>[Example:</p>																

Attributes	Description
urn:schemas-microsoft-com:office:office	<pre><v:shape ... o:userdrawn="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
userhidden (Hide Script Anchors) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies whether a script anchor is hidden. Default is false. If true, script anchors stay hidden even if the shape is otherwise visible. A script anchor is the visual representation of a script that when displayed in an application.</p> <p>[Example:</p> <pre><v:shape ... o:userhidden="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
wrapcoords (Shape Bounding Polygon)	<p>Specifies the bounding polygon that surrounds a shape. This is specified using a comma-delimited list of x and y coordinates: "x1,y1,x2,y2,x3,y3,..." This is used when text is tightly wrapped around a shape. Default is no value until the mso-wrap-mode style attribute is set to tight or through.</p> <p>[Example:</p> <pre><v:shape ... wrapcoords="0,0 0,200, 200,200, 200,0" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Shapetype](#)) is located in §A.6.1. *end note]*

19.1.2.21 stroke (Line Stroke Settings)

This element describes how to draw the path if something beyond solid line with a solid color is desired. The attributes of the stroke element can be used to describe a powerful set of stroke properties. Extensions to the VML stroke definition are encoded as sub-elements of stroke.



[Example:



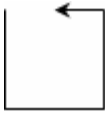
```
<v:polyline points="0pt,0pt,50pt,0pt,50pt,35pt,15pt,35pt,
15pt,15pt,75pt,15pt">
  <v:stroke startarrow="classic" endarrow="classic"
    startarrowwidth="wide" endarrowwidth="wide" dashstyle="dashdot"
    weight="2pt" color="teal" linestyle="thinThin"/>
</v:polyline>
```


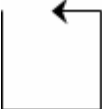





end example]

Attributes	Description
althref (Alternate Image Reference) Namespace: urn:schemas-microsoft-com:office:office	Specifies an alternate reference for an image in Macintosh PICT format. [Example: <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
color (Stroke Color)	Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors. [Example: The shape stroke is blue: <pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
color2 (Stroke Alternate Pattern Color)	Specifies a second color for strokes, used when filltype is pattern. Default is no value. When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.



Attributes	Description
	<p>[<i>Example</i>: This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre> <v:background fillcolor="yellow"/> <v:shape style="width:60;height:50" strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape> </pre>  <p>, where myimage.gif is: </p> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
dashstyle (Stroke Dash Pattern)	<p>Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p> <ul style="list-style-type: none"> • solid • shortdash • shortdot • shortdashdot • shortdashdotdot • dot • dash • longdash • dashdot • longdashdot • longdashdotdot <p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is</p>


Attributes	Description
	<p>fourfold symmetrical (with round end caps, this is a circle).</p> <p>[Example:</p> <pre data-bbox="451 394 1062 491"><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre>  <pre data-bbox="451 680 1029 777"><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>endarrow (Line End Arrowhead)</p>	<p>Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul data-bbox="461 1218 613 1436" style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre data-bbox="451 1575 932 1604"><v:stroke endarrow="classic"/></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple</p>


Attributes	Description
endarrowlength (Line End Arrowhead Length)	<p>type (§19.1.3.8).</p> <p>Specifies the length of the arrowhead at the end of a line. Default is <code>medium</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>short</code> • <code>medium</code> • <code>long</code> <p>[Example:</p> <pre><v:stroke ... endarrowlength="long" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_StrokeArrowLength</code> simple type (§19.1.3.7).</p>
endarrowwidth (Line End Arrowhead Width)	<p>Specifies the width of the arrowhead at the end of a line. Default is <code>medium</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>narrow</code> • <code>medium</code> • <code>wide</code> <p>[Example:</p> <pre><v:stroke ... endarrowwidth="wide" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_StrokeArrowWidth</code> simple type (§19.1.3.9).</p>
endcap (Line End Cap)	<p>Specifies the cap style for the end of a stroke. Default is <code>flat</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>flat</code> • <code>square</code>



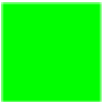
Attributes	Description
	<ul style="list-style-type: none"> • round <p>[Example:</p> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p>endcap="flat"</p> <p>endcap="square"</p> <p>endcap="round"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
filltype (Stroke Image Style)	<p>Specifies the kind of fill used for the background of a stroke. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • solid - The fill pattern is solid. • tile - The fill image is tiled. • pattern - The fill image is stretched to form a pattern. • frame - The fill image becomes a border for the shape. <p>[Example:</p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre>  <p>, where border.gif is: </p> <p>end example]</p>


Attributes	Description
	The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).
<p>forcedash (Force Dashed Outline)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>href (Original Image Reference)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>id (Relationship)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID of the relationship to the image used for the stroke. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information:</p> <pre>< ... r:id="rId10" /></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
<p>id (Unique Identifier)</p>	<p>Specifies a unique identifier that can be used to reference a VML object.</p>



Attributes	Description				
	<p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p>imagealignshape (Stoke Image Alignment)</p>	<p>Specifies the alignment of the stroke image. If true, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is true.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre> <div data-bbox="451 1098 609 1255">  </div> <p>imagealignshape="false"</p> <div data-bbox="451 1291 609 1449">  </div> <p>imagealignshape="false"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
<p>imageaspect (Stroke Image Aspect Ratio)</p>	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table border="1" data-bbox="418 1753 1318 1879"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>ignore</td><td>Ignore aspect issues.</td></tr> </tbody> </table>	Value	Description	ignore	Ignore aspect issues.
Value	Description				
ignore	Ignore aspect issues.				

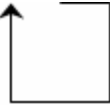
Attributes	Description				
	<table><tr><td>atleast</td><td>Image is at least as big as imagesize.</td></tr><tr><td>atmost</td><td>Image is no bigger than imagesize.</td></tr></table> <p>[Example:</p> <pre><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre> <div></div> <p>imagealignshape="ignore"</p> <p>imagealignshape="atleast"</p> <p>imagealignshape="atmost"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>	atleast	Image is at least as big as imagesize.	atmost	Image is no bigger than imagesize.
atleast	Image is at least as big as imagesize.				
atmost	Image is no bigger than imagesize.				
imagesize (Stroke Image Size)	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre><v:stroke ... imagesize="10pt,10pt" ... /></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre>				

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>joinstyle (Line End Join Style)</p>	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none"> • round • bevel • miter <p>[Example:</p> <pre><v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline></pre>  <p>joinstyle="round"</p> <p>joinstyle="bevel"</p> <p>joinstyle="miter"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p>
<p>linestyle (Stroke Line Style)</p>	<p>Specifies the line style of the stroke. Default is single.</p> <ul style="list-style-type: none"> • single • thinThin • thinThick • thickThin • thickBetweenThin <p>[Example:</p>

Attributes	Description
	<pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§19.1.3.12).</p>
miterlimit (Miter Joint Limit)	<p>Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p> <p>[Example:</p> <pre><v:stroke jointstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
on (Stroke Toggle)	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

Attributes	Description
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
relid (Relationship to Part) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the relationship ID of the relationship to the image. The specified relationship shall be of type http://schemas.openxmlformats.org/officeDocument/2006/relationships/image or the document shall be considered non-conformant.</p> <p>[Example: The markup specifies the associated relationship part with relationship ID rId10 contains the corresponding relationship information:</p> <pre><v:stroke ... o:relid="rId10" ...> </v:stroke></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
src (Stroke Image Location)	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre><v:stroke ... src="myimage.gif" ... > </v:stroke></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
startarrow (Line Start Arrowhead)	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p>

Attributes	Description
	<ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke startarrow="classic"/></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... startarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium

Attributes	Description
	<ul style="list-style-type: none"> • wide <p>[Example:</p> <pre><v:stroke ... startarrowwidth="wide" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
<p>title (Stroke Title)</p> <p>Namespace: urn:schemas- microsoft- com:office:office</p>	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>weight (Stroke Weight)</p>	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Stroke](#)) is located in §A.6.1. *end note*]

19.1.2.22 [textbox \(Text Box\)](#)

This element is used to define text that appears inside the shape. This text can contain rich formatting and is rendered to fit inside the textboxrect defined by the path element (§19.1.2.14).


[Example:

```
<v:shape style="width=200;height=200" coordsize="400,400"
  fillcolor="yellow" strokecolor="maroon"
```

```
path="m 119,0 l 148,86 238,86 166,140 192,226 119,175 46,226
72,140 0,86 90,86 x e">
<v:textbox inset="32pt,35pt,, ">VML</v:textbox>
</v:shape>
```



end example]

Attributes	Description
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
inset (Text Box Inset)	<p>Specifies inner margin values for textbox text. Default is "0.1in, 0.05in, 0.1in, 0.05in". Missing values are set to the default. This is used if insetmode is custom.</p> <p>The internal text margin value is specified as a string containing four values, each separated by commas or spaces. The values measure inset from the left, top, right, and bottom edges of the box specified by the textboxrect attribute of the path element (§19.1.2.14).</p> <p>[Example: The text is set toward the lower right of a small square:</p> <pre><v:textbox inset="20pt,30pt,10pt,10pt"> VML</v:textbox></pre> 

Attributes	Description				
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether the application calculates the internal text margin instead of using the inset attribute. Default is custom.</p> <p>[Example:</p> <pre><v:textbox ... o:insetmode="auto" ... > </v:textbox></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§19.2.3.17).</p>				
<p>singleclick (Text Box Single-Click Selection Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies whether text is selectable with a single click. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
<p>style (Shape Styling Properties)</p>	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table border="1" data-bbox="415 1661 1479 1854"> <thead> <tr> <th data-bbox="415 1661 664 1707">Property</th><th data-bbox="664 1661 1479 1707">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1707 664 1854">flip</td><td data-bbox="664 1707 1479 1854"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. </td></tr> </tbody> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates.
Property	Description				
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. 				

Attributes	Description	
		<ul style="list-style-type: none"> • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-</p>

Attributes	Description	
	relative	<p>position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies the distance from the top of the shape to the text that</p>

Attributes	Description	
	distance-top	wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are: <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are: <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are: <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.

Attributes	Description							
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none">hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.inherit - The visibility state is inherited from the parent of the shape.						
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">auto - Default position of an element in the flow of the page.<units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.<percentage>- Value expressed as a percentage of the parent object's width.						
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">auto - Uses the order that the shapes appear in the page, bottom to top.<order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.						
<p>The following properties are only used by the textbox element (§19.1.2.22):</p> <table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p><ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.</td></tr><tr><td>layout-flow</td><td><p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p></td></tr></table>			Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p>
Property	Description							
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.							
layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p>							

Attributes	Description	
		<ul style="list-style-type: none"> • <code>horizontal</code> - Text is displayed horizontally. • <code>vertical</code> - Text is displayed vertically. • <code>vertical-ideographic</code> - Ideographic text is displayed vertically. • <code>horizontal-ideographic</code> - Ideographic text is displayed horizontally.
	<code>mso-direction-alt</code>	Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code> .
	<code>mso-fit-shape-to-text</code>	Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code> .
	<code>mso-fit-text-to-shape</code>	Specifies whether the text stretches to fit the textbox. Default is <code>false</code> .
	<code>mso-layout-flow-alt</code>	Specifies the alternate layout flow for text in textboxes. This property is used instead of <code>layout-flow</code> when the layout flow is from bottom to top for non-ideographic languages. Its only value is <code>bottom-to-top</code> .
	<code>mso-next-textbox</code>	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	<code>mso-rotate</code>	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	<code>mso-text-scale</code>	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if <code>mso-fit-text-to-shape</code> is <code>true</code> .
	<code>v-text-anchor</code>	<p>Specifies the vertical anchoring of text in a textbox. Default is <code>top</code>. The alignment of a text anchor only becomes evident if <code>mso-fit-text-to-shape</code> is <code>false</code>. This property is different from the <code>vertical-align</code> CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> • <code>top</code> • <code>middle</code> • <code>bottom</code> • <code>top-center</code>

Attributes	Description							
		<ul style="list-style-type: none">• middle-center• bottom-center• top-baseline• bottom-baseline• top-center-baseline• bottom-center-baseline						
	The following properties are only used by the textpath element (§19.1.2.23):							
	Property	Description						
	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.						
	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.						
	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.						
	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none">• normal• italic• oblique - Treated the same as italic.						
	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none">• normal• small-caps						
	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>normal</td><td>Treated as non-bold.</td></tr><tr><td>lighter</td><td></td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	
	Value	Description						
normal	Treated as non-bold.							
lighter								

Attributes	Description	
		100 200 300 400
		bold bolder 500 600 700 800 900 Treated as bold.
mso-text-shadow		Specifies whether a shadow is applied to the text on a text path. Default is false.
text-decoration		Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are: <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
v-rotate-letters		Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
v-same-letter-heights		Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
v-text-align		Specifies the alignment of text. Default is left. Allowed values are: <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between

Attributes	Description	
		<p>the letters.</p> <ul style="list-style-type: none"> stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> tightening tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none"> top left width height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> flip height left margin-left margin-top position rotation top visibility width z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		

[Note: The W3C XML Schema definition of this element's content model ([CT Textbox](#)) is located in §A.6.1. *end note*]

19.1.2.23 `textpath` (Text Layout Path)







This element is used to define a vector path based on the text data, font and font styles supplied. The path which results is then mapped into the region defined by the `v` attribute of the shape's path (§19.1.2.14).


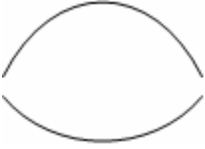
[Example:

```
<v:curve from="50,100" to="400,100"
  control1="200,200" control2="300,200">
  <v:stroke color="blue"/>
  <v:fill color="yellow" color2="green" type="gradient"/>
  <v:path textpathok="true"/>
  <v:textpath on="true" style="font:normal normal normal 36pt Arial"
    fitpath="true" string="Hello, VML!"/>
</v:curve>
```



end example]

Attributes	Description				
fitpath (Path Fit Toggle)	<p>Specifies whether the text fits the path of a shape. If true, sizes the text to fill the path it lies out on. Default is false.</p> <p>[Example:</p> <pre><v:textpath on="true" fitpath="true" string="VML"> </v:textpath></pre> <table border="1"> <tr> <td></td><td>fitpath="true"</td></tr> <tr> <td></td><td>fitpath="false"</td></tr> </table> <p><i>end example</i>]</p>		fitpath="true"		fitpath="false"
	fitpath="true"				
	fitpath="false"				

Attributes	Description
fitshape (Shape Fit Toggle)	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p> <p>Specifies whether the text fits the bounding box of a shape. If true, the text is stretched out to the edges of the box that defines the entire shape. Default is false.</p> <p>[Example: When fitshape is false, the text is drawn along the first part of the path. When true, the text is stretched to fit the entire enclosed area of the shape.</p> <pre> <v:shape style="width:100;height:100" path="m 0,500 c 250,0 750,0 1000,500 e m 0,600 c 250,900 750,900 1000,600 e" fillcolor="yellow" strokecolor="maroon"> <v:path textpathok="t"/> <v:textpath on="t" fitshape="t" string="VML"/> </v:shape> </pre>  <p>The raw path stroke is:</p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
id (Unique Identifier)	<p>Specifies a unique identifier that can be used to reference a VML object.</p> <p>Default is no value.</p> <p>[Example:</p> <pre> <v:shape ... id="myShape" ... > </v:shape> </pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
on (Text Path Toggle)	<p>Specifies whether the text is displayed on the textpath. Default is false. The textpathok attribute of the path element (§19.1.2.14) overrides this.</p>

Attributes	Description						
	<p>[Example:</p> <pre> <v:line from="50,100" to="100,100"> <v:path textpathok="false"/> <v:textpath on="true" string="VML"/> </v:line> </pre> <p>_____</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>						
string (Text Path Text)	<p>Specifies the text of the text path. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>						
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre> <v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape> </pre> <p>end example]</p> <table border="1" data-bbox="415 1438 1479 1862"> <thead> <tr> <th data-bbox="415 1438 664 1484">Property</th><th data-bbox="664 1438 1479 1484">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1484 664 1755">flip</td><td data-bbox="664 1484 1479 1755"> <p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. </td></tr> <tr> <td data-bbox="415 1755 664 1862">height</td><td data-bbox="664 1755 1479 1862"> <p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> </td></tr> </tbody> </table>	Property	Description	flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>
Property	Description						
flip	<p>Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis. 						
height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>						

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the

Attributes	Description	
		parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-relative	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text

Attributes	Description	
		<ul style="list-style-type: none"> • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-top	<p>Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-	<p>Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this</p>

Attributes	Description	
	edited	property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	<p>Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is inherit. Allowed values are:</p> <ul style="list-style-type: none"> • hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not


Attributes	Description	
		<p>processed.</p> <ul style="list-style-type: none"> • <code>inherit</code> - The visibility state is inherited from the parent of the shape.
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • <code>auto</code> - Default position of an element in the flow of the page. • <code><units></code>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <code><percentage></code>- Value expressed as a percentage of the parent object's width.
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • <code>auto</code> - Uses the order that the shapes appear in the page, bottom to top. • <code><order></code>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.
The following properties are only used by the textbox element (§19.1.2.22):		
	Property	Description
	direction	<p>Specifies the direction of the text in the textbox. Default is <code>ltr</code>. This property is superceded by the <code>mso-direction-alt</code> property if that is specified. Allowed values are:</p> <ul style="list-style-type: none"> • <code>ltr</code> - Text is displayed left-to-right. • <code>rtl</code> - Text is displayed right-to-left.
	layout-flow	<p>Determines the flow of the text layout in a textbox. Default is <code>horizontal</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>horizontal</code> - Text is displayed horizontally. • <code>vertical</code> - Text is displayed vertically. • <code>vertical-ideographic</code> - Ideographic text is displayed vertically. • <code>horizontal-ideographic</code> - Ideographic text is displayed

Attributes	Description	
		horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> • 0 • 90 • 180 • -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> • top • middle • bottom • top-center • middle-center • bottom-center • top-baseline • bottom-baseline • top-center-baseline • bottom-center-baseline

Attributes	Description																									
	<p>The following properties are only used by the textpath element (§19.1.2.23):</p> <table border="1"> <tr> <th data-bbox="415 352 664 401">Property</th><th data-bbox="664 352 1479 401">Description</th></tr> <tr> <td data-bbox="415 401 664 554">font</td><td data-bbox="664 401 1479 554">Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr> <tr> <td data-bbox="415 554 664 638">font-family</td><td data-bbox="664 554 1479 638">Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr> <tr> <td data-bbox="415 638 664 758">font-size</td><td data-bbox="664 638 1479 758">Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr> <tr> <td data-bbox="415 758 664 1031">font-style</td><td data-bbox="664 758 1479 1031"> Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. </td></tr> <tr> <td data-bbox="415 1031 664 1255">font-variant</td><td data-bbox="664 1031 1479 1255"> Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps </td></tr> <tr> <td data-bbox="415 1255 664 1871">font-weight</td><td data-bbox="664 1255 1479 1871"> Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <tr> <th data-bbox="677 1404 878 1453">Value</th><th data-bbox="878 1404 1466 1453">Description</th></tr> <tr> <td data-bbox="677 1453 878 1808">normal</td><td data-bbox="878 1453 1466 1808" rowspan="6">Treated as non-bold.</td></tr> <tr> <td data-bbox="677 1520 878 1556">lighter</td></tr> <tr> <td data-bbox="677 1583 878 1619">100</td></tr> <tr> <td data-bbox="677 1646 878 1682">200</td></tr> <tr> <td data-bbox="677 1709 878 1745">300</td></tr> <tr> <td data-bbox="677 1772 878 1808">400</td></tr> <tr> <td data-bbox="677 1808 878 1871">bold</td><td data-bbox="878 1808 1466 1871">Treated as bold.</td></tr> </table> </td></tr> </table>	Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. 	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps 	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <tr> <th data-bbox="677 1404 878 1453">Value</th><th data-bbox="878 1404 1466 1453">Description</th></tr> <tr> <td data-bbox="677 1453 878 1808">normal</td><td data-bbox="878 1453 1466 1808" rowspan="6">Treated as non-bold.</td></tr> <tr> <td data-bbox="677 1520 878 1556">lighter</td></tr> <tr> <td data-bbox="677 1583 878 1619">100</td></tr> <tr> <td data-bbox="677 1646 878 1682">200</td></tr> <tr> <td data-bbox="677 1709 878 1745">300</td></tr> <tr> <td data-bbox="677 1772 878 1808">400</td></tr> <tr> <td data-bbox="677 1808 878 1871">bold</td><td data-bbox="878 1808 1466 1871">Treated as bold.</td></tr> </table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.
Property	Description																									
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.																									
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.																									
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.																									
font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. 																									
font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps 																									
font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <tr> <th data-bbox="677 1404 878 1453">Value</th><th data-bbox="878 1404 1466 1453">Description</th></tr> <tr> <td data-bbox="677 1453 878 1808">normal</td><td data-bbox="878 1453 1466 1808" rowspan="6">Treated as non-bold.</td></tr> <tr> <td data-bbox="677 1520 878 1556">lighter</td></tr> <tr> <td data-bbox="677 1583 878 1619">100</td></tr> <tr> <td data-bbox="677 1646 878 1682">200</td></tr> <tr> <td data-bbox="677 1709 878 1745">300</td></tr> <tr> <td data-bbox="677 1772 878 1808">400</td></tr> <tr> <td data-bbox="677 1808 878 1871">bold</td><td data-bbox="878 1808 1466 1871">Treated as bold.</td></tr> </table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.														
Value	Description																									
normal	Treated as non-bold.																									
lighter																										
100																										
200																										
300																										
400																										
bold	Treated as bold.																									

Attributes	Description	
		<div> <div>bolder</div> <div>500</div> <div>600</div> <div>700</div> <div>800</div> <div>900</div> </div>
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.
	text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.

Attributes	Description	
	v-text-spacing-mode	Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are: <ul style="list-style-type: none">• tightening• tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties: <ul style="list-style-type: none">• top• left• width• height	
	The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute: <ul style="list-style-type: none">• flip• height• left• margin-left• margin-top• position• rotation• top• visibility• width• z-index	
	The possible values for this attribute are defined by the W3C XML Schema string datatype.	
trim (Text Path Trim Toggle)	Specifies whether extra space is removed above and below the text. If true, space reserved for ascenders and descenders is removed. Default is false. [Example: The shape path is duplicated as a second shape and overlaid on the textpath for illustrative purposes: <v:shape style=" width:100;height:100" path="m 0,500 c 250,0 750,0 1000,500 e	

Attributes	Description
	<pre> m 0,600 c 250,900 750,900 1000,600 e" fillcolor="yellow" strokecolor="maroon"> <v:path textpathok="true"/> <v:textpath on="true" fitshape="true" string="vml" trim="true"/> </v:shape> </pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
xscale (Text X-Scaling)	<p>Specifies whether a straight text path is used instead of the shape path. If true, the text runs along a path from left to right along the x value of the lower boundary of the shape. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_TextPath](#)) is located in §A.6.1. *end note*]

19.1.3 Simple Types

The following additional simple type information in the urn:schemas-microsoft-com:vml namespace is used for documents of a transitional conformance class.

19.1.3.1 ST_EditAs (Shape Grouping Types)

This simple type specifies the different meanings of a group of shapes.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
bullseye (Bullseye Diagram)	Specifies that the group represents a bulls-eye diagram.

Enumeration Value	Description
canvas (Shape Canvas)	Specifies that the group is a regular group and does not represent a diagram.
cycle (Cycle Diagram)	Specifies that the group represents a cycle diagram.
orgchart (Organization Chart Diagram)	Specifies that the group represents an organization chart.
radial (Radial Diagram)	Specifies that the group represents a radial diagram.
stacked (Pyramid Diagram)	Specifies that the group represents a pyramid diagram.
venn (Venn Diagram)	Specifies that the group represents a Venn diagram.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_EditAs](#)) is located in §A.6.1. *end note*]

19.1.3.2 ST_Ext (VML Extension Handling Behaviors)

This simple type specifies VML extension handling behaviors.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
backwardCompatible (Renderable)	Specifies that the VML entity may be rendered by ignoring the extension information. If edited, the extension information must be discarded.
edit (Editable)	Specifies that the VML entity may be safely rendered and edited without invalidating the extension information.
view (Not renderable)	Specifies that the VML entity is not be renderable without understanding the extension information. If the extension information cannot be understood, the downlevel image should be used to render the object.




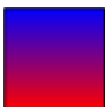
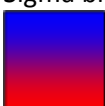
[Note: The W3C XML Schema definition of this simple type's content model ([ST_Ext](#)) is located in §A.6.1. *end note*]

19.1.3.3 ST_FillMethod (Gradient Fill Computation Type)

This simple type specifies ways in which a gradient fill is computed.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
any (Application Default Fill)	Default blend




Enumeration Value	Description
	
linear (Linear Fill)	Linear blend 
linear sigma (Linear Sigma Fill)	Linear sigma blend 
none (No Gradient Fill)	No blend 
sigma (Sigma Fill)	Sigma blend 




[Note: The W3C XML Schema definition of this simple type's content model ([ST_FillMethod](#)) is located in §A.6.1. end note]

19.1.3.4 ST_FillType (Shape Fill Type)

This simple type specifies the types for fills applied to a shape.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
frame (Stretch Image to Fit)	The image is stretched to fill the shape. 
gradient (Linear Gradient)	The fill colors blend together in a linear gradient from bottom to top. 
gradientRadial (Radial Gradient)	The fill colors blend together in a radial gradient. 
pattern (Image Pattern)	The image is used to create a pattern using the fill




Enumeration Value	Description
	colors. 
solid (Solid Fill)	The fill pattern is a solid color. 
tile (Tiled Image)	The fill image is tiled. 

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_FillType](#)) is located in §A.6.1. *end note*]

19.1.3.5 [ST_ImageAspect](#) (Image Scaling Behavior)

This simple type specifies the scaling behaviors for an image applied to a stroke.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
atLeast (At Least)	Image is at least as big as imagesize. 
atMost (At Most)	Image is no bigger than imagesize. 
ignore (Ignore Aspect Ratio)	Ignore aspect issues. 

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_ImageAspect](#)) is located in §A.6.1. *end note*]

19.1.3.6 [ST_ShadowType](#) (Shadow Type)

This simple type specifies the types of shadows applied to a shape.

This simple type's contents are a restriction of the W3C XML Schema string datatype.




Enumeration Value	Description
double (Double Shadow)	Double shadow. color2 and offset2 are used for the second shadow's color and offset.
emboss (Embossed Shadow)	The shadow has an embossed look. Similar to double.
perspective (Perspective Shadow)	Perspective shadow.
single (Single Shadow)	Single shadow.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ShadowType](#)) is located in §A.6.1. *end note*]

19.1.3.7 [ST_StrokeArrowLength \(Stroke Arrowhead Length\)](#)

This simple type specifies the lengths of a stroke arrowhead.

This simple type's contents are a restriction of the W3C XML Schema string datatype.




Enumeration Value	Description
long (Long Arrowhead)	Long length 
medium (Medium Arrowhead)	Medium length 
short (Short Arrowhead)	Short length 




[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowLength](#)) is located in §A.6.1. *end note*]

19.1.3.8 [ST_StrokeArrowType \(Stroke Arrowhead Type\)](#)

This simple type specifies the types of arrowhead for a stroke.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
block (Block Arrowhead)	Block arrowhead 
classic (Classic Arrowhead)	Classic curved arrowhead 
diamond (Diamond Arrowhead)	Diamond arrowhead 




Enumeration Value	Description
none (No Arrowhead)	No arrowhead 
open (Open Arrowhead)	Open arrowhead 
oval (Oval Arrowhead)	Round arrowhead 

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowType](#)) is located in §A.6.1. *end note*]

19.1.3.9 ST_StrokeArrowWidth (Stroke Arrowhead Width)

This simple type specifies the widths of a stroke arrowhead.

This simple type's contents are a restriction of the W3C XML Schema string datatype.


Enumeration Value	Description
medium (Medium Arrowhead)	Medium width 
narrow (Narrow Arrowhead)	Narrow width 
wide (Wide Arrowhead)	Wide width 



[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowWidth](#)) is located in §A.6.1. *end note*]

19.1.3.10 ST_StrokeEndCap (Stroke End Cap Type)

This simple type specifies the styles for the end of a stroke.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
flat (Flat End)	Flat end 
round (Round End)	Round end




Enumeration Value	Description
	
square (Square End)	Square end 

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeEndCap](#)) is located in §A.6.1. *end note*]

19.1.3.11 [ST_StrokeJoinStyle](#) (Line Join Type)

This simple type specifies the join styles for a polyline (§19.1.2.15).

This simple type's contents are a restriction of the W3C XML Schema string datatype.






Enumeration Value	Description
bevel (Bevel Joint)	Bevel joint 
miter (Miter Joint)	Miter joint 
round (Round Joint)	Round joint 

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeJoinStyle](#)) is located in §A.6.1. *end note*]

19.1.3.12 [ST_StrokeLineStyle](#) (Stroke Line Style)

This simple type specifies the line styles for a stroke.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
single (Single Line)	Single line 
thickBetweenThin (Thck Line Between Thin Lines)	Thick line between thin lines 
thickThin (Thick Line Outside Thin Line)	Thick line outside thin line 
thinThick (Thin Line Outside Thick Line)	Thin line outside thick line 
thinThin (Two Thin Lines)	Two thin lines 

[Note: The W3C XML Schema definition of this simple type’s content model ([ST StrokeLineStyle](#)) is located in §A.6.1. *end note*]

19.2 VML - Office Drawing

It is possible to include graphical VML objects in Office Open XML documents. The elements describing the core graphical objects are defined in the VML namespace. Additional elements that describe certain advanced shape effects, metadata and relationships are defined in this namespace.

[Note: The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a transitional format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML *end note*]

[Example: Assume the following shape exists in a document:



The basic speech bubble shape is defined using VML. The 3-D effect is defined using the extrusion element in this namespace. The specular attribute defines the subtle sharp reflection on the edge of the shape. The color attribute sets the extrusion to a different color than the face of the shape. The rotationangle attribute sets the shape's rotation about the X- and Y-axes. The lightposition and lightposition2 attributes set the positions of the light sources that illuminate the shape.

```
<o:extrusion v:ext="view" specularity="80000f" color="#c4bc96 [2414]" on="t"
rotationangle="-5,15" lightposition="0,-50000" lightposition2="0,50000"
type="perspective"/>
```

This element is a child of the primary shape definition:

```
<v:shape id="_x0000_s1030" type="#_x0000_t62"
style="position:absolute;left:0;text-align:left;margin-left:35.25pt;
margin-top:60pt;width:69pt;height:57pt;z-index:251658240" adj="1675,27171"
fillcolor="#ddd8c2 [2894]">
<o:extrusion ... />
</v:shape>
```

end example]

Throughout VML, numeric values that are allowed to take units can be specified in: cm (centimeters), mm (millimeters), in (inches), pt (points), pc (picas), px (pixels).

19.2.1 Table of Contents

This subclause is informative.

19.2.2	Elements	680
19.2.2.1	bottom (Text Box Bottom Stroke)	681
19.2.2.2	callout (Callout)	693
19.2.2.3	clippath (Shape Clipping Path).....	695
19.2.2.4	colormenu (UI Default Colors).....	698
19.2.2.5	colormru (Most Recently Used Colors)	699
19.2.2.6	column (Text Box Interior Stroke)	700
19.2.2.7	complex (Complex)	712
19.2.2.8	diagram (VML Diagram).....	712
19.2.2.9	entry (Regroup Entry)	716
19.2.2.10	equationxml (Storage for Alternate Math Content).....	717
19.2.2.11	extrusion (3D Extrusion)	718
19.2.2.12	FieldCodes (WordprocessingML Field Switches)	733
19.2.2.13	fill (Shape Fill Extended Properties)	733
19.2.2.14	idmap (Shape ID Map)	734
19.2.2.15	ink (Ink)	735
19.2.2.16	left (Text Box Left Stroke)	736
19.2.2.17	LinkType (Embedded Object Alternate Image Request)	748
19.2.2.18	lock (Shape Protections)	749

19.2.2.19	LockedField (Embedded Object Cannot Be Refreshed).....	750
19.2.2.20	OLEObject (Embedded OLE Object).....	751
19.2.2.21	proxy (Shape Reference)	753
19.2.2.22	r (Rule)	754
19.2.2.23	regrouptable (Shape Grouping History)	756
19.2.2.24	rel (Diagram Relationship).....	757
19.2.2.25	relationtable (Diagram Relationship Table).....	759
19.2.2.26	right (Text Box Right Stroke).....	759
19.2.2.27	rules (Rule Set).....	771
19.2.2.28	shapedefaults (New Shape Defaults)	772
19.2.2.29	shapelayout (Shape Layout Properties)	783
19.2.2.30	signatureline (Digital Signature Line)	784
19.2.2.31	skew (Skew Transform)	788
19.2.2.32	top (Text Box Top Stroke)	789
19.2.3	Simple Types	801
19.2.3.1	ST_AlternateMathContentType (Alternate Math Content Type)	801
19.2.3.2	ST_Angle (Callout Angles).....	801
19.2.3.3	ST_BWMode (Black And White Modes)	802
19.2.3.4	ST_CalloutDrop (Callout Drop Location)	802
19.2.3.5	ST_CalloutPlacement (Callout Placement)	802
19.2.3.6	ST_ColorMode (Extrusion Color Types).....	803
19.2.3.7	ST_ConnectorType (Connector Type).....	803
19.2.3.8	ST_ConnectType (Connection Locations Type)	804
19.2.3.9	ST_ContentType (Content Type)	804
19.2.3.10	ST_DiagramLayout (Diagram Layout Type)	804
19.2.3.11	ST_ExtrusionPlane (Extrusion Planes)	805
19.2.3.12	ST_ExtrusionRender (Extrusion Rendering Types)	805
19.2.3.13	ST_ExtrusionType (Extrusion Type)	806
19.2.3.14	ST_FillType (Shape Fill Type)	806
19.2.3.15	ST_How (Alignment Type)	807
19.2.3.16	ST_HrAlign (Alignment Type).....	808
19.2.3.17	ST_InsetMode (Inset Margin Type)	808
19.2.3.18	ST_OLEDrawAspect (Embedded Object Representations).....	808
19.2.3.19	ST_OLELinkType (Embedded Object Alternate Image Request Types).....	809
19.2.3.20	ST_OLEType (Embedded Connection Type)	809
19.2.3.21	ST_OLEUpdateMode (Embedded Object Update Method Type).....	809
19.2.3.22	ST_RType (Rule Type)	810
19.2.3.23	ST_ScreenSize (Screen Sizes Type)	810

End of informative text.

19.2.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:office:office namespace:

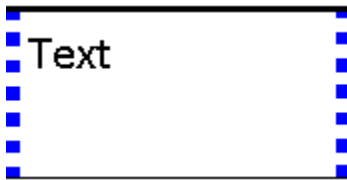
[*Note:* As the VML format is a format provided for backward compatibility, those VML elements defined in the same urn:schemas-microsoft-com:office:office namespace remain in that namespace as it is already used by millions of documents already using VML. *end note*]

19.2.2.1 bottom (Text Box Bottom Stroke)

This element specifies the stroke properties for the bottom border of a text box. It entirely supercedes its parent stroke element if its on attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the on attribute is false or not specified, the border is not shown.


[*Example:* The text box borders are set independently. Note that the bottom border does not inherit the weight from the parent stroke element.



```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```

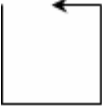

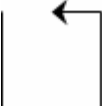



end example]



Attributes	Description
althref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p>[<i>Example:</i></p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
color (Stroke Color)	<p>Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors.</p> <p>[<i>Example:</i> The shape stroke is blue:</p>


Attributes	Description
	<pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>color2 (Stroke Alternate Pattern Color)</p>	<p>Specifies a second color for strokes, used when filltype is pattern. Default is no value.</p> <p>When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.</p> <p>[<i>Example:</i> This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre><v:background fillcolor="yellow"/> <v:shape style="width:60;height:50" strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape></pre> <div data-bbox="414 1241 594 1402" data-label="Image"> </div> <p>, where myimage.gif is: </p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>dashstyle (Stroke Dash Pattern)</p>	<p>Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p> <ul style="list-style-type: none"> • solid • shortdash • shortdot • shortdashdot • shortdashdotdot • dot


Attributes	Description
	<ul style="list-style-type: none"> • dash • longdash • dashdot • longdashdot • longdashdotdot <p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is fourfold symmetrical (with round end caps, this is a circle).</p> <p>[Example:</p> <pre><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre>  <pre><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
endarrow (Line End Arrowhead)	<p>Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open



Attributes	Description
	<p>[Example:</p> <pre><v:stroke endarrow="classic"/></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
<p>endarrowlength (Line End Arrowhead Length)</p>	<p>Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... endarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
<p>endarrowwidth (Line End Arrowhead Width)</p>	<p>Specifies the width of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... endarrowwidth="wide" ... /></pre> 


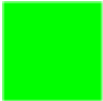
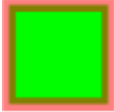
Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
<p>endcap (Line End Cap)</p>	<p>Specifies the cap style for the end of a stroke. Default is flat. Allowed values are:</p> <ul style="list-style-type: none"> • flat • square • round <p>[Example:</p> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p>endcap="flat"</p> <p>endcap="square"</p> <p>endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vml</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale]</i></p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • solid - The fill pattern is solid. • tile - The fill image is tiled. • pattern - The fill image is stretched to form a pattern.

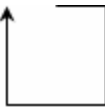
Attributes	Description
	<ul style="list-style-type: none"> • frame - The fill image becomes a border for the shape. <p>[Example:</p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre> <div style="display: flex; align-items: center; justify-content: center;">  , where border.gif is:  </div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).</p>
forcedash (Force Dashed Outline)	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
href (Original Image Reference)	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>


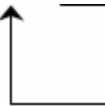
Attributes	Description								
imagealignshape (Stroke Image Alignment)	<p>Specifies the alignment of the stroke image. If true, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is true.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre>  <p style="text-align: center;">imagealignshape="false"</p> <p style="text-align: center;">imagealignshape="false"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table border="1" data-bbox="418 1354 1320 1549"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>ignore</td><td>Ignore aspect issues.</td></tr> <tr> <td>atleast</td><td>Image is at least as big as imagesize.</td></tr> <tr> <td>atmost</td><td>Image is no bigger than imagesize.</td></tr> </tbody> </table> <p>[Example:</p> <pre><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre>	Value	Description	ignore	Ignore aspect issues.	atleast	Image is at least as big as imagesize.	atmost	Image is no bigger than imagesize.
Value	Description								
ignore	Ignore aspect issues.								
atleast	Image is at least as big as imagesize.								
atmost	Image is no bigger than imagesize.								

Attributes	Description
	 <p><code>imagealignshape="ignore"</code></p> <p><code>imagealignshape="atleast"</code></p> <p><code>imagealignshape="atmost"</code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>
imagesize (Stroke Image Size)	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre><v:stroke ... imagesize="10pt,10pt" ... /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
joinstyle (Line End Join Style))	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none"> • round • bevel • miter <p>[Example:</p>

Attributes	Description
	<pre> <v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline> </pre>  <p>joinstyle="round"</p> <p>joinstyle="bevel"</p> <p>joinstyle="miter"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p>
linestyle (Stroke Line Style)	<p>Specifies the line style of the stroke. Default is single.</p> <ul style="list-style-type: none"> • single • thinThin • thinThick • thickThin • thickBetweenThin <p>[Example:</p> <pre> <v:stroke linestyle="thickThin" weight="5pt"> </v:stroke> </pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§19.1.3.12).</p>
miterlimit (Miter Joint Limit)	<p>Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p>

Attributes	Description
	<p>[Example:</p> <pre><v:stroke joinstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
on (Stroke Toggle)	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>  <p>end example]</p>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Stroke Image Location)	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre><v:stroke ... src="myimage.gif" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
startarrow (Line Start Arrowhead)	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke startarrow="classic"/></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p>

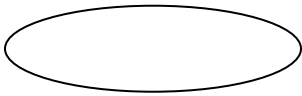
Attributes	Description
	<p><code><v:stroke ... startarrowlength="long" ... /></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <p><code><v:stroke ... startarrowwidth="wide" ... /></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
title (Stroke Title)	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <p><code><v:fill ... o:title="alt text" ... ></code> <code></v:fill></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
weight (Stroke Weight)	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_StrokeChild](#)) is located in §A.6.2.
end note]

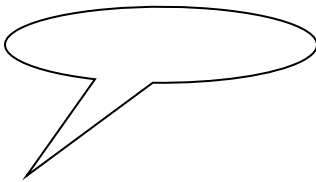
19.2.2.2 callout (Callout)

This element specifies the automatic behavior and layout parameters of callout shapes. Callout shapes are standard VML shapes that behave as callouts, providing an additional callout object which can be used to point at another location:

[Example: Consider the following VML shape:



If this shape is made a callout shape by adding the callout element to its shape definition, then the shape has a callout object, for example:



end example]

Attributes	Description
accentbar (Callout accent bar toggle)	Specifies whether an accent bar is used with the callout. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
angle (Callout angle)	Specifies the angle that the callout makes with respect to the bounding box of the shape. Default is no value. The possible values for this attribute are defined by the ST_Angle simple type (§19.2.3.2).
distance (Callout drop distance)	Specifies the drop distance of a callout. The drop distance of a callout is measured from the edge of the shape where the pointer line starts and continues the absolute length of the distance value. If specified with no units, EMUs are assumed. Default is no value. The possible values for this attribute are defined by the W3C XML Schema string datatype.
drop (Callout drop position)	Specifies where the drop of a callout is placed. The possible values for this attribute are defined by the ST_CalloutDrop simple type (§19.2.3.4).

Attributes	Description
dropauto (Callout automatic drop toggle)	<p>Specifies whether the callout has an automatic drop.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:VML	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
gap (Callout gap)	<p>Specifies the distance of the callout line from the bounding rectangle of the callout. Default value is one-twelfth of an inch, in EMUs (76200).</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
length (Callout length)	<p>Specifies the length of the first part of a multi-segmented callout line. If specified with no units, EMUs are assumed. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lengthspecified (Callout length toggle)	<p>Specifies whether the length attribute is used for the callout. Default is false. If true, the length attribute is used. If false, a best fit is used.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
minusx (Callout flip x)	<p>Specifies whether the callout flips to the other side of the drop tip along the x-axis when moved or resized. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
minusy (Callout flip y)	<p>Specifies whether the callout flips to the other side of the drop tip along the y-axis when moved or resized. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
on (Callout toggle)	<p>Specifies whether a shape is a callout. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
textborder (Callout text border toggle)	<p>Specifies whether a callout has a text border. Default is true.</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
type (Callout type)	<p>Specifies the type of callout. Default is rectangle. Allowed values are:</p> <ul style="list-style-type: none"> rectangle roundedrectangle oval cloud <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Callout](#)) is located in §A.6.2. *end note*]

19.2.2.3 [clippath \(Shape Clipping Path\)](#)

This element specifies the path of the clipping polygon for the shape.

[Example:

```
<v:rect ... wrapcoords="-207 -433 -207 21925 21807 21925 21807 -433 -207 -433"
o:clip="t" o:cliptowrap="t">
  <o:clippath o:v="m-207,-433r,22358121807,21925r,-223581-207,-433xe"/>
</v:rect>
```

end example]

Attributes	Description
v (Path Definition)	<p>Specifies a string containing the commands that define the shape's path. This value consists of commands followed by zero or more parameters. Default is no value.</p> <p>The following rules apply to path strings:</p> <ul style="list-style-type: none"> Commas or spaces delimit parameters for each command. Both "m 0,0" and "m0 0" are acceptable. A parameter that is omitted using commas is treated as having a value of zero. Thus, "c 10,10,0,0,25,13" and "c 10,10,,,25,13" are equivalent. Parameterized paths are also allowed. In this case, the shape shall also have a formulas element (§19.1.2.6) with a list of formulas that are substituted into the path using the @ symbol followed by the number of the formula. The adj property of the shape contains the input parameters for these formulas. For example, "moveto @1@4". The evaluations of the formulas are substituted into the appropriate positions. @ also serves as a delimiter.

Attributes	Description			
	The allowed commands are given below. An asterisk (*) indicates that the command is allowed to be repeated. For the qb command, the controlpoint parameter is also allowed to be repeated.			
	Command	Name	Parameters	Description
	m	moveto	2	Start a new sub-path at the given (x,y) coordinate.
	l	lineto	2*	Draw a line from the current point to the given (x,y) coordinate which becomes the new current point. Specifying a number of coordinate pairs forms a polyline.
	c	curveto	6*	Draw a cubic bézier curve from the current point to the coordinate given by the final two parameters. The control points are given by the first four parameters.
	x	close	0	Close the current sub-path by drawing a straight line from the current point to the original moveto point.
	e	end	0	End the current set of sub-paths. A given set of sub-paths (as delimited by end) is filled. Subsequent sets of sub-paths are filled independently and superimposed on existing ones.
	t	rmoveto	2*	Start a new sub-path at a coordinate relative to the current point, cp (cpx+x, cpy+y).
	r	rlineto	2*	Draw a line from the current point to the given relative coordinate (cpx+x, cpy+y).
	v	rcurveto	6*	Cubic bézier curve using the given coordinate relative to the current point.
	nf	nofill	0	The current set of sub-paths (delimited by e) is not filled.
	ns	nostroke	0	The current set of sub-paths (delimited by e) is not stroked.

Attributes	Description			
	ae	angleellipseto	6*	Draws a segment of an ellipse as described using these parameters. A straight line is drawn from the current point to the start point of the segment. The parameters are: center (x,y), size(w,h), start angle, end angle.
	al	angleellipse	6*	Same as angleellipseto except that there is an implied moveto the starting point of the segment.
	at	arcto	8*	A segment of the ellipse is drawn which starts at the angle defined by the start radius vector and ends at the angle defined by the end vector. A straight line is drawn from the current point to the start of the arc. The arc is always drawn in a counterclockwise direction. The parameters are: left, top, right, bottom, start(x,y), end(x,y). The first four values define the bounding box of an ellipse. The last four define two radial vectors.
	ar	arc	8*	Same as arcto except there is an implied moveto the start point of the arc.
	wa	clockwisearco	8*	Same as arcto but the arc is drawn in a clockwise direction.
	wr	clockwisearc	8*	Same as arc but the arc is drawn in a clockwise direction
	qx	ellipticalquadrantx	2*	A quarter ellipse is drawn from the current point to the given end point. The elliptical segment is initially tangential to a line parallel to the x-axis. (i.e. the segment starts out horizontal). The parameters are: end(x,y).
	qy	ellipticalquadranty	2*	Same as ellipticalquadrantx except that the elliptical segment is initially tangential to a line parallel to the y-axis (i.e. the segment starts out vertical).

Attributes	Description			
	qb	quadraticbezier	2+2*	Defines one or more quadratic bézier curves by means of control points and an end point. Intermediate (on-curve) points are obtained by interpolation between successive control points as in the OpenType font specification. The sub-path need not be started in which case the sub-path is closed. In this case the last point of the sub-path defines the start point of the quadratic bézier. The parameters are: controlpoint(x,y)*, end(x,y).
	The possible values for this attribute are defined by the W3C XML Schema string datatype.			

[Note: The W3C XML Schema definition of this element’s content model ([CT_ClipPath](#)) is located in §A.6.2. *end note*]

19.2.2.4 colormenu (UI Default Colors)

This element determines the default colors for different types of colors that can be applied to VML shapes.

[*Rationale*: An application can choose to retain default colors or the last color choices a user made and present those in parts of its user interface. *end rationale*]

[*Example*:

```
<o:shapedefaults ... >
  <o:colormenu v:ext="edit" fillcolor="none" extrusioncolor="#36f"/>
</o:shapedefaults>
```

end example]

Attributes	Description
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vml	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. [<i>Rationale</i> : This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>] The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).
extrusioncolor	The default color associated with the 3D extrusion of a VML shape. Default is "#000000".

Attributes	Description
(Default extrusion color)	The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
fillcolor (Default fill color)	The default color associated with the fill of a VML shape. Default is "#0000FF". The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
shadowcolor (Default shadow color)	The default color associated with the shadow of a VML shape. Default is "#80800C". The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).
strokecolor (Default stroke color)	The default color associated with the stroke of a VML shape. Default is "#FFFF00". The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_ColorMenu](#)) is located in §A.6.2. *end note*]

19.2.2.5 colormru (Most Recently Used Colors)

This element defines a list of up to eight colors which represent the colors most recently used by the user.

[*Rationale*: An application can choose to retain the last color choices a user made, regardless of where on VML shapes they are used, and present those in parts of its user interface. *end rationale*]

[*Example*:

```
<o:shapedefaults ... >
  <o:colormru v:ext="edit" colors="#a01aae,#456b69,#06f,#a1ae24,#d57811"/>
</o:shapedefaults>
```

end example]

Attributes	Description
colors (Recent colors)	A comma-separated list of up to eight most recently used colors. Default is no value. Colors should be defined using hexadecimal notation - see the ST_ColorType simple type (§20.1.2.3) for a full description. [<i>Example</i> : <pre><o:colormru v:ext="edit" colors="#a01aae,#456b69,#06f,#a1ae24,#d57811"/></pre> <i>end example</i>]

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vml	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. [<i>Rationale</i> : This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>] The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).

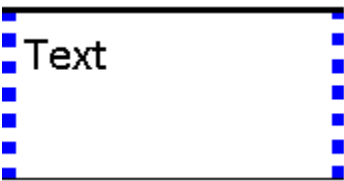
[*Note*: The W3C XML Schema definition of this element’s content model ([CT_ColorMru](#)) is located in §A.6.2. *end note*]

19.2.2.6 column (Text Box Interior Stroke)

This element specifies the stroke properties for the interior border of a text box. It entirely supercedes its parent stroke element if its on attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the on attribute is false or not specified, the border is not shown. [*Note*: This element is ignored if an implementation does not support multi-column text boxes. *end note*]

[*Example*: The text box borders are set independently. Note that the bottom border does not inherit the weight from the parent stroke element.

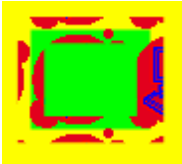


```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```


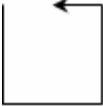



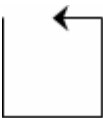

end example]


Attributes	Description
althref (Alternate Image Reference)	Specifies an alternate reference for an image in Macintosh PICT format.



Attributes	Description
	<p>[Example:</p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
color (Stroke Color)	<p>Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors.</p> <p>[Example: The shape stroke is blue:</p> <pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
color2 (Stroke Alternate Pattern Color)	<p>Specifies a second color for strokes, used when filltype is pattern. Default is no value.</p> <p>When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.</p> <p>[Example: This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre><v:background fillcolor="yellow"/> <v:shape style="width:60;height:50" strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape></pre>


Attributes	Description
	<div data-bbox="414 245 594 407"></div> , where myimage.gif is: <div data-bbox="873 310 972 407"></div> <p data-bbox="414 453 574 485"><i>end example]</i></p> <p data-bbox="414 525 1396 590">The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
dashstyle (Stroke Dash Pattern)	<p data-bbox="414 609 1477 640">Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p> <ul data-bbox="462 680 730 1041" style="list-style-type: none">• solid• shortdash• shortdot• shortdashdot• shortdashdotdot• dot• dash• longdash• dashdot• longdashdot• longdashdotdot <p data-bbox="414 1081 1455 1360">A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is fourfold symmetrical (with round end caps, this is a circle).</p> <p data-bbox="414 1400 535 1432"><i>[Example:</i></p> <div data-bbox="453 1472 1062 1570"><pre data-bbox="453 1472 1062 1570"><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre></div> <div data-bbox="414 1604 522 1719"></div> <div data-bbox="453 1755 1029 1854"><pre data-bbox="453 1755 1029 1854"><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre></div>


Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
endarrow (Line End Arrowhead)	<p>Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke endarrow="classic"/></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
endarrowlength (Line End Arrowhead Length)	<p>Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... endarrowlength="long" ... /></pre>  <p><i>end example]</i></p>




Attributes	Description
	<p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
endarrowwidth (Line End Arrowhead Width)	<p>Specifies the width of the arrowhead at the end of a line. Default is <code>medium</code>. Allowed values are:</p> <ul style="list-style-type: none">• <code>narrow</code>• <code>medium</code>• <code>wide</code> <p>[Example:</p> <pre><v:stroke ... endarrowwidth="wide" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
endcap (Line End Cap)	<p>Specifies the cap style for the end of a stroke. Default is <code>flat</code>. Allowed values are:</p> <ul style="list-style-type: none">• <code>flat</code>• <code>square</code>• <code>round</code> <p>[Example:</p> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p><code>endcap="flat"</code></p> <p><code>endcap="square"</code></p> <p><code>endcap="round"</code></p> <p><i>end example]</i></p>


Attributes	Description
	<p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm1</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • <code>solid</code> - The fill pattern is solid. • <code>tile</code> - The fill image is tiled. • <code>pattern</code> - The fill image is stretched to form a pattern. • <code>frame</code> - The fill image becomes a border for the shape. <p>[<i>Example</i>:</p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre> <div data-bbox="414 1266 513 1367" data-label="Image"> </div> <p>, where border.gif is: </p> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).</p>
<p>forcedash (Force Dashed Outline)</p>	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[<i>Example</i>:</p> <pre><v:shape ... o:forcedash="true" ... ></pre>


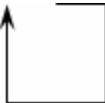

Attributes	Description
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
href (Original Image Reference)	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
imagealignshape (Stoke Image Alignment)	<p>Specifies the alignment of the stroke image. If true, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is true.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre> <div data-bbox="451 1360 609 1518">  </div> <p><code>imagealignshape="false"</code></p> <div data-bbox="451 1556 609 1713">  </div> <p><code>imagealignshape="false"</code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

Attributes	Description								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table data-bbox="418 352 1320 548"> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>ignore</td><td>Ignore aspect issues.</td></tr> <tr> <td>atleast</td><td>Image is at least as big as imagesize.</td></tr> <tr> <td>atmost</td><td>Image is no bigger than imagesize.</td></tr> </table> <p>[Example:</p> <pre data-bbox="456 657 1110 789"><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre> <div data-bbox="451 825 604 1171">  </div> <p>imagealignshape="ignore"</p> <p>imagealignshape="atleast"</p> <p>imagealignshape="atmost"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>	Value	Description	ignore	Ignore aspect issues.	atleast	Image is at least as big as imagesize.	atmost	Image is no bigger than imagesize.
Value	Description								
ignore	Ignore aspect issues.								
atleast	Image is at least as big as imagesize.								
atmost	Image is no bigger than imagesize.								
imagesize (Stroke Image Size)	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre data-bbox="456 1514 1065 1545"><v:stroke ... imagesize="10pt,10pt" ... /></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>								
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p>								

Attributes	Description
	<p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>joinstyle (Line End Join Style))</p>	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none"> • round • bevel • miter <p>[Example:</p> <pre><v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline></pre> <div style="display: flex; align-items: center; margin-top: 20px;">  <div style="margin-left: 20px;"> <p>joinstyle="round"</p> <p>joinstyle="bevel"</p> <p>joinstyle="miter"</p> </div> </div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p>
<p>linestyle (Stroke Line Style)</p>	<p>Specifies the line style of the stroke. Default is single.</p> <ul style="list-style-type: none"> • single • thinThin • thinThick • thickThin • thickBetweenThin

Attributes	Description
	<p>[Example:</p> <pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§19.1.3.12).</p>
<p>miterlimit (Miter Joint Limit)</p>	<p>Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p> <p>[Example:</p> <pre><v:stroke joinstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
<p>on (Stroke Toggle)</p>	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p>end example]</p>

Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>opacity (Stroke Opacity)</p>	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>src (Stroke Image Location)</p>	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre><v:stroke ... src="myimage.gif" ... > </v:stroke></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>startarrow (Line Start Arrowhead)</p>	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke startarrow="classic"/></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... startarrowlength="long" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... startarrowwidth="wide" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>

Attributes	Description
title (Stroke Title)	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
weight (Stroke Weight)	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_StrokeChild](#)) is located in §A.6.2. end note]

19.2.2.7 complex (Complex)

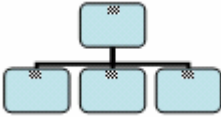
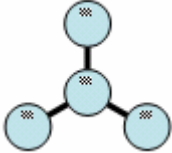
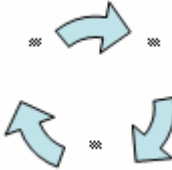


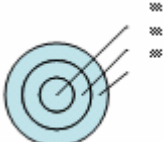
This element specifies that a shapetype contains fragments.

Attributes	Description
ext (VML Extension Handling Behavior)	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p>
Namespace: urn:schemas-microsoft-com:vml	<p>[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end rationale]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Complex](#)) is located in §A.6.2. end note]

19.2.2.8 diagram (VML Diagram)

This element specifies semantic information for a limited set of structured diagrams that have VML representations. Diagrams should be defined using DrawingML; this representation is included for compatibility with applications that rely on VML. The following diagram types have VML representations:

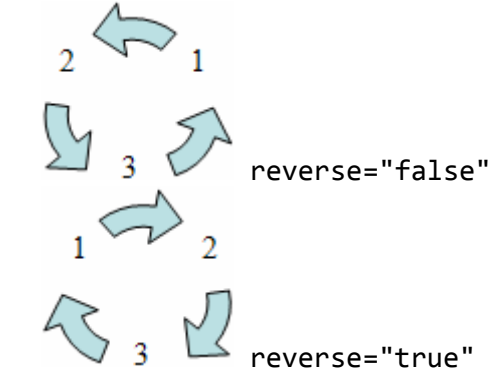
Diagram Type	Example (non-normative)
Organization chart	
Radial	
Cycle	
Pyramid	
Venn	
Bulls-eye	

Each of these types of diagrams contains shapes that are positioned relative to one another. Each shape also has optional associated text.

Attributes	Description
autoformat (Diagram Automatic Format)	<p>Specifies whether the diagram is formatted automatically by the application and user overrides are locked. Default is false.</p> <p>[Example:</p> <pre><o:diagram ... autoformat="true"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

Attributes	Description
autolayout (Diagram Automatic Layout)	<p>Specifies whether the diagram elements are laid out automatically by the application and user overrides are locked. Default is true.</p> <p>[Example:</p> <pre><o:diagram ... autolayout="false"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
constrainbounds (Diagram Layout Extents)	<p>Specifies an optional, application-specific parameter related to the diagram's extents intended to be used by the application to assist laying out the diagram.</p> <p>[Example:</p> <pre><o:diagram ... constrainbounds="2910,2696,9773,9558"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
dgmbasetextscale (Diagram Base Font Size)	<p>Specifies the diagram's original font size. This is used in subsequent font size recalculations. If the most recent diagram font size is used to calculate the font size after a rescale, the font size would be wrong after non-isometric diagram rescalings.</p> <p>[Example:</p> <pre><o:diagram ... dgmbasetextscale="12"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
dgmfontsize (Diagram Font Size)	<p>Specifies the font size for attached text when a new diagram node is added.</p> <p>[Example:</p> <pre><o:diagram ... dgmfontsize="12"> </o:diagram></pre> <p>end example]</p>

Attributes	Description
	<p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>dgmscalex (Diagram Layout X Scale)</p>	<p>Specifies an optional, application-specific parameter related to the horizontal scaling of the diagram that is intended to be used by the application to assist laying out the diagram.</p> <p>[Example:</p> <pre><o:diagram ... dgmscalex="50000"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>dgmscaley (Diagram Layout Y Scale)</p>	<p>Specifies an optional, application-specific parameter related to the vertical scaling of the diagram that is intended to be used by the application to assist laying out the diagram.</p> <p>[Example:</p> <pre><o:diagram ... dgmscaley="75000"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>dgmstyle (Diagram Style Options)</p>	<p>Specifies an optional, application-specific parameter related to the styling of the diagram that is intended to be used by the application to assist in formatting the diagram.</p> <p>[Example:</p> <pre><o:diagram ... dgmstyle="1"> </o:diagram></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end</p>

Attributes	Description
microsoft-com:uml	<i>rationale</i> The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).
reverse (Diagram Reverse Direction)	<p>Specifies whether the order of the diagram nodes is reversed. This is only relevant to diagrams that have linear ordering.</p> <p>[Example:</p> <pre><o:diagram ... reverse="true"> </o:diagram></pre>  <p>end example</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>

[Note: The W3C XML Schema definition of this element’s content model ([CT_Diagram](#)) is located in §A.6.2. end note]

19.2.2.9 entry (Regroup Entry)

This element specifies a single entry in a regrouptable (§19.2.2.23). Each entry is a pair mapping a current regroupid value to an old one. This is used to restore regrouping information on the regrouped object. A value of zero indicates no previous group.

[Example: The zero value of the old attribute indicates that if the shapes with regroupid 1 are regrouped, the restored group was not previously grouped with any other shapes:

```
<o:regrouptable v:ext="edit">
  <o:entry new="1" old="0"/>
</o:regrouptable>
```

end example]

Attributes	Description
new (New Group ID)	Specifies the ID of the new group. Default is 0. The possible values for this attribute are defined by the W3C XML Schema int datatype.
old (Old Group ID)	Specifies the ID of the old group. Default is 0. The possible values for this attribute are defined by the W3C XML Schema int datatype.

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Entry](#)) is located in §A.6.2. *end note*]

19.2.2.10 `equationxml` (Storage for Alternate Math Content)

This element specifies XML markup for mathematical text which can be used in place of the shape data. [*Note:* Applications are encouraged to use an open format, such as the Math format defined in ECMA-376-1, or the MathML format, a Recommendation from the World Wide Web Consortium, available at <http://www.w3.org/TR/MathML/>. *end note*]

[*Example:* Consider a VML object which specifies alternate math content using MathML. This object might contain the following XML markup:

```
<v:shape>
...
<o:equationXml contentType="mathml">
  <mrow>
    <mrow>
      <msup>
        <mi>x</mi>
        <mn>2</mn>
      </msup>
      <mo>+</mo>
    </mrow>
    <mrow>
      <mn>4</mn>
      <mo>*</mo>
      <mi>x</mi>
    </mrow>
    <mo>+</mo>
    <mn>4</mn>
  </mrow>
  <mo>=</mo>
  <mn>0</mn>
</mrow>
</o:equationXml>
</v:shape>
```

The embedded MathML markup is stored within the `equationxml` element. *end example*

If a producer that wants interoperability supports equations, it should use one of the following standard formats:

- Office Open XML Math (Part 1, §22.1
- W3C MathML 2.0

Attributes	Description
contentType (Content Type of Alternate Math Content)	Specifies the syntax of the markup used for the alternate math content stored in the <code>equationxml</code> attribute. The possible values for this attribute are defined by the <code>ST_AlternateMathContentType</code> simple type (§19.2.3.1).

[Note: The W3C XML Schema definition of this element’s content model ([CT_EquationXml](#)) is located in §A.6.2. *end note*]

19.2.2.11 [extrusion \(3D Extrusion\)](#)

This element specifies a parallel or perspective extrusion of a 2-D shape, creating the appearance of a 3-D shape. Lighting is controlled via two independent point light sources. Extrusions are defined as either perspective or parallel.



[Example:



```
<v:polyline points="0pt,75pt 20pt,45pt 10pt,50pt 30pt,10pt
50pt,50pt 40pt,45pt 60pt,75pt 0pt,75pt" fillcolor="#00a000">
  <o:extrusion on="t" backdepth="20pt"
    lightposition="30000,10000,10000"/>
</v:polyline>
```









end example]




Attributes	Description
autorotationcenter (Center of Rotation Toggle)	Specifies whether the center of rotation is the geometric center of the extrusion. Default is <code>false</code> . If <code>true</code> , the geometric center of an extruded shape is (0,0,0). If <code>false</code> , the center of rotation is determined by the <code>rotationcenter</code> attribute.





Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>backdepth (Backward Extrusion Depth)</p>	<p>Specifies the amount of backward extrusion. Default is 36 pt, default units are points.</p> <p>[Example:</p> <pre data-bbox="451 474 1097 541"><o:extrusion on="true" backdepth="15pt"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>brightness (Brightness)</p>	<p>Specifies the overall brightness of a scene. Default is 0.3. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied (as "f" indicates the value is a fraction). For example, a value of "52429f" represents 52429/65536 or 0.8.</p> <p>This quantity is not specified using units. The numeric values are in the range 0 to 1 (0f to 65536f), where 0 implies darkness and 1 implies light saturation.</p> <p>[Example:</p> <pre data-bbox="451 1205 1097 1272"><o:extrusion on="true" brightness="0.4"> </o:extrusion></pre>  <p>brightness="0"</p> <p>brightness="25000f"</p> <p>brightness="0.4"</p> <p>brightness="0.75"</p>


Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>color (Extrusion Color)</p>	<p>Specifies the color of the extrusion faces. This attribute is only used when colormode is custom. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre data-bbox="451 688 1015 787"><o:extrusion on="true" color="lime" colormode="custom"> </o:extrusion></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
<p>colormode (Extrusion Color Mode)</p>	<p>Specifies whether the extrusion color is defined by the color attribute or is the same as the shape's fill color. Default is auto.</p> <p>[Example:</p> <pre data-bbox="451 1333 1015 1432"><o:extrusion on="true" color="lime" colormode="auto"> </o:extrusion></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorMode simple type (§19.2.3.6).</p>
<p>diffusivity (Diffuse Reflection)</p>	<p>Specifies the amount of diffusion of reflected light from an extruded shape, defined as the ratio of incident light to diffused reflected light. Default is 1. Normal values are in the range 0 to 1. This numeric value can also be specified in 1/65536-ths if a trailing "f" is</p>

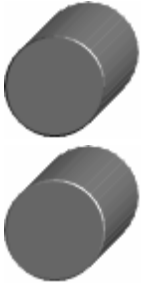
Attributes	Description
	<p>supplied (as "f" indicates the value is a fraction). <i>[Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p>This quantity is not specified using units. The numeric values are in the range 0 to 1 (0f to 65536f), where 0 implies all reflected light is diffuse and 1 implies no reflected light is diffuse.</p> <p>Specularity and diffusity should be considered together as it is possible, though physically incorrect, to define more reflected light than incident light. This is the case if the amount of specularly reflected light and diffusely reflected light add up to more than the amount of incident light.</p> <p><i>[Example:</i></p> <pre><o:extrusion on="true" diffusity=".75"> </o:extrusion></pre> <div data-bbox="456 848 591 982">  </div> <p>diffusity="0"</p> <div data-bbox="456 995 591 1129">  </div> <p>diffusity="0.5"</p> <div data-bbox="456 1142 591 1276">  </div> <p>diffusity="0.75"</p> <div data-bbox="456 1289 591 1423">  </div> <p>diffusity="1"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
edge (Simulated Bevel)	<p>Specifies the apparent bevel of the extrusion edges. Default is 1 point.</p> <p><i>[Example:</i></p> <pre><o:extrusion on="true" edge="2pt"> </o:extrusion></pre>


Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale]</i></p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>facet (Faceting Quality)</p>	<p>Specifies the quality with which the application approximates curved surfaces of an extrusion. A higher facet value produces shapes with smoother curves. A lower value reduces smoothing, resulting in curves with sharper, jagged edges. Default is 30000.</p> <p>Allowed values are in the range 1 to 65536, where 1 implies extremely low quality curve approximation and 65536 implies extremely high quality.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 1192 1047 1255"><o:extrusion on="true" facet="65536"> </o:extrusion></pre> <div data-bbox="451 1293 914 1583">  </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>foredepth (Forward Extrusion)</p>	<p>Specifies the amount of forward extrusion. Default is 0 pt, default units are points.</p> <p>[<i>Example</i>:</p>


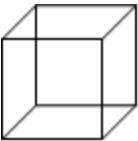
Attributes	Description
	<pre data-bbox="451 285 1097 348"><o:extrusion on="true" foredepth="25pt"> </o:extrusion></pre>  <p data-bbox="415 583 574 615"><i>end example]</i></p> <p data-bbox="415 653 1377 716">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lightface (Shape Face Lighting Toggle)	<p data-bbox="415 741 1451 804">Specifies whether the front face of the extrusion responds to changes in the lighting. If false, the front face does not respond when a lighting value changes. Default is true.</p> <p data-bbox="415 846 1484 909"><i>[Example: The front face is colored as if the shape were not extruded and lit by a 3-D light source:</i></p> <pre data-bbox="451 951 1114 1014"><o:extrusion on="true" lightface="false"> </o:extrusion></pre>  <p data-bbox="415 1228 574 1260"><i>end example]</i></p> <p data-bbox="415 1297 1390 1360">The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
lightharsh (Primary Light Harshness Toggle)	<p data-bbox="415 1383 1425 1446">Specifies whether the primary light source is harsh. If false, shadow boundaries are diffused. Default is true.</p> <p data-bbox="415 1488 1414 1520"><i>[Example: The secondary light source is turned off so only the primary has an effect:</i></p> <pre data-bbox="451 1562 1110 1659"><o:extrusion on="true" lightharsh="false" lightlevel2="0"> </o:extrusion></pre>  <p data-bbox="607 1803 893 1835">lightharsh="false"</p>


Attributes	Description
	<div><code>lightharsh="true"</code></div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
lightharsh2 (Secondary Light Harshness Toggle)	<p>Specifies whether the secondary light source is harsh. If false, shadow boundaries defined by the secondary light source are diffused. Default is false.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
lightlevel (Primary Light Intensity)	<p>Specifies the intensity of the primary light source for the scene. Default is 0.6. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied (as "f" indicates the value is a fraction). <i>[Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p>This quantity is not specified using units. The numeric values are in the range 0 to 1 (0f to 65536f), where 0 implies no direct light and 1 implies saturated direct light.</p> <p><i>[Example: The secondary light source is turned off so only the primary has an effect:</i></p> <div><pre><o:extrusion on="true" lightlevel=".5" lightlevel2="0"> </o:extrusion></pre></div> <div><div><code>lightlevel="1"</code></div><div><code>lightlevel="0.5"</code></div><div><code>lightlevel="0"</code></div></div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>



Attributes	Description
lightlevel2 (Secondary Light Intensity)	<p>Specifies the intensity of the secondary light source for the scene. Default is 0.6. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied (as "f" indicates the value is a fraction). <i>[Example: A value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p>This quantity is not specified using units. The numeric values are in the range 0 to 1 (0f to 65536f), where 0 implies no direct light and 1 implies saturated direct light.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lightposition (Primary Light Position)	<p>Specifies the normalized X,Y,Z position of the primary light in a scene in 1/65536-ths. Default is "50000,0,10000". The use of a normalized vector from the shape origin effectively establishes the direction of the light relative to the shape. The distance of the light from the shape is irrelevant as the light source is treated as a directional light.</p> <p>The position "0,0,0" is at the center of the shape. Positive numbers move the light to the right, down and toward the viewer, respectively.</p> <p><i>[Example: The secondary light source is turned off so only the primary has an effect:</i></p> <pre><o:extrusion on="true" lightlevel2="0" lightposition="7000,-13000,20000"> </o:extrusion></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lightposition2 (Secondary Light Position)	<p>Specifies the X,Y,Z position of the secondary light in a scene in 1/65536-ths. Default is "-50000,0,10000". The use of a normalized vector from the shape origin effectively establishes the direction of the light relative to the shape. The distance of the light from the shape is irrelevant as the light source is treated as a directional light.</p> <p>The position "0,0,0" is at the center of the shape. Positive numbers move the light to the right, down and toward the viewer, respectively.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lockrotationcenter (Rotation Toggle)	<p>Specifies whether the rotation of the extruded object is specified by the rotationangle attribute. If false, the rotation is specified by the orientation attribute. Default is true.</p>

Attributes	Description
	<p>[Example: The following snippets are equivalent:</p> <pre><o:extrusion lockrotationcenter="false" orientationangle="45" orientation="0,1,0"> </o:extrusion></pre> <pre><o:extrusion lockrotationcenter=true rotationangle="45"/> </o:extrusion></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>metal (Metallic Surface Toggle)</p>	<p>Specifies whether the surface of the extruded shape resembles metal. Default is false.</p> <p>If true, this attribute causes the specularly reflected light to be the material color instead of the light source color, making the object seem more metallic. To further approximate a metallic material requires that specularity be relatively high (about 1.2) and diffusivity be relatively low (about 0.6).</p> <p>[Example:</p> <pre><o:extrusion on="true" metal="true" lightposition="10000,-10000,10000" lightlevel2="0" specularity="1.2" diffusivity="0.6"> </o:extrusion></pre> <div data-bbox="451 1310 812 1596">  <p>metal="true"</p> <p>metal="false"</p> </div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>on (Extrusion Toggle)</p>	<p>Specifies whether an extrusion is displayed. Default is false.</p> <p>[Example:</p>

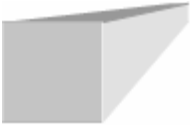

Attributes	Description
	<pre><v:rect style="width=50;height=50"> <o:extrusion /> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
orientation (Rotation Axis)	<p>Specifies a vector in 3D space around which the shape is rotated, as given by the orientationangle attribute. Default is "100,0,0".</p> <p>The position "0,0,0" is at the center of the shape. Positive numbers are to the right, down and toward the viewer, respectively.</p> <p>[Example:</p> <pre><o:extrusion ... orientation="200,0,0"> </o:extrusion></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
orientationangle (Rotation Around Axis)	<p>Specifies the angle, in degrees, that an extrusion rotates around the orientation. Default is 0.</p> <p>[Example:</p> <pre><o:extrusion ... orientationangle="30"> </o:extrusion></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
plane (Extrusion Direction)	<p>Specifies the plane that is at right angles to the extrusion. Default is xy. Allowed values are:</p> <ul style="list-style-type: none"> • xy • zx

Attributes	Description								
	<ul style="list-style-type: none">yz <p>[Example:</p> <pre><o:extrusion on="true" plane="yz" backdepth="100pt"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ExtrusionPlane simple type (§19.2.3.11).</p>								
render (Extrusion Render Mode)	<p>Specifies the rendering mode of the extrusion. Default is <code>solid</code>. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td><code>solid</code></td><td>Rendering displays a solid shape.</td></tr><tr><td><code>wireframe</code></td><td>Rendering displays a wireframe shape.</td></tr><tr><td><code>boundingcube</code></td><td>Rendering displays the bounding cube that contains the shape.</td></tr></table> <p>[Example:</p> <pre><o:extrusion on="true" render="wireframe"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ExtrusionRender simple type (§19.2.3.12).</p>	Value	Description	<code>solid</code>	Rendering displays a solid shape.	<code>wireframe</code>	Rendering displays a wireframe shape.	<code>boundingcube</code>	Rendering displays the bounding cube that contains the shape.
Value	Description								
<code>solid</code>	Rendering displays a solid shape.								
<code>wireframe</code>	Rendering displays a wireframe shape.								
<code>boundingcube</code>	Rendering displays the bounding cube that contains the shape.								
rotationangle (X-Y Rotation Angle)	<p>Specifies the rotation of the object about the x- and y-axes, in degrees. Default is "0,0". Positive angles are measured clockwise around the axis (as if viewing from the positive axis).</p>								

Attributes	Description
	<p>The rotation of the object is defined by a rotation angle about the y-axis followed by the rotation angle about the x-axis. The z-axis angle is controlled by the value of the CSS style attribute's rotation property.</p> <p>[Example:</p> <pre data-bbox="451 499 1172 596"><o:extrusion on="t" lockrotationcenter="true" rotationangle="10,20"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>rotationcenter (Rotation Center)</p>	<p>Specifies the center of rotation for a shape if autorotationcenter is false. The offset of the rotation is specified in terms of fractions of the shape's size. Default is "0,0,0".</p> <p>The position "0,0,0" is at the center of the shape. Positive numbers are to the right, down and toward the viewer, respectively.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>shininess (Shininess)</p>	<p>Specifies the concentration of the reflected light on an extrusion surface. Default is 5. The range of values should be constrained to 0-10. Reflection intensity typically grows exponentially with the shininess value.</p> <p>High values (8-10) approximate the shininess of a mirror and low values (2-3) approximate a speckled effect. Reflections do not mirror other objects; only pinpoint light sources are reflected.</p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
<p>skewamt (Extrusion Skew)</p>	<p>Specifies the amount of skew, or length, of a parallel extrusion. Default is 50%. Applies only if the extrusion type is parallel. This attribute and backdepth interact to create the actual extrusion length. Allowed values are in the range 0 (0%) to 1 (100%).</p> <p>[Example:</p> <pre data-bbox="451 1801 1062 1864"><o:extrusion on="true" skewamt="100%"> </o:extrusion></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
skewangle (Extrusion Skew Angle)	<p>Specifies the angle of the skew of a parallel extrusion. Default is 225 degrees. Angles are measured in degrees, counterclockwise from the negative x-axis. Applies only if the extrusion type is parallel.</p> <p>[Example:</p> <pre data-bbox="451 821 1062 888"><o:extrusion on="true" skewangle="25"> </o:extrusion></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
specularity (Specularity)	<p>Specifies the specularity of an extruded shape, defined as the ratio of incident light to specularly reflected light. Default is 0. Normal values are in the range 0 to 1. This numeric value can also be specified in 1/65536-ths if a trailing "f" is supplied (as "f" indicates the value is a fraction). [Example: A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>Specularity and diffusity should be considered together as it is possible, though physically incorrect, to define more reflected light than incident light. This is the case if the amount of specularly reflected light and diffusely reflected light add up to more than the amount of incident light.</p> <p>[Example: The secondary light source is turned off so only the primary has an effect. Although the effect is subtle, the first cylinder has a sharper specular reflection on its edge:</p> <pre data-bbox="451 1738 1062 1869"><o:extrusion on="true" specularity="1" lightposition="10000,-10000,10000" lightlevel2="0"> </o:extrusion></pre>

Attributes	Description						
	<div data-bbox="451 279 589 422" data-label="Image"> </div> <div data-bbox="607 390 846 426" data-label="Text"> <p>specularity="1"</p> </div> <div data-bbox="451 426 589 569" data-label="Image"> </div> <div data-bbox="607 537 846 573" data-label="Text"> <p>specularity="0"</p> </div> <p data-bbox="414 606 574 642"><i>end example]</i></p> <p data-bbox="414 678 1375 745">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>						
<p data-bbox="139 762 326 829">type (Extrusion Type)</p>	<p data-bbox="414 762 1446 793">Specifies the way that the shape is extruded. Default is parallel. Allowed values are:</p> <table data-bbox="414 829 1260 1155"> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>parallel</td><td>Extrusion is rendered so that the center of projection is infinitely far away; that is, the extrusion lines do not converge (unlike perspective projections).</td></tr> <tr> <td>perspective</td><td>Extrusion is rendered to a center of projection, which is the same as the vanishing point for unrotated objects.</td></tr> </table> <p data-bbox="414 1192 534 1224"><i>[Example:</i></p> <pre data-bbox="451 1262 1062 1360"> <o:extrusion on="true" type="parallel" backdepth="100pt"> </o:extrusion> </pre> <div data-bbox="451 1398 647 1591" data-label="Image"> </div> <div data-bbox="664 1566 902 1602" data-label="Text"> <p>type="parallel"</p> </div> <div data-bbox="451 1602 589 1734" data-label="Image"> </div> <div data-bbox="589 1707 875 1743" data-label="Text"> <p>type="perspective"</p> </div> <p data-bbox="414 1780 574 1816"><i>end example]</i></p> <p data-bbox="414 1852 1446 1883">The possible values for this attribute are defined by the ST_ExtrusionType simple type</p>	Value	Description	parallel	Extrusion is rendered so that the center of projection is infinitely far away; that is, the extrusion lines do not converge (unlike perspective projections).	perspective	Extrusion is rendered to a center of projection, which is the same as the vanishing point for unrotated objects.
Value	Description						
parallel	Extrusion is rendered so that the center of projection is infinitely far away; that is, the extrusion lines do not converge (unlike perspective projections).						
perspective	Extrusion is rendered to a center of projection, which is the same as the vanishing point for unrotated objects.						

Attributes	Description
viewpoint (Extrusion Viewpoint)	<p>(§19.2.3.13).</p> <p>Specifies the viewpoint of the observer in EMUs. This is effectively the end of a vector extending from the viewpointorigin.</p> <p>The position "0,0,0" is at the center of the shape. Positive numbers are to the right, down and toward the viewer, respectively.</p> <p>[Example:</p> <pre><o:extrusion on="true" type="perspective" viewpoint="500000,-100000,100000"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
viewpointorigin (Extrusion Viewpoint Origin)	<p>Specifies the origin of the viewpoint vector for perspective extrusions. This is the origin of the vector whose opposite end is given by the viewpoint attribute. This origin is always within the bounding box of the shape. Default is "0.5,-0.5".</p> <p>The viewpoint is specified in terms of the x and y values of the original shape. The x and y values are in the range 0.5 to -0.5 (50% to -50% of the shape's coordinate origin). Larger numbers move the viewpoint outside the bounding box.</p> <p>[Example:</p> <pre><o:extrusion on="true" type="perspective" viewpoint="500000,-100000,100000" viewpointorigin="0,1"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Extrusion](#)) is located in §A.6.2. *end note*]

19.2.2.12 FieldCodes (WordprocessingML Field Switches)

This element specifies the WordprocessingML field switches which shall be stored with an embedded object, using the set of field switches defined by the LINK field, as specified in Part 1, §17.16. This element shall only be used within a WordprocessingML document, and shall specify the exact field switches for the field which represents the object..

[*Rationale:* Legacy word processors used fields to represent embedded objects – this element stores the field switches not explicitly defined using individual Office VML Drawing elements for embeddings so as not to use the fidelity of their contents. *end rationale*]

[*Example:* The following example inserts an embedded object and specifies additional properties as defined by the LINK field.

```
<o:OLEObject ...>
  <o:FieldCodes>\f 0</o:FieldCodes>
</o:OLEObject>
```

This embedded object specifies additional LINK field code values of \f 0, which specifies that the embedded object shall retain its source formatting (as defined in Part 1, §17.16).

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.2.2.13 fill (Shape Fill Extended Properties)

This element specifies additional properties for fills. It is used to identify additional types of gradient fills beyond those specified in the fill element (§19.1.2.5).

Attributes	Description
ext (VML Extension Handling Behavior)	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.
Namespace: urn:schemas-microsoft-com:vml	<p>[<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
type (Fill Type)	<p>Specifies the type of fill. If specified, this overrides the value of the type attribute in the parent fill element.</p> <p>[<i>Example:</i> The gradientCenter value overrides gradientRadial:</p>

Attributes	Description
	<pre><v:fill color2="black" focus="100%" type="gradientRadial"> <o:fill v:ext="view" type="gradientCenter"/> </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§19.2.3.14).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Fill](#)) is located in §A.6.2. *end note*]

19.2.2.14 idmap (Shape ID Map)

This element specifies how shape IDs in the document have been generated. This is an optional element included to allow applications a mechanism for storing information they need to persist related to generating shape IDs.

Attributes	Description
data (Shape IDs)	<p>Specifies the data the application uses to generate shape IDs.</p> <p>[Example: An application might choose to reserve blocks of shape ID numbers for each part in the package. Each block of 1024 shape IDs could be referred to by index and this index stored in the data attribute. The data value for a given part might then be:</p> <pre><o:idmap v:ext="edit" data="1"/></pre> <p>indicating that all the IDs in block 1 are reserved by this part (meaning shape IDs from 1 to 1024 cannot be used). The application's internal constraint would be that each part reserve a different set of IDs. Another part, that contains more shapes, might use:</p> <pre><o:idmap v:ext="edit" data="2,3"/></pre> <p>In this case, shape IDs from 1025 to 3072 [3 x 1024] cannot be used).</p> <p>Another implementation might choose to store more verbose information in this attribute. Yet another implementation might ignore this element completely.</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
ext (VML Extension Handling Behavior)	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.
Namespace:	[Rationale: This part of the original VML specification is included to assist applications

Attributes	Description
urn:schemas-microsoft-com:VML	that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i> The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).

[Note: The W3C XML Schema definition of this element's content model ([CT_IdMap](#)) is located in §A.6.2. *end note*]

19.2.2.15 **ink (Ink)**

This element specifies the presence of an ink object. An ink object is a VML object which allows applications to store data for ink annotations. [Note: Applications are encouraged to use an open-ink format, such as the Ink Markup Language (InkML). *end note*]

[Example:

```
<v:shape ... >
  <o:ink i="..." annotation="t" contentType="application/inkml+xml"/>
</v:shape>
<v:shape ... >
  <o:ink i="AMgFHQSWC+YFASAAaAwAAAAAAMA..." annotation="t"
    contentType="application/x-ms-ink"/>
</v:shape>
```

end example]

Attributes	Description
annotation (Annotation Flag)	Specifies whether the ink object was created as an annotation rather than through pen input. Default is false. [<i>Rationale</i> This allows an application to treat annotation ink objects as any other annotation. For example, if annotations are hidden, the application can hide the ink object. An ink object that represents primary user input through a pen can be left visible. <i>end rationale</i>] [Example: <pre><o:ink ... annotation="true"> </o:ink></pre> <i>end example</i>] The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
contentType (Content Type)	Specifies the format of the ink content stored in the i attribute. The syntax is a content type as defined in IETF RFC 2616.

Attributes	Description
	<p>If this attribute is omitted, the application should attempt to determine the content type by reading the contents of the <code>i</code> element.</p> <p>The possible values for this attribute are defined by the <code>ST_ContentType</code> simple type (§19.2.3.9).</p>
i (Ink Data)	<p>Specifies additional ink object information which shall be associated with the parent VML shape. The VML shape specifies the information necessary to render the ink, and this attribute can be used to store additional data about the VML shape(s) representing ink. This attribute's contents are formatted as specified by the <code>contentType</code> attribute, but are optional and can be ignored if not recognized.</p> <p>[Example:</p> <pre data-bbox="451 758 695 821"><o:ink ... i="..."> </o:ink></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

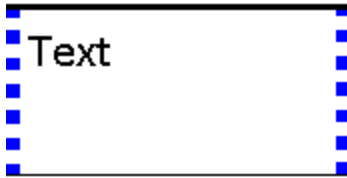
[Note: The W3C XML Schema definition of this element's content model ([CT_Ink](#)) is located in §A.6.2. end note]

19.2.2.16 left (Text Box Left Stroke)

This element specifies the stroke properties for the left border of a text box. It entirely supercedes its parent stroke element if its `on` attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the `on` attribute is `false` or not specified, the border is not shown.

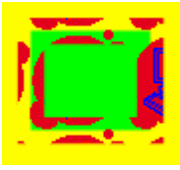

[Example: The text box borders are set independently. The bottom border does not inherit the weight from the parent stroke element.



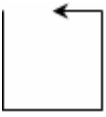
```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```


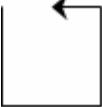




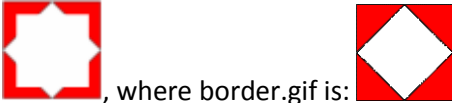
end example]


Attributes	Description
althref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p>[Example:</p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
color (Stroke Color)	<p>Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors.</p> <p>[Example: The shape stroke is blue:</p> <pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
color2 (Stroke Alternate Pattern Color)	<p>Specifies a second color for strokes, used when filltype is pattern. Default is no value.</p> <p>When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.</p> <p>[Example: This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre><v:background fillcolor="yellow"/> <v:shape style="width:60;height:50"</pre>





Attributes	Description
	<pre>strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape></pre>  <p>, where myimage.gif is: </p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
dashstyle (Stroke Dash Pattern)	<p>Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p> <ul style="list-style-type: none"> • solid • shortdash • shortdot • shortdashdot • shortdashdotdot • dot • dash • longdash • dashdot • longdashdot • longdashdotdot <p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is fourfold symmetrical (with round end caps, this is a circle).</p> <p>[Example:</p> <pre><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre>


Attributes	Description
	 <pre data-bbox="454 388 1031 493"><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p data-bbox="414 672 576 703"><i>end example]</i></p> <p data-bbox="414 745 1372 808">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
endarrow (Line End Arrowhead)	<p data-bbox="414 829 1404 892">Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul data-bbox="462 934 625 1134" style="list-style-type: none"> • none • block • classic • diamond • oval • open <p data-bbox="414 1165 544 1197"><i>[Example:</i></p> <pre data-bbox="454 1228 933 1270"><v:stroke endarrow="classic"/></pre>  <p data-bbox="414 1449 576 1480"><i>end example]</i></p> <p data-bbox="414 1522 1437 1585">The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
endarrowlength (Line End Arrowhead Length)	<p data-bbox="414 1606 1437 1669">Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul data-bbox="462 1711 609 1816" style="list-style-type: none"> • short • medium • long <p data-bbox="414 1848 544 1879"><i>[Example:</i></p>



Attributes	Description
	<p data-bbox="456 283 1062 315"><v:stroke ... endarrowlength="long" ... /></p>  <p data-bbox="415 499 574 531"><i>end example]</i></p> <p data-bbox="415 573 1455 636">The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
<p data-bbox="142 657 363 751">endarrowwidth (Line End Arrowhead Width)</p>	<p data-bbox="415 657 1429 720">Specifies the width of the arrowhead at the end of a line. Default is <code>medium</code>. Allowed values are:</p> <ul data-bbox="464 762 600 856" style="list-style-type: none"> • <code>narrow</code> • <code>medium</code> • <code>wide</code> <p data-bbox="415 898 535 930">[Example:</p> <p data-bbox="456 972 1045 1003"><v:stroke ... endarrowwidth="wide" ... /></p>  <p data-bbox="415 1188 574 1220"><i>end example]</i></p> <p data-bbox="415 1262 1446 1325">The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
<p data-bbox="142 1346 350 1409">endcap (Line End Cap)</p>	<p data-bbox="415 1346 1398 1377">Specifies the cap style for the end of a stroke. Default is <code>flat</code>. Allowed values are:</p> <ul data-bbox="464 1419 600 1514" style="list-style-type: none"> • <code>flat</code> • <code>square</code> • <code>round</code> <p data-bbox="415 1556 535 1587">[Example:</p> <p data-bbox="456 1629 1175 1661"><v:stroke ... endcap="round" weight="10pt" ... /></p>  <p data-bbox="597 1791 805 1822">endcap="flat"</p>

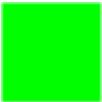
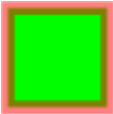
Attributes	Description
	 <p>endcap="square"</p> <p>endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm1</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale]</i></p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • solid - The fill pattern is solid. • tile - The fill image is tiled. • pattern - The fill image is stretched to form a pattern. • frame - The fill image becomes a border for the shape. <p>[<i>Example:</i></p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre>  <p>, where border.gif is:</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).</p>

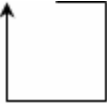
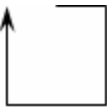
Attributes	Description
forcedash (Force Dashed Outline)	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code>.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
href (Original Image Reference)	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
imagealignshape (Stoke Image Alignment)	<p>Specifies the alignment of the stroke image. If true, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is true.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre>  <p>imagealignshape="false"</p>

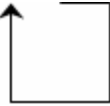
Attributes	Description								
	 <code>imagealignshape="false"</code> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is <code>ignore</code>. Allowed values are:</p> <table border="1" data-bbox="418 703 1320 898"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td><code>ignore</code></td><td>Ignore aspect issues.</td></tr> <tr> <td><code>atleast</code></td><td>Image is at least as big as <code>imagesize</code>.</td></tr> <tr> <td><code>atmost</code></td><td>Image is no bigger than <code>imagesize</code>.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke> </pre>  <code>imagealignshape="ignore"</code>  <code>imagealignshape="atleast"</code>  <code>imagealignshape="atmost"</code> <i>end example]</i> The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).	Value	Description	<code>ignore</code>	Ignore aspect issues.	<code>atleast</code>	Image is at least as big as <code>imagesize</code> .	<code>atmost</code>	Image is no bigger than <code>imagesize</code> .
Value	Description								
<code>ignore</code>	Ignore aspect issues.								
<code>atleast</code>	Image is at least as big as <code>imagesize</code> .								
<code>atmost</code>	Image is no bigger than <code>imagesize</code> .								
imagesize (Stroke Image Size)	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre> <v:stroke ... imagesize="10pt,10pt" ... /> </pre>								

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
joinstyle (Line End Join Style))	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none">• round• bevel• miter <p>[Example:</p> <pre><v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline></pre> <div><p>joinstyle="round"</p><p>joinstyle="bevel"</p><p>joinstyle="miter"</p></div> <p><i>end example]</i></p>

Attributes	Description
linestyle (Stroke Line Style)	<p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p> <p>Specifies the line style of the stroke. Default is single.</p> <ul style="list-style-type: none"> • single • thinThin • thinThick • thickThin • thickBetweenThin <p>[Example:</p> <pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§19.1.3.12).</p>
miterlimit (Miter Joint Limit)	<p>Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p> <p>[Example:</p> <pre><v:stroke joinstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
on (Stroke Toggle)	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p>

Attributes	Description
	<pre><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Stroke Image Location)	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre><v:stroke ... src="myimage.gif" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
startarrow (Line Start Arrowhead)	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none

Attributes	Description
	<ul style="list-style-type: none"> • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke startarrow="classic"/></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... startarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... startarrowwidth="wide" ... /></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
title (Stroke Title)	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre data-bbox="451 751 971 819"><v:fill ... o:title="alt text" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
weight (Stroke Weight)	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_StrokeChild](#)) is located in §A.6.2.
end note]

19.2.2.17 LinkType (Embedded Object Alternate Image Request)

This element specifies the kind of image which shall be requested from an embedded object's host application when the contents of a linked image are updated within a document. When linked images are stored in documents, the only items stored in the document are an image representation and a link to the source. This element specifies the kind of image which shall be requested from the source on update.

[Note: The formats available can vary based on the kind of embedded object - this information is typically queried from the embedded object's application before it is stored. This setting can be omitted, and is usually stored for performance reasons, so it is not queried on each update of the linked object. *end note]*

The possible values for this element are defined by the ST_OLELinkType simple type (§19.2.3.19).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_OLELinkType](#)) is located in §A.6.2.
end note]

19.2.2.18 lock (Shape Protections)

This element specifies locks against actions that can be effected in the UI of an authoring application or programmatically through an object model.

[*Example:* The following snippet locks the shape's aspect ratio and text from user edits.

```
<v:shape ... >
  <o:lock v:ext="edit" aspectratio="t" text="t"/>
</v:shape>
```

end example]

Attributes	Description
adjusthandles (Handles Lock)	Specifies whether the handles of a shape are locked from being edited. Default is <i>false</i> . The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
aspectratio (Aspect Ratio Lock)	Specifies whether the aspect ratio of a shape is locked from being edited. Default is <i>false</i> . The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
cropping (Cropping Lock)	Specifies whether cropping of a shape is locked from being edited. Default is <i>false</i> . The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vml	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. [<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>] The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).
grouping (Grouping Lock)	Specifies whether a shape is locked from being grouped. Default is <i>false</i> . The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
position (Position Lock)	Specifies whether the position of a shape is locked from being edited. Default is <i>false</i> . The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).

Attributes	Description
rotation (Rotation Lock)	<p>Specifies whether the rotation of a shape is locked from being edited. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
selection (Selection Lock)	<p>Specifies whether the shape is locked from being selectable in an editor. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
shapetype (AutoShape Type Lock)	<p>Specifies whether the AutoShape type is locked from being edited. Default is <code>false</code>. If <code>true</code>, the type of an AutoShape cannot be changed in a graphical editor.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
text (Text Lock)	<p>Specifies whether the text attached to a shape is locked from being edited. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
ungrouping (Ungrouping Lock)	<p>Specifies whether a grouped shape is locked from being ungrouped. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>
vertices (Vertices Lock)	<p>Specifies whether the vertices of a path are locked from being edited. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Lock](#)) is located in §A.6.2. *end note*]

19.2.2.19 LockedField (Embedded Object Cannot Be Refreshed)

This element specifies that the embedded object's appearance is locked - that is, that the object's current representation shall be locked to prevent any user interaction or automatic application behavior from modifying its contents.

This element shall contain no content - its presence indicates that the embedded object is locked, and its omission allows the field to be updated.

The possible values for this element are defined by the `ST_TrueFalseBlank` simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.2.2.20 OLEObject (Embedded OLE Object)

This element specifies an embedded object.

[*Example:* The following markup defines a reference to an embedded object using Bonobo. The ProgId attribute contains the shared library that contains the widget. The content type of the referred part identifies the referenced Bonobo object.

```
<OLEObject r:id="rb1" ProgId="OAFIID:Bonobo_Sample_Calculator">
...
</OLEObject>
```

The target of the relationship with ID rb1, defines the Bonobo object itself. This example shows a link to a sample Bonobo widget taken from the article <http://www.ibm.com/developerworks/webservices/library/co-bnbo2.html>, which also provides an introduction to Bonobo. *end example*]

[*Example:* The following demonstrates a video file embedded in a WordprocessingML document:

```
<w:object ... >
  <v:shape id="_x0000_i1025" type="#_x0000_t75"
    style="width:1in;height:24pt" o:ole="">
    <v:imagedata r:id="rId4" o:title=""/>
  </v:shape>
  <o:OLEObject Type="Embed" ProgID="AVIFile" ShapeID="_x0000_i1025"
    DrawAspect="Content" ObjectID="_1219561732" r:id="rId5"/>
</w:object>
```

end example]

Attributes	Description
DrawAspect (Embedded Object Representation)	<p>Specifies how the embedded object is represented visually in the application.</p> <p>[<i>Example:</i></p> <pre><o:OLEObject ... DrawAspect="Content"> </o:OLEObject></pre> <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_OLEDrawAspect simple type (§19.2.3.18).</p>
id (Relationship) Namespace: .../officeDocument /2006/relationships	<p>Specifies the actual OLE object using a standard part relationship lookup.</p> <p>[<i>Example:</i></p> <pre><o:OLEObject ... r:id="rId5"> </o:OLEObject></pre>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_RelationshipId simple type (Part 1, §22.8.2.1).</p>
ObjectID (Unique ID for Embedded Object)	<p>Specifies a unique ID identifying the embedded object.</p> <p>[<i>Example:</i> The following markup defines a reference to a linked object using KParts. The name attribute contains the shared library that contains the plugin. The item element contains the name of the plugin. The content type of the referred part would identifies the referenced KParts object.</p> <pre><oleLink r:id="rKp1" progId="libhtmlvalidatorplugin"> ... </oleLink></pre> <p>The following XML, contained in the target of the relationship with ID rKp1, defines the KPart object, and follows the kpartgui DTD:</p> <pre><!DOCTYPE kpartgui SYSTEM "kpartgui.dtd"> <kpartgui library="libhtmlvalidatorplugin" name="htmlvalidatorplugin" version="1" > <MenuBar> <Menu name="tools"><Text>&Tools</Text> <Action name="validatewebpage"/> </Menu> </MenuBar> </kpartgui></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
ProgID (Object Link Identifier)	<p>Specifies the embedded object server application associated with the embedded object.</p> <p>[<i>Example:</i></p> <pre><o:OLEObject ... ProgID="AVIFile"> </o:OLEObject></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
ShapeID (Embedded Object)	<p>Specifies the shape with which the embedded object is associated. A VML shape provides the visual placeholder for an embedded object and this attribute is set to the id</p>

Attributes	Description
Shape)	<p>of the placeholder shape.</p> <p>[Example:</p> <pre><o:OLEObject ... ShapeID="_x0000_i1025"> </o:OLEObject></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
Type (Embedded Object Type)	<p>Specifies the kind of embedded object connection.</p> <p>[Example:</p> <pre><o:OLEObject ... Type="Embed"> </o:OLEObject></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_OLEType simple type (§19.2.3.20).</p>
UpdateMode (Update Mode for Embedded Object)	<p>Specifies how the object is updated with new data if the Type is Link - automatically or on-demand by the user.</p> <p>[Example:</p> <pre><o:OLEObject ... UpdateMode="Always"> </o:OLEObject></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_OLEUpdateMode simple type (§19.2.3.21).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_OLEObject](#)) is located in §A.6.2. end note]

19.2.2.21 proxy (Shape Reference)

This element specifies an entry in a `r` element rule that contains a reference to one or more shapes that are participating in the rule.

[Example: The following rule defines a connection between two shapes. The shape with id `_s1036` connects shape `_s1033` to `_s1032`:

```
<o:shapelayout v:ext="edit">
  <o:rules v:ext="edit">
    <o:r id="V:Rule1" type="connector" idref="#_s1036">
      <o:proxy start="" idref="#_s1033" connectloc="0"/>
      <o:proxy end="" idref="#_s1032" connectloc="2"/>
    </o:r>
  </o:rules>
</o:shapelayout>
```

end example]

Attributes	Description
connectloc (Connection Location)	<p>Specifies the location on the shape where the connector is attached. The value is an index into the list of connection points defined in the shape - see the connectlocs attribute. Default is 0. Only used in a connector rule.</p> <p>The possible values for this attribute are defined by the W3C XML Schema int datatype.</p>
end (End Point Connection Flag)	<p>Specifies whether the connector's end point is connected to the shape. Default is false. Only used in a connector rule.</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>
idref (Proxy Shape Reference)	<p>Specifies a reference to a shape in the current document. Default is no value. A shape name is used as the reference mechanism; this is not a relationship ID.</p> <p>This attribute indicates that the referenced shape is part of this rule. Two or more proxy elements are used for an alignment rule. A connector rule uses one or two, indicating which shapes the connector is attached to.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
start (Start Point Connection Flag)	<p>Specifies whether the connector's start point is connected to the shape. Default is false. Only used in a connector rule. If both start and end are specified the later one takes precedence.</p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Proxy](#)) is located in §A.6.2. *end note*]

19.2.2.22 r (Rule)

This element specifies a rule entry in a rules element rule set that describes how a certain shape or set of shapes behaves during editing.

[*Example:* The following rule defines a connection between two shapes. The shape with id _s1036 connects shape _s1033 to _s1032:

```
<o:shapelayout v:ext="edit">
  <o:rules v:ext="edit">
    <o:r id="V:Rule1" type="connector" idref="#_s1036">
      <o:proxy start="" idref="#_s1033" connectloc="0"/>
      <o:proxy end="" idref="#_s1032" connectloc="2"/>
    </o:r>
  </o:rules>
</o:shapelayout>
```

end example]

Attributes	Description
how (Alignment Rule Type)	<p>Specifies the kind of alignment for an alignment rule. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • top • middle • bottom • left • center • right <p>The possible values for this attribute are defined by the ST_How simple type (§19.2.3.15).</p>
id (Rule ID)	<p>Specifies an identifier for the rule. Default is no value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
idref (Rule Shape Reference)	<p>Specifies a reference to a shape in the current document that is the primary shape in the rule. [<i>Example:</i> For a connector rule, the connector. <i>end example]</i></p> <p>Default is no value. A shape name is used as the reference mechanism; this is not a relationship ID.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
type (Rule Type)	<p>Specifies the kind of the rule. Default is no value. Allowed values are:</p> <ul style="list-style-type: none"> • arc • callout • connector • align

Attributes	Description
	The possible values for this attribute are defined by the ST_RType simple type (§19.2.3.22).

[Note: The W3C XML Schema definition of this element’s content model (CT_R) is located in §A.6.2. *end note*]

19.2.2.23 regrouptable (Shape Grouping History)

This element specifies a list of entries which describe how shapes were previously grouped so they can be regrouped. The regroupid attribute of shapes indicates which shapes belong together when a regroup is performed. The regrouptable tracks the previous regroupid that should be assigned to all shapes with the given current regroupid.

[Example: Consider a document containing two rectangles and a circle. The rectangles are grouped together, then that group is grouped with the circle. This new group is then ungrouped, leaving the circle and grouped rectangles. The document might contain the following snippets:

```
<v:oval ... o:regroupid="1"/>
<v:group ... o:regroupid="1"/>
  <v:rect ... />
  <v:rect ... />
</v:group>

<o:regrouptable v:ext="edit">
  <o:entry new="1" old="0"/>
</o:regrouptable>
```

The regroupid attribute indicates that the shapes with regroupid 1 were previously grouped together. The entry indicates that if those shapes are regrouped, the new group formed should not have a regroupid value as it was not previously ungrouped.

If the two rectangles are ungrouped, the document reflects that the rectangles were previously grouped and that their old group was previously grouped:

```
<v:oval ... o:regroupid="1"/>
<v:rect ... o:regroupid="2"/>
<v:rect ... o:regroupid="2"/>

<o:regrouptable v:ext="edit">
  <o:entry new="1" old="0"/>
  <o:entry new="2" old="1"/>
</o:regrouptable>
```

end example]

Attributes	Description
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:VML	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. <i>[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end rationale]</i> The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).

[Note: The W3C XML Schema definition of this element's content model ([CT_RegroupTable](#)) is located in §A.6.2. end note]

19.2.2.24 rel (Diagram Relationship)

This element specifies a relationship between two diagram nodes. An optional third node that exists between the primary two can also be included. The relationship has an implicit order since it describes the source and destination nodes.

[Example: In the cycle diagram below, shape 1036 (the shape that is the text box for the text "2") is the first node. A relationship exists between shape 1036 and shape 1044 (the text box containing "1"). In between those shapes is shape 1038 (the yellow arrow).

```

<o:relationtable v:ext="edit">
  <o:rel v:ext="edit" idsrc="#_s1036" iddest="#_s1036"/>
  <o:rel v:ext="edit" idsrc="#_s1042" iddest="#_s1036" idcntr="#_s1043"/>
  <o:rel v:ext="edit" idsrc="#_s1044" iddest="#_s1042" idcntr="#_s1045"/>
  <o:rel v:ext="edit" idsrc="#_s1036" iddest="#_s1044" idcntr="#_s1038"/>
</o:relationtable>

<v:rect id="_s1036" ... >
  <v:textbox ... ><...>2</...></v:textbox>
</v:rect>

<v:rect id="_s1044" ... >
  <v:textbox ... ><...>1</...></v:textbox>
</v:rect>

<v:shape id="_s1038" ... />

```



end example]

Attributes	Description
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:VML</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>idcntr (Diagram Relationship Center Shape)</p>	<p>Specifies the optional identifier of the shape that exists between the source and destination shapes. This is omitted if the relationship does not have a shape between the source and destination shapes.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 1016 870 1079"><o:rel ... idcntr="#s_1038"> </o:rel></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>iddest (Diagram Relationship Destination Shape)</p>	<p>Specifies the identifier of the shape at the destination of the relationship.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 1417 870 1480"><o:rel ... iddest="#s_1044"> </o:rel></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>idsrc (Diagram Relationship Source Shape)</p>	<p>Specifies the identifier of the shape at the source of the relationship.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 1818 854 1881"><o:rel ... idsrc="#s_1036"> </o:rel></pre>

Attributes	Description
	<i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Relation](#)) is located in §A.6.2. *end note]*

19.2.2.25 relationtable (Diagram Relationship Table)

This element specifies a list that describes the relationships among diagram nodes.

[*Example:* The following table describes the parent-child relationships for shapes in an organization chart. The first entry describes the top-level shape in the diagram. The next two rows describe that the shapes are subordinates to the first shape. Shape 1029 is a subordinate of shape 1028. Shape 1032, a connector in this case, is in between the two.

```
<o:relationtable v:ext="edit">
  <o:rel v:ext="edit" idsrc="#_s1028" iddest="#_s1028"/>
  <o:rel v:ext="edit" idsrc="#_s1029" iddest="#_s1028" idcntr="#_s1032"/>
  <o:rel v:ext="edit" idsrc="#_s1030" iddest="#_s1028" idcntr="#_s1033"/>
</o:relationtable>
```

end example]

Attributes	Description
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vmf	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. [<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale]</i> The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).

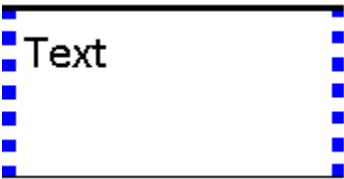
[*Note:* The W3C XML Schema definition of this element's content model ([CT_RelationTable](#)) is located in §A.6.2. *end note]*

19.2.2.26 right (Text Box Right Stroke)

This element specifies the stroke properties for the right border of a text box. It entirely supercedes its parent stroke element if its on attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the on attribute is false or not specified, the border is not shown.

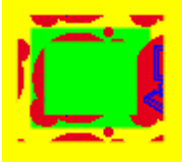

[*Example:* The text box borders are set independently. The bottom border does not inherit the weight from the parent stroke element.



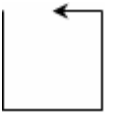
```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```


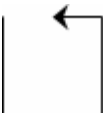



end example]



Attributes	Description
althref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p>[<i>Example:</i></p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
color (Stroke Color)	<p>Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors.</p> <p>[<i>Example:</i> The shape stroke is blue:</p> <pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
color2 (Stroke Alternate Pattern)	<p>Specifies a second color for strokes, used when filltype is pattern. Default is no value.</p>



Attributes	Description
Color)	<p>When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.</p> <p>[<i>Example:</i> This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre> <v:background fillcolor="yellow"/> <v:shape style="width:60;height:50" strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape> </pre>  <p>, where myimage.gif is: </p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
dashstyle (Stroke Dash Pattern)	<p>Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p> <ul style="list-style-type: none"> • solid • shortdash • shortdot • shortdashdot • shortdashdotdot • dot • dash • longdash • dashdot • longdashdot • longdashdotdot <p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated</p>


Attributes	Description
	<p>to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is fourfold symmetrical (with round end caps, this is a circle).</p> <p>[Example:</p> <pre><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre>  <pre><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>endarrow (Line End Arrowhead)</p>	<p>Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke endarrow="classic"/></pre>  <p>end example]</p>

Attributes	Description
<p>endarrowlength (Line End Arrowhead Length)</p>	<p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p> <p>Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... endarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
<p>endarrowwidth (Line End Arrowhead Width)</p>	<p>Specifies the width of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... endarrowwidth="wide" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
<p>endcap (Line End Cap)</p>	<p>Specifies the cap style for the end of a stroke. Default is flat. Allowed values are:</p> <ul style="list-style-type: none"> • flat • square • round

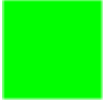
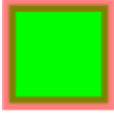
Attributes	Description
	<p>[Example:</p> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p>endcap="flat"</p> <p>endcap="square"</p> <p>endcap="round"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vmf</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end rationale]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is solid. Allowed values are:</p> <ul style="list-style-type: none"> • solid - The fill pattern is solid. • tile - The fill image is tiled. • pattern - The fill image is stretched to form a pattern. • frame - The fill image becomes a border for the shape. <p>[Example:</p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre>

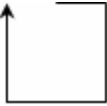

Attributes	Description
	 , where border.gif is:  <i>end example]</i> The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).
forcedash (Force Dashed Outline)	Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is <code>false</code> . Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape. <i>[Example:</i> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
href (Original Image Reference)	Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value. <i>[Example:</i> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema string datatype.
imagealignshape (Stoke Image Alignment)	Specifies the alignment of the stroke image. If <code>true</code> , the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is <code>true</code> . <i>[Example:</i> The top position offset shifts the image alignment relative to the containing window: <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"></pre>

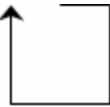
Attributes	Description								
	<div><pre><v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre></div> <div></div> <div>imagealignshape="false"</div> <div>end example]</div> <div>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</div>								
imageaspect (Stroke Image Aspect Ratio)	<div>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</div> <table><tr><th>Value</th><th>Description</th></tr><tr><td>ignore</td><td>Ignore aspect issues.</td></tr><tr><td>atleast</td><td>Image is at least as big as imagesize.</td></tr><tr><td>atmost</td><td>Image is no bigger than imagesize.</td></tr></table> <div>[Example:</div> <div><pre><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre></div> <div></div> <div>imagealignshape="ignore"</div> <div>imagealignshape="atleast"</div> <div>imagealignshape="atmost"</div>	Value	Description	ignore	Ignore aspect issues.	atleast	Image is at least as big as imagesize.	atmost	Image is no bigger than imagesize.
Value	Description								
ignore	Ignore aspect issues.								
atleast	Image is at least as big as imagesize.								
atmost	Image is no bigger than imagesize.								

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>
<p>imagesize (Stroke Image Size)</p>	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre><v:stroke ... imagesize="10pt,10pt" ... /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>insetpen (Inset Border From Path)</p>	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>joinstyle (Line End Join Style))</p>	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none"> • round • bevel • miter <p>[Example:</p> <pre><v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline></pre> <div data-bbox="451 1690 683 1822">  </div> <p>joinstyle="round"</p>

Attributes	Description
	<div data-bbox="451 243 683 373"></div> <div data-bbox="703 344 972 378">joinstyle="bevel"</div> <div data-bbox="451 380 683 510"></div> <div data-bbox="703 485 972 518">joinstyle="miter"</div> <p data-bbox="415 560 574 590"><i>end example]</i></p> <p data-bbox="415 630 1458 695">The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p>
linestyle (Stroke Line Style)	<p data-bbox="415 716 1068 745">Specifies the line style of the stroke. Default is single.</p> <ul data-bbox="462 787 743 947" style="list-style-type: none">• single• thinThin• thinThick• thickThin• thickBetweenThin <p data-bbox="415 989 534 1018">[Example:</p> <div data-bbox="451 1058 1174 1121"><pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre></div> <div data-bbox="415 1159 583 1251"></div> <p data-bbox="415 1293 574 1323"><i>end example]</i></p> <p data-bbox="415 1362 1463 1428">The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§19.1.3.12).</p>
miterlimit (Miter Joint Limit)	<p data-bbox="415 1446 1471 1547">Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p> <p data-bbox="415 1589 534 1619">[Example:</p> <div data-bbox="451 1659 1109 1759"><pre><v:stroke joinstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre></div> <div data-bbox="415 1793 516 1885"></div>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
on (Stroke Toggle)	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p> <pre data-bbox="454 619 1242 745"><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre data-bbox="454 1228 1096 1354"><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Stroke Image Location)	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre data-bbox="454 1837 982 1869"><v:stroke ... src="myimage.gif" ... ></pre>

Attributes	Description
	<p><code></v:stroke></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>startarrow (Line Start Arrowhead)</p>	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <p><code><v:stroke startarrow="classic"/></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
<p>startarrowlength (Line Start Arrowhead Length)</p>	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <p><code><v:stroke ... startarrowlength="long" ... /></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple</p>

Attributes	Description
startarrowwidth (Line Start Arrowhead Width)	<p>type (§19.1.3.7).</p> <p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... startarrowwidth="wide" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
title (Stroke Title)	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
weight (Stroke Weight)	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_StrokeChild](#)) is located in §A.6.2.
end note]

19.2.2.27 rules (Rule Set)

This element specifies a list of rule entries which describe how a certain shape or sets of shapes should behave during editing.

[Example: The following rule defines a connection between two shapes. The shape with id _s1036 connects shape _s1033 to _s1032:

```
<o:shapelayout v:ext="edit">
  <o:rules v:ext="edit">
    <o:r id="V:Rule1" type="connector" idref="#_s1036">
      <o:proxy start="" idref="#_s1033" connectloc="0"/>
      <o:proxy end="" idref="#_s1032" connectloc="2"/>
    </o:r>
  </o:rules>
</o:shapelayout>
```

end example]

Attributes	Description
ext (VML Extension Handling Behavior)	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.
Namespace: urn:schemas-microsoft-com:vml	[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end rationale] The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).

[Note: The W3C XML Schema definition of this element’s content model ([CT_Rules](#)) is located in §A.6.2. end note]

19.2.2.28 [shapedefaults \(New Shape Defaults\)](#)


This element specifies the defaults that are used when creating new shapes. These defaults are stored once per document.

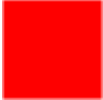

[Example: Consider a case in which an application chooses to store the highest shape ID it has used in the document thus far. This could be used to support the generation of new shape IDs:

```
<o:shapedefaults v:ext="edit" spidmax="1029"/>
```

end example]

Attributes	Description
allowincell (Allow in Table Cell)	Specifies whether the shape is allowed to be placed in a table cell. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
ext (VML Extension	Specifies an optional value that indicates how applications that implement VML should

Attributes	Description
Handling Behavior) Namespace: urn:schemas- microsoft-com:vm1	<p>interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
fill (Shape Fill Toggle)	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden by the fill on attribute.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 688 1062 751"><v:shape ... fill="f" fillcolor="red" ...> </v:shape></pre>  <p><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
fillcolor (Default Fill Color)	<p>Specifies the default shape fill color. Default is no value. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
spidmax (Shape ID Optional Storage)	<p>Specifies an optional value that allows applications a mechanism for storing information they need to persist related to shape IDs. Default is 0.</p> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>
stroke (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§19.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[<i>Example</i>:</p> <pre data-bbox="451 1751 1045 1848"><v:shape ... fillcolor="red" stroke="false" strokecolor="blue"...> </v:shape></pre>

Attributes	Description				
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>				
strokecolor (Shape Stroke Color)	<p>Specifies the primary color of the brush to use to stroke the path of the shape. Default is black. The color attribute of the stroke element (§19.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits representing the red, green and blue values of the color, such as #00FF30. Full details are specified in the simple type description.</p> <p>[Example:</p> <pre data-bbox="451 825 951 888"><v:shape ... strokecolor="red" ...> </v:shape></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>				
style (Shape Styling Properties)	<p>Specifies the CSS2 styling properties of the shape. The CSS2 (Cascading Style Sheets, Level 2) specification, a Recommendation of the World Wide Web Consortium, is available here: http://www.w3.org/TR/REC-CSS2.</p> <p>This attribute uses a semi-colon delimited list of “name:value” pairs, the syntax defined by section 4.1.8 of the CSS2 specification for grouped declarations (without the surrounding braces). The following tables define the allowed properties and the VML treatment of each.</p> <p>[Example:</p> <pre data-bbox="451 1577 1451 1671"><v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape></pre> <p><i>end example]</i></p> <table data-bbox="414 1711 1477 1873"> <tr> <th data-bbox="414 1711 664 1761">Property</th><th data-bbox="664 1711 1477 1761">Description</th></tr> <tr> <td data-bbox="414 1761 664 1873">flip</td><td data-bbox="664 1761 1477 1873">Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</td></tr> </table>	Property	Description	flip	Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:
Property	Description				
flip	Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:				

Attributes	Description	
		<ul style="list-style-type: none"> • x - Flip along the y-axis, reversing the x-coordinates. • y - Flip along the x-axis, reversing the y-coordinates. • xy - Flip along both the y- and x-axis. • yx - Flip along both the x- and y-axis.
	height	<p>Specifies the height of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	left	<p>Specifies the position of the left of the containing block of the shape relative to the element left of it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-bottom	<p>Specifies the position of the bottom of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	margin-left	<p>Specifies the position of the left of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p>

Attributes	Description	
		<ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-right	<p>Specifies the position of the right of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units> - A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's width.
	margin-top	<p>Specifies the position of the top of the containing block of the shape relative to the shape anchor. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	mso-position-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • left • center • right • inside • outside
	mso-position-horizontal-	<p>Specifies relative horizontal position data for objects in WordprocessingML documents. This modifies the mso-</p>

Attributes	Description	
	relative	<p>position-horizontal property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • char
	mso-position-vertical	<p>Specifies the vertical positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-relative	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-position-vertical property. Default is text. Allowed values are:</p> <ul style="list-style-type: none"> • margin • page • text • line
	mso-wrap-distance-bottom	<p>Specifies the distance from the bottom of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-left	<p>Specifies the distance from the left side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-right	<p>Specifies the distance from the right side of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape to include the margin areas. This property does not change the origin.</p>
	mso-wrap-distance-top	<p>Specifies the distance from the top of the shape to the text that wraps around it. Default is 0 pt. This property is different from the CSS margin property, which changes the origin of the shape</p>

Attributes	Description	
		to include the margin areas. This property does not change the origin.
	mso-wrap-edited	Specifies whether the wrap coordinates were customized by the user. If the wrap coordinates are generated by an editor, this property is true; otherwise they were customized by a user. Default is false.
	mso-wrap-style	<p>Specifies the wrapping mode for text in shapes in WordprocessingML documents. Default is square. Allowed values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.
	position	<p>Specifies the kind of positioning used to place an element. Default is static. When the element is contained inside a group, this property shall be absolute. Allowed values are:</p> <ul style="list-style-type: none"> • static - The element is positioned according to the normal flow of the page. The top and left properties are ignored. If the object is anchored inline, this value is used. • absolute - The element is positioned relative to the parent, using the top and left properties. • relative - The element is positioned according to the normal flow of the page, but the top and left properties are used. The overlap of overlapping elements is governed by the z-index property.
	rotation	Specifies the angle that a shape is rotated, in degrees. Default is 0. Positive angles are clockwise.
	top	<p>Specifies the position of the top of the containing block of the shape relative to the element above it in the flow of the page. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none"> • auto - Default position of an element in the flow of the page. • <units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed. • <percentage>- Value expressed as a percentage of the parent object's height.
	visibility	Specifies whether a shape is displayed. Only inherit and hidden are used; any other values are mapped to inherit. Default is

Attributes	Description						
		<p>inherit. Allowed values are:</p> <ul style="list-style-type: none">hidden - The shape is not visible, but is still part of the flow of the objects in the browser. Mouse events are not processed.inherit - The visibility state is inherited from the parent of the shape.					
	width	<p>Specifies the width of the containing block of the shape. Default is 0. It is specified in CSS units or, for elements in a group, in the coordinate system of the parent element. Allowed values are:</p> <ul style="list-style-type: none">auto - Default position of an element in the flow of the page.<units>- A number with an absolute units designator (cm, mm, in, pt, pc, or px) or a relative units designator (em or ex). If no units are given, pixels (px) is assumed.<percentage>- Value expressed as a percentage of the parent object's width.					
	z-index	<p>Specifies the display order of overlapping shapes. Default is 0. This property shall not be used for shapes anchored inline. Allowed values are:</p> <ul style="list-style-type: none">auto - Uses the order that the shapes appear in the page, bottom to top.<order>- A number that represents the stacking precedence. Shapes with higher numbers are placed on top of those with lower numbers. Negative numbers are allowed.					
	<p>The following properties are only used by the textbox element (§19.1.2.22):</p> <table><tr><th>Property</th><th>Description</th></tr><tr><td>direction</td><td><p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p><ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.</td></tr><tr><td>layout-flow</td><td><p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p><ul style="list-style-type: none">horizontal - Text is displayed horizontally.</td></tr></table>		Property	Description	direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.	layout-flow
Property	Description						
direction	<p>Specifies the direction of the text in the textbox. Default is ltr. This property is superceded by the mso-direction-alt property if that is specified. Allowed values are:</p> <ul style="list-style-type: none">ltr - Text is displayed left-to-right.rtl - Text is displayed right-to-left.						
layout-flow	<p>Determines the flow of the text layout in a textbox. Default is horizontal. Allowed values are:</p> <ul style="list-style-type: none">horizontal - Text is displayed horizontally.						

Attributes	Description	
		<ul style="list-style-type: none"> vertical - Text is displayed vertically. vertical-ideographic - Ideographic text is displayed vertically. horizontal-ideographic - Ideographic text is displayed horizontally.
	mso-direction-alt	Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.
	mso-fit-shape-to-text	Specifies whether the shape stretches to fit the text in the textbox. Default is false.
	mso-fit-text-to-shape	Specifies whether the text stretches to fit the textbox. Default is false.
	mso-layout-flow-alt	Specifies the alternate layout flow for text in textboxes. This property is used instead of layout-flow when the layout flow is from bottom to top for non-ideographic languages. Its only value is bottom-to-top.
	mso-next-textbox	Specifies the ID of the next textbox in a series. Used to keep track of a set of linked textboxes. Default is no value.
	mso-rotate	<p>Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are:</p> <ul style="list-style-type: none"> 0 90 180 -90
	mso-text-scale	Specifies the scaling factor for fitting text to shapes. Default is 0. This property is only used if mso-fit-text-to-shape is true.
	v-text-anchor	<p>Specifies the vertical anchoring of text in a textbox. Default is top. The alignment of a text anchor only becomes evident if mso-fit-text-to-shape is false. This property is different from the vertical-align CSS property, which is used for ideographic languages. Allowed values are:</p> <ul style="list-style-type: none"> top middle bottom top-center middle-center bottom-center top-baseline bottom-baseline top-center-baseline bottom-center-baseline

Attributes	Description																				
	<p>The following properties are only used by the textpath element (§19.1.2.23):</p> <table border="1"> <thead> <tr> <th data-bbox="418 352 662 399">Property</th><th data-bbox="662 352 1481 399">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="418 399 662 554">font</td><td data-bbox="662 399 1481 554">Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.</td></tr> <tr> <td data-bbox="418 554 662 638">font-family</td><td data-bbox="662 554 1481 638">Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.</td></tr> <tr> <td data-bbox="418 638 662 756">font-size</td><td data-bbox="662 638 1481 756">Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.</td></tr> <tr> <td data-bbox="418 756 662 1016">font-style</td><td data-bbox="662 756 1481 1016"> Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. </td></tr> <tr> <td data-bbox="418 1016 662 1243">font-variant</td><td data-bbox="662 1016 1481 1243"> Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps </td></tr> <tr> <td data-bbox="418 1243 662 1877">font-weight</td><td data-bbox="662 1243 1481 1877"> Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <thead> <tr> <th data-bbox="678 1390 876 1436">Value</th><th data-bbox="876 1390 1464 1436">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="678 1436 876 1638">normal lighter 100 200 300 400</td><td data-bbox="876 1436 1464 1638">Treated as non-bold.</td></tr> <tr> <td data-bbox="678 1638 876 1877">bold bolder 500 600 700 800 900</td><td data-bbox="876 1638 1464 1877">Treated as bold.</td></tr> </tbody> </table> </td></tr> </tbody> </table>	Property	Description	font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.	font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.	font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.	font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. 	font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps 	font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <thead> <tr> <th data-bbox="678 1390 876 1436">Value</th><th data-bbox="876 1390 1464 1436">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="678 1436 876 1638">normal lighter 100 200 300 400</td><td data-bbox="876 1436 1464 1638">Treated as non-bold.</td></tr> <tr> <td data-bbox="678 1638 876 1877">bold bolder 500 600 700 800 900</td><td data-bbox="876 1638 1464 1877">Treated as bold.</td></tr> </tbody> </table>	Value	Description	normal lighter 100 200 300 400	Treated as non-bold.	bold bolder 500 600 700 800 900	Treated as bold.
Property	Description																				
font	Specifies a compound value of font settings. Default is no value. The values are the same as those of the CSS font property. The order of definitions in the string is: font-style, font-variant, font-weight, font-size, line-height, font-family.																				
font-family	Specifies the family of the font. Default is no value. The values are the same as those of the CSS font-family property.																				
font-size	Specifies the size of the font. Default is no value. The font size is defined in points. The values are the same as those of the CSS font-size property.																				
font-style	Specifies the amount of slant for a font. Default is normal. The values are the same as those of the CSS font-style property. Allowed values are: <ul style="list-style-type: none"> • normal • italic • oblique - Treated the same as italic. 																				
font-variant	Specifies the variant style of a font. Default is normal. The values are the same as those of the CSS font-variant property. Allowed values are: <ul style="list-style-type: none"> • normal • small-caps 																				
font-weight	Specifies the thickness of the letters of the font. Default is normal. The values are the same as those of the CSS font-weight property. Allowed values are: <table border="1"> <thead> <tr> <th data-bbox="678 1390 876 1436">Value</th><th data-bbox="876 1390 1464 1436">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="678 1436 876 1638">normal lighter 100 200 300 400</td><td data-bbox="876 1436 1464 1638">Treated as non-bold.</td></tr> <tr> <td data-bbox="678 1638 876 1877">bold bolder 500 600 700 800 900</td><td data-bbox="876 1638 1464 1877">Treated as bold.</td></tr> </tbody> </table>	Value	Description	normal lighter 100 200 300 400	Treated as non-bold.	bold bolder 500 600 700 800 900	Treated as bold.														
Value	Description																				
normal lighter 100 200 300 400	Treated as non-bold.																				
bold bolder 500 600 700 800 900	Treated as bold.																				

Attributes	Description	
	mso-text-shadow	Specifies whether a shadow is applied to the text on a text path. Default is false.
	text-decoration	<p>Specifies the style of text decoration. Default is none. The values are the same as those of the CSS text-decoration property. Allowed values are:</p> <ul style="list-style-type: none"> • none • underline • overline • line-through • blink
	v-rotate-letters	Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.
	v-same-letter-heights	Specifies whether all letters are the same height regardless of initial case. If true, the lowercase letters are stretched to the height of the uppercase letters. Default is false.
	v-text-align	<p>Specifies the alignment of text. Default is left. Allowed values are:</p> <ul style="list-style-type: none"> • left • right • center • justify • letter-justify - Distributes the extra space between the letters. • stretch-justify - Stretches the letters to fill in the space.
	v-text-kern	Specifies whether kerning is turned on. Default is false.
	v-text-reverse	Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.
	v-text-spacing-mode	<p>Specifies the mode for letter spacing. Default is tightening. This property determines whether space is removed between each letter (tightening) or added between each letter (tracking). The amount of letter spacing change is defined by the v-text-spacing property. Allowed values are:</p> <ul style="list-style-type: none"> • tightening • tracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	<p>The line (§19.1.2.12), polyline (§19.1.2.15) and curve (§19.1.2.3) elements ignore the following properties:</p>	

Attributes	Description
	<ul style="list-style-type: none"> • top • left • width • height <p>The following properties are not inherited by an element that references a shapetype element (§19.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[*Note:* The W3C XML Schema definition of this element’s content model ([CT_ShapeDefaults](#)) is located in §A.6.2. *end note*]

19.2.2.29 [shapelayout](#) (Shape Layout Properties)

This element contains child elements that store information used in the editing and layout of shapes.

Attributes	Description
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vml	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>

[*Note:* The W3C XML Schema definition of this element’s content model ([CT_ShapeLayout](#)) is located in §A.6.2. *end note*]

19.2.2.30 signatureline (Digital Signature Line)

This element specifies a signature line in a document. A signature line provides a visual representation of a signature in a document that is digitally signed. The signature line element indicates that the VML shape in which it appears acts as that visual representation. Typically, the VML shape is an image.

[Example:

```
<v:shape ... >
  <v:imagedata ... />
  <o:signatureline v:ext="edit" id="{11979195-DE54-414B-ABD6-5F63607C648B}"
    provid="{00000000-0000-0000-0000-000000000000}" o:suggestedsigner="John Doe"
    o:suggestedsigner2="Manager" o:suggestedsigneremail=johndoe@example.com
    allowcomments="t" issignatureline="t"/>
</v:shape>
```

The signature line in the document might look like this:



end example]

Attributes	Description
addlxml (Additional Signature Information)	<p>Specifies an optional string that is used to store additional information about the digital signature. Default is no value. [<i>Rationale</i>: Some digital signature software stores, for example, server and region information with the signature. <i>end rationale</i>]</p> <p>[Example:</p> <pre><o:signatureline ... o:addlxml="..."> </o:signatureline></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
allowcomments (User-specified Comments Flag)	<p>Specifies whether the user can attach comments to the signature line at signing time. Default is false.</p> <p>[Example:</p>

Attributes	Description
	<pre><o:signatureline ... allowcomments="true"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm1</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale</i>: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>id (Unique ID)</p>	<p>Specifies a unique ID for the signature line. Default is no value.</p> <p>[<i>Example</i>:</p> <pre><o:signatureline ... id="{11979195-DE54-414B-ABD6-5F63607C648B}"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_Guid simple type (Part 1, §22.9.2.4).</p>
<p>issignatureline (Signature Line Flag)</p>	<p>Specifies whether the image is a signature line. Default is true.</p> <p>[<i>Example</i>:</p> <pre><o:signatureline ... issignatureline="true"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
<p>provid (Signature Provider ID)</p>	<p>Specifies a unique ID identifying which signature provider created the signature line. Default is no value. [<i>Guidance</i> The GUID is typically the CLSID of the provider COM add-in. <i>end guidance</i>]</p> <p>[<i>Example</i>:</p> <pre><o:signatureline ...</pre>

Attributes	Description
	<pre> provid="{00000000-0000-0000-0000-000000000000}"> </o:signatureline> <i>end example</i> The possible values for this attribute are defined by the ST_Guid simple type (Part 1, §22.9.2.4). </pre>
showsigndate (Show Signed Date Flag)	Specifies whether the signed signature line image generated should include the date of signing. Default is true. [Example: <pre> <o:signatureline ... showsigndate="false"> </o:signatureline> <i>end example</i> </pre> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
signinginstructions (Instructions for Signing)	Specifies text shown to the user at signing time. Default is no value. [Example: <pre> <o:signatureline ... o:signinginstructions="Sign here"> </o:signatureline> <i>end example</i> </pre> The possible values for this attribute are defined by the W3C XML Schema string datatype.
signinginstructionsset (Use Signing Instructions Flag)	Specifies whether there is data set in the signinginstructions attribute. Default is false. [Example: <pre> <o:signatureline ... signinginstructionsset="true"> </o:signatureline> <i>end example</i> </pre> The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).
sigprovurl (Signature Provider Download URL)	Specifies the URL for downloading the signature provider. Default is no value. [Example:

Attributes	Description
	<pre><o:signatureline ... o:sigprovurl="http://www.example.com"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
suggestedsigner (Suggested Signer Line 1)	<p>Specifies the first line of information of who should sign the signature line. Default is no value.</p> <p>[Example:</p> <pre><o:signatureline ... o:suggestedsigner="John Doe"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
suggestedsigner2 (Suggested Signer Line 2)	<p>Specifies the second line of information of who should sign the signature line. Default is no value.</p> <p>[Example:</p> <pre><o:signatureline ... o:suggestedsigner2="Title"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
suggestedsigneremail (Suggested Signer E-mail Address)	<p>Specifies the e-mail address of who should sign the signature line. Default is no value.</p> <p>[Example:</p> <pre><o:signatureline ... o:suggestedsigneremail="johndoe@example.com"> </o:signatureline></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[*Note: The W3C XML Schema definition of this element's content model ([CT_SignatureLine](#)) is located in §A.6.2. end note*]

19.2.2.31 skew (Skew Transform)

This element specifies a perspective skew effect on a shape. The skew is applied to vector graphics, not image data on the shape in picture fills or image elements. The `on` attribute shall be true and a permitted value assigned to the `matrix` attribute.

Attributes	Description
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vml	Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML. <i>[Rationale: This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. end rationale]</i> The possible values for this attribute are defined by the <code>ST_Ext</code> simple type (§19.1.3.2).
id (Skew ID)	Specifies a name that provides a unique identifier for a skew. Default is no value. The possible values for this attribute are defined by the W3C XML Schema string datatype.
matrix (Skew Perspective Matrix)	Specifies a perspective transform of a skew. Default is "1,0,0,1,0,0". The matrix is given in the form " s_{xx} , s_{xy} , s_{yx} , s_{yy} , p_x , p_y " where s = scale and p = perspective. If the <code>offset</code> attribute is in absolute units then p_x , p_y are in 1/EMU units; otherwise they are an inverse fraction of the shape size. The possible values for this attribute are defined by the W3C XML Schema string datatype.
offset (Skew Offset)	Specifies the amount of x,y offset from the shape's location. Default is "2pt,2pt". Positive values are measured from the upper left of the face of the shape. Values are specified as either an absolute measurement or a fractional value of the shape's dimensions (−0.5 to +0.5). The possible values for this attribute are defined by the W3C XML Schema string datatype.
on (Skew Toggle)	Specifies whether a skew is displayed. Default is <code>false</code> . The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§20.1.2.5).
origin (Skew Origin)	Specifies the origin of the skew. Default is "0,0". Values are typically a percentage of the shape's size and range from -0.5 to +0.5. Larger

Attributes	Description
	<p>values are allowed that give offsets as multiples of the shape's size.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

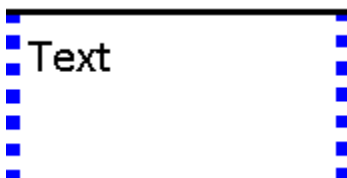
[Note: The W3C XML Schema definition of this element's content model ([CT_Skew](#)) is located in §A.6.2. *end note*]

19.2.2.32 top (Text Box Top Stroke)

This element specifies the stroke properties for the top border of a text box. It entirely supercedes its parent stroke element if its on attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the on attribute is false or not specified, the border is not shown.


[Example: The text box borders are set independently. Note that the bottom border does not inherit the weight from the parent stroke element.



```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```

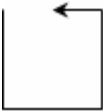



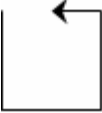
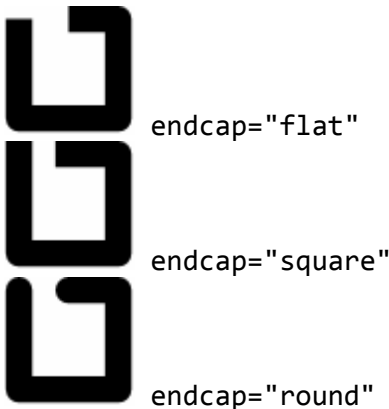
end example]


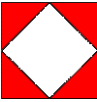
Attributes	Description
althref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p>[Example:</p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>



Attributes	Description
	datatype.
color (Stroke Color)	<p>Specifies the stroke color. Overrides the strokecolor attribute of a shape. Default is black. See the fillcolor attribute for a list of supported named colors.</p> <p>[Example: The shape stroke is blue:</p> <pre><v:shape ... strokecolor="red" ... > <v:stroke color="blue"/> </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
color2 (Stroke Alternate Pattern Color)	<p>Specifies a second color for strokes, used when filltype is pattern. Default is no value.</p> <p>When a pattern fill is used for the stroke, the stroke color is used in colored parts of the source image. The color2 defines an alternate color to use in place of black in the source image.</p> <p>[Example: This unusual example is intended to demonstrate how the image and colors interact to create a patterned stroke. The yellow background shows transparency. The non-square shape and square image create an effective offset. The heavy stroke weight shows more of the image. The green shape fill shows how the stroke is overlaid on the shape.</p> <pre><v:background fillcolor="yellow"/> <v:shape style="width:60;height:50" strokecolor="red" fillcolor="lime" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="pattern" weight="10pt" src="myimage.gif" color2="blue"/> </v:shape></pre> <div data-bbox="414 1465 596 1629" data-label="Image"> </div> <p>, where myimage.gif is: </p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§20.1.2.3).</p>
dashstyle (Stroke Dash Pattern)	Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:


Attributes	Description
	<ul style="list-style-type: none"> • solid • shortdash • shortdot • shortdashdot • shortdashdotdot • dot • dash • longdash • dashdot • longdashdot • longdashdotdot <p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. The endcap style is applied to each dash but the arrow style is not. The string defines the length of the dash then the length of the space. This can be repeated to form complex dash styles. The string should always contain a pair of numbers; if it contains an odd number of numbers the last is disregarded. 0 implies a dot that is fourfold symmetrical (with round end caps, this is a circle).</p> <p>[Example:</p> <pre><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre>  <pre><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
endarrow (Line End Arrowhead)	<p>Specifies an arrowhead for the end of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none



Attributes	Description
	<ul style="list-style-type: none"> • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke endarrow="classic"/></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
endarrowlength (Line End Arrowhead Length)	<p>Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium • long <p>[Example:</p> <pre><v:stroke ... endarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
endarrowwidth (Line End Arrowhead Width)	<p>Specifies the width of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p>


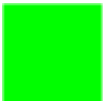
Attributes	Description
	<p><code><v:stroke ... endarrowwidth="wide" ... /></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
<p>endcap (Line End Cap)</p>	<p>Specifies the cap style for the end of a stroke. Default is <code>flat</code>. Allowed values are:</p> <ul style="list-style-type: none"> • <code>flat</code> • <code>square</code> • <code>round</code> <p>[Example:</p> <p><code><v:stroke ... endcap="round" weight="10pt" ... /></code></p>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§19.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vml</p>	<p>Specifies an optional value that indicates how applications that implement VML should interpret extensions not defined as part of the original specification of core VML.</p> <p>[<i>Rationale:</i> This part of the original VML specification is included to assist applications that leverage existing VML support in implementing the Office Open XML Format. <i>end rationale]</i></p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§19.1.3.2).</p>
<p>filltype (Stroke</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is <code>solid</code>. Allowed</p>



Attributes	Description
Image Style)	<p>values are:</p> <ul style="list-style-type: none"> • solid - The fill pattern is solid. • tile - The fill image is tiled. • pattern - The fill image is stretched to form a pattern. • frame - The fill image becomes a border for the shape. <p>[Example:</p> <pre><v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape></pre> <div data-bbox="414 814 863 915">  , where border.gif is:  </div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§19.1.3.4).</p>
forcedash (Force Dashed Outline)	<p>Specifies whether a dashed outline is used to draw a shape when a shape has no line or fill. Default is false.</p> <p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
href (Original Image Reference)	<p>Specifies the URL to the original image file. Used only if the picture has been linked and embedded. Default is no value.</p> <p>[Example:</p> <pre><v:fill ... o:href="myimage.gif" ... > </v:fill></pre>

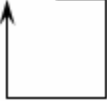
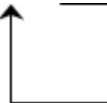
Attributes	Description								
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>								
<p>imagealignshape (Stroke Image Alignment)</p>	<p>Specifies the alignment of the stroke image. If true, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is true.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre> <v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape> </pre>  <p>imagealignshape="false"</p>  <p>imagealignshape="false"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>								
<p>imageaspect (Stroke Image Aspect Ratio)</p>	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table border="1" data-bbox="418 1543 1318 1738"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>ignore</td><td>Ignore aspect issues.</td></tr> <tr> <td>atleast</td><td>Image is at least as big as imagesize.</td></tr> <tr> <td>atmost</td><td>Image is no bigger than imagesize.</td></tr> </tbody> </table> <p>[Example:</p> <pre> <v:stroke filltype="frame" weight="10pt" </pre>	Value	Description	ignore	Ignore aspect issues.	atleast	Image is at least as big as imagesize.	atmost	Image is no bigger than imagesize.
Value	Description								
ignore	Ignore aspect issues.								
atleast	Image is at least as big as imagesize.								
atmost	Image is no bigger than imagesize.								

Attributes	Description
	<pre>src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre>  <p>imagealignshape="ignore"</p> <p>imagealignshape="atleast"</p> <p>imagealignshape="atmost"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§19.1.3.5).</p>
imagesize (Stroke Image Size)	<p>Specifies the size of the image for the stroke. Default is the size of the image.</p> <p>[Example:</p> <pre><v:stroke ... imagesize="10pt,10pt" ... /></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
insetpen (Inset Border From Path)	<p>Specifies that the border shall be displayed inside of the path defining the shape, rather than along the path (the default border placement), or outside of the path as might be done with an image.</p> <p>[Example:</p> <pre><v:shape ... insetpen="true" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
joinstyle (Line End Join Style)	<p>Specifies the join style for line ends. Default is round.</p> <ul style="list-style-type: none"> • round

Attributes	Description
	<ul style="list-style-type: none"> • bevel • miter <p>[Example:</p> <pre><v:polyline strokeweight="10pt" strokecolor="navy" points="10pt,10pt,50pt,50pt,90pt,10pt"> <v:stroke joinstyle="bevel"/> </v:polyline></pre>  <p>joinstyle="round"</p> <p>joinstyle="bevel"</p> <p>joinstyle="miter"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§19.1.3.11).</p>
linestyle (Stroke Line Style)	<p>Specifies the line style of the stroke. Default is single.</p> <ul style="list-style-type: none"> • single • thinThin • thinThick • thickThin • thickBetweenThin <p>[Example:</p> <pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type</p>

Attributes	Description
miterlimit (Miter Joint Limit)	<p>(§19.1.3.12).</p> <p>Specifies the smoothness of the miter joint, or the maximum distance between the inner point and outer point of a joint. This number is a multiple of the thickness of the line. Default is 8.</p> <p>[Example:</p> <pre><v:stroke joinstyle="miter" weight="10pt" miterlimit="2"> </v:stroke></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
on (Stroke Toggle)	<p>Specifies whether the stroke is displayed. Default is true. This attribute overrides the shape's stroke attribute.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" stroked="true" fillcolor="lime" strokecolor="red"> <v:stroke on="false" weight="5pt"/> </v:rect></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§20.1.2.5).</p>
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p> <p>[Example:</p> <pre><v:rect style="width:50;height:50" fillcolor="lime" strokecolor="red"> <v:stroke weight="5pt" opacity="50%"/> </v:rect></pre>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Stroke Image Location)	<p>Specifies the source image to load for a stroke fill. Default is no value.</p> <p>[Example:</p> <pre><v:stroke ... src="myimage.gif" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
startarrow (Line Start Arrowhead)	<p>Specifies an arrowhead for the start of a line. Default is none. The path shall not be closed with the x command for the arrowhead to show. Allowed values are:</p> <ul style="list-style-type: none"> • none • block • classic • diamond • oval • open <p>[Example:</p> <pre><v:stroke startarrow="classic"/></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§19.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • short • medium

Attributes	Description
	<ul style="list-style-type: none"> • long <p>[Example:</p> <pre><v:stroke ... startarrowlength="long" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§19.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values are:</p> <ul style="list-style-type: none"> • narrow • medium • wide <p>[Example:</p> <pre><v:stroke ... startarrowwidth="wide" ... /></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§19.1.3.9).</p>
title (Stroke Title)	<p>Specifies the title of an embedded stroke image. This is typically set to the comment property of the image, which is often blank.</p> <p>[Example:</p> <pre><v:fill ... o:title="alt text" ... > </v:fill></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
weight (Stroke	<p>Specifies the thickness of a stroke. Default is 1. This attribute overrides the shape's</p>

Attributes	Description
Weight)	<p>strokeweight attribute.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_StrokeChild](#)) is located in §A.6.2. *end note*]

19.2.3 Simple Types

The following additional simple type information in the urn:schemas-microsoft-com:office:office namespace is used for documents of a transitional conformance class.

19.2.3.1 ST_AlternateMathContentType (Alternate Math Content Type)

This simple type specifies the content type of the XML markup stored within the equationxml element.

The following values are reserved:

Value	Meaning
officeopenxmlmath	Specifies that the data has been stored using the Office Open XML Math syntax defined in Part 1, §22.1.
mathml	Specifies that the data has been stored using the MathML syntax.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_AlternateMathContentType](#)) is located in §A.6.2. *end note*]

19.2.3.2 ST_Angle (Callout Angles)

This simple type specifies values for the angle attribute of the callout element (§19.2.2.2).

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
30 (30 degrees)	30 degrees.
45 (45 degrees)	45 degrees.
60 (60 degrees)	60 degrees.
90 (90 degrees)	90 degrees.
any (Any Angle)	Unconstrained angle.
auto (Automatic Angle)	The application chooses an appropriate angle.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_Angle](#)) is located in §A.6.2. end note*]

19.2.3.3 [ST_BWMode \(Black And White Modes\)](#)

This simple type specifies the ways in which a shape renders in a black and white context.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
auto (Automatic)	Use the bwpure or bwnormal attributes based on the type of output being generated.
black (Black)	Use black only.
blackTextAndLines (Black Text And Lines)	Use shades of gray, except for text and lines, which are black.
color (Color)	Do not use grayscale or black and white.
grayOutline (Gray Outlines)	Use gray and white only.
grayScale (Grayscale)	Use shades of gray only.
hide (Hide Object When Displayed in Black and White)	Do not display the object when rendering in only black and white.
highContrast (Black And White)	Use black and white only, no grays.
inverseGray (Inverse Grayscale)	Use shades of gray only, but invert light and dark grays.
lightGrayscale (Light grayscale)	Use light shades of gray only.
undrawn (Do Not Show)	Do not show the object.
white (White)	Use white only.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_BWMode](#)) is located in §A.6.2. end note*]

19.2.3.4 [ST_CalloutDrop \(Callout Drop Location\)](#)

This simple type specifies location values for the drop attribute of the callout element (§19.2.2.2).

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_CalloutDrop](#)) is located in §A.6.2. end note*]

19.2.3.5 [ST_CalloutPlacement \(Callout Placement\)](#)

This type defines location values used by the drop attribute of the callout element (§19.2.2.2).

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
bottom (Bottom placement)	Bottom of the shape.
center (Center placement)	Vertical center of the shape.
top (Top placement)	Top of the shape.
user (User-defined placement)	User-defined placement.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_CalloutPlacement](#)) is located in §A.6.2. *end note*]

19.2.3.6 ST_ColorMode (Extrusion Color Types)

This simple type specifies ways that the extrusion color is defined.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
auto (Use Shape Fill Color)	Specifies that the color of the extrusion is the same as the fill color of the shape.
custom (Use Custom Color)	Specifies that the extrusion is the color of the color attribute.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_ColorMode](#)) is located in §A.6.2. *end note*]

19.2.3.7 ST_ConnectorType (Connector Type)

This simple type specifies types of connectors.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
curved (Curved Connector)	A curved connector.
elbow (Elbow Connector)	An elbow-shaped connector.
none (No Connector)	No connector.
straight (Straight Connector)	A straight connector.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_ConnectorType](#)) is located in §A.6.2. *end note*]

19.2.3.8 ST_ConnectType (Connection Locations Type)

This simple type specifies types of connection locations.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
custom (Custom Connections)	A custom array of connection locations.
none (No)	No connection locations.
rect (Four Connections)	Standard four connection points at midpoints of top, bottom, left, and right sides.
segments (Edit Point Connections)	The edit points of the shape are used. Edit points are the black dots in a graphical editor that are used to select parts of a shape.

[Note: The W3C XML Schema definition of this simple type’s content model ([ST_ConnectType](#)) is located in §A.6.2. end note]

19.2.3.9 ST_ContentType (Content Type)

This simple type specifies a content type.

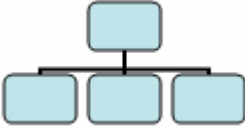
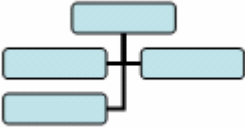
This simple type's contents are a restriction of the W3C XML Schema string datatype.

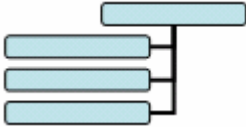
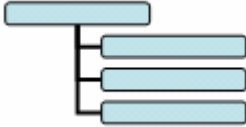
[Note: The W3C XML Schema definition of this simple type’s content model ([ST_ContentType](#)) is located in §A.6.2. end note]

19.2.3.10 ST_DiagramLayout (Diagram Layout Type)

This simple type specifies the style of automatic layout to apply to a node in a diagram.

This simple type's contents are a restriction of the W3C XML Schema integer datatype.

Enumeration Value	Description
0 (Top-down Centered)	Top-down, centered layout. 
1 (Hanging Both Sides)	Hanging, both sides layout. 
2 (Hanging Right Side)	Hanging, right side layout.

Enumeration Value	Description
	
3 (Hanging Left Side)	Hanging, left side layout. 

[Note: The W3C XML Schema definition of this simple type's content model ([ST_DiagramLayout](#)) is located in §A.6.2. *end note*]

19.2.3.11 ST_ExtrusionPlane (Extrusion Planes)

This simple type specifies three axis-aligned planes.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
XY (XY Plane)	The xy plane.
YZ (YZ Plane)	The yz plane.
ZX (ZX Plane)	The zx plane.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionPlane](#)) is located in §A.6.2. *end note*]

19.2.3.12 ST_ExtrusionRender (Extrusion Rendering Types)

This simple type specifies different rendering modes.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
boundingCube (Bounding Cube)	Rendering displays the bounding cube that contains the shape.
solid (Solid)	Rendering displays a solid shape.
wireFrame (Wireframe)	Rendering displays a wireframe shape.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionRender](#)) is located in §A.6.2. *end note*]

19.2.3.13 ST_ExtrusionType (Extrusion Type)

This simple type specifies types of extrusions.

This simple type's contents are a restriction of the W3C XML Schema string datatype.




Enumeration Value	Description
parallel (Parallel Projection)	Extrusion is rendered so that the center of projection is infinitely far away; the extrusion lines do not converge.
perspective (Perspective Projection)	Extrusion is rendered to a center of projection, which is the same as the vanishing point for unrotated objects.

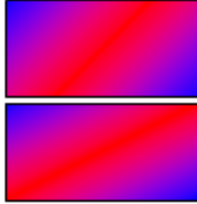

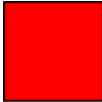

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionType](#)) is located in §A.6.2. *end note*]

19.2.3.14 ST_FillType (Shape Fill Type)

This simple type specifies the types for fills applied to a shape.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
background (Use Background Fill)	Use the fill properties of the background of the object on which the shape exists, such as the page.
frame (Stretch Image to Fit)	The image is stretched to fill the shape. 
gradient (Linear Gradient)	The fill colors blend together in a linear gradient from bottom to top. 
gradientCenter (Centered Radial Gradient)	This indicates that the gradient runs across the center of the shape for a gradient that is defined as gradientRadial in the parent fill element (§19.1.2.5) that is defined in the VML namespace.
gradientRadial (Radial Gradient)	The fill colors blend together in a radial gradient. 
gradientUnscaled (Unscaled Gradient)	The gradient angle is not scaled relative to the aspect ratio of the shape.

Enumeration Value	Description
	<p>[Example: The shapes below are twice as wide as they are tall. The first shape uses an unscaled gradient and the second uses a regular scaled gradient:</p>  <p>end example]</p>
pattern (Image Pattern)	<p>The image is used to create a pattern using the fill colors.</p> 
solid (Solid Fill)	<p>The fill pattern is a solid color.</p> 
tile (Tiled Image)	<p>The fill image is tiled.</p> 

[Note: The W3C XML Schema definition of this simple type's content model ([ST_FillType](#)) is located in §A.6.2.
end note]

19.2.3.15 ST_How (Alignment Type)

This simple type specifies types of alignment.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
bottom (Bottom Alignment)	Bottom vertical alignment.
center (Center Alignment)	Center horizontal alignment.
left (Left Alignment)	Left horizontal alignment.
middle (Middle Alignment)	Middle vertical alignment.
right (Right Alignment)	Right horizontal alignment.
top (Top Alignment)	Top vertical alignment.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_How](#)) is located in §A.6.2. end note*]

19.2.3.16 [ST_HrAlign](#) (Alignment Type)

This simple type specifies alignments for horizontal rules.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
center (Center Alignment)	Center aligned.
left (Left Alignment)	Left aligned.
right (Right Alignment)	Right aligned.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_HrAlign](#)) is located in §A.6.2. end note*]

19.2.3.17 [ST_InsetMode](#) (Inset Margin Type)

This simple type specifies how inner text margins are obtained.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
auto (Automatic Margins)	Inner text margins are calculated by the application.
custom (Custom Margins)	Inner text margins are specified by the shape.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_InsetMode](#)) is located in §A.6.2. end note*]

19.2.3.18 [ST_OLEDrawAspect](#) (Embedded Object Representations)

This simple type specifies the ways in which embedded objects are displayed in the application.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
Content (Snapshot)	The object's presentation is a picture of the contained document (provided by the embedded object server technology).
Icon (Icon)	The object's presentation is an icon.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_OLEDrawAspect](#)) is located in §A.6.2. end note*]

19.2.3.19 ST_OLELinkType (Embedded Object Alternate Image Request Types)

This simple type specifies the kind of image that shall be requested from the application which hosts embedded object data for a linked object. This simple type allows any image format to be specified; however, the following values are reserved:

Enumeration Value	Description
Bitmap	Specifies that a bitmap should be requested.
EnhancedMetaFile	Specifies that a metafile (non-raster) image should be requested.
Jpeg	Specifies an image which should use the JPEG format.
Picture	Specifies that any image format can be requested. [<i>Example: PNG or CGM (ISO/IEC 8632). end example</i>]
Png	Specifies an image which should use the Portable Network Graphics format.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_OLELinkType](#)) is located in §A.6.2. end note*]

19.2.3.20 ST_OLEType (Embedded Connection Type)

This simple type specifies whether the embedded object is included in the package (that is, embedded) or is stored outside the package (that is, linked).

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
Embed (Embedded Object)	Embedded object.
Link (Linked Object)	Linked object.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_OLEType](#)) is located in §A.6.2. end note*]

19.2.3.21 ST_OLEUpdateMode (Embedded Object Update Method Type)

This simple type specifies how an embedded object is updated.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
Always (Server Application Update)	The object is updated whenever the server application using the embedded object indicates there is new data available.
OnCall (User Update)	The object is updated when the user chooses to update it.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_OLEUpdateMode](#)) is located in §A.6.2. *end note*]

19.2.3.22 ST_RType (Rule Type)

This simple type specifies types of rules.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
align (Alignment Rule)	Alignment rule.
arc (Arc Rule)	Arc rule.
callout (Callout Rule)	Callout rule.
connector (Connector Rule)	Connector rule.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_RType](#)) is located in §A.6.2. *end note*]

19.2.3.23 ST_ScreenSize (Screen Sizes Type)

This simple type specifies screen sizes.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
1024,768 (1024x768 pixels)	1024x768 pixels.
1152,862 (1152x862 pixels)	1152x862 pixels.
544,376 (544x376 pixels)	544x376 pixels.
640,480 (640x480 pixels)	640x480 pixels.
720,512 (720x512 pixels)	720x512 pixels.
800,600 (800x600 pixels)	800x600 pixels.

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ScreenSize](#)) is located in §A.6.2. *end note*]

19.3 VML - WordprocessingML Drawing

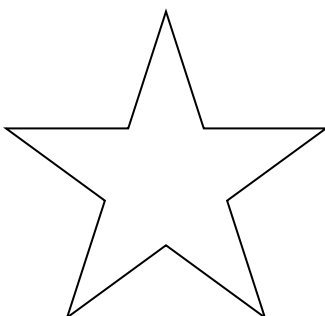
Within a WordprocessingML document, it is possible to include graphical VML objects. When these objects are present in a word processing document, it is necessary to include information about the object which is specific to their presence in a word processing document.

[*Note:* The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a transitional format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. *end note*]

The VML WordprocessingML Drawing namespace acts in this capacity, specifying all information necessary to anchor and display VML objects within a word processing document.

All elements defined in this subclause shall only appear in a WordprocessingML document.

[*Example:* Consider a 5-point star added to a WordprocessingML document, for example:



This object allows surrounding text to wrap around its top and bottom, but not to either side, so this interaction with the surrounding document text (which is specific to a word processing document) is stored in the WordprocessingML Drawing namespace as follows:

```
<v:shape ... >
...
  <wd:wrap wd:type="topAndBottom" />
</v:shape>
```

The wrap element specifies how surrounding WordprocessingML document content must wrap around the floating VML object - in this case, by wrapping to its top and bottom extents via the type attribute value of topAndBottom. *end example*

19.3.1 Table of Contents

This subclause is informative.

19.3.2 Elements	812
19.3.2.1 anchorlock (Anchor Location Is Locked)	812

19.3.2.2	borderbottom (Bottom Border)	812
19.3.2.3	borderleft (Left Border)	814
19.3.2.4	borderright (Right Border)	815
19.3.2.5	bordertop (Top Border)	816
19.3.2.6	wrap (Text Wrapping)	818
19.3.3	Simple Types	820
19.3.3.1	ST_BorderShadow (Border Shadow Type)	820
19.3.3.2	ST_BorderType (Border Type)	820
19.3.3.3	ST_HorizontalAnchor (Horizontal Anchor Type)	823
19.3.3.4	ST_VerticalAnchor (Vertical Anchor Type)	824
19.3.3.5	ST_WrapSide (Text Wrapping Side)	825
19.3.3.6	ST_WrapType (Text Wrapping Type)	825

End of informative text.

19.3.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:office:word namespace:

[*Note*: As the VML format is a format provided for backward compatibility, those VML elements defined in the same urn:schemas-microsoft-com:office:word namespace remain in that namespace as it is already used by millions of documents already using VML. *end note*]

19.3.2.1 anchorlock (Anchor Location Is Locked)

This element specifies that the anchor location for this object shall not be modified at runtime when an application edits the contents of this document. [*Guidance*: An application might have automatic behaviors which reposition the anchor for a VML object based on user interaction - for example, moving it from one page to another as needed. This element must tell applications not to perform any such behaviors. *end guidance*]

If this element is omitted, then the anchor shall not be locked for the parent VML object.

[*Example*: Consider a floating VML object which must have its anchor locked at the current location. This setting is specified as follows:

```
<wd:anchorLock/>
```

The anchorLock element's presence specifies that the VML object's current anchor location must not be changed by applications editing this content. *end example*].

[*Note*: The W3C XML Schema definition of this element's content model ([CT_AnchorLock](#)) is located in §A.6.3. *end note*]

19.3.2.2 borderbottom (Bottom Border)

This element specifies the properties for the bottom border of a VML object.

Attributes	Description
shadow (Border shadow)	<p>Specifies whether this border should be modified to create the appearance of a shadow.</p> <p>For the right and bottom borders, this is accomplished by duplicating the border below and right of the normal border location. For the left and top borders, this is accomplished by moving the border down and to the right of its original location.</p> <p>If this attribute is omitted, then the border is not given the shadow effect.</p> <p>[<i>Example:</i> Consider a top border which must appear with a shadow effect, resulting in the following content:</p> <pre><wd:bordertop wd:shadow="true" ... /></pre> <p>This element's shadow attribute is <code>true</code>, indicating that the shadow effect must be applied to the border. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_BorderShadow simple type (§19.3.3.1).</p>
type (Border Style)	<p>Specifies the style of border used on this object.</p> <p>See the simple type definition for a description of each border style.</p> <p>[<i>Example:</i> Consider a left border resulting in the following WordprocessingML:</p> <pre><wd:borderleft wd:type="single" .../></pre> <p>This border's type is <code>single</code>, indicating that the border style is a single line. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_BorderType simple type (§19.3.3.2).</p>
width (Border Width)	<p>Specifies the width of the current border.</p> <p>The width of this border is specified in measurements of eighths of a point, with a minimum value of two (one-fourth of a point) and a maximum value of 96 (twelve points). Any values outside this range can be reassigned to a more appropriate value.</p> <p>[<i>Example:</i> Consider a document with a three point wide dashed line border on all sides, resulting in the following WordprocessingML markup:</p> <pre><wd:bordertop wd:type="dashed" wd:width="24" .../> <wd:borderleft wd:type="dashed" wd:width="24" .../> <wd:borderbottom wd:type="dashed" wd:width="24" .../> <wd:boarderright wd:type="dashed" wd:width="24" .../></pre> <p>The width attribute specifies the size in eighths of a point (24 eighths of a point = 3 points). <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema <code>positiveInteger</code> datatype.

[Note: The W3C XML Schema definition of this element's content model ([CT_Border](#)) is located in §A.6.3. *end note*]

19.3.2.3 `borderleft` (Left Border)

This element represents the properties for the left border of a VML object.

Attributes	Description
shadow (Border shadow)	<p>Specifies whether this border should be modified to create the appearance of a shadow.</p> <p>For the right and bottom borders, this is accomplished by duplicating the border below and right of the normal border location. For the left and top borders, this is accomplished by moving the border down and to the right of its original location.</p> <p>If this attribute is omitted, then the border is not given the shadow effect.</p> <p>[Example: Consider a top border which must appear with a shadow effect, resulting in the following content:</p> <pre><wd:bordertop wd:shadow="true" ... /></pre> <p>This element's shadow attribute is <code>true</code>, indicating that the shadow effect must be applied to the border. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_BorderShadow</code> simple type (§19.3.3.1).</p>
type (Border Style)	<p>Specifies the style of border used on this object.</p> <p>See the simple type definition for a description of each border style.</p> <p>[Example: Consider a left border resulting in the following WordprocessingML:</p> <pre><wd:borderleft wd:type="single" .../></pre> <p>This border's type is <code>single</code>, indicating that the border style is a single line. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_BorderType</code> simple type (§19.3.3.2).</p>
width (Border Width)	Specifies the width of the current border.

Attributes	Description
	<p>The width of this border is specified in measurements of eighths of a point, with a minimum value of two (one-fourth of a point) and a maximum value of 96 (twelve points). Any values outside this range can be reassigned to a more appropriate value.</p> <p>[<i>Example:</i> Consider a document with a three point wide dashed line border on all sides, resulting in the following WordprocessingML markup:</p> <pre><wd:bordertop wd:type="dashed" wd:width="24" .../> <wd:borderleft wd:type="dashed" wd:width="24" .../> <wd:borderbottom wd:type="dashed" wd:width="24" .../> <wd:borderright wd:type="dashed" wd:width="24" .../></pre> <p>The width attribute specifies the size in eighths of a point (24 eighths of a point = 3 points). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema positiveInteger datatype.</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Border](#)) is located in §A.6.3. *end note*]

19.3.2.4 [borderright \(Right Border\)](#)

This element specifies the properties for the right border of a VML object.

Attributes	Description
shadow (Border shadow)	<p>Specifies whether this border should be modified to create the appearance of a shadow.</p> <p>For the right and bottom borders, this is accomplished by duplicating the border below and right of the normal border location. For the left and top borders, this is accomplished by moving the border down and to the right of its original location.</p> <p>If this attribute is omitted, then the border is not given the shadow effect.</p> <p>[<i>Example:</i> Consider a top border which must appear with a shadow effect, resulting in the following content:</p> <pre><wd:bordertop wd:shadow="true" ... /></pre> <p>This element's shadow attribute is true, indicating that the shadow effect must be applied to the border. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_BorderShadow simple type (§19.3.3.1).</p>
type (Border Style)	Specifies the style of border used on this object.

Attributes	Description
	<p>See the simple type definition for a description of each border style.</p> <p>[<i>Example:</i> Consider a left border resulting in the following WordprocessingML:</p> <pre data-bbox="456 426 1016 457"><wd:borderleft wd:type="single" .../></pre> <p>This border's type is <code>single</code>, indicating that the border style is a single line. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the <code>ST_BorderType</code> simple type (§19.3.3.2).</p>
width (Border Width)	<p>Specifies the width of the current border.</p> <p>The width of this border is specified in measurements of eighths of a point, with a minimum value of two (one-fourth of a point) and a maximum value of 96 (twelve points). Any values outside this range can be reassigned to a more appropriate value.</p> <p>[<i>Example:</i> Consider a document with a three point wide dashed line border on all sides, resulting in the following WordprocessingML markup:</p> <pre data-bbox="456 972 1273 1108"><wd:bordertop wd:type="dashed" wd:width="24" .../> <wd:borderleft wd:type="dashed" wd:width="24" .../> <wd:borderbottom wd:type="dashed" wd:width="24" .../> <wd:boarderright wd:type="dashed" wd:width="24" .../></pre> <p>The width attribute specifies the size in eighths of a point (24 eighths of a point = 3 points). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema <code>positiveInteger</code> datatype.</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Border](#)) is located in §A.6.3. *end note*]

19.3.2.5 [bordertop \(Top Border\)](#)

This element specifies the properties for the top border of a VML object.

Attributes	Description
shadow (Border shadow)	<p>Specifies whether this border should be modified to create the appearance of a shadow.</p> <p>For the right and bottom borders, this is accomplished by duplicating the border below and right of the normal border location. For the left and top borders, this is accomplished by moving the border down and to the right of its original location.</p>

Attributes	Description
	<p>If this attribute is omitted, then the border is not given the shadow effect.</p> <p>[<i>Example</i>: Consider a top border which must appear with a shadow effect, resulting in the following content:</p> <pre><wd:bordertop wd:shadow="true" ... /></pre> <p>This element's shadow attribute is <code>true</code>, indicating that the shadow effect must be applied to the border. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_BorderShadow simple type (§19.3.3.1).</p>
type (Border Style)	<p>Specifies the style of border used on this object.</p> <p>See the simple type definition for a description of each border style.</p> <p>[<i>Example</i>: Consider a left border resulting in the following WordprocessingML:</p> <pre><wd:borderleft wd:type="single" .../></pre> <p>This border's type is <code>single</code>, indicating that the border style is a single line. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_BorderType simple type (§19.3.3.2).</p>
width (Border Width)	<p>Specifies the width of the current border.</p> <p>The width of this border is specified in measurements of eighths of a point, with a minimum value of two (one-fourth of a point) and a maximum value of 96 (twelve points). Any values outside this range can be reassigned to a more appropriate value.</p> <p>[<i>Example</i>: Consider a document with a three point wide dashed line border on all sides, resulting in the following WordprocessingML markup:</p> <pre><wd:bordertop wd:type="dashed" wd:width="24" .../> <wd:borderleft wd:type="dashed" wd:width="24" .../> <wd:borderbottom wd:type="dashed" wd:width="24" .../> <wd:borderright wd:type="dashed" wd:width="24" .../></pre> <p>The width attribute specifies the size in eighths of a point (24 eighths of a point = 3 points). <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema <code>positiveInteger</code> datatype.</p>

[*Note*: The W3C XML Schema definition of this element's content model ([CT_Border](#)) is located in §A.6.3. *end note*]

19.3.2.6 wrap (Text Wrapping)

This element specifies the type of text wrapping which should be allowed around the contents of this VML object.

If this element is omitted, then no text wrapping shall be performed (i.e. the object shall be presented in line with text).

[*Example*: Consider the following VML object:

```
<v:shape ... >
...
  <wd:wrap wd:type="square" />
</v:shape>
```

The wrap element specifies how surrounding WordprocessingML document content must wrap around the floating VML object - in this case, by wrapping around its extents in a square via the type attribute value of square. *end example*].

Attributes	Description
anchorx (Horizontal Positioning Base)	<p>Specifies the base object from which the horizontal positioning of the object should be calculated.</p> <p>A VML object can be horizontally positioned relative to:</p> <ul style="list-style-type: none"> • The vertical edge of the page before any runs of text (the left edge for left-to-right paragraphs, the right edge for right-to-left paragraphs) • The vertical edge of the text margin before any runs of text (the left edge for left-to-right paragraphs, the right edge for right-to-left paragraphs) • The vertical edge of the text in the paragraph containing the VML object • The position of anchor for the floating VML object in the text. <p>If this attribute is omitted, then its value shall be assumed to be page.</p> <p>[<i>Example</i>: Consider a VML object which should be positioned relative to the page edges, which is specified as follows:</p> <pre><wd:wrap wd:anchorx="page" wd:anchory="page" /></pre> <p>The anchorx attribute specifies that horizontal anchoring is relative to the edge of the page. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_HorizontalAnchor simple type (§19.3.3.3).</p>
anchory (Vertical)	Specifies the base object from which the vertical positioning of the object should be

Attributes	Description
Positioning Base)	<p>calculated.</p> <p>A VML object can be vertically positioned relative to:</p> <ul style="list-style-type: none"> • The horizontal top edge of the page • The horizontal edge of the top text margin before any runs of text • The horizontal top edge of line containing the VML object • The horizontal top edge of the paragraph containing the text. <p>If this attribute is omitted, then its value shall be assumed to be page.</p> <p>[<i>Example:</i> Consider a VML object which should be positioned relative to the page edges, which is specified as follows:</p> <pre><wd:wrap wd:anchorx="page" wd:anchory="page" /></pre> <p>The anchory attribute specifies that horizontal anchoring is relative to the edge of the page. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_VerticalAnchor simple type (§19.3.3.4).</p>
side (Wrapping side)	<p>Specifies how text shall wrap around the object's left and right sides.</p> <p>[<i>Example:</i> Consider a floating DrawingML object which must allow text to wrap around its left side only. This setting is specified as follows:</p> <pre><wd:wrap side="left" ... /></pre> <p>The side attribute value of left specifies that text must only wrap around the left side of the object. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_WrapSide simple type (§19.3.3.5).</p>
type (Wrapping type)	<p>Specifies the type of wrapping - see the simple type definition for a description of each type.</p> <p>[<i>Example:</i> Consider the following VML object:</p> <pre><v:shape ... > ... <wd:wrap wd:type="topAndBottom" /> </v:shape></pre> <p>The wrap element specifies how surrounding WordprocessingML document content must wrap around the floating VML object - in this case, by wrapping around its top and bottom extents via the type attribute value of topAndBottom. <i>end example</i>]</p>

Attributes	Description
	The possible values for this attribute are defined by the ST_WrapType simple type (§19.3.3.6).

[Note: The W3C XML Schema definition of this element's content model ([CT_Wrap](#)) is located in §A.6.3. *end note*]

19.3.3 Simple Types

The following additional simple type information in the urn:schemas-microsoft-com:office:word namespace is used for documents of a transitional conformance class.

19.3.3.1 ST_BorderShadow (Border Shadow Type)

This simple type specifies logical true and false values.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
f (False)	Logical false.
false (False)	Logical false.
t (True)	Logical true.
true (True)	Logical true.



[Note: The W3C XML Schema definition of this simple type's content model ([ST_BorderShadow](#)) is located in §A.6.3. *end note*]



19.3.3.2 ST_BorderType (Border Type)


This type defines which types of borders are supported.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
dash (pecifies a line border consisting of a dashed line around the parent object.)	Specifies a line border consisting of a dashed line around the parent object.
dashDotDot (Dash Dot Dot Border)	Specifies a line border consisting of a alternating dotted, dotted, dashed line around the parent object.
dashDotStroked (Stroked Dash Dot Border)	Specifies a line border consisting of a line with a series of alternating thin and thick strokes around the parent object.
dashedSmall (Small Dash Border)	Specifies a line border consisting of a dashed line with

Enumeration Value	Description
	small gaps around the parent object.
dot (Dotted Border)	Specifies a line border consisting of a dotted line around the parent object.
dotDash (Dot Dash Border)	Specifies a line border consisting of a alternating dotted and dashed line around the parent object.
double (Double Line Border)	Specifies a line border consisting of a double line around the parent object.
doubleWave (Double Wavy Lines Border)	Specifies a line border consisting of a double wavy line around the parent object.
hairline (Hairline Border)	Specifies a line border consisting of a very thin line.
HTMLInset (Inset Border)	<p>Specifies a line border consisting of an inset set of lines around the parent object.</p> <p>[Example:</p>  <p>end example]</p>
HTMLOutset (Outset Border)	<p>Specifies a line border consisting of an outset set of lines around the parent object.</p> <p>[Example:</p>  <p>end example]</p>
none (No Border)	Specifies that no border shall be applied to the current item.
single (Single Line Border)	Specifies a line border consisting of a single line around the parent object.
thick (Thick Line Border)	Specifies a line border consisting of a single line around the parent object.
thickBetweenThin (Thin-thick-thin Border)	Specifies a line border consisting of a thick line contained within a thin line with a medium sized intermediate gap around the parent object.
thickBetweenThinLarge (Large thin-thick-thin Border)	Specifies a line border consisting of a thin line contained within a thick line, contained within a thin line with a medium sized intermediate gap around the parent object.
thickBetweenThinSmall (Small thin-thick-thin Lines Border)	Specifies a line border consisting of a thin line contained within a thick line, contained within a thin

Enumeration Value	Description
	line with a small intermediate gap around the parent object.
thickThin (Thick Thin Line Border)	Specifies a line border consisting of a thick line contained within a thin line with a medium sized intermediate gap around the parent object.
thickThinLarge (Thick Thin Large Gap Border)	Specifies a line border consisting of a thick line contained within a thin line with a large sized intermediate gap around the parent object.
thickThinSmall (Small thick-thin lines border)	Specifies a line border consisting of a thick line contained within a thin line with a small intermediate gap around the parent object.
thinThick (Thin Thick Line Border)	Specifies a line border consisting of a thin line contained within a thick line contained within a thick thin with a medium sized intermediate gap between each around the parent object.
thinThickLarge (Thin Thick Large Gap Border)	Specifies a line border consisting of a thin line contained within a thick line contained within a thick thin with a large sized intermediate gap between each around the parent object.
thinThickSmall (Thin Thick Small Gap Border)	Specifies a line border consisting of a thin line contained within a thick line contained within a thick thin with a small intermediate gap between each around the parent object.
threeDEmboss (3D Embossed Border)	<p>Specifies a line border consisting of three staged gradient lines around the parent object, getting darker towards the object.</p> <p>[Example:</p>  <p>end example]</p>
threeDEngrave (3D Engraved Border)	<p>Specifies a line border consisting of three staged gradient lines around the parent object, getting darker away from the object.</p> <p>[Example:</p>  <p>end example]</p>
triple (Triple Line Border)	Specifies a line border consisting of a triple line around

Enumeration Value	Description
	the parent object.
wave (Wavy Border)	<p>Specifies a line border consisting of a wavy line around the parent object.</p> <p>[Example:</p>  <p>end example]</p>

[Note: The W3C XML Schema definition of this simple type's content model ([ST_BorderType](#)) is located in §A.6.3.
end note]

19.3.3.3 ST_HorizontalAnchor (Horizontal Anchor Type)

This simple type specifies the horizontal position to which the parent object has been anchored in the document. This anchor position shall be used as the base location to determine the final horizontal position of the object in the document.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
char (Character)	Specifies that the parent object shall be horizontally anchored based on the position of the anchor within the text flow.
margin (Margin)	<p>Specifies that the parent object shall be horizontally anchored to the text margins.</p> <p>This shall be used to specify that any horizontal positioning values shall be calculated with respect to the location of the text margin.</p>
page (Page)	<p>Specifies that the parent object shall be horizontally anchored to the page edge.</p> <p>This shall be used to specify that any horizontal positioning values shall be calculated with respect to the location of the edge of the page.</p>
text (Text)	<p>Specifies that the parent object shall be horizontally anchored to the text extents.</p> <p>This shall be used to specify that any horizontal positioning values shall be calculated with respect to the location of the edge of the text in the anchor</p>

Enumeration Value	Description
	paragraph (including text indentations on that paragraph within the text margins).

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_HorizontalAnchor](#)) is located in §A.6.3. *end note*]

19.3.3.4 ST_VerticalAnchor (Vertical Anchor Type)

This simple type specifies the vertical position to which the parent object has been anchored in the document. This anchor position shall be used as the base location to determine the final vertical position of the object in the document.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
line (Line)	<p>Specifies that the parent object shall be vertically anchored to the line on which its anchor appears.</p> <p>This shall be used to specify that any vertical positioning values shall be calculated with respect to the location of the top edge of the anchor's line in the anchor paragraph.</p>
margin (Margin)	<p>Specifies that the parent object shall be vertically anchored to the text margins.</p> <p>This shall be used to specify that any vertical positioning values shall be calculated with respect to the location of the text margin.</p>
page (Page)	<p>Specifies that the parent object shall be vertically anchored to the page edge.</p> <p>This shall be used to specify that any vertical positioning values shall be calculated with respect to the location of the edge of the page.</p>
text (Text)	<p>Specifies that the parent object shall be vertically anchored to the text extents.</p> <p>This shall be used to specify that any vertical positioning values shall be calculated with respect to the location of the top edge of the text in the anchor paragraph.</p>

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_VerticalAnchor](#)) is located in §A.6.3. end note*]

19.3.3.5 [ST_WrapSide \(Text Wrapping Side\)](#)

This simple type defines which sides text can wrap around a VML object.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
both (Both sides)	Wrap text on both sides.
largest (Largest side)	Wrap text on largest side.
left (Left side)	Wrap text on left side.
right (Right side)	Wrap text on right side.

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_WrapSide](#)) is located in §A.6.3. end note*]

19.3.3.6 [ST_WrapType \(Text Wrapping Type\)](#)

This simple type specifies the type of text wrapping which shall be allowed around a VML object within a document.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
none (No wrapping)	Specifies that text shall not be allowed to wrap around the remaining space on each lines around this VML object.
square (Square wrapping)	Specifies that text shall be allowed to wrap around the remaining space on each line around this text frame in the document using a rectangle touching each of the object's furthest edges.
through (Through wrapping)	Specifies that text shall be allowed to wrap around the remaining space on each line around this text frame in the document, including any holes in the object.
tight (Tight wrapping)	Specifies that text shall be allowed to tightly wrap around the remaining space on each line around this text frame in the document.
topAndBottom (Top and bottom wrapping)	Specifies that text shall not be allowed to wrap around the remaining space on each lines around the VML object. Any text content shall therefore be placed on the next

Enumeration Value	Description
	line following the object which does not intersect with the object's extents.

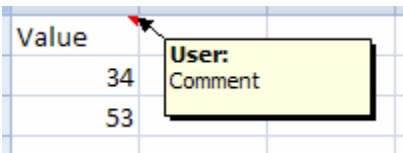
[Note: The W3C XML Schema definition of this simple type’s content model ([ST WrapType](#)) is located in §A.6.3. end note]

19.4 VML - SpreadsheetML Drawing

It is possible to attach user interface controls, such as comments, combo boxes (dropdowns) and embedded controls, to a SpreadsheetML document. VML is used to define certain aspects of the control, such as size and visual appearance. Additional information describing the control shall also be included. The VML SpreadsheetML Drawing namespace provides the additional information necessary to define the object type, settings and behavior for the control.

[Note: The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a transitional format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. end note]

[Example: Assume the comment below exists on a spreadsheet:



The following defines the additional information necessary to describe the comment. The ObjectType attribute describes the object as a comment. The Anchor element defines that its edges are anchored to the first and fourth rows and the second and fourth columns. The Row and Column elements indicate that it points to the cell in the first row, first column.

```
<x:ClientData ObjectType="Note">
  <x:MoveWithCells/>
  <x:SizeWithCells/>
  <x:Anchor>1, 13, 0, 12, 2, 52, 2, 10</x:Anchor>
  <x:AutoFill>False</x:AutoFill>
  <x:Row>0</x:Row>
  <x:Column>0</x:Column>
  <x:Visible/>
</x:ClientData>
```

This additional comment data exists inside the VML shape that defines the comment object:

```

<v:shape id="_x0000_s1025" type="#_x0000_t202" style='position:absolute;margin
left:57.75pt;margin-top:9pt;width:77.25pt;height:28.5pt;z-index:1;mso-wrap-
style:tight' fillcolor="#ffffe1" o:insetmode="auto">
  <v:fill color2="#ffffe1"/>
  <v:shadow on="t" color="black" obscured="t"/>
  <v:path o:connecttype="none"/>
  <v:textbox style='mso-direction-alt:auto'>
    <div style='text-align:left'></div>
  </v:textbox>
  <x:ClientData ObjectType="Note"> ... </x:ClientData>
</v:shape>

```

end example]

19.4.1 Table of Contents

This subclause is informative.

19.4.2	Elements	829
19.4.2.1	Accel (Primary Keyboard Accelerator)	829
19.4.2.2	Accel2 (Secondary Keyboard Accelerator)	829
19.4.2.3	Anchor (Anchor)	829
19.4.2.4	AutoFill (AutoFill)	830
19.4.2.5	AutoLine (AutoLine)	831
19.4.2.6	AutoPict (Automatically Size)	831
19.4.2.7	AutoScale (Font AutoScale)	832
19.4.2.8	Camera (Camera Tool)	832
19.4.2.9	Cancel (Cancel Button)	833
19.4.2.10	CF (Clipboard Format)	833
19.4.2.11	Checked (Checked)	833
19.4.2.12	ClientData (Attached Object Data)	834
19.4.2.13	ColHidden (Comment's Column is Hidden)	835
19.4.2.14	Colored (Dropdown Color Toggle)	835
19.4.2.15	Column (Comment Column Target)	836
19.4.2.16	DDE (Dynamic Data Exchange)	836
19.4.2.17	Default (Default Button)	836
19.4.2.18	DefaultSize (Default Size Toggle)	837
19.4.2.19	Disabled (Macro Disable Toggle)	837
19.4.2.20	Dismiss (Dismiss Button)	838
19.4.2.21	DropLines (Dropdown Maximum Lines)	838
19.4.2.22	DropStyle (Dropdown Style)	838
19.4.2.23	Dx (Scroll Bar Width)	839
19.4.2.24	FirstButton (First Radio Button)	839
19.4.2.25	FmlaGroup (Linked Formula - Group Box)	839
19.4.2.26	FmlaLink (Linked Formula)	840
19.4.2.27	FmlaMacro (Reference to Custom Function)	840
19.4.2.28	FmlaPict (Camera Source Range)	841

19.4.2.29	FmlaRange (List Items Source Range)	841
19.4.2.30	FmlaTxbx (Text Formula)	841
19.4.2.31	Help (Help Button)	841
19.4.2.32	Horiz (Scroll Bar Orientation)	842
19.4.2.33	Inc (Scroll Bar Increment)	842
19.4.2.34	JustLastX (Far East Alignment Toggle)	842
19.4.2.35	LCT (Callback Type)	843
19.4.2.36	ListItem (Non-linked List Item)	843
19.4.2.37	Locked (Lock Toggle)	843
19.4.2.38	LockText (Text Lock)	844
19.4.2.39	MapOCX (Embedded Control)	844
19.4.2.40	Max (Scroll Bar Maximum)	845
19.4.2.41	Min (Scroll Bar Minimum)	845
19.4.2.42	MoveWithCells (Move with Cells)	845
19.4.2.43	MultiLine (Multi-line)	846
19.4.2.44	MultiSel (Multiple Selections)	846
19.4.2.45	NoThreeD (Disable 3D)	846
19.4.2.46	NoThreeD2 (Disable 3D)	847
19.4.2.47	Page (Scroll Bar Page Increment)	847
19.4.2.48	PrintObject (Print Toggle)	847
19.4.2.49	RecalcAlways (Recalculation Toggle)	848
19.4.2.50	Row (Comment Row Target)	848
19.4.2.51	RowHidden (Comment's Row is Hidden)	848
19.4.2.52	ScriptExtended (HTML Script Attributes)	849
19.4.2.53	ScriptLanguage (HTML Script Language)	849
19.4.2.54	ScriptLocation (HTML Script Location)	850
19.4.2.55	ScriptText (HTML Script Text)	850
19.4.2.56	SecretEdit (Password Edit)	850
19.4.2.57	Sel (Selected Entry)	851
19.4.2.58	SelType (Selection Type)	851
19.4.2.59	SizeWithCells (Resize with Cells)	851
19.4.2.60	TextHAlign (Horizontal Text Alignment)	852
19.4.2.61	TextVAlign (Vertical Text Alignment)	852
19.4.2.62	UIObj (UI Object Toggle)	852
19.4.2.63	Val (Scroll bar position)	853
19.4.2.64	ValidIds (Valid ID)	853
19.4.2.65	Visible (Comment Visibility Toggle)	853
19.4.2.66	VScroll (Vertical Scroll)	854
19.4.2.67	VTEdit (Validation Type)	854
19.4.2.68	WidthMin (Minimum Width)	855
19.4.3	Simple Types	855
19.4.3.1	ST_CF (Clipboard Format Type)	855
19.4.3.2	ST_ObjectType (Object Type)	856

End of informative text.

19.4.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:office:excel namespace:

[*Note:* As the VML format is a format provided for backward compatibility, those VML elements defined in the same urn:schemas-microsoft-com:office:excel namespace remain in that namespace as it is already used by millions of documents already using VML. *end note*]

19.4.2.1 Accel (Primary Keyboard Accelerator)

This element specifies the primary keyboard accelerator for an object. The value is the decimal value of the Unicode character corresponding to the accelerator key. This element is used for buttons, checkboxes, radio buttons and group boxes.

[*Example:* The primary accelerator key is 'A' (65 is the decimal value for 'A' (U+0041))]:

```
<x:ClientData ... >
  <x:Accel>65</x:Accel>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.2 Accel2 (Secondary Keyboard Accelerator)

This element specifies the secondary keyboard accelerator for an object. The value is the decimal value of the Unicode character corresponding to the accelerator key. This element is used for buttons, checkboxes, radio buttons and group boxes.

[*Example:* The secondary accelerator key is 'A' (65 is the decimal value for 'A' (U+0041))]:

```
<x:ClientData>
  <x:Accel2>65</x:Accel2>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.3 Anchor (Anchor)

This element specifies the anchor location for the object. This is a general-use element.

The value is a comma-separated list of data written out as: LeftColumn, LeftOffset, TopRow, TopOffset, RightColumn, RightOffset, BottomRow, BottomOffset.

Value	Description
LeftColumn	The left anchor column of the object (left-most column is 0). [<i>Example:</i>

Value	Description
	An object whose left anchor was off of the third column has a LeftColumn value of 2. <i>end example</i>
LeftOffset	The offset of the object's left edge from the left edge of the left anchor column. This value is measured in pixels.
TopRow	The top anchor row of the object (top-most column is 0). [<i>Example</i> : An object whose top anchor was off of the fifth row has a TopRow value of 4. <i>end example</i>]
TopOffset	The offset of the object's top edge from the top edge of the top anchor row. This value is measured in pixels.
RightColumn	The right anchor column of the object (left-most column is 0). [<i>Example</i> : An object whose right anchor was off of the tenth column has a RightColumn value of 9. <i>end example</i>]
RightOffset	The offset of the object's right edge from the left edge of the right anchor column. This value is measured in pixels.
BottomRow	The bottom anchor row of the object (top-most column is 0). [<i>Example</i> : An object whose bottom anchor was off of the tenth row has a BottomRow value of 9. <i>end example</i>]
BottomOffset	The offset of the object's bottom edge from the bottom edge of the bottom anchor row. This value is measured in pixels.

[*Example*: The left side of the object is 15 pixels to the right of the left edge of the second column. The top edge is 2 pixels below the upper edge of the first row. The right side is 15 pixels to the right of the left edge of the fourth column. The bottom edge is 16 pixels below the top of the fourth row.]

```
<x:ClientData>
  <x:Anchor>1, 15, 0, 2, 3, 15, 3, 16</x:Anchor>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.4 AutoFill (AutoFill)

This element specifies that the object's fill properties are automatically provided by the application and are not overridden with a specific fill color or style. [*Rationale*: An application can choose to display objects with certain visual properties that are appropriate to the application environment. *end rationale*] If this element is specified without a value, it is assumed to be true. This is a general-use element.

[*Example*:


```
<x:ClientData> ...
  <x:AutoFill>False</x:AutoFill>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.5 AutoLine (AutoLine)

This element specifies that the object's line properties are automatically provided by the application and are not overridden with a specific line color, style, or width. [*Rationale:* An application can choose to display objects with certain visual properties that are appropriate to the application environment. *end rationale]* If this element is specified without a value, it is assumed to be true. This is a general-use element.

[*Example:*

```
<x:ClientData> ...
  <x:AutoLine>False</x:AutoLine>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.6 AutoPict (Automatically Size)

This element specifies whether the object's aspect ratio is locked when rendered in different views by the application. If this element is specified without a value, it is assumed to be true. This is a general-use element for objects that use an image representation, denoted by the Pict value of ST_ObjectType. These objects are: embedded objects, embedded controls, cameras and signature lines.

[*Example:*

```
<x:ClientData> ...
  <x:AutoPict>True</x:AutoPict>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.7 AutoScale (Font AutoScale)

This element specifies whether the object's font is automatically scaled by the application when the object is resized. If this element is specified without a value, it is assumed to be true. This element is used for attached text. Attached text refers to a class of objects that have text associated with them. The following values defined by the ST_ObjectType simple type are attached text objects: Button, Checkbox, Dialog, Edit, GBox, Label, Note and Radio.

[Example:

```
<x:ClientData> ...  
  <x:AutoScale>True</x:AutoScale>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

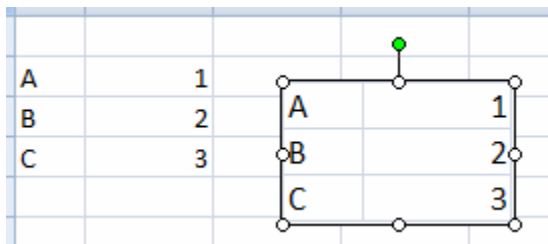
19.4.2.8 Camera (Camera Tool)

This element specifies that the object is a camera object. A camera object is a shape that is filled with a live view of a cell range in the same spreadsheet, including all applied styles. The cell range is defined by the fmlaPict element (§19.4.2.28), which shall be present. Shape properties such as the position and size of the camera object are defined by the shape. The shape shall be a rectangle. The view of the cell range is scaled vertically and horizontally to fill the rectangle exactly.

If this element is specified without a value, it is assumed to be true.

[Example:

```
<x:ClientData> ...  
  <x:FmlaPict>$A$2:$B$4</x:FmlaPict>  
  <x:Camera>True</x:Camera>  
</x:ClientData>
```



end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.4.2.9 Cancel (Cancel Button)

This element specifies that the object is a cancel button. If this element is specified without a value, it is assumed to be true. This element is used for buttons.

[*Example:*

```
<x:ClientData> ...
  <x:Cancel/>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.4.2.10 CF (Clipboard Format)

This element specifies the clipboard format used to render the object. This is a general-use element for objects that use an image representation, such as embedded objects, embedded controls, cameras and signature lines.

[*Example:*

```
<x:ClientData> ...
  <x:CF>Pict</x:CF>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_CF simple type (§19.4.3.1).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_CF](#)) is located in §A.6.4. *end note*]

19.4.2.11 Checked (Checked)

This element specifies that the checkbox is checked or the radio button is selected. This element is used for checkboxes and radio buttons. Permitted values are:

Value	Description
0	Unchecked / unselected

Value	Description
1	Checked / selected
2	Mixed selection

[Example:

```
<x:ClientData> ...
  <x:Checked>2</x:Checked>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.12 ClientData (Attached Object Data)

This element specifies data associated with objects attached to a spreadsheet. While this element might contain any of the child elements below, only certain combinations are meaningful. The ObjectType attribute determines the kind of object the element represents and which subset of child elements is appropriate. Relevant groups are identified for each child element.

[Example: The following defines additional information for a comment. Its edges are anchored to the first and fourth rows and the second and fourth columns. It points to the cell in the first row, first column.

```
<x:ClientData ObjectType="Note">
  <x:MoveWithCells/>
  <x:SizeWithCells/>
  <x:Anchor>1, 15, 0, 2, 3, 15, 3, 16</x:Anchor>
  <x:AutoFill>False</x:AutoFill>
  <x:Row>0</x:Row>
  <x:Column>0</x:Column>
  <x:Visible/>
</x:ClientData>
```

end example]

[Example: The following defines additional information for a radio button. It is the first in a series of radio buttons and selected by default. The accelerator key is 'A' (65 is the decimal value for 'A' (U+0041)) and it is linked to the cell at column A, row 1 of the first sheet.

```

<x:ClientData ObjectType=3D"Radio">
  <x:SizeWithCells/>
  <x:AutoFill>False</x:AutoFill>
  <x:AutoLine>False</x:AutoLine>
  <x:TextVAlign>Center</x:TextVAlign>
  <x:Checked>1</x:Checked>
  <x:Accel>65</x:Accel>
  <x:FmlaLink>Sheet1!$A$1</x:FmlaLink>
  <x:FirstButton/>
</x:ClientData>

```

end example]

Attributes	Description
ObjectType (Object type)	<ul style="list-style-type: none"> Specifies the kind of the object. Different sets of child elements are appropriate for different types of objects. <p>The possible values for this attribute are defined by the ST_ObjectType simple type (§19.4.3.2).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_ClientData](#)) is located in §A.6.4. *end note*]

19.4.2.13 ColHidden (Comment's Column is Hidden)

This element specifies that the column of the cell to which this comment points is hidden. If this element is specified without a value, it is assumed to be true. This element is used for comments.

[Example:

```

<x:ClientData> ...
  <x:ColHidden>True</x:ColHidden>
</x:ClientData>

```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.4.2.14 Colored (Dropdown Color Toggle)

This element specifies that the dropdown is colored. If this element is specified without a value, it is assumed to be true. This element is used for dropdowns.

[Example:

```
<x:ClientData> ...
  <x:Colored>True</x:Colored>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.15 Column (Comment Column Target)

This element specifies the column a comment points to. The column index is 0-based. This element is used for comments.

[Example:

```
<x:ClientData> ...
  <x:Column>0</x:Column>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.16 DDE (Dynamic Data Exchange)

This element specifies that the object is a DDE (Dynamic Data Exchange) link. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:DDE>True</x:DDE>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.17 Default (Default Button)

This element specifies that the object is a default (OK) button . If this element is specified without a value, it is assumed to be true. This element is used for buttons.

[Example:

```
<x:ClientData> ...
  <x:Default>True</x:Default>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.18 DefaultSize (Default Size Toggle)

This element specifies that the object is at its default size. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[*Example:*

```
<x:ClientData> ...
  <x:DefaultSize>True</x:DefaultSize>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.19 Disabled (Macro Disable Toggle)

This element specifies that the object cannot run an attached macro. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[*Example:*

```
<x:ClientData> ...
  <x:Disabled>True</x:Disabled>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.20 Dismiss (Dismiss Button)

This element specifies that the object is a dismiss button. If this element is specified without a value, it is assumed to be true. This element is used for buttons.

[Example:

```
<x:ClientData> ...  
  <x:Dismiss>True</x:Dismiss>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.21 DropLines (Dropdown Maximum Lines)

This element specifies the maximum number of lines in the dropdown before scrollbars are added. This element is used for dropdowns.

If this element is omitted, one line is shown.

[Example:

```
<x:ClientData> ...  
  <x:DropLines>8</x:DropLines>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.22 DropStyle (Dropdown Style)

This element specifies the style of the dropdown. Allowed values are:

Value	Description
Combo	Standard combo box
ComboEdit	Editable combo box
Simple	Standard combo box with only the dropdown button visible when the box is not expanded

This element is used for dropdowns.

[Example:


```
<x:ClientData> ...
  <x:DropStyle>Combo</x:DropStyle>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.23 Dx (Scroll Bar Width)

This element specifies the width of the scroll bar in screen pixels. This element is used for scroll bars and spinners. [Note: It is possible for other controls, such as combo boxes and list boxes, to use scroll bars and this element is permitted for those controls. *end note*]

[Example:

```
<x:ClientData> ...
  <x:Dx>16</x:Dx>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.24 FirstButton (First Radio Button)

This element specifies that the object is the first radio button in a set of radio buttons. If this element is specified without a value, it is assumed to be true. This element is used for radio buttons.

[Example:

```
<x:ClientData> ...
  <x:FirstButton>True</x:FirstButton>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.4.2.25 FmlaGroup (Linked Formula - Group Box)

This element specifies the cell the object is linked to, using standard cell reference syntax. This element is used for group boxes. This overrides the FmlaLink for any radio buttons enclosed in the group box. The value in the linked cell and the index of the selected radio button are linked together. The formula syntax is described in Part 1, §18.17 of the SpreadsheetML reference.

[Example:

```
<x:ClientData> ...
  <x:FmlaGroup>$A$1</x:FmlaGroup>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.26 FmlaLink (Linked Formula)

This element specifies the cell the object is linked to, using standard cell reference syntax. This element is used for checkboxes, radio buttons, scroll bars, spinners, dropdowns and list boxes. The value in the linked cell and the index of the selected item in the object are linked together. This link is ignored if the control allows multiple selections. The formula syntax is described in Part 1, §18.17 of the SpreadsheetML reference.

[Example:

```
<x:ClientData> ...
  <x:FmlaLink>$A$4</x:FmlaLink>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.27 FmlaMacro (Reference to Custom Function)

This element specifies the custom function associated with the object. [Example: A macro script, add-in function, and so on. *end example]*

This element applies to objects defined by all values of the ST_ObjectType simple type, except: LineA, Note, RectA.

The format of this string shall be application-defined, and should be ignored if not understood.

[Example:

```
<x:ClientData> ...
  <x:FmlaMacro>Button1_Click()</x:FmlaMacro>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.28 FmlaPict (Camera Source Range)

This element specifies the range of source data cells visible in the camera object (§19.4.2.8). This element is used for cameras. The formula syntax is described in Part 1, §18.17 of the SpreadsheetML reference.

This element is ignored if the Camera element is absent.

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.29 FmlaRange (List Items Source Range)

This element specifies the range of source data cells used to populate the list box, using standard cell reference syntax. This element is used for list boxes. The formula syntax is described in Part 1, §18.17 of the SpreadsheetML reference.

[Example:

```
<x:ClientData> ...
  <x:FmlaRange>$A$1:$A$15</x:FmlaRange>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.30 FmlaTxbx (Text Formula)

This element defines the formula associated with the object's text. This element is used for attached text.

[Example:

```
<x:ClientData> ...
  <x:FmlaTxbx>$D$9</x:FmlaTxbx>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.31 Help (Help Button)

This element specifies that the object is a help button. If this element is specified without a value, it is assumed to be true. This element is used for buttons.

[Example:

```
<x:ClientData> ...
  <x:Help>True</x:Help>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.32 Horiz (Scroll Bar Orientation)

This element specifies that the scroll bar is horizontal. If omitted, the scroll bar is vertical. If this element is specified without a value, it is assumed to be true. This element is used for scroll bars and spinners.

[*Example:*

```
<x:ClientData> ...
  <x:Horiz>True</x:Horiz>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.33 Inc (Scroll Bar Increment)

This element specifies the number of lines to move the scroll bar on an increment click. If omitted, the increment is 0. This element is used for scroll bars and spinners.

[*Example:*

```
<x:ClientData> ...
  <x:Inc>1</x:Inc>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.34 JustLastX (Far East Alignment Toggle)

This element specifies that Far East alignment is set for the last line in the text. Typically, justified text in Far East environments leaves the last line unjustified. Specifying this element also justifies the last line. If this element is specified without a value, it is assumed to be true. This element is used for attached text.

[*Example:*

```
<x:ClientData> ...
  <x:JustLastX>True</x:JustLastX>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.35 LCT (Callback Type)

This element specifies the kind of list box callback. The application should use the callback to determine how to handle user actions on the list box. The only allowed value is Normal. This element is used for list boxes.

[*Example:*

```
<x:ClientData> ...
  <x:LCT>Normal</x:LCT>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.36 ListItem (Non-linked List Item)

This element specifies a non-linked list item that shall be persisted with the list. This element is used for list boxes. [*Rationale:* This is a place for applications to store optional information associated with the list box. For example, an item to be shown in the list box that is not linked from another set of data. *end rationale]*

[*Example:*

```
<x:ClientData> ...
  <x:ListItem>TheItem</x:ListItem>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.37 Locked (Lock Toggle)

This element specifies that the object is locked when the sheet is protected. If omitted, the object is assumed to be locked. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[*Example:*

```
<x:ClientData> ...
  <x:Locked>False</x:Locked>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.38 LockText (Text Lock)

This element specifies that the object's text is locked. If omitted, the object's text is assumed to be locked. If this element is specified without a value, it is assumed to be true. This element is used for attached text.

[*Example:*

```
<x:ClientData> ...
  <x:LockText>False</x:LockText>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.39 MapOCX (Embedded Control)

This element specifies that the object is an embedded control. If this element is specified without a value, it is assumed to be true. This element is used for all embedded controls.

[*Example:*

```
<x:ClientData>...
  <x:MapOCX>True</x:MapOCX>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.40 Max (Scroll Bar Maximum)

This element specifies the maximum scroll bar position as the index of the list item just above the item at the top of the view when the control is scrolled all the way down. The list indexes are 1-based. If omitted, the value is assumed to be that which allows the last item to be viewed when the control is scrolled all the way down. This element is used for scroll bars and spinners.

[Example: Item 21 is the first item visible in the list when the object is scrolled all the way down.]

```
<x:ClientData> ...
  <x:Max>20</x:Max>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.41 Min (Scroll Bar Minimum)

This element specifies the minimum scroll bar position as the index of the list item just above the item at the top of the view when the control is scrolled all the way up, typically 0. The list indexes are 1-based. If omitted, the value is assumed to be 0. This element is used for scroll bars and spinners.

[Example: The first item in the list is visible when the object is scrolled all the way up:]

```
<x:ClientData> ...
  <x:Min>0</x:Min>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.42 MoveWithCells (Move with Cells)

This element specifies that the object moves with its underlying cells. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:MoveWithCells>True</x:MoveWithCells>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.43 MultiLine (Multi-line)

This element specifies that the control is multiline. If this element is specified without a value, it is assumed to be true. This element is used for edit controls.

[*Example:*

```
<x:ClientData> ...
  <x:Multiline>True</x:Multiline>
</x:ClientData>
```

end example]

The possible values for this element are defined by the `ST_TrueFalseBlank` simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.44 MultiSel (Multiple Selections)

This element specifies a comma-delimited list of selected items. This element overrides the `Sel` element (§19.4.2.57). This element is used for list boxes that allow multiple selections. See also the `SelType` element (§19.4.2.58).

[*Example:*

```
<x:ClientData> ...
  <x:MultiSel>3, 5, 6</x:MultiSel>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.45 NoThreeD (Disable 3D)

This element specifies that 3D effects are disabled. If this element is specified without a value, it is assumed to be true. This element is used for checkboxes, radio buttons, group boxes and scroll bars.

[*Example:*

```
<x:ClientData> ...
  <x>NoThreeD>True</x>NoThreeD>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. end note]

19.4.2.46 NoThreeD2 (Disable 3D)

This element specifies that 3D effects are disabled. If this element is specified without a value, it is assumed to be true. This element is used for dropdowns and list boxes.

[Example:

```
<x:ClientData> ...
  <x:NoThreeD2>True</x:NoThreeD2>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. end note]

19.4.2.47 Page (Scroll Bar Page Increment)

This element specifies the number of lines to move the scroll bar on a page click. This element is used for scroll bars and spinners.

[Example:

```
<x:ClientData> ...
  <x:Page>9</x:Page>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.48 PrintObject (Print Toggle)

This element specifies that the object is printed when the document is printed. If omitted, it is assumed the object prints when the document is printed. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:PrintObject>False</x:PrintObject>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.49 RecalcAlways (Recalculation Toggle)

This element defines whether the object is always included in recalculation. If this element is specified without a value, it is assumed to be true. This is used by controls that reference cells in the spreadsheet to update themselves when the spreadsheet changes.

[*Example:*

```
<x:ClientData> ...
  <x:RecalcAlways>True</x:RecalcAlways>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.50 Row (Comment Row Target)

This element specifies the row a comment points to. The row index is 0-based. This element is used for comments.

[*Example:*

```
<x:ClientData> ...
  <x:Row>0</x:Row>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.51 RowHidden (Comment's Row is Hidden)

This element specifies that the row of the cell to which this comment points is hidden. If this element is specified without a value, it is assumed to be true. This element is used for comments.

[*Example:*

```
<x:ClientData> ...
  <x:RowHidden>True</x:RowHidden>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.52 ScriptExtended (HTML Script Attributes)

This element specifies custom extended attributes associated with the HTML script tag. The language and id are not included in the extended attributes. If the document contains no HTML script, this element should be ignored.

[*Example:* The extended script attribute is " src="file.js"":

```
<x:ClientData> ...
  <x:ScriptExtended>src=&quot;file.js&quot;</x:ScriptExtended>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.53 ScriptLanguage (HTML Script Language)

This element specifies the language of the custom function. If the document contains no HTML script, this element should be ignored. Allowed values are:

Value	Description
1	Java
2	Visual Basic
3	ASP
4	Other

[*Example:*

```
<x:ClientData> ...
  <x:ScriptLanguage>1</x:ScriptLanguage>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema nonNegativeInteger datatype.

19.4.2.54 **ScriptLocation (HTML Script Location)**

This element specifies the location of the custom function. If the document contains no HTML script, this element should be ignored. Allowed values are:

Value	Description
1	Head
2	Body

[Example:

```
<x:ClientData> ...  
  <x:ScriptLocation>2</x:ScriptLocation>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema nonNegativeInteger datatype.

19.4.2.55 **ScriptText (HTML Script Text)**

This element specifies the script text (comment) associated with a block of HTML script in the document. If the document contains no HTML script, this element should be ignored.

[Example: The script text reads: "<!-- Comment -->":

```
<x:ClientData> ...  
  <x:ScriptText>&lt;!-- Comment -->&gt;</x:ScriptText>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.56 **SecretEdit (Password Edit)**

This element specifies that the object represents a password edit field. If this element is specified without a value, it is assumed to be true. This element is used for attached text.

[Example:

```
<x:ClientData> ...  
  <x:SecretEdit>True</x:SecretEdit>  
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

19.4.2.57 Sel (Selected Entry)

This element specifies the index of the selected item. The list indexes are 1-based. If omitted or set to a value of 0, no items are selected. This element is used for list boxes.

[Example:

```
<x:ClientData>...
  <x:Sel>1</x:Sel>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.58 SelType (Selection Type)

This element specifies the kind of selection for the list box. If omitted, the control is assumed to be Single. Allowed values are:

Value	Description
Single	The listbox shall only have one selected item.
Multi	The listbox can have multiple items selected by clicking on each item.
Extend	The listbox can have multiple items selected by holding a control key and clicking on each item.

This element is used for list boxes.

[Example:

```
<x:ClientData> ...
  <x:SelType>Single</x:SelType>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.59 SizeWithCells (Resize with Cells)

This element specifies that the object resizes with its underlying cells. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:SizeWithCells>True</x:SizeWithCells>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.60 TextHAlign (Horizontal Text Alignment)

This element specifies the horizontal text alignment for the object. Permitted values are Left, Justify, Center, Right and Distributed. If omitted, the alignment is assumed to be Left. This element is used for attached text.

[Example:

```
<x:ClientData> ...
  <x:TextHAlign>Right</x:TextHAlign>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.61 TextVAlign (Vertical Text Alignment)

This element specifies the horizontal text alignment for the object. Permitted values are Top, Justify, Center, Bottom and Distributed. If omitted, the alignment is assumed to be Top. This element is used for attached text.

[Example:

```
<x:ClientData> ...
  <x:TextVAlign>Center</x:TextVAlign>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema string datatype.

19.4.2.62 UIObj (UI Object Toggle)

This element defines whether the object is a UI object. If this element is specified without a value, it is assumed to be true. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:UIObj>True</x:UIObj>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.63 Val (Scroll bar position)

This element specifies the scroll bar position as the index of the list item just above the item at the top of the view, given the current scroll position. The list indexes are 1-based. If omitted, the value is assumed to be 0. This element is used for scroll bars and spinners.

[Example: The first list item (item 1) is just off the top of the view. The second list item is at the top of the view.

```
<x:ClientData> ...
  <x:Val>1</x:Val>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.64 ValidIds (Valid ID)

This element specifies that the ID of a linked object is correct. This is a general-use element.

[Example:

```
<x:ClientData> ...
  <x:ValidIds>True</x:ValidIds>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.65 Visible (Comment Visibility Toggle)

This element specifies that a comment is visible. If omitted, the comment is assumed to be invisible. If this element is specified without a value, it is assumed to be true. This element is used for comments.

[Example:

```
<x:ClientData> ...
  <x:Visible>True</x:Visible>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.66 VScroll (Vertical Scroll)

This element specifies that the object has a vertical scroll. If omitted, a vertical scroll is not used. If this element is specified without a value, it is assumed to be true. This element is used for edit controls.

[Example:

```
<x:ClientData> ...
  <x:VScroll>True</x:VScroll>
</x:ClientData>
```

end example]

The possible values for this element are defined by the ST_TrueFalseBlank simple type (§20.1.2.6).

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

19.4.2.67 VTEdit (Validation Type)

This element specifies the kind of semantic validation to use for data input to the control. If omitted, the value is assumed to be Text. Permitted values are:

Value	Description
0	Text
1	Integer
2	Number
3	Reference
4	Formula

This element is used for edit controls.

[Example:


```
<x:ClientData> ...
  <x:VTEdit>True</x:VTEdit>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.2.68 WidthMin (Minimum Width)

This element specifies the smallest width allowed for the dropdown window in screen pixels. This element is used for list boxes and dropdowns.

[Example:

```
<x:ClientData ... > ...
  <x:WidthMin>78</x:WidthMin>
</x:ClientData>
```

end example]

The possible values for this element are defined by the W3C XML Schema integer datatype.

19.4.3 Simple Types

The following additional simple type information in the urn:schemas-microsoft-com:office:excel namespace is used for documents of a transitional conformance class.

19.4.3.1 ST_CF (Clipboard Format Type)

This simple type specifies the allowed clipboard formats. This simple type allows any image format to be specified; however, the following values are reserved:

Value	Description
Bitmap	Bitmap.
Jpeg	An image which should use the JPEG format.
Pict	Any picture format. [Example: SVG or JPEG. <i>end example]</i>
PictOld	Any picture format, but preferably one that is more likely to be supported by legacy applications.
PictPrint	An image rendered using the default printer's settings. This is typically of higher resolution and scaled differently compared to a picture created for on-screen rendering.
PictScreen	An image rendered using screen settings. This is typically lower resolution than an image created for printing.

Value	Description
Png	An image which should use the Portable Network Graphics format.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_CF](#)) is located in §A.6.4. *end note*]

19.4.3.2 ST_ObjectType (Object Type)

This simple type specifies the objects that a ClientData element can represent.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
Button (Pushbutton)	A pushbutton control.
Checkbox (Checkbox)	A checkbox control.
Dialog (Dialog)	A dialog.
Drop (Dropdown Box)	A dropdown (combo box) control.
Edit (Editable Text Field)	An editable text field control.
GBox (Group Box)	A group box control.
Group (Group)	A group of objects, such as a group of checkboxes.
Label (Label)	A label control.
LineA (Auditing Line)	A formula auditing arrow.
List (List Box)	A list control.
Movie (Movie)	A movie object in Mac format.
Note (Comment)	A comment.
Pict (Image)	A placeholder image.
Radio (Radio Button)	A radio button control.
Rect (Plain Rectangle)	A rectangle shape that is not a control.
RectA (Auditing Rectangle)	A formula auditing rectangle.
Scroll (Scroll Bar)	A scroll bar.
Shape (Plain Shape)	A general shape that is not a control.
Spin (Spin Button)	A spin button (spinner) control.

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_ObjectType](#)) is located in §A.6.4. *end note*]

19.5 VML - PresentationML Drawing

This section describes additional information attached to VML shapes that is specific to usage with PresentationML.

[*Note:* The VML format is a legacy format originally introduced with Office 2000 and is included and fully defined in ECMA-376 for backwards compatibility reasons. The DrawingML format is a newer and richer format created with the goal of eventually replacing any uses of VML in the Office Open XML formats. VML should be considered a transitional format included in Office Open XML for legacy reasons only and new applications that need a file format for drawings are strongly encouraged to use preferentially DrawingML. *end note*]

[*Example:* Assume the following annotation was drawn on a slide during a presentation and saved into the presentation:

- Bullet point

The red circle annotation is stored as a VML shape that is an ink annotation. For brevity, the specific path and ink data are omitted.

```
<v:shape id="_x0000_s1029" style='position:absolute;left:126pt;
top:327.375pt;width:27.625pt;height:24.75pt' coordorigin="4445,11549"
coordsize="973,874" path="..." filled="f" strokecolor="red"
strokeweight="1.5pt">
  <v:stroke endcap="round"/>
  <v:path shadowok="f" o:extrusionok="f" fillok="f" insetpenok="f"/>
  <o:lock v:ext="edit" rotation="t" aspectratio="t" verticies="t" text="t"
    shapetype="t"/>
  <o:ink i="..." annotation="t"/>
  <pvm1:iscomment/>
</v:shape>
```

end example]

19.5.1 Table of Contents

This subclause is informative.

19.5.2 Elements	858
19.5.2.1 iscomment (Ink Annotation Flag)	858
19.5.2.2 textdata (VML Diagram Text)	858

End of informative text.

19.5.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:office:powerpoint namespace:

[*Note*: As the VML format is a format provided for backward compatibility, those VML elements defined in the same urn:schemas-microsoft-com:office:powerpoint namespace remain in that namespace as it is already used by millions of documents already using VML. *end note*]

19.5.2.1 iscomment (Ink Annotation Flag)

Specifies that the object was created as an ink annotation. Default is `false`. If this element is specified without a value, it is assumed to be true. This element is only used with PresentationML. [*Rationale* This allows an application to treat annotation ink objects as any other annotation. For example, if annotations are hidden, the application can hide the ink object. *end rationale*]

[*Example*:

```
<v:shape ... >
  <o:ink ... annotation="true"/>
  <pvm1:iscomment/>
</v:shape>
```

• **Bullet point**

end example]

[*Note*: The W3C XML Schema definition of this element's content model (CT_Empty) is located in §A.6.5. *end note*]

19.5.2.2 textdata (VML Diagram Text)

This element specifies optional supplementary text information associated with a legacy VML shape that is a node in a VML diagram when it cannot otherwise be stored within the DrawingML framework.

[*Note*: An application could use this to preserve a specific diagram format for backward compatibility, but it is strongly recommended to upgrade all VML shapes to DrawingML shapes. *end note*]

Attributes	Description
id (Text Reference)	<p>Specifies the identifier that is used in conjunction with a corresponding relationship file to resolve the location of the diagram shape text.</p> <p>[<i>Example</i>:</p> <pre><v:shape ... o:dgmnodekind="0" > <v:textbox inset="0,0,0,0"/> <pvm1:textdata id="rId1"/></pre>

Attributes	Description
	<p data-bbox="456 247 613 279"></v:shape></p> <p data-bbox="415 317 574 348"><i>end example]</i></p> <p data-bbox="415 386 1377 459">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Rel](#)) is located in §A.6.5. *end note*]

20. Shared MLs Reference Material

20.1 Shared Simple Types

20.1.1 Table of Contents

This subclause is informative.

20.1.2	Simple Types	860
20.1.2.1	ST_AlgClass (Cryptographic Algorithm Classes)	860
20.1.2.2	ST_AlgType (Cryptographic Algorithm Types)	861
20.1.2.3	ST_ColorType (Color Type)	862
20.1.2.4	ST_CryptProv (Cryptographic Provider Types)	863
20.1.2.5	ST_TrueFalse (Boolean Value)	863
20.1.2.6	ST_TrueFalseBlank (Boolean Value with Blank [False] State)	864
20.4.1	Changed attribute for sources element (Part 1, §22.6.2.60)	864

End of informative text.

20.1.2 Simple Types

The following additional simple type information in the <http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes> namespace is used for documents of a transitional conformance class.

20.1.2.1 ST_AlgClass (Cryptographic Algorithm Classes)

This simple type specifies the possible classes of cryptographic algorithm used by protection. [*Note: The initial version of ECMA-376 only supports a single version - hash - but future versions may expand this as necessary. end note*]

[*Note: Omitting this attribute is logically equivalent to assigning it the value custom. end note*]

[*Example: Consider a WordprocessingML document with the following information stored in one of its protection elements:*

```
<... w:cryptAlgorithmClass="hash"
      w:cryptAlgorithmType="typeAny"
      w:cryptAlgorithmSid="1"
      w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" />
```

The cryptAlgorithmClass attribute value of hash specifies that the algorithm used for the password is a hashing algorithm. *end example*]

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
custom (Custom Algorithm)	Specifies that a custom algorithm class, specified within the parent element's <code>algIdExt</code> attribute, generated the hash value.
hash (Hashing)	Specifies that the algorithm is a hashing function, which creates a hash value for user-supplied input that is very difficult to reverse-engineer.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST AlgClass](#)) is located in §A.7.9. *end note*]

20.1.2.2 ST_AlgorithmType (Cryptographic Algorithm Types)

This simple type specifies the possible values for the type of cryptographic algorithm used by protection. [*Note:* The initial version of ECMA-376 only supports a single type - `typeAny` - but future versions may expand this as necessary. *end note*]

[*Note:* Omitting this attribute is logically equivalent to assigning it the value `custom`. *end note*]

[*Example:* Consider a WordprocessingML document with the following information stored in one of its protection elements:

```
<... w:cryptAlgorithmClass="hash"
      w:cryptAlgorithmType="typeAny"
      w:cryptAlgorithmSid="1"
      w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" />
```

The `cryptAlgorithmType` attribute value of `typeAny` specifies that any type of algorithm may have been used for the password. *end example*]

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
custom (Custom Algorithm)	Specifies that a custom algorithm type, specified within the parent element's <code>algIdExt</code> attribute, generated the hash value.
typeAny (Any Predefined Type)	Specifies that one of the predefined cryptographic algorithms, specified by the parent element's <code>cryptAlgorithmSid</code> attribute, generated the hash value.

















[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_AlgType](#)) is located in §A.7.9.
end note]

20.1.2.3 ST_ColorType (Color Type)

This simple type specifies a color. Colors are specified in one of three ways - named color, hexadecimal RGB or color palette entry. An optional index can be stored in square brackets following the color and a space.

[*Rationale:* An application might store the color's index in a system color palette using this means. *end rationale*]

A named color is specified using the name of the color. The following named colors are supported:

- Black (#000000) 
- Silver (#C0C0C0) 
- Gray (#808080) 
- White (#FFFFFF) 
- Maroon (#800000) 
- Red (#FF0000) 
- Purple (#800080) 
- Fuchsia (#FF00FF) 
- Green (#008000) 
- Lime (#00FF00) 
- Olive (#808000) 
- Yellow (#FFFF00) 
- Navy (#000080) 
- Blue (#0000FF) 
- Teal (#008080) 
- Aqua (#00FFFF) 

[*Example:*

```
<... color="red" ... >
```

end example]

Hexadecimal RGB is specified using a hash symbol (#) followed by six hexadecimal characters, where each pair represents the red, green and blue component of the color.

[*Example:*

```
< ... color="#5f2726" ... >
```

end example]

A color palette entry is specified using the name of the color in the palette.

[*Example:*

```
<... color="buttonFace [67]" ... >
```


end example]

This simple type's contents are a restriction of the W3C XML Schema string datatype.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_ColorType](#)) is located in §A.7.9. *end note*]

20.1.2.4 ST_CryptProv (Cryptographic Provider Types)

This simple type specifies the possible types of cryptographic providers which may be used.

[*Note:* Omitting this attribute is logically equivalent to assigning it the value custom. *end note*]

[*Example:* Consider a WordprocessingML document with the following information stored in one of its protection elements:

```
<... w:cryptProviderType="rsaAES"
      w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" />
```

The cryptProviderType attribute value of rsaAES specifies that the cryptographic provider type shall be an Advanced Encryption Standard provider. *end example*

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
custom (Custom Provider)	Specifies that a custom algorithm type, specified within the parent element's algIdExt attribute, generated the hash value.
rsaAES (AES Provider)	Specifies that the provider shall support the Advanced Encryption Algorithm standard.
rsaFull (Any Provider)	Specifies that any suitable provider shall be used.

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_CryptProv](#)) is located in §A.7.9. *end note*]

20.1.2.5 ST_TrueFalse (Boolean Value)

This type specifies logical true and false.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
f (False)	Logical false.
false (False)	Logical false.
t (True)	Logical true.

Enumeration Value	Description
true (True)	Logical true.

[Note: The W3C XML Schema definition of this simple type's content model (ST_TrueFalse) is located in §A.7.9. *end note*]

20.1.2.6 ST_TrueFalseBlank (Boolean Value with Blank [False] State)

This simple type specifies a boolean value with a third state, using a blank attribute, which specifies that the value be false.

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Enumeration Value	Description
(Blank – Logical False)	Logical false.
f (Logical False)	Logical false.
false (Logical False)	Logical false.
t (Logical True)	Logical true.
true (Logical True)	Logical true.

[Note: The W3C XML Schema definition of this simple type's content model (ST_TrueFalseBlank) is located in §A.7.9. *end note*]

20.2 Extended Properties (Part 1, §22.2)

When used in a document of the Transitional conformance class, extended properties are stored within an Extended File Properties part with a source relationship of <http://schemas.openxmlformats.org/officeDocument/2006/relationships/extended-properties>.

20.3 Custom Properties (Part 1, §22.3)

When used in a document of the Transitional conformance class, custom properties are stored within a Custom File Properties part with a source relationship of <http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties>.

20.4 Changed attributes

The following attributes, which are defined in subclauses within Part 1, §22, “Shared MLs Reference Material”, have different source relationships when used in documents of the Transitional conformance class:

20.4.1 Changed attribute for sources element (Part 1, §22.6.2.60)

Attributes	Description
SelectedStyle	Specifies the filename of a file which can be used to format the bibliographies and

Attributes	Description
(Selected Style)	<p>citations within this document.</p> <p>If this file is of an unknown form or cannot be located, then the other attributes on this element can be used to determine the format to use.</p> <p>[Example:</p> <pre data-bbox="451 499 1468 569"><b:Sources SelectedStyle="\APA.XSL" StyleName="APA" URI="http://schemas.openxmlformats.org/bibliographicStyle/APA"></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p>

Annex A.

(normative)

Schemas – W3C XML Schema

This Office Open XML specification includes a family of schemas defined using the W3C XML Schema 1.0 syntax. The normative definitions of these schemas follow below, and they also reside in an accompanying file named OfficeOpenXML-XMLSchema-Transitional.zip, which is distributed in electronic form.

A.1 WordprocessingML

This schema is available in the file wml.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:m="http://schemas.openxmlformats.org/officeDocument/2006/math"
3   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4   xmlns:sl="http://schemas.openxmlformats.org/schemaLibrary/2006/main"
5   xmlns:wp="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
6   xmlns="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
7   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
8   elementFormDefault="qualified" attributeFormDefault="qualified" blockDefault="#all"
9   targetNamespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main">
10   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
11     schemaLocation="dml-wordprocessingDrawing.xsd"/>
12   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/math"
13     schemaLocation="shared-math.xsd"/>
14   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
15     schemaLocation="shared-relationshipReference.xsd"/>
16   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
17     schemaLocation="shared-commonSimpleTypes.xsd"/>
18   <xsd:import namespace="http://schemas.openxmlformats.org/schemaLibrary/2006/main"
19     schemaLocation="shared-customXmlSchemaProperties.xsd"/>
20   <xsd:import namespace="http://www.w3.org/XML/1998/namespace"/>
21   <xsd:complexType name="CT_Empty"/>
22   <xsd:complexType name="CT_OnOff">
23     <xsd:attribute name="val" type="s:ST_OnOff"/>
24   </xsd:complexType>
25   <xsd:simpleType name="ST_LongHexNumber">
26     <xsd:restriction base="xsd:hexBinary">
27       <xsd:length value="4"/>
28     </xsd:restriction>
29   </xsd:simpleType>
30   <xsd:complexType name="CT_LongHexNumber">
31     <xsd:attribute name="val" type="ST_LongHexNumber" use="required"/>
32   </xsd:complexType>
33   <xsd:simpleType name="ST_ShortHexNumber">

```

```

34     <xsd:restriction base="xsd:hexBinary">
35         <xsd:length value="2"/>
36     </xsd:restriction>
37 </xsd:simpleType>
38 <xsd:simpleType name="ST_UcharHexNumber">
39     <xsd:restriction base="xsd:hexBinary">
40         <xsd:length value="1"/>
41     </xsd:restriction>
42 </xsd:simpleType>
43 <xsd:complexType name="CT_Charset">
44     <xsd:attribute name="val" type="ST_UcharHexNumber" use="optional"/>
45     <xsd:attribute name="characterSet" type="s:ST_String" use="optional"/>
46 </xsd:complexType>
47 <xsd:simpleType name="ST_DecimalNumberOrPercent">
48     <xsd:union memberTypes="ST_UnqualifiedPercentage s:ST_Percentage"/>
49 </xsd:simpleType>
50 <xsd:simpleType name="ST_UnqualifiedPercentage">
51     <xsd:restriction base="xsd:integer"/>
52 </xsd:simpleType>
53 <xsd:simpleType name="ST_DecimalNumber">
54     <xsd:restriction base="xsd:integer"/>
55 </xsd:simpleType>
56 <xsd:complexType name="CT_DecimalNumber">
57     <xsd:attribute name="val" type="ST_DecimalNumber" use="required"/>
58 </xsd:complexType>
59 <xsd:complexType name="CT_UnsignedDecimalNumber">
60     <xsd:attribute name="val" type="s:ST_UnsignedDecimalNumber" use="required"/>
61 </xsd:complexType>
62 <xsd:complexType name="CT_DecimalNumberOrPrecent">
63     <xsd:attribute name="val" type="ST_DecimalNumberOrPercent" use="required"/>
64 </xsd:complexType>
65 <xsd:complexType name="CT_TwipsMeasure">
66     <xsd:attribute name="val" type="s:ST_TwipsMeasure" use="required"/>
67 </xsd:complexType>
68 <xsd:simpleType name="ST_SignedTwipsMeasure">
69     <xsd:union memberTypes="xsd:integer s:ST_UniversalMeasure"/>
70 </xsd:simpleType>
71 <xsd:complexType name="CT_SignedTwipsMeasure">
72     <xsd:attribute name="val" type="ST_SignedTwipsMeasure" use="required"/>
73 </xsd:complexType>
74 <xsd:simpleType name="ST_PixelsMeasure">
75     <xsd:restriction base="s:ST_UnsignedDecimalNumber"/>
76 </xsd:simpleType>
77 <xsd:complexType name="CT_PixelsMeasure">
78     <xsd:attribute name="val" type="ST_PixelsMeasure" use="required"/>
79 </xsd:complexType>
80 <xsd:simpleType name="ST_HpsMeasure">
81     <xsd:union memberTypes="s:ST_UnsignedDecimalNumber s:ST_PositiveUniversalMeasure"/>
82 </xsd:simpleType>
83 <xsd:complexType name="CT_HpsMeasure">
84     <xsd:attribute name="val" type="ST_HpsMeasure" use="required"/>
85 </xsd:complexType>
86 <xsd:simpleType name="ST_SignedHpsMeasure">

```

```

87     <xsd:union memberTypes="xsd:integer s:ST_UniversalMeasure"/>
88 </xsd:simpleType>
89 <xsd:complexType name="CT_SignedHpsMeasure">
90     <xsd:attribute name="val" type="ST_SignedHpsMeasure" use="required"/>
91 </xsd:complexType>
92 <xsd:simpleType name="ST_DateTime">
93     <xsd:restriction base="xsd:dateTime"/>
94 </xsd:simpleType>
95 <xsd:simpleType name="ST_MacroName">
96     <xsd:restriction base="xsd:string">
97         <xsd:maxLength value="33"/>
98     </xsd:restriction>
99 </xsd:simpleType>
100 <xsd:complexType name="CT_MacroName">
101     <xsd:attribute name="val" use="required" type="ST_MacroName"/>
102 </xsd:complexType>
103 <xsd:simpleType name="ST_EighthPointMeasure">
104     <xsd:restriction base="s:ST_UnsignedDecimalNumber"/>
105 </xsd:simpleType>
106 <xsd:simpleType name="ST_PointMeasure">
107     <xsd:restriction base="s:ST_UnsignedDecimalNumber"/>
108 </xsd:simpleType>
109 <xsd:complexType name="CT_String">
110     <xsd:attribute name="val" type="s:ST_String" use="required"/>
111 </xsd:complexType>
112 <xsd:simpleType name="ST_TextScale">
113     <xsd:union memberTypes="ST_TextScalePercent ST_TextScaleDecimal"/>
114 </xsd:simpleType>
115 <xsd:simpleType name="ST_TextScalePercent">
116     <xsd:restriction base="xsd:string">
117         <xsd:pattern value="0*(600|([0-5]?[0-9]?[0-9]))%"/>
118     </xsd:restriction>
119 </xsd:simpleType>
120 <xsd:simpleType name="ST_TextScaleDecimal">
121     <xsd:restriction base="xsd:integer">
122         <xsd:minInclusive value="0"/>
123         <xsd:maxInclusive value="600"/>
124     </xsd:restriction>
125 </xsd:simpleType>
126 <xsd:complexType name="CT_TextScale">
127     <xsd:attribute name="val" type="ST_TextScale"/>
128 </xsd:complexType>
129 <xsd:simpleType name="ST_HighlightColor">
130     <xsd:restriction base="xsd:string">
131         <xsd:enumeration value="black"/>
132         <xsd:enumeration value="blue"/>
133         <xsd:enumeration value="cyan"/>
134         <xsd:enumeration value="green"/>
135         <xsd:enumeration value="magenta"/>
136         <xsd:enumeration value="red"/>
137         <xsd:enumeration value="yellow"/>
138         <xsd:enumeration value="white"/>
139         <xsd:enumeration value="darkBlue"/>

```

```

140      <xsd:enumeration value="darkCyan"/>
141      <xsd:enumeration value="darkGreen"/>
142      <xsd:enumeration value="darkMagenta"/>
143      <xsd:enumeration value="darkRed"/>
144      <xsd:enumeration value="darkYellow"/>
145      <xsd:enumeration value="darkGray"/>
146      <xsd:enumeration value="lightGray"/>
147      <xsd:enumeration value="none"/>
148    </xsd:restriction>
149  </xsd:simpleType>
150  <xsd:complexType name="CT_Highlight">
151    <xsd:attribute name="val" type="ST_HighlightColor" use="required"/>
152  </xsd:complexType>
153  <xsd:simpleType name="ST_HexColorAuto">
154    <xsd:restriction base="xsd:string">
155      <xsd:enumeration value="auto"/>
156    </xsd:restriction>
157  </xsd:simpleType>
158  <xsd:simpleType name="ST_HexColor">
159    <xsd:union memberTypes="ST_HexColorAuto s:ST_HexColorRGB"/>
160  </xsd:simpleType>
161  <xsd:complexType name="CT_Color">
162    <xsd:attribute name="val" type="ST_HexColor" use="required"/>
163    <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
164    <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
165    <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
166  </xsd:complexType>
167  <xsd:complexType name="CT_Lang">
168    <xsd:attribute name="val" type="s:ST_Lang" use="required"/>
169  </xsd:complexType>
170  <xsd:complexType name="CT_Guid">
171    <xsd:attribute name="val" type="s:ST_Guid"/>
172  </xsd:complexType>
173  <xsd:simpleType name="ST_Underline">
174    <xsd:restriction base="xsd:string">
175      <xsd:enumeration value="single"/>
176      <xsd:enumeration value="words"/>
177      <xsd:enumeration value="double"/>
178      <xsd:enumeration value="thick"/>
179      <xsd:enumeration value="dotted"/>
180      <xsd:enumeration value="dottedHeavy"/>
181      <xsd:enumeration value="dash"/>
182      <xsd:enumeration value="dashedHeavy"/>
183      <xsd:enumeration value="dashLong"/>
184      <xsd:enumeration value="dashLongHeavy"/>
185      <xsd:enumeration value="dotDash"/>
186      <xsd:enumeration value="dashDotHeavy"/>
187      <xsd:enumeration value="dotDotDash"/>
188      <xsd:enumeration value="dashDotDotHeavy"/>
189      <xsd:enumeration value="wave"/>
190      <xsd:enumeration value="wavyHeavy"/>
191      <xsd:enumeration value="wavyDouble"/>
192      <xsd:enumeration value="none"/>

```



```

193     </xsd:restriction>
194 </xsd:simpleType>
195 <xsd:complexType name="CT_Underline">
196     <xsd:attribute name="val" type="ST_Underline" use="optional"/>
197     <xsd:attribute name="color" type="ST_HexColor" use="optional"/>
198     <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
199     <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
200     <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
201 </xsd:complexType>
202 <xsd:simpleType name="ST_TextEffect">
203     <xsd:restriction base="xsd:string">
204         <xsd:enumeration value="blinkBackground"/>
205         <xsd:enumeration value="lights"/>
206         <xsd:enumeration value="antsBlack"/>
207         <xsd:enumeration value="antsRed"/>
208         <xsd:enumeration value="shimmer"/>
209         <xsd:enumeration value="sparkle"/>
210         <xsd:enumeration value="none"/>
211     </xsd:restriction>
212 </xsd:simpleType>
213 <xsd:complexType name="CT_TextEffect">
214     <xsd:attribute name="val" type="ST_TextEffect" use="required"/>
215 </xsd:complexType>
216 <xsd:simpleType name="ST_Border">
217     <xsd:restriction base="xsd:string">
218         <xsd:enumeration value="nil"/>
219         <xsd:enumeration value="none"/>
220         <xsd:enumeration value="single"/>
221         <xsd:enumeration value="thick"/>
222         <xsd:enumeration value="double"/>
223         <xsd:enumeration value="dotted"/>
224         <xsd:enumeration value="dashed"/>
225         <xsd:enumeration value="dotDash"/>
226         <xsd:enumeration value="dotDotDash"/>
227         <xsd:enumeration value="triple"/>
228         <xsd:enumeration value="thinThickSmallGap"/>
229         <xsd:enumeration value="thickThinSmallGap"/>
230         <xsd:enumeration value="thinThickThinSmallGap"/>
231         <xsd:enumeration value="thinThickMediumGap"/>
232         <xsd:enumeration value="thickThinMediumGap"/>
233         <xsd:enumeration value="thinThickThinMediumGap"/>
234         <xsd:enumeration value="thinThickLargeGap"/>
235         <xsd:enumeration value="thickThinLargeGap"/>
236         <xsd:enumeration value="thinThickThinLargeGap"/>
237         <xsd:enumeration value="wave"/>
238         <xsd:enumeration value="doubleWave"/>
239         <xsd:enumeration value="dashSmallGap"/>
240         <xsd:enumeration value="dashDotStroked"/>
241         <xsd:enumeration value="threeDEmboss"/>
242         <xsd:enumeration value="threeDEngrave"/>
243         <xsd:enumeration value="outset"/>
244         <xsd:enumeration value="inset"/>
245         <xsd:enumeration value="apples"/>

```

```

246     <xsd:enumeration value="archedScallops"/>
247     <xsd:enumeration value="babyPacifier"/>
248     <xsd:enumeration value="babyRattle"/>
249     <xsd:enumeration value="balloons3Colors"/>
250     <xsd:enumeration value="balloonsHotAir"/>
251     <xsd:enumeration value="basicBlackDashes"/>
252     <xsd:enumeration value="basicBlackDots"/>
253     <xsd:enumeration value="basicBlackSquares"/>
254     <xsd:enumeration value="basicThinLines"/>
255     <xsd:enumeration value="basicWhiteDashes"/>
256     <xsd:enumeration value="basicWhiteDots"/>
257     <xsd:enumeration value="basicWhiteSquares"/>
258     <xsd:enumeration value="basicWideInline"/>
259     <xsd:enumeration value="basicWideMidline"/>
260     <xsd:enumeration value="basicWideOutline"/>
261     <xsd:enumeration value="bats"/>
262     <xsd:enumeration value="birds"/>
263     <xsd:enumeration value="birdsFlight"/>
264     <xsd:enumeration value="cabins"/>
265     <xsd:enumeration value="cakeSlice"/>
266     <xsd:enumeration value="candyCorn"/>
267     <xsd:enumeration value="celticKnotwork"/>
268     <xsd:enumeration value="certificateBanner"/>
269     <xsd:enumeration value="chainLink"/>
270     <xsd:enumeration value="champagneBottle"/>
271     <xsd:enumeration value="checkedBarBlack"/>
272     <xsd:enumeration value="checkedBarColor"/>
273     <xsd:enumeration value="checkered"/>
274     <xsd:enumeration value="christmasTree"/>
275     <xsd:enumeration value="circlesLines"/>
276     <xsd:enumeration value="circlesRectangles"/>
277     <xsd:enumeration value="classicalWave"/>
278     <xsd:enumeration value="clocks"/>
279     <xsd:enumeration value="compass"/>
280     <xsd:enumeration value="confetti"/>
281     <xsd:enumeration value="confettiGrays"/>
282     <xsd:enumeration value="confettiOutline"/>
283     <xsd:enumeration value="confettiStreamers"/>
284     <xsd:enumeration value="confettiWhite"/>
285     <xsd:enumeration value="cornerTriangles"/>
286     <xsd:enumeration value="couponCutoutDashes"/>
287     <xsd:enumeration value="couponCutoutDots"/>
288     <xsd:enumeration value="crazyMaze"/>
289     <xsd:enumeration value="creaturesButterfly"/>
290     <xsd:enumeration value="creaturesFish"/>
291     <xsd:enumeration value="creaturesInsects"/>
292     <xsd:enumeration value="creaturesLadyBug"/>
293     <xsd:enumeration value="crossStitch"/>
294     <xsd:enumeration value="cup"/>
295     <xsd:enumeration value="decoArch"/>
296     <xsd:enumeration value="decoArchColor"/>
297     <xsd:enumeration value="decoBlocks"/>
298     <xsd:enumeration value="diamondsGray"/>

```

```

299     <xsd:enumeration value="doubleD"/>
300     <xsd:enumeration value="doubleDiamonds"/>
301     <xsd:enumeration value="earth1"/>
302     <xsd:enumeration value="earth2"/>
303     <xsd:enumeration value="earth3"/>
304     <xsd:enumeration value="eclipsingSquares1"/>
305     <xsd:enumeration value="eclipsingSquares2"/>
306     <xsd:enumeration value="eggsBlack"/>
307     <xsd:enumeration value="fans"/>
308     <xsd:enumeration value="film"/>
309     <xsd:enumeration value="firecrackers"/>
310     <xsd:enumeration value="flowersBlockPrint"/>
311     <xsd:enumeration value="flowersDaisies"/>
312     <xsd:enumeration value="flowersModern1"/>
313     <xsd:enumeration value="flowersModern2"/>
314     <xsd:enumeration value="flowersPansy"/>
315     <xsd:enumeration value="flowersRedRose"/>
316     <xsd:enumeration value="flowersRoses"/>
317     <xsd:enumeration value="flowersTeacup"/>
318     <xsd:enumeration value="flowersTiny"/>
319     <xsd:enumeration value="gems"/>
320     <xsd:enumeration value="gingerbreadMan"/>
321     <xsd:enumeration value="gradient"/>
322     <xsd:enumeration value="handmade1"/>
323     <xsd:enumeration value="handmade2"/>
324     <xsd:enumeration value="heartBalloon"/>
325     <xsd:enumeration value="heartGray"/>
326     <xsd:enumeration value="hearts"/>
327     <xsd:enumeration value="heebieJeebies"/>
328     <xsd:enumeration value="holly"/>
329     <xsd:enumeration value="houseFunky"/>
330     <xsd:enumeration value="hypnotic"/>
331     <xsd:enumeration value="iceCreamCones"/>
332     <xsd:enumeration value="lightBulb"/>
333     <xsd:enumeration value="lightning1"/>
334     <xsd:enumeration value="lightning2"/>
335     <xsd:enumeration value="mapPins"/>
336     <xsd:enumeration value="mapleLeaf"/>
337     <xsd:enumeration value="mapleMuffins"/>
338     <xsd:enumeration value="marquee"/>
339     <xsd:enumeration value="marqueeToothed"/>
340     <xsd:enumeration value="moons"/>
341     <xsd:enumeration value="mosaic"/>
342     <xsd:enumeration value="musicNotes"/>
343     <xsd:enumeration value="northwest"/>
344     <xsd:enumeration value="ovals"/>
345     <xsd:enumeration value="packages"/>
346     <xsd:enumeration value="palmsBlack"/>
347     <xsd:enumeration value="palmsColor"/>
348     <xsd:enumeration value="paperClips"/>
349     <xsd:enumeration value="papyrus"/>
350     <xsd:enumeration value="partyFavor"/>
351     <xsd:enumeration value="partyGlass"/>

```

```

352     <xsd:enumeration value="pencils"/>
353     <xsd:enumeration value="people"/>
354     <xsd:enumeration value="peopleWaving"/>
355     <xsd:enumeration value="peopleHats"/>
356     <xsd:enumeration value="poinsettias"/>
357     <xsd:enumeration value="postageStamp"/>
358     <xsd:enumeration value="pumpkin1"/>
359     <xsd:enumeration value="pushPinNote2"/>
360     <xsd:enumeration value="pushPinNote1"/>
361     <xsd:enumeration value="pyramids"/>
362     <xsd:enumeration value="pyramidsAbove"/>
363     <xsd:enumeration value="quadrants"/>
364     <xsd:enumeration value="rings"/>
365     <xsd:enumeration value="safari"/>
366     <xsd:enumeration value="sawtooth"/>
367     <xsd:enumeration value="sawtoothGray"/>
368     <xsd:enumeration value="scaredCat"/>
369     <xsd:enumeration value="seattle"/>
370     <xsd:enumeration value="shadowedSquares"/>
371     <xsd:enumeration value="sharksTeeth"/>
372     <xsd:enumeration value="shorebirdTracks"/>
373     <xsd:enumeration value="skyrocket"/>
374     <xsd:enumeration value="snowflakeFancy"/>
375     <xsd:enumeration value="snowflakes"/>
376     <xsd:enumeration value="sombrero"/>
377     <xsd:enumeration value="southwest"/>
378     <xsd:enumeration value="stars"/>
379     <xsd:enumeration value="starsTop"/>
380     <xsd:enumeration value="stars3d"/>
381     <xsd:enumeration value="starsBlack"/>
382     <xsd:enumeration value="starsShadowed"/>
383     <xsd:enumeration value="sun"/>
384     <xsd:enumeration value="swirligig"/>
385     <xsd:enumeration value="tornPaper"/>
386     <xsd:enumeration value="tornPaperBlack"/>
387     <xsd:enumeration value="trees"/>
388     <xsd:enumeration value="triangleParty"/>
389     <xsd:enumeration value="triangles"/>
390     <xsd:enumeration value="triangle1"/>
391     <xsd:enumeration value="triangle2"/>
392     <xsd:enumeration value="triangleCircle1"/>
393     <xsd:enumeration value="triangleCircle2"/>
394     <xsd:enumeration value="shapes1"/>
395     <xsd:enumeration value="shapes2"/>
396     <xsd:enumeration value="twistedLines1"/>
397     <xsd:enumeration value="twistedLines2"/>
398     <xsd:enumeration value="vine"/>
399     <xsd:enumeration value="waveline"/>
400     <xsd:enumeration value="weavingAngles"/>
401     <xsd:enumeration value="weavingBraid"/>
402     <xsd:enumeration value="weavingRibbon"/>
403     <xsd:enumeration value="weavingStrips"/>
404     <xsd:enumeration value="whiteFlowers"/>

```

```

405     <xsd:enumeration value="woodwork"/>
406     <xsd:enumeration value="xIllusions"/>
407     <xsd:enumeration value="zanyTriangles"/>
408     <xsd:enumeration value="zigZag"/>
409     <xsd:enumeration value="zigZagStitch"/>
410     <xsd:enumeration value="custom"/>
411 </xsd:restriction>
412 </xsd:simpleType>
413 <xsd:complexType name="CT_Border">
414     <xsd:attribute name="val" type="ST_Border" use="required"/>
415     <xsd:attribute name="color" type="ST_HexColor" use="optional"/>
416     <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
417     <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
418     <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
419     <xsd:attribute name="sz" type="ST_EighthPointMeasure" use="optional"/>
420     <xsd:attribute name="space" type="ST_PointMeasure" use="optional"/>
421     <xsd:attribute name="shadow" type="s:ST_OnOff" use="optional"/>
422     <xsd:attribute name="frame" type="s:ST_OnOff" use="optional"/>
423 </xsd:complexType>
424 <xsd:simpleType name="ST_Shd">
425     <xsd:restriction base="xsd:string">
426         <xsd:enumeration value="nil"/>
427         <xsd:enumeration value="clear"/>
428         <xsd:enumeration value="solid"/>
429         <xsd:enumeration value="horzStripe"/>
430         <xsd:enumeration value="vertStripe"/>
431         <xsd:enumeration value="reverseDiagStripe"/>
432         <xsd:enumeration value="diagStripe"/>
433         <xsd:enumeration value="horzCross"/>
434         <xsd:enumeration value="diagCross"/>
435         <xsd:enumeration value="thinHorzStripe"/>
436         <xsd:enumeration value="thinVertStripe"/>
437         <xsd:enumeration value="thinReverseDiagStripe"/>
438         <xsd:enumeration value="thinDiagStripe"/>
439         <xsd:enumeration value="thinHorzCross"/>
440         <xsd:enumeration value="thinDiagCross"/>
441         <xsd:enumeration value="pct5"/>
442         <xsd:enumeration value="pct10"/>
443         <xsd:enumeration value="pct12"/>
444         <xsd:enumeration value="pct15"/>
445         <xsd:enumeration value="pct20"/>
446         <xsd:enumeration value="pct25"/>
447         <xsd:enumeration value="pct30"/>
448         <xsd:enumeration value="pct35"/>
449         <xsd:enumeration value="pct37"/>
450         <xsd:enumeration value="pct40"/>
451         <xsd:enumeration value="pct45"/>
452         <xsd:enumeration value="pct50"/>
453         <xsd:enumeration value="pct55"/>
454         <xsd:enumeration value="pct60"/>
455         <xsd:enumeration value="pct62"/>
456         <xsd:enumeration value="pct65"/>
457         <xsd:enumeration value="pct70"/>

```

```

458         <xsd:enumeration value="pct75"/>
459         <xsd:enumeration value="pct80"/>
460         <xsd:enumeration value="pct85"/>
461         <xsd:enumeration value="pct87"/>
462         <xsd:enumeration value="pct90"/>
463         <xsd:enumeration value="pct95"/>
464     </xsd:restriction>
465 </xsd:simpleType>
466 <xsd:complexType name="CT_Shdt">
467     <xsd:attribute name="val" type="ST_Shdt" use="required"/>
468     <xsd:attribute name="color" type="ST_HexColor" use="optional"/>
469     <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
470     <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
471     <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
472     <xsd:attribute name="fill" type="ST_HexColor" use="optional"/>
473     <xsd:attribute name="themeFill" type="ST_ThemeColor" use="optional"/>
474     <xsd:attribute name="themeFillTint" type="ST_UcharHexNumber" use="optional"/>
475     <xsd:attribute name="themeFillShade" type="ST_UcharHexNumber" use="optional"/>
476 </xsd:complexType>
477 <xsd:complexType name="CT_VerticalAlignRun">
478     <xsd:attribute name="val" type="s:ST_VerticalAlignRun" use="required"/>
479 </xsd:complexType>
480 <xsd:complexType name="CT_FitText">
481     <xsd:attribute name="val" type="s:ST_TwipsMeasure" use="required"/>
482     <xsd:attribute name="id" type="ST_DecimalNumber" use="optional"/>
483 </xsd:complexType>
484 <xsd:simpleType name="ST_Em">
485     <xsd:restriction base="xsd:string">
486         <xsd:enumeration value="none"/>
487         <xsd:enumeration value="dot"/>
488         <xsd:enumeration value="comma"/>
489         <xsd:enumeration value="circle"/>
490         <xsd:enumeration value="underDot"/>
491     </xsd:restriction>
492 </xsd:simpleType>
493 <xsd:complexType name="CT_Em">
494     <xsd:attribute name="val" type="ST_Em" use="required"/>
495 </xsd:complexType>
496 <xsd:complexType name="CT_Language">
497     <xsd:attribute name="val" type="s:ST_Lang" use="optional"/>
498     <xsd:attribute name="eastAsia" type="s:ST_Lang" use="optional"/>
499     <xsd:attribute name="bidi" type="s:ST_Lang" use="optional"/>
500 </xsd:complexType>
501 <xsd:simpleType name="ST_CombineBrackets">
502     <xsd:restriction base="xsd:string">
503         <xsd:enumeration value="none"/>
504         <xsd:enumeration value="round"/>
505         <xsd:enumeration value="square"/>
506         <xsd:enumeration value="angle"/>
507         <xsd:enumeration value="curly"/>
508     </xsd:restriction>
509 </xsd:simpleType>
510 <xsd:complexType name="CT_EastAsianLayout">

```

```

511     <xsd:attribute name="id" type="ST_DecimalNumber" use="optional"/>
512     <xsd:attribute name="combine" type="s:ST_OnOff" use="optional"/>
513     <xsd:attribute name="combineBrackets" type="ST_CombineBrackets" use="optional"/>
514     <xsd:attribute name="vert" type="s:ST_OnOff" use="optional"/>
515     <xsd:attribute name="vertCompress" type="s:ST_OnOff" use="optional"/>
516 </xsd:complexType>
517 <xsd:simpleType name="ST_HeightRule">
518     <xsd:restriction base="xsd:string">
519         <xsd:enumeration value="auto"/>
520         <xsd:enumeration value="exact"/>
521         <xsd:enumeration value="atLeast"/>
522     </xsd:restriction>
523 </xsd:simpleType>
524 <xsd:simpleType name="ST_Wrap">
525     <xsd:restriction base="xsd:string">
526         <xsd:enumeration value="auto"/>
527         <xsd:enumeration value="notBeside"/>
528         <xsd:enumeration value="around"/>
529         <xsd:enumeration value="tight"/>
530         <xsd:enumeration value="through"/>
531         <xsd:enumeration value="none"/>
532     </xsd:restriction>
533 </xsd:simpleType>
534 <xsd:simpleType name="ST_VAnchor">
535     <xsd:restriction base="xsd:string">
536         <xsd:enumeration value="text"/>
537         <xsd:enumeration value="margin"/>
538         <xsd:enumeration value="page"/>
539     </xsd:restriction>
540 </xsd:simpleType>
541 <xsd:simpleType name="ST_HAnchor">
542     <xsd:restriction base="xsd:string">
543         <xsd:enumeration value="text"/>
544         <xsd:enumeration value="margin"/>
545         <xsd:enumeration value="page"/>
546     </xsd:restriction>
547 </xsd:simpleType>
548 <xsd:simpleType name="ST_DropCap">
549     <xsd:restriction base="xsd:string">
550         <xsd:enumeration value="none"/>
551         <xsd:enumeration value="drop"/>
552         <xsd:enumeration value="margin"/>
553     </xsd:restriction>
554 </xsd:simpleType>
555 <xsd:complexType name="CT_FramePr">
556     <xsd:attribute name="dropCap" type="ST_DropCap" use="optional"/>
557     <xsd:attribute name="lines" type="ST_DecimalNumber" use="optional"/>
558     <xsd:attribute name="w" type="s:ST_TwipsMeasure" use="optional"/>
559     <xsd:attribute name="h" type="s:ST_TwipsMeasure" use="optional"/>
560     <xsd:attribute name="vSpace" type="s:ST_TwipsMeasure" use="optional"/>
561     <xsd:attribute name="hSpace" type="s:ST_TwipsMeasure" use="optional"/>
562     <xsd:attribute name="wrap" type="ST_Wrap" use="optional"/>
563     <xsd:attribute name="hAnchor" type="ST_HAnchor" use="optional"/>

```

```

564     <xsd:attribute name="vAnchor" type="ST_VAnchor" use="optional"/>
565     <xsd:attribute name="x" type="ST_SignedTwipsMeasure" use="optional"/>
566     <xsd:attribute name="xAlign" type="s:ST_XAlign" use="optional"/>
567     <xsd:attribute name="y" type="ST_SignedTwipsMeasure" use="optional"/>
568     <xsd:attribute name="yAlign" type="s:ST_YAlign" use="optional"/>
569     <xsd:attribute name="hRule" type="ST_HeightRule" use="optional"/>
570     <xsd:attribute name="anchorLock" type="s:ST_OnOff" use="optional"/>
571 </xsd:complexType>
572 <xsd:simpleType name="ST_TabJc">
573     <xsd:restriction base="xsd:string">
574         <xsd:enumeration value="clear"/>
575         <xsd:enumeration value="start"/>
576         <xsd:enumeration value="center"/>
577         <xsd:enumeration value="end"/>
578         <xsd:enumeration value="decimal"/>
579         <xsd:enumeration value="bar"/>
580         <xsd:enumeration value="num"/>
581         <xsd:enumeration value="left"/>
582         <xsd:enumeration value="right"/>
583     </xsd:restriction>
584 </xsd:simpleType>
585 <xsd:simpleType name="ST_TabTlc">
586     <xsd:restriction base="xsd:string">
587         <xsd:enumeration value="none"/>
588         <xsd:enumeration value="dot"/>
589         <xsd:enumeration value="hyphen"/>
590         <xsd:enumeration value="underscore"/>
591         <xsd:enumeration value="heavy"/>
592         <xsd:enumeration value="middleDot"/>
593     </xsd:restriction>
594 </xsd:simpleType>
595 <xsd:complexType name="CT_TabStop">
596     <xsd:attribute name="val" type="ST_TabJc" use="required"/>
597     <xsd:attribute name="leader" type="ST_TabTlc" use="optional"/>
598     <xsd:attribute name="pos" type="ST_SignedTwipsMeasure" use="required"/>
599 </xsd:complexType>
600 <xsd:simpleType name="ST_LineSpacingRule">
601     <xsd:restriction base="xsd:string">
602         <xsd:enumeration value="auto"/>
603         <xsd:enumeration value="exact"/>
604         <xsd:enumeration value="atLeast"/>
605     </xsd:restriction>
606 </xsd:simpleType>
607 <xsd:complexType name="CT_Spacing">
608     <xsd:attribute name="before" type="s:ST_TwipsMeasure" use="optional"/>
609     <xsd:attribute name="beforeLines" type="ST_DecimalNumber" use="optional"/>
610     <xsd:attribute name="beforeAutospacing" type="s:ST_OnOff" use="optional"/>
611     <xsd:attribute name="after" type="s:ST_TwipsMeasure" use="optional"/>
612     <xsd:attribute name="afterLines" type="ST_DecimalNumber" use="optional"/>
613     <xsd:attribute name="afterAutospacing" type="s:ST_OnOff" use="optional"/>
614     <xsd:attribute name="line" type="ST_SignedTwipsMeasure" use="optional"/>
615     <xsd:attribute name="lineRule" type="ST_LineSpacingRule" use="optional"/>
616 </xsd:complexType>

```



```

617 <xsd:complexType name="CT_Ind">
618   <xsd:attribute name="start" type="ST_SignedTwipsMeasure" use="optional"/>
619   <xsd:attribute name="startChars" type="ST_DecimalNumber" use="optional"/>
620   <xsd:attribute name="end" type="ST_SignedTwipsMeasure" use="optional"/>
621   <xsd:attribute name="endChars" type="ST_DecimalNumber" use="optional"/>
622   <xsd:attribute name="left" type="ST_SignedTwipsMeasure" use="optional"/>
623   <xsd:attribute name="leftChars" type="ST_DecimalNumber" use="optional"/>
624   <xsd:attribute name="right" type="ST_SignedTwipsMeasure" use="optional"/>
625   <xsd:attribute name="rightChars" type="ST_DecimalNumber" use="optional"/>
626   <xsd:attribute name="hanging" type="s:ST_TwipsMeasure" use="optional"/>
627   <xsd:attribute name="hangingChars" type="ST_DecimalNumber" use="optional"/>
628   <xsd:attribute name="firstLine" type="s:ST_TwipsMeasure" use="optional"/>
629   <xsd:attribute name="firstLineChars" type="ST_DecimalNumber" use="optional"/>
630 </xsd:complexType>
631 <xsd:simpleType name="ST_Jc">
632   <xsd:restriction base="xsd:string">
633     <xsd:enumeration value="start"/>
634     <xsd:enumeration value="center"/>
635     <xsd:enumeration value="end"/>
636     <xsd:enumeration value="both"/>
637     <xsd:enumeration value="mediumKashida"/>
638     <xsd:enumeration value="distribute"/>
639     <xsd:enumeration value="numTab"/>
640     <xsd:enumeration value="highKashida"/>
641     <xsd:enumeration value="lowKashida"/>
642     <xsd:enumeration value="thaiDistribute"/>
643     <xsd:enumeration value="left"/>
644     <xsd:enumeration value="right"/>
645   </xsd:restriction>
646 </xsd:simpleType>
647 <xsd:simpleType name="ST_JcTable">
648   <xsd:restriction base="xsd:string">
649     <xsd:enumeration value="center"/>
650     <xsd:enumeration value="end"/>
651     <xsd:enumeration value="left"/>
652     <xsd:enumeration value="right"/>
653     <xsd:enumeration value="start"/>
654   </xsd:restriction>
655 </xsd:simpleType>
656 <xsd:complexType name="CT_Jc">
657   <xsd:attribute name="val" type="ST_Jc" use="required"/>
658 </xsd:complexType>
659 <xsd:complexType name="CT_JcTable">
660   <xsd:attribute name="val" type="ST_JcTable" use="required"/>
661 </xsd:complexType>
662 <xsd:simpleType name="ST_View">
663   <xsd:restriction base="xsd:string">
664     <xsd:enumeration value="none"/>
665     <xsd:enumeration value="print"/>
666     <xsd:enumeration value="outline"/>
667     <xsd:enumeration value="masterPages"/>
668     <xsd:enumeration value="normal"/>
669     <xsd:enumeration value="web"/>

```

```

670     </xsd:restriction>
671 </xsd:simpleType>
672 <xsd:complexType name="CT_View">
673     <xsd:attribute name="val" type="ST_View" use="required"/>
674 </xsd:complexType>
675 <xsd:simpleType name="ST_Zoom">
676     <xsd:restriction base="xsd:string">
677         <xsd:enumeration value="none"/>
678         <xsd:enumeration value="fullPage"/>
679         <xsd:enumeration value="bestFit"/>
680         <xsd:enumeration value="textFit"/>
681     </xsd:restriction>
682 </xsd:simpleType>
683 <xsd:complexType name="CT_Zoom">
684     <xsd:attribute name="val" type="ST_Zoom" use="optional"/>
685     <xsd:attribute name="percent" type="ST_DecimalNumberOrPercent" use="required"/>
686 </xsd:complexType>
687 <xsd:complexType name="CT_WritingStyle">
688     <xsd:attribute name="lang" type="s:ST_Lang" use="required"/>
689     <xsd:attribute name="vendorID" type="s:ST_String" use="required"/>
690     <xsd:attribute name="dllVersion" type="s:ST_String" use="required"/>
691     <xsd:attribute name="nlCheck" type="s:ST_OnOff" use="optional"/>
692     <xsd:attribute name="checkStyle" type="s:ST_OnOff" use="required"/>
693     <xsd:attribute name="appName" type="s:ST_String" use="required"/>
694 </xsd:complexType>
695 <xsd:simpleType name="ST_Proof">
696     <xsd:restriction base="xsd:string">
697         <xsd:enumeration value="clean"/>
698         <xsd:enumeration value="dirty"/>
699     </xsd:restriction>
700 </xsd:simpleType>
701 <xsd:complexType name="CT_Proof">
702     <xsd:attribute name="spelling" type="ST_Proof" use="optional"/>
703     <xsd:attribute name="grammar" type="ST_Proof" use="optional"/>
704 </xsd:complexType>
705 <xsd:simpleType name="ST_DocType">
706     <xsd:restriction base="xsd:string"/>
707 </xsd:simpleType>
708 <xsd:complexType name="CT_DocType">
709     <xsd:attribute name="val" type="ST_DocType" use="required"/>
710 </xsd:complexType>
711 <xsd:simpleType name="ST_DocProtect">
712     <xsd:restriction base="xsd:string">
713         <xsd:enumeration value="none"/>
714         <xsd:enumeration value="readOnly"/>
715         <xsd:enumeration value="comments"/>
716         <xsd:enumeration value="trackedChanges"/>
717         <xsd:enumeration value="forms"/>
718     </xsd:restriction>
719 </xsd:simpleType>
720 <xsd:attributeGroup name="AG_Password">
721     <xsd:attribute name="algorithmName" type="s:ST_String" use="optional"/>
722     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>

```

```

723     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
724     <xsd:attribute name="spinCount" type="ST DecimalNumber" use="optional"/>
725 </xsd:attributeGroup>
726 <xsd:attributeGroup name="AG_TransitionalPassword">
727     <xsd:attribute name="cryptProviderType" type="s:ST CryptProv"/>
728     <xsd:attribute name="cryptAlgorithmClass" type="s:ST AlgClass"/>
729     <xsd:attribute name="cryptAlgorithmType" type="s:ST AlgType"/>
730     <xsd:attribute name="cryptAlgorithmSid" type="ST DecimalNumber"/>
731     <xsd:attribute name="cryptSpinCount" type="ST DecimalNumber"/>
732     <xsd:attribute name="cryptProvider" type="s:ST String"/>
733     <xsd:attribute name="algIdExt" type="ST LongHexNumber"/>
734     <xsd:attribute name="algIdExtSource" type="s:ST String"/>
735     <xsd:attribute name="cryptProviderTypeExt" type="ST LongHexNumber"/>
736     <xsd:attribute name="cryptProviderTypeExtSource" type="s:ST String"/>
737     <xsd:attribute name="hash" type="xsd:base64Binary"/>
738     <xsd:attribute name="salt" type="xsd:base64Binary"/>
739 </xsd:attributeGroup>
740 <xsd:complexType name="CT_DocProtect">
741     <xsd:attribute name="edit" type="ST DocProtect" use="optional"/>
742     <xsd:attribute name="formatting" type="s:ST OnOff" use="optional"/>
743     <xsd:attribute name="enforcement" type="s:ST OnOff"/>
744     <xsd:attributeGroup ref="AG Password"/>
745     <xsd:attributeGroup ref="AG TransitionalPassword"/>
746 </xsd:complexType>
747 <xsd:simpleType name="ST_MailMergeDocType">
748     <xsd:restriction base="xsd:string">
749         <xsd:enumeration value="catalog"/>
750         <xsd:enumeration value="envelopes"/>
751         <xsd:enumeration value="mailingLabels"/>
752         <xsd:enumeration value="formLetters"/>
753         <xsd:enumeration value="email"/>
754         <xsd:enumeration value="fax"/>
755     </xsd:restriction>
756 </xsd:simpleType>
757 <xsd:complexType name="CT_MailMergeDocType">
758     <xsd:attribute name="val" type="ST MailMergeDocType" use="required"/>
759 </xsd:complexType>
760 <xsd:simpleType name="ST_MailMergeDataType">
761     <xsd:restriction base="xsd:string"/>
762 </xsd:simpleType>
763 <xsd:complexType name="CT_MailMergeDataType">
764     <xsd:attribute name="val" type="ST MailMergeDataType" use="required"/>
765 </xsd:complexType>
766 <xsd:simpleType name="ST_MailMergeDest">
767     <xsd:restriction base="xsd:string">
768         <xsd:enumeration value="newDocument"/>
769         <xsd:enumeration value="printer"/>
770         <xsd:enumeration value="email"/>
771         <xsd:enumeration value="fax"/>
772     </xsd:restriction>
773 </xsd:simpleType>
774 <xsd:complexType name="CT_MailMergeDest">
775     <xsd:attribute name="val" type="ST MailMergeDest" use="required"/>

```

```

776 </xsd:complexType>
777 <xsd:simpleType name="ST_MailMergeOdsoFMDFieldType">
778   <xsd:restriction base="xsd:string">
779     <xsd:enumeration value="null"/>
780     <xsd:enumeration value="dbColumn"/>
781   </xsd:restriction>
782 </xsd:simpleType>
783 <xsd:complexType name="CT_MailMergeOdsoFMDFieldType">
784   <xsd:attribute name="val" type="ST_MailMergeOdsoFMDFieldType" use="required"/>
785 </xsd:complexType>
786 <xsd:complexType name="CT_TrackChangesView">
787   <xsd:attribute name="markup" type="s:ST_OnOff" use="optional"/>
788   <xsd:attribute name="comments" type="s:ST_OnOff" use="optional"/>
789   <xsd:attribute name="insDel" type="s:ST_OnOff" use="optional"/>
790   <xsd:attribute name="formatting" type="s:ST_OnOff" use="optional"/>
791   <xsd:attribute name="inkAnnotations" type="s:ST_OnOff" use="optional"/>
792 </xsd:complexType>
793 <xsd:complexType name="CT_Kinsoku">
794   <xsd:attribute name="lang" type="s:ST_Lang" use="required"/>
795   <xsd:attribute name="val" type="s:ST_String" use="required"/>
796 </xsd:complexType>
797 <xsd:simpleType name="ST_TextDirection">
798   <xsd:restriction base="xsd:string">
799     <xsd:enumeration value="tb"/>
800     <xsd:enumeration value="rl"/>
801     <xsd:enumeration value="lr"/>
802     <xsd:enumeration value="tbV"/>
803     <xsd:enumeration value="rlV"/>
804     <xsd:enumeration value="lrV"/>
805     <xsd:enumeration value="btLr"/>
806     <xsd:enumeration value="lrTb"/>
807     <xsd:enumeration value="lrTbV"/>
808     <xsd:enumeration value="tbLrV"/>
809     <xsd:enumeration value="tbRl"/>
810     <xsd:enumeration value="tbRlV"/>
811   </xsd:restriction>
812 </xsd:simpleType>
813 <xsd:complexType name="CT_TextDirection">
814   <xsd:attribute name="val" type="ST_TextDirection" use="required"/>
815 </xsd:complexType>
816 <xsd:simpleType name="ST_TextAlignment">
817   <xsd:restriction base="xsd:string">
818     <xsd:enumeration value="top"/>
819     <xsd:enumeration value="center"/>
820     <xsd:enumeration value="baseline"/>
821     <xsd:enumeration value="bottom"/>
822     <xsd:enumeration value="auto"/>
823   </xsd:restriction>
824 </xsd:simpleType>
825 <xsd:complexType name="CT_TextAlignment">
826   <xsd:attribute name="val" type="ST_TextAlignment" use="required"/>
827 </xsd:complexType>
828 <xsd:simpleType name="ST_DisplacedByCustomXml">

```

```

829     <xsd:restriction base="xsd:string">
830         <xsd:enumeration value="next"/>
831         <xsd:enumeration value="prev"/>
832     </xsd:restriction>
833 </xsd:simpleType>
834 <xsd:simpleType name="ST_AnnotationVMerge">
835     <xsd:restriction base="xsd:string">
836         <xsd:enumeration value="cont"/>
837         <xsd:enumeration value="rest"/>
838     </xsd:restriction>
839 </xsd:simpleType>
840 <xsd:complexType name="CT_Markup">
841     <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
842 </xsd:complexType>
843 <xsd:complexType name="CT_TrackChange">
844     <xsd:complexContent>
845         <xsd:extension base="CT_Markup">
846             <xsd:attribute name="author" type="s:ST_String" use="required"/>
847             <xsd:attribute name="date" type="ST_DateTime" use="optional"/>
848         </xsd:extension>
849     </xsd:complexContent>
850 </xsd:complexType>
851 <xsd:complexType name="CT_CellMergeTrackChange">
852     <xsd:complexContent>
853         <xsd:extension base="CT_TrackChange">
854             <xsd:attribute name="vMerge" type="ST_AnnotationVMerge" use="optional"/>
855             <xsd:attribute name="vMergeOrig" type="ST_AnnotationVMerge" use="optional"/>
856         </xsd:extension>
857     </xsd:complexContent>
858 </xsd:complexType>
859 <xsd:complexType name="CT_TrackChangeRange">
860     <xsd:complexContent>
861         <xsd:extension base="CT_TrackChange">
862             <xsd:attribute name="displacedByCustomXml" type="ST_DisplacedByCustomXml"
863                 use="optional"/>
864         </xsd:extension>
865     </xsd:complexContent>
866 </xsd:complexType>
867 <xsd:complexType name="CT_MarkupRange">
868     <xsd:complexContent>
869         <xsd:extension base="CT_Markup">
870             <xsd:attribute name="displacedByCustomXml" type="ST_DisplacedByCustomXml"
871                 use="optional"/>
872         </xsd:extension>
873     </xsd:complexContent>
874 </xsd:complexType>
875 <xsd:complexType name="CT_BookmarkRange">
876     <xsd:complexContent>
877         <xsd:extension base="CT_MarkupRange">
878             <xsd:attribute name="colFirst" type="ST_DecimalNumber" use="optional"/>
879             <xsd:attribute name="colLast" type="ST_DecimalNumber" use="optional"/>
880         </xsd:extension>
881     </xsd:complexContent>

```

```

882 </xsd:complexType>
883 <xsd:complexType name="CT_Bookmark">
884   <xsd:complexContent>
885     <xsd:extension base="CT_BookmarkRange">
886       <xsd:attribute name="name" type="s:ST_String" use="required"/>
887     </xsd:extension>
888   </xsd:complexContent>
889 </xsd:complexType>
890 <xsd:complexType name="CT_MoveBookmark">
891   <xsd:complexContent>
892     <xsd:extension base="CT_Bookmark">
893       <xsd:attribute name="author" type="s:ST_String" use="required"/>
894       <xsd:attribute name="date" type="ST_DateTime" use="required"/>
895     </xsd:extension>
896   </xsd:complexContent>
897 </xsd:complexType>
898 <xsd:complexType name="CT_Comment">
899   <xsd:complexContent>
900     <xsd:extension base="CT_TrackChange">
901       <xsd:sequence>
902         <xsd:group ref="EG_BlockLevelElts" minOccurs="0" maxOccurs="unbounded"/>
903       </xsd:sequence>
904       <xsd:attribute name="initials" type="s:ST_String" use="optional"/>
905     </xsd:extension>
906   </xsd:complexContent>
907 </xsd:complexType>
908 <xsd:complexType name="CT_TrackChangeNumbering">
909   <xsd:complexContent>
910     <xsd:extension base="CT_TrackChange">
911       <xsd:attribute name="original" type="s:ST_String" use="optional"/>
912     </xsd:extension>
913   </xsd:complexContent>
914 </xsd:complexType>
915 <xsd:complexType name="CT_TblPrExChange">
916   <xsd:complexContent>
917     <xsd:extension base="CT_TrackChange">
918       <xsd:sequence>
919         <xsd:element name="tblPrEx" type="CT_TblPrExBase" minOccurs="1"/>
920       </xsd:sequence>
921     </xsd:extension>
922   </xsd:complexContent>
923 </xsd:complexType>
924 <xsd:complexType name="CT_TcPrChange">
925   <xsd:complexContent>
926     <xsd:extension base="CT_TrackChange">
927       <xsd:sequence>
928         <xsd:element name="tcPr" type="CT_TcPrInner" minOccurs="1"/>
929       </xsd:sequence>
930     </xsd:extension>
931   </xsd:complexContent>
932 </xsd:complexType>
933 <xsd:complexType name="CT_TrPrChange">
934   <xsd:complexContent>

```

```

935     <xsd:extension base="CT_TrackChange">
936         <xsd:sequence>
937             <xsd:element name="trPr" type="CT_TrPrBase" minOccurs="1"/>
938         </xsd:sequence>
939     </xsd:extension>
940 </xsd:complexContent>
941 </xsd:complexType>
942 <xsd:complexType name="CT_TblGridChange">
943     <xsd:complexContent>
944         <xsd:extension base="CT_Markup">
945             <xsd:sequence>
946                 <xsd:element name="tblGrid" type="CT_TblGridBase"/>
947             </xsd:sequence>
948         </xsd:extension>
949     </xsd:complexContent>
950 </xsd:complexType>
951 <xsd:complexType name="CT_TblPrChange">
952     <xsd:complexContent>
953         <xsd:extension base="CT_TrackChange">
954             <xsd:sequence>
955                 <xsd:element name="tblPr" type="CT_TblPrBase"/>
956             </xsd:sequence>
957         </xsd:extension>
958     </xsd:complexContent>
959 </xsd:complexType>
960 <xsd:complexType name="CT_SectPrChange">
961     <xsd:complexContent>
962         <xsd:extension base="CT_TrackChange">
963             <xsd:sequence>
964                 <xsd:element name="sectPr" type="CT_SectPrBase" minOccurs="0"/>
965             </xsd:sequence>
966         </xsd:extension>
967     </xsd:complexContent>
968 </xsd:complexType>
969 <xsd:complexType name="CT_PPrChange">
970     <xsd:complexContent>
971         <xsd:extension base="CT_TrackChange">
972             <xsd:sequence>
973                 <xsd:element name="pPr" type="CT_PPrBase" minOccurs="1"/>
974             </xsd:sequence>
975         </xsd:extension>
976     </xsd:complexContent>
977 </xsd:complexType>
978 <xsd:complexType name="CT_RPrChange">
979     <xsd:complexContent>
980         <xsd:extension base="CT_TrackChange">
981             <xsd:sequence>
982                 <xsd:element name="rPr" type="CT_RPrOriginal" minOccurs="1"/>
983             </xsd:sequence>
984         </xsd:extension>
985     </xsd:complexContent>
986 </xsd:complexType>
987 <xsd:complexType name="CT_ParaRPrChange">

```

```

988     <xsd:complexContent>
989         <xsd:extension base="CT_TrackChange">
990             <xsd:sequence>
991                 <xsd:element name="rPr" type="CT_ParaRPrOriginal" minOccurs="1"/>
992             </xsd:sequence>
993         </xsd:extension>
994     </xsd:complexContent>
995 </xsd:complexType>
996 <xsd:complexType name="CT_RunTrackChange">
997     <xsd:complexContent>
998         <xsd:extension base="CT_TrackChange">
999             <xsd:choice minOccurs="0" maxOccurs="unbounded">
1000                 <xsd:group ref="EG_ContentRunContent"/>
1001                 <xsd:group ref="m:EG_OMathMathElements"/>
1002             </xsd:choice>
1003         </xsd:extension>
1004     </xsd:complexContent>
1005 </xsd:complexType>
1006 <xsd:group name="EG_PContentMath">
1007     <xsd:choice>
1008         <xsd:group ref="EG_PContentBase" minOccurs="0" maxOccurs="unbounded" />
1009         <xsd:group ref="EG_ContentRunContentBase" minOccurs="0"
1010             maxOccurs="unbounded" />
1011     </xsd:choice>
1012 </xsd:group>
1013 <xsd:group name="EG_PContentBase">
1014     <xsd:choice>
1015         <xsd:element name="customXml" type="CT_CustomXmlRun"/>
1016         <xsd:element name="fldSimple" type="CT_SimpleField" minOccurs="0"
1017             maxOccurs="unbounded"/>
1018         <xsd:element name="hyperlink" type="CT_Hyperlink"/>
1019     </xsd:choice>
1020 </xsd:group>
1021 <xsd:group name="EG_ContentRunContentBase">
1022     <xsd:choice>
1023         <xsd:element name="smartTag" type="CT_SmartTagRun"/>
1024         <xsd:element name="sdt" type="CT_SdtRun"/>
1025         <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded" />
1026     </xsd:choice>
1027 </xsd:group>
1028 <xsd:group name="EG_CellMarkupElements">
1029     <xsd:choice>
1030         <xsd:element name="cellIns" type="CT_TrackChange" minOccurs="0"/>
1031         <xsd:element name="cellDel" type="CT_TrackChange" minOccurs="0"/>
1032         <xsd:element name="cellMerge" type="CT_CellMergeTrackChange" minOccurs="0"/>
1033     </xsd:choice>
1034 </xsd:group>
1035 <xsd:group name="EG_RangeMarkupElements">
1036     <xsd:choice>
1037         <xsd:element name="bookmarkStart" type="CT_Bookmark"/>
1038         <xsd:element name="bookmarkEnd" type="CT_MarkupRange"/>
1039         <xsd:element name="moveFromRangeStart" type="CT_MoveBookmark"/>
1040         <xsd:element name="moveFromRangeEnd" type="CT_MarkupRange"/>

```



```

1041     <xsd:element name="moveToRangeStart" type="CT_MoveBookmark"/>
1042     <xsd:element name="moveToRangeEnd" type="CT_MarkupRange"/>
1043     <xsd:element name="commentRangeStart" type="CT_MarkupRange"/>
1044     <xsd:element name="commentRangeEnd" type="CT_MarkupRange"/>
1045     <xsd:element name="customXmlInsRangeStart" type="CT_TrackChange"/>
1046     <xsd:element name="customXmlInsRangeEnd" type="CT_Markup"/>
1047     <xsd:element name="customXmlDelRangeStart" type="CT_TrackChange"/>
1048     <xsd:element name="customXmlDelRangeEnd" type="CT_Markup"/>
1049     <xsd:element name="customXmlMoveFromRangeStart" type="CT_TrackChange"/>
1050     <xsd:element name="customXmlMoveFromRangeEnd" type="CT_Markup"/>
1051     <xsd:element name="customXmlMoveToRangeStart" type="CT_TrackChange"/>
1052     <xsd:element name="customXmlMoveToRangeEnd" type="CT_Markup"/>
1053   </xsd:choice>
1054 </xsd:group>
1055 <xsd:complexType name="CT_NumPr">
1056   <xsd:sequence>
1057     <xsd:element name="ilvl" type="CT_DecimalNumber" minOccurs="0"/>
1058     <xsd:element name="numId" type="CT_DecimalNumber" minOccurs="0"/>
1059     <xsd:element name="numberingChange" type="CT_TrackChangeNumbering" minOccurs="0"/>
1060     <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
1061   </xsd:sequence>
1062 </xsd:complexType>
1063 <xsd:complexType name="CT_PBdr">
1064   <xsd:sequence>
1065     <xsd:element name="top" type="CT_Border" minOccurs="0"/>
1066     <xsd:element name="left" type="CT_Border" minOccurs="0"/>
1067     <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
1068     <xsd:element name="right" type="CT_Border" minOccurs="0"/>
1069     <xsd:element name="between" type="CT_Border" minOccurs="0"/>
1070     <xsd:element name="bar" type="CT_Border" minOccurs="0"/>
1071   </xsd:sequence>
1072 </xsd:complexType>
1073 <xsd:complexType name="CT_Tabs">
1074   <xsd:sequence>
1075     <xsd:element name="tab" type="CT_TabStop" minOccurs="1" maxOccurs="unbounded"/>
1076   </xsd:sequence>
1077 </xsd:complexType>
1078 <xsd:simpleType name="ST_TextboxTightWrap">
1079   <xsd:restriction base="xsd:string">
1080     <xsd:enumeration value="none"/>
1081     <xsd:enumeration value="allLines"/>
1082     <xsd:enumeration value="firstAndLastLine"/>
1083     <xsd:enumeration value="firstLineOnly"/>
1084     <xsd:enumeration value="lastLineOnly"/>
1085   </xsd:restriction>
1086 </xsd:simpleType>
1087 <xsd:complexType name="CT_TextboxTightWrap">
1088   <xsd:attribute name="val" type="ST_TextboxTightWrap" use="required"/>
1089 </xsd:complexType>
1090 <xsd:complexType name="CT_PPr">
1091   <xsd:complexContent>
1092     <xsd:extension base="CT_PPrBase">
1093       <xsd:sequence>

```

```

1094         <xsd:element name="rPr" type="CT_ParaRPr" minOccurs="0"/>
1095         <xsd:element name="sectPr" type="CT_SectPr" minOccurs="0"/>
1096         <xsd:element name="pPrChange" type="CT_PPrChange" minOccurs="0"/>
1097     </xsd:sequence>
1098 </xsd:extension>
1099 </xsd:complexContent>
1100 </xsd:complexType>
1101 <xsd:complexType name="CT_PPrBase">
1102     <xsd:sequence>
1103         <xsd:element name="pStyle" type="CT_String" minOccurs="0"/>
1104         <xsd:element name="keepNext" type="CT_OnOff" minOccurs="0"/>
1105         <xsd:element name="keepLines" type="CT_OnOff" minOccurs="0"/>
1106         <xsd:element name="pageBreakBefore" type="CT_OnOff" minOccurs="0"/>
1107         <xsd:element name="framePr" type="CT_FramePr" minOccurs="0"/>
1108         <xsd:element name="widowControl" type="CT_OnOff" minOccurs="0"/>
1109         <xsd:element name="numPr" type="CT_NumPr" minOccurs="0"/>
1110         <xsd:element name="suppressLineNumbers" type="CT_OnOff" minOccurs="0"/>
1111         <xsd:element name="pBdr" type="CT_PBdr" minOccurs="0"/>
1112         <xsd:element name="shd" type="CT_Shdt" minOccurs="0"/>
1113         <xsd:element name="tabs" type="CT_Tabs" minOccurs="0"/>
1114         <xsd:element name="suppressAutoHyphens" type="CT_OnOff" minOccurs="0"/>
1115         <xsd:element name="kinsoku" type="CT_OnOff" minOccurs="0"/>
1116         <xsd:element name="wordWrap" type="CT_OnOff" minOccurs="0"/>
1117         <xsd:element name="overflowPunct" type="CT_OnOff" minOccurs="0"/>
1118         <xsd:element name="topLinePunct" type="CT_OnOff" minOccurs="0"/>
1119         <xsd:element name="autoSpaceDE" type="CT_OnOff" minOccurs="0"/>
1120         <xsd:element name="autoSpaceDN" type="CT_OnOff" minOccurs="0"/>
1121         <xsd:element name="bidi" type="CT_OnOff" minOccurs="0"/>
1122         <xsd:element name="adjustRightInd" type="CT_OnOff" minOccurs="0"/>
1123         <xsd:element name="snapToGrid" type="CT_OnOff" minOccurs="0"/>
1124         <xsd:element name="spacing" type="CT_Spacing" minOccurs="0"/>
1125         <xsd:element name="ind" type="CT_Ind" minOccurs="0"/>
1126         <xsd:element name="contextualSpacing" type="CT_OnOff" minOccurs="0"/>
1127         <xsd:element name="mirrorIndents" type="CT_OnOff" minOccurs="0"/>
1128         <xsd:element name="suppressOverlap" type="CT_OnOff" minOccurs="0"/>
1129         <xsd:element name="jc" type="CT_Jc" minOccurs="0"/>
1130         <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0"/>
1131         <xsd:element name="textAlignment" type="CT_TextAlignment" minOccurs="0"/>
1132         <xsd:element name="textboxTightWrap" type="CT_TextboxTightWrap" minOccurs="0"/>
1133         <xsd:element name="outlineLvl" type="CT_DecimalNumber" minOccurs="0"/>
1134         <xsd:element name="divId" type="CT_DecimalNumber" minOccurs="0"/>
1135         <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
1136     </xsd:sequence>
1137 </xsd:complexType>
1138 <xsd:complexType name="CT_PPrGeneral">
1139     <xsd:complexContent>
1140         <xsd:extension base="CT_PPrBase">
1141             <xsd:sequence>
1142                 <xsd:element name="pPrChange" type="CT_PPrChange" minOccurs="0"/>
1143             </xsd:sequence>
1144         </xsd:extension>
1145     </xsd:complexContent>
1146 </xsd:complexType>

```

```

1147 <xsd:complexType name="CT_Control">
1148   <xsd:attribute name="name" type="s:ST String" use="optional"/>
1149   <xsd:attribute name="shapeid" type="s:ST String" use="optional"/>
1150   <xsd:attribute ref="r:id" use="optional"/>
1151 </xsd:complexType>
1152 <xsd:complexType name="CT_Background">
1153   <xsd:sequence>
1154     <xsd:sequence maxOccurs="unbounded">
1155       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1156         maxOccurs="unbounded"/>
1157       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1158         minOccurs="0" maxOccurs="unbounded"/>
1159     </xsd:sequence>
1160     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0"/>
1161   </xsd:sequence>
1162   <xsd:attribute name="color" type="ST HexColor" use="optional"/>
1163   <xsd:attribute name="themeColor" type="ST ThemeColor" use="optional"/>
1164   <xsd:attribute name="themeTint" type="ST UcharHexNumber" use="optional"/>
1165   <xsd:attribute name="themeShade" type="ST UcharHexNumber" use="optional"/>
1166 </xsd:complexType>
1167 <xsd:complexType name="CT_Rel">
1168   <xsd:attribute ref="r:id" use="required"/>
1169 </xsd:complexType>
1170 <xsd:complexType name="CT_Object">
1171   <xsd:sequence>
1172     <xsd:sequence maxOccurs="unbounded">
1173       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1174         maxOccurs="unbounded"/>
1175       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1176         minOccurs="0" maxOccurs="unbounded"/>
1177     </xsd:sequence>
1178     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0"/>
1179     <xsd:choice minOccurs="0">
1180       <xsd:element name="control" type="CT_Control"/>
1181       <xsd:element name="objectLink" type="CT_ObjectLink"/>
1182       <xsd:element name="objectEmbed" type="CT_ObjectEmbed"/>
1183       <xsd:element name="movie" type="CT_Rel"/>
1184     </xsd:choice>
1185   </xsd:sequence>
1186   <xsd:attribute name="dxaOrig" type="s:ST TwipsMeasure" use="optional"/>
1187   <xsd:attribute name="dyaOrig" type="s:ST TwipsMeasure" use="optional"/>
1188 </xsd:complexType>
1189 <xsd:complexType name="CT_Picture">
1190   <xsd:sequence>
1191     <xsd:sequence maxOccurs="unbounded">
1192       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1193         maxOccurs="unbounded"/>
1194       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1195         minOccurs="0" maxOccurs="unbounded"/>
1196     </xsd:sequence>
1197     <xsd:element name="movie" type="CT_Rel" minOccurs="0"/>
1198     <xsd:element name="control" type="CT_Control" minOccurs="0"/>
1199   </xsd:sequence>

```

```

1200 </xsd:complexType>
1201 <xsd:complexType name="CT_ObjectEmbed">
1202   <xsd:attribute name="drawAspect" type="ST_ObjectDrawAspect" use="optional"/>
1203   <xsd:attribute ref="r:id" use="required"/>
1204   <xsd:attribute name="progId" type="s:ST_String" use="optional"/>
1205   <xsd:attribute name="shapeId" type="s:ST_String" use="optional"/>
1206   <xsd:attribute name="fieldCodes" type="s:ST_String" use="optional"/>
1207 </xsd:complexType>
1208 <xsd:simpleType name="ST_ObjectDrawAspect">
1209   <xsd:restriction base="xsd:string">
1210     <xsd:enumeration value="content"/>
1211     <xsd:enumeration value="icon"/>
1212   </xsd:restriction>
1213 </xsd:simpleType>
1214 <xsd:complexType name="CT_ObjectLink">
1215   <xsd:complexContent>
1216     <xsd:extension base="CT_ObjectEmbed">
1217       <xsd:attribute name="updateMode" type="ST_ObjectUpdateMode" use="required"/>
1218       <xsd:attribute name="lockedField" type="s:ST_OnOff" use="optional"/>
1219     </xsd:extension>
1220   </xsd:complexContent>
1221 </xsd:complexType>
1222 <xsd:simpleType name="ST_ObjectUpdateMode">
1223   <xsd:restriction base="xsd:string">
1224     <xsd:enumeration value="always"/>
1225     <xsd:enumeration value="onCall"/>
1226   </xsd:restriction>
1227 </xsd:simpleType>
1228 <xsd:complexType name="CT_Drawing">
1229   <xsd:choice minOccurs="1" maxOccurs="unbounded">
1230     <xsd:element ref="wp:anchor" minOccurs="0"/>
1231     <xsd:element ref="wp:inline" minOccurs="0"/>
1232   </xsd:choice>
1233 </xsd:complexType>
1234 <xsd:complexType name="CT_SimpleField">
1235   <xsd:sequence>
1236     <xsd:element name="fldData" type="CT_Text" minOccurs="0" maxOccurs="1"/>
1237     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
1238   </xsd:sequence>
1239   <xsd:attribute name="instr" type="s:ST_String" use="required"/>
1240   <xsd:attribute name="fldLock" type="s:ST_OnOff"/>
1241   <xsd:attribute name="dirty" type="s:ST_OnOff"/>
1242 </xsd:complexType>
1243 <xsd:simpleType name="ST_FldCharType">
1244   <xsd:restriction base="xsd:string">
1245     <xsd:enumeration value="begin"/>
1246     <xsd:enumeration value="separate"/>
1247     <xsd:enumeration value="end"/>
1248   </xsd:restriction>
1249 </xsd:simpleType>
1250 <xsd:simpleType name="ST_InfoTextType">
1251   <xsd:restriction base="xsd:string">
1252     <xsd:enumeration value="text"/>

```

```

1253     <xsd:enumeration value="autoText"/>
1254   </xsd:restriction>
1255 </xsd:simpleType>
1256 <xsd:simpleType name="ST_FFHelpTextVal">
1257   <xsd:restriction base="xsd:string">
1258     <xsd:maxLength value="256"/>
1259   </xsd:restriction>
1260 </xsd:simpleType>
1261 <xsd:simpleType name="ST_FFStatusTextVal">
1262   <xsd:restriction base="xsd:string">
1263     <xsd:maxLength value="140"/>
1264   </xsd:restriction>
1265 </xsd:simpleType>
1266 <xsd:simpleType name="ST_FFName">
1267   <xsd:restriction base="xsd:string">
1268     <xsd:maxLength value="65"/>
1269   </xsd:restriction>
1270 </xsd:simpleType>
1271 <xsd:simpleType name="ST_FFTextType">
1272   <xsd:restriction base="xsd:string">
1273     <xsd:enumeration value="regular"/>
1274     <xsd:enumeration value="number"/>
1275     <xsd:enumeration value="date"/>
1276     <xsd:enumeration value="currentTime"/>
1277     <xsd:enumeration value="currentDate"/>
1278     <xsd:enumeration value="calculated"/>
1279   </xsd:restriction>
1280 </xsd:simpleType>
1281 <xsd:complexType name="CT_FFTextType">
1282   <xsd:attribute name="val" type="ST_FFTextType" use="required"/>
1283 </xsd:complexType>
1284 <xsd:complexType name="CT_FFName">
1285   <xsd:attribute name="val" type="ST_FFName"/>
1286 </xsd:complexType>
1287 <xsd:complexType name="CT_FldChar">
1288   <xsd:choice>
1289     <xsd:element name="fldData" type="CT_Text" minOccurs="0" maxOccurs="1"/>
1290     <xsd:element name="ffData" type="CT_FFData" minOccurs="0" maxOccurs="1"/>
1291     <xsd:element name="numberingChange" type="CT_TrackChangeNumbering" minOccurs="0"/>
1292   </xsd:choice>
1293   <xsd:attribute name="fldCharType" type="ST_FldCharType" use="required"/>
1294   <xsd:attribute name="fldLock" type="s:ST_OnOff"/>
1295   <xsd:attribute name="dirty" type="s:ST_OnOff"/>
1296 </xsd:complexType>
1297 <xsd:complexType name="CT_Hyperlink">
1298   <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
1299   <xsd:attribute name="tgtFrame" type="s:ST_String" use="optional"/>
1300   <xsd:attribute name="tooltip" type="s:ST_String" use="optional"/>
1301   <xsd:attribute name="docLocation" type="s:ST_String" use="optional"/>
1302   <xsd:attribute name="history" type="s:ST_OnOff" use="optional"/>
1303   <xsd:attribute name="anchor" type="s:ST_String" use="optional"/>
1304   <xsd:attribute ref="r:id"/>
1305 </xsd:complexType>

```

```

1306 <xsd:complexType name="CT_FFData">
1307   <xsd:choice maxOccurs="unbounded">
1308     <xsd:element name="name" type="CT_FFName"/>
1309     <xsd:element name="label" type="CT_DecimalNumber" minOccurs="0"/>
1310     <xsd:element name="tabIndex" type="CT_UnsignedDecimalNumber" minOccurs="0"/>
1311     <xsd:element name="enabled" type="CT_OnOff"/>
1312     <xsd:element name="calcOnExit" type="CT_OnOff"/>
1313     <xsd:element name="entryMacro" type="CT_MacroName" minOccurs="0" maxOccurs="1"/>
1314     <xsd:element name="exitMacro" type="CT_MacroName" minOccurs="0" maxOccurs="1"/>
1315     <xsd:element name="helpText" type="CT_FFHelpText" minOccurs="0" maxOccurs="1"/>
1316     <xsd:element name="statusText" type="CT_FFStatusText" minOccurs="0" maxOccurs="1"/>
1317     <xsd:choice>
1318       <xsd:element name="checkBox" type="CT_FFCheckBox"/>
1319       <xsd:element name="ddList" type="CT_FFDDList"/>
1320       <xsd:element name="textInput" type="CT_FFTextInput"/>
1321     </xsd:choice>
1322   </xsd:choice>
1323 </xsd:complexType>
1324 <xsd:complexType name="CT_FFHelpText">
1325   <xsd:attribute name="type" type="ST_InfoTextType"/>
1326   <xsd:attribute name="val" type="ST_FFHelpTextVal"/>
1327 </xsd:complexType>
1328 <xsd:complexType name="CT_FFStatusText">
1329   <xsd:attribute name="type" type="ST_InfoTextType"/>
1330   <xsd:attribute name="val" type="ST_FFStatusTextVal"/>
1331 </xsd:complexType>
1332 <xsd:complexType name="CT_FFCheckBox">
1333   <xsd:sequence>
1334     <xsd:choice>
1335       <xsd:element name="size" type="CT_HpsMeasure"/>
1336       <xsd:element name="sizeAuto" type="CT_OnOff"/>
1337     </xsd:choice>
1338     <xsd:element name="default" type="CT_OnOff" minOccurs="0"/>
1339     <xsd:element name="checked" type="CT_OnOff" minOccurs="0"/>
1340   </xsd:sequence>
1341 </xsd:complexType>
1342 <xsd:complexType name="CT_FFDDList">
1343   <xsd:sequence>
1344     <xsd:element name="result" type="CT_DecimalNumber" minOccurs="0"/>
1345     <xsd:element name="default" type="CT_DecimalNumber" minOccurs="0"/>
1346     <xsd:element name="listEntry" type="CT_String" minOccurs="0" maxOccurs="unbounded"/>
1347   </xsd:sequence>
1348 </xsd:complexType>
1349 <xsd:complexType name="CT_FFTextInput">
1350   <xsd:sequence>
1351     <xsd:element name="type" type="CT_FFTextType" minOccurs="0"/>
1352     <xsd:element name="default" type="CT_String" minOccurs="0"/>
1353     <xsd:element name="maxLength" type="CT_DecimalNumber" minOccurs="0"/>
1354     <xsd:element name="format" type="CT_String" minOccurs="0"/>
1355   </xsd:sequence>
1356 </xsd:complexType>
1357 <xsd:simpleType name="ST_SectionMark">
1358   <xsd:restriction base="xsd:string">

```

```

1359         <xsd:enumeration value="nextPage"/>
1360         <xsd:enumeration value="nextColumn"/>
1361         <xsd:enumeration value="continuous"/>
1362         <xsd:enumeration value="evenPage"/>
1363         <xsd:enumeration value="oddPage"/>
1364     </xsd:restriction>
1365 </xsd:simpleType>
1366 <xsd:complexType name="CT_SectType">
1367     <xsd:attribute name="val" type="ST_SectionMark"/>
1368 </xsd:complexType>
1369 <xsd:complexType name="CT_PaperSource">
1370     <xsd:attribute name="first" type="ST_DecimalNumber"/>
1371     <xsd:attribute name="other" type="ST_DecimalNumber"/>
1372 </xsd:complexType>
1373 <xsd:simpleType name="ST_NumberFormat">
1374     <xsd:restriction base="xsd:string">
1375         <xsd:enumeration value="decimal"/>
1376         <xsd:enumeration value="upperRoman"/>
1377         <xsd:enumeration value="lowerRoman"/>
1378         <xsd:enumeration value="upperLetter"/>
1379         <xsd:enumeration value="lowerLetter"/>
1380         <xsd:enumeration value="ordinal"/>
1381         <xsd:enumeration value="cardinalText"/>
1382         <xsd:enumeration value="ordinalText"/>
1383         <xsd:enumeration value="hex"/>
1384         <xsd:enumeration value="chicago"/>
1385         <xsd:enumeration value="ideographDigital"/>
1386         <xsd:enumeration value="japaneseCounting"/>
1387         <xsd:enumeration value="aiueo"/>
1388         <xsd:enumeration value="iroha"/>
1389         <xsd:enumeration value="decimalFullWidth"/>
1390         <xsd:enumeration value="decimalHalfWidth"/>
1391         <xsd:enumeration value="japaneseLegal"/>
1392         <xsd:enumeration value="japaneseDigitalTenThousand"/>
1393         <xsd:enumeration value="decimalEnclosedCircle"/>
1394         <xsd:enumeration value="decimalFullWidth2"/>
1395         <xsd:enumeration value="aiueoFullWidth"/>
1396         <xsd:enumeration value="irohaFullWidth"/>
1397         <xsd:enumeration value="decimalZero"/>
1398         <xsd:enumeration value="bullet"/>
1399         <xsd:enumeration value="ganada"/>
1400         <xsd:enumeration value="chosung"/>
1401         <xsd:enumeration value="decimalEnclosedFullstop"/>
1402         <xsd:enumeration value="decimalEnclosedParen"/>
1403         <xsd:enumeration value="decimalEnclosedCircleChinese"/>
1404         <xsd:enumeration value="ideographEnclosedCircle"/>
1405         <xsd:enumeration value="ideographTraditional"/>
1406         <xsd:enumeration value="ideographZodiac"/>
1407         <xsd:enumeration value="ideographZodiacTraditional"/>
1408         <xsd:enumeration value="taiwaneseCounting"/>
1409         <xsd:enumeration value="ideographLegalTraditional"/>
1410         <xsd:enumeration value="taiwaneseCountingThousand"/>
1411         <xsd:enumeration value="taiwaneseDigital"/>

```

```

1412     <xsd:enumeration value="chineseCounting"/>
1413     <xsd:enumeration value="chineseLegalSimplified"/>
1414     <xsd:enumeration value="chineseCountingThousand"/>
1415     <xsd:enumeration value="koreanDigital"/>
1416     <xsd:enumeration value="koreanCounting"/>
1417     <xsd:enumeration value="koreanLegal"/>
1418     <xsd:enumeration value="koreanDigital2"/>
1419     <xsd:enumeration value="vietnameseCounting"/>
1420     <xsd:enumeration value="russianLower"/>
1421     <xsd:enumeration value="russianUpper"/>
1422     <xsd:enumeration value="none"/>
1423     <xsd:enumeration value="numberInDash"/>
1424     <xsd:enumeration value="hebrew1"/>
1425     <xsd:enumeration value="hebrew2"/>
1426     <xsd:enumeration value="arabicAlpha"/>
1427     <xsd:enumeration value="arabicAbjad"/>
1428     <xsd:enumeration value="hindiVowels"/>
1429     <xsd:enumeration value="hindiConsonants"/>
1430     <xsd:enumeration value="hindiNumbers"/>
1431     <xsd:enumeration value="hindiCounting"/>
1432     <xsd:enumeration value="thaiLetters"/>
1433     <xsd:enumeration value="thaiNumbers"/>
1434     <xsd:enumeration value="thaiCounting"/>
1435     <xsd:enumeration value="bahtText"/>
1436     <xsd:enumeration value="dollarText"/>
1437     <xsd:enumeration value="custom"/>
1438   </xsd:restriction>
1439 </xsd:simpleType>
1440 <xsd:simpleType name="ST_PageOrientation">
1441   <xsd:restriction base="xsd:string">
1442     <xsd:enumeration value="portrait"/>
1443     <xsd:enumeration value="landscape"/>
1444   </xsd:restriction>
1445 </xsd:simpleType>
1446 <xsd:complexType name="CT_PageSz">
1447   <xsd:attribute name="w" type="s:ST_TwipsMeasure"/>
1448   <xsd:attribute name="h" type="s:ST_TwipsMeasure"/>
1449   <xsd:attribute name="orient" type="ST_PageOrientation" use="optional"/>
1450   <xsd:attribute name="code" type="ST_DecimalNumber" use="optional"/>
1451 </xsd:complexType>
1452 <xsd:complexType name="CT_PageMar">
1453   <xsd:attribute name="top" type="ST_SignedTwipsMeasure" use="required"/>
1454   <xsd:attribute name="right" type="s:ST_TwipsMeasure" use="required"/>
1455   <xsd:attribute name="bottom" type="ST_SignedTwipsMeasure" use="required"/>
1456   <xsd:attribute name="left" type="s:ST_TwipsMeasure" use="required"/>
1457   <xsd:attribute name="header" type="s:ST_TwipsMeasure" use="required"/>
1458   <xsd:attribute name="footer" type="s:ST_TwipsMeasure" use="required"/>
1459   <xsd:attribute name="gutter" type="s:ST_TwipsMeasure" use="required"/>
1460 </xsd:complexType>
1461 <xsd:simpleType name="ST_PageBorderZOrder">
1462   <xsd:restriction base="xsd:string">
1463     <xsd:enumeration value="front"/>
1464     <xsd:enumeration value="back"/>

```



```

1465     </xsd:restriction>
1466 </xsd:simpleType>
1467 <xsd:simpleType name="ST_PageBorderDisplay">
1468     <xsd:restriction base="xsd:string">
1469         <xsd:enumeration value="allPages"/>
1470         <xsd:enumeration value="firstPage"/>
1471         <xsd:enumeration value="notFirstPage"/>
1472     </xsd:restriction>
1473 </xsd:simpleType>
1474 <xsd:simpleType name="ST_PageBorderOffset">
1475     <xsd:restriction base="xsd:string">
1476         <xsd:enumeration value="page"/>
1477         <xsd:enumeration value="text"/>
1478     </xsd:restriction>
1479 </xsd:simpleType>
1480 <xsd:complexType name="CT_PageBorders">
1481     <xsd:sequence>
1482         <xsd:element name="top" type="CT_TopPageBorder" minOccurs="0"/>
1483         <xsd:element name="left" type="CT_PageBorder" minOccurs="0"/>
1484         <xsd:element name="bottom" type="CT_BottomPageBorder" minOccurs="0"/>
1485         <xsd:element name="right" type="CT_PageBorder" minOccurs="0"/>
1486     </xsd:sequence>
1487     <xsd:attribute name="zOrder" type="ST_PageBorderZOrder" use="optional"/>
1488     <xsd:attribute name="display" type="ST_PageBorderDisplay" use="optional"/>
1489     <xsd:attribute name="offsetFrom" type="ST_PageBorderOffset" use="optional"/>
1490 </xsd:complexType>
1491 <xsd:complexType name="CT_PageBorder">
1492     <xsd:complexContent>
1493         <xsd:extension base="CT_Border">
1494             <xsd:attribute ref="r:id" use="optional"/>
1495         </xsd:extension>
1496     </xsd:complexContent>
1497 </xsd:complexType>
1498 <xsd:complexType name="CT_BottomPageBorder">
1499     <xsd:complexContent>
1500         <xsd:extension base="CT_PageBorder">
1501             <xsd:attribute ref="r:bottomLeft" use="optional"/>
1502             <xsd:attribute ref="r:bottomRight" use="optional"/>
1503         </xsd:extension>
1504     </xsd:complexContent>
1505 </xsd:complexType>
1506 <xsd:complexType name="CT_TopPageBorder">
1507     <xsd:complexContent>
1508         <xsd:extension base="CT_PageBorder">
1509             <xsd:attribute ref="r:topLeft" use="optional"/>
1510             <xsd:attribute ref="r:topRight" use="optional"/>
1511         </xsd:extension>
1512     </xsd:complexContent>
1513 </xsd:complexType>
1514 <xsd:simpleType name="ST_ChapterSep">
1515     <xsd:restriction base="xsd:string">
1516         <xsd:enumeration value="hyphen"/>
1517         <xsd:enumeration value="period"/>

```

```

1518         <xsd:enumeration value="colon"/>
1519         <xsd:enumeration value="emDash"/>
1520         <xsd:enumeration value="enDash"/>
1521     </xsd:restriction>
1522 </xsd:simpleType>
1523 <xsd:simpleType name="ST_LineNumberRestart">
1524     <xsd:restriction base="xsd:string">
1525         <xsd:enumeration value="newPage"/>
1526         <xsd:enumeration value="newSection"/>
1527         <xsd:enumeration value="continuous"/>
1528     </xsd:restriction>
1529 </xsd:simpleType>
1530 <xsd:complexType name="CT_LineNumber">
1531     <xsd:attribute name="countBy" type="ST_DecimalNumber" use="optional"/>
1532     <xsd:attribute name="start" type="ST_DecimalNumber" use="optional"/>
1533     <xsd:attribute name="distance" type="s:ST_TwipsMeasure" use="optional"/>
1534     <xsd:attribute name="restart" type="ST_LineNumberRestart" use="optional"/>
1535 </xsd:complexType>
1536 <xsd:complexType name="CT_PageNumber">
1537     <xsd:attribute name="fmt" type="ST_NumberFormat" use="optional"/>
1538     <xsd:attribute name="start" type="ST_DecimalNumber" use="optional"/>
1539     <xsd:attribute name="chapStyle" type="ST_DecimalNumber" use="optional"/>
1540     <xsd:attribute name="chapSep" type="ST_ChapterSep" use="optional"/>
1541 </xsd:complexType>
1542 <xsd:complexType name="CT_Column">
1543     <xsd:attribute name="w" type="s:ST_TwipsMeasure" use="optional"/>
1544     <xsd:attribute name="space" type="s:ST_TwipsMeasure" use="optional"/>
1545 </xsd:complexType>
1546 <xsd:complexType name="CT_Columns">
1547     <xsd:sequence minOccurs="0">
1548         <xsd:element name="col" type="CT_Column" maxOccurs="45"/>
1549     </xsd:sequence>
1550     <xsd:attribute name="equalWidth" type="s:ST_OnOff" use="optional"/>
1551     <xsd:attribute name="space" type="s:ST_TwipsMeasure" use="optional"/>
1552     <xsd:attribute name="num" type="ST_DecimalNumber" use="optional"/>
1553     <xsd:attribute name="sep" type="s:ST_OnOff" use="optional"/>
1554 </xsd:complexType>
1555 <xsd:simpleType name="ST_VerticalJc">
1556     <xsd:restriction base="xsd:string">
1557         <xsd:enumeration value="top"/>
1558         <xsd:enumeration value="center"/>
1559         <xsd:enumeration value="both"/>
1560         <xsd:enumeration value="bottom"/>
1561     </xsd:restriction>
1562 </xsd:simpleType>
1563 <xsd:complexType name="CT_VerticalJc">
1564     <xsd:attribute name="val" type="ST_VerticalJc" use="required"/>
1565 </xsd:complexType>
1566 <xsd:simpleType name="ST_DocGrid">
1567     <xsd:restriction base="xsd:string">
1568         <xsd:enumeration value="default"/>
1569         <xsd:enumeration value="lines"/>
1570         <xsd:enumeration value="linesAndChars"/>

```

```

1571     <xsd:enumeration value="snapToChars"/>
1572   </xsd:restriction>
1573 </xsd:simpleType>
1574 <xsd:complexType name="CT_DocGrid">
1575   <xsd:attribute name="type" type="ST_DocGrid"/>
1576   <xsd:attribute name="linePitch" type="ST_DecimalNumber"/>
1577   <xsd:attribute name="charSpace" type="ST_DecimalNumber"/>
1578 </xsd:complexType>
1579 <xsd:simpleType name="ST_HdrFtr">
1580   <xsd:restriction base="xsd:string">
1581     <xsd:enumeration value="even"/>
1582     <xsd:enumeration value="default"/>
1583     <xsd:enumeration value="first"/>
1584   </xsd:restriction>
1585 </xsd:simpleType>
1586 <xsd:simpleType name="ST_FtnEdn">
1587   <xsd:restriction base="xsd:string">
1588     <xsd:enumeration value="normal"/>
1589     <xsd:enumeration value="separator"/>
1590     <xsd:enumeration value="continuationSeparator"/>
1591     <xsd:enumeration value="continuationNotice"/>
1592   </xsd:restriction>
1593 </xsd:simpleType>
1594 <xsd:complexType name="CT_HdrFtrRef">
1595   <xsd:complexContent>
1596     <xsd:extension base="CT_Rel">
1597       <xsd:attribute name="type" type="ST_HdrFtr" use="required"/>
1598     </xsd:extension>
1599   </xsd:complexContent>
1600 </xsd:complexType>
1601 <xsd:group name="EG_HdrFtrReferences">
1602   <xsd:choice>
1603     <xsd:element name="headerReference" type="CT_HdrFtrRef" minOccurs="0"/>
1604     <xsd:element name="footerReference" type="CT_HdrFtrRef" minOccurs="0"/>
1605   </xsd:choice>
1606 </xsd:group>
1607 <xsd:complexType name="CT_HdrFtr">
1608   <xsd:group ref="EG_BlockLevelElt" minOccurs="1" maxOccurs="unbounded"/>
1609 </xsd:complexType>
1610 <xsd:group name="EG_SectPrContents">
1611   <xsd:sequence>
1612     <xsd:element name="footnotePr" type="CT_FtnProps" minOccurs="0"/>
1613     <xsd:element name="endnotePr" type="CT_EdnProps" minOccurs="0"/>
1614     <xsd:element name="type" type="CT_SectType" minOccurs="0"/>
1615     <xsd:element name="pgSz" type="CT_PageSz" minOccurs="0"/>
1616     <xsd:element name="pgMar" type="CT_PageMar" minOccurs="0"/>
1617     <xsd:element name="paperSrc" type="CT_PaperSource" minOccurs="0"/>
1618     <xsd:element name="pgBorders" type="CT_PageBorders" minOccurs="0"/>
1619     <xsd:element name="lnNumType" type="CT_LineNumber" minOccurs="0"/>
1620     <xsd:element name="pgNumType" type="CT_PageNumber" minOccurs="0"/>
1621     <xsd:element name="cols" type="CT_Columns" minOccurs="0"/>
1622     <xsd:element name="formProt" type="CT_OnOff" minOccurs="0"/>
1623     <xsd:element name="vAlign" type="CT_VerticalJc" minOccurs="0"/>

```

```

1624     <xsd:element name="noEndnote" type="CT_OnOff" minOccurs="0"/>
1625     <xsd:element name="titlePg" type="CT_OnOff" minOccurs="0"/>
1626     <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0"/>
1627     <xsd:element name="bidi" type="CT_OnOff" minOccurs="0"/>
1628     <xsd:element name="rtlGutter" type="CT_OnOff" minOccurs="0"/>
1629     <xsd:element name="docGrid" type="CT_DocGrid" minOccurs="0"/>
1630     <xsd:element name="printerSettings" type="CT_Rel" minOccurs="0"/>
1631   </xsd:sequence>
1632 </xsd:group>
1633 <xsd:attributeGroup name="AG_SectPrAttributes">
1634   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
1635   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
1636   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
1637   <xsd:attribute name="rsidSect" type="ST_LongHexNumber"/>
1638 </xsd:attributeGroup>
1639 <xsd:complexType name="CT_SectPrBase">
1640   <xsd:sequence>
1641     <xsd:group ref="EG_SectPrContents" minOccurs="0"/>
1642   </xsd:sequence>
1643   <xsd:attributeGroup ref="AG_SectPrAttributes"/>
1644 </xsd:complexType>
1645 <xsd:complexType name="CT_SectPr">
1646   <xsd:sequence>
1647     <xsd:group ref="EG_HdrFtrReferences" minOccurs="0" maxOccurs="6"/>
1648     <xsd:group ref="EG_SectPrContents" minOccurs="0"/>
1649     <xsd:element name="sectPrChange" type="CT_SectPrChange" minOccurs="0"/>
1650   </xsd:sequence>
1651   <xsd:attributeGroup ref="AG_SectPrAttributes"/>
1652 </xsd:complexType>
1653 <xsd:simpleType name="ST_BrType">
1654   <xsd:restriction base="xsd:string">
1655     <xsd:enumeration value="page"/>
1656     <xsd:enumeration value="column"/>
1657     <xsd:enumeration value="textWrapping"/>
1658   </xsd:restriction>
1659 </xsd:simpleType>
1660 <xsd:simpleType name="ST_BrClear">
1661   <xsd:restriction base="xsd:string">
1662     <xsd:enumeration value="none"/>
1663     <xsd:enumeration value="left"/>
1664     <xsd:enumeration value="right"/>
1665     <xsd:enumeration value="all"/>
1666   </xsd:restriction>
1667 </xsd:simpleType>
1668 <xsd:complexType name="CT_Br">
1669   <xsd:attribute name="type" type="ST_BrType" use="optional"/>
1670   <xsd:attribute name="clear" type="ST_BrClear" use="optional"/>
1671 </xsd:complexType>
1672 <xsd:simpleType name="ST_PTabAlignment">
1673   <xsd:restriction base="xsd:string">
1674     <xsd:enumeration value="left"/>
1675     <xsd:enumeration value="center"/>
1676     <xsd:enumeration value="right"/>

```

```

1677     </xsd:restriction>
1678 </xsd:simpleType>
1679 <xsd:simpleType name="ST_PTabRelativeTo">
1680     <xsd:restriction base="xsd:string">
1681         <xsd:enumeration value="margin"/>
1682         <xsd:enumeration value="indent"/>
1683     </xsd:restriction>
1684 </xsd:simpleType>
1685 <xsd:simpleType name="ST_PTabLeader">
1686     <xsd:restriction base="xsd:string">
1687         <xsd:enumeration value="none"/>
1688         <xsd:enumeration value="dot"/>
1689         <xsd:enumeration value="hyphen"/>
1690         <xsd:enumeration value="underscore"/>
1691         <xsd:enumeration value="middleDot"/>
1692     </xsd:restriction>
1693 </xsd:simpleType>
1694 <xsd:complexType name="CT_PTab">
1695     <xsd:attribute name="alignment" type="ST_PTabAlignment" use="required"/>
1696     <xsd:attribute name="relativeTo" type="ST_PTabRelativeTo" use="required"/>
1697     <xsd:attribute name="leader" type="ST_PTabLeader" use="required"/>
1698 </xsd:complexType>
1699 <xsd:complexType name="CT_Sym">
1700     <xsd:attribute name="font" type="s:ST_String"/>
1701     <xsd:attribute name="char" type="ST_ShortHexNumber"/>
1702 </xsd:complexType>
1703 <xsd:simpleType name="ST_ProofErr">
1704     <xsd:restriction base="xsd:string">
1705         <xsd:enumeration value="spellStart"/>
1706         <xsd:enumeration value="spellEnd"/>
1707         <xsd:enumeration value="gramStart"/>
1708         <xsd:enumeration value="gramEnd"/>
1709     </xsd:restriction>
1710 </xsd:simpleType>
1711 <xsd:complexType name="CT_ProofErr">
1712     <xsd:attribute name="type" type="ST_ProofErr" use="required"/>
1713 </xsd:complexType>
1714 <xsd:simpleType name="ST_EdGrp">
1715     <xsd:restriction base="xsd:string">
1716         <xsd:enumeration value="none"/>
1717         <xsd:enumeration value="everyone"/>
1718         <xsd:enumeration value="administrators"/>
1719         <xsd:enumeration value="contributors"/>
1720         <xsd:enumeration value="editors"/>
1721         <xsd:enumeration value="owners"/>
1722         <xsd:enumeration value="current"/>
1723     </xsd:restriction>
1724 </xsd:simpleType>
1725 <xsd:complexType name="CT_Perm">
1726     <xsd:attribute name="id" type="s:ST_String" use="required"/>
1727     <xsd:attribute name="displacedByCustomXml" type="ST_DisplacedByCustomXml" use="optional"/>
1728 </xsd:complexType>
1729 <xsd:complexType name="CT_PermStart">

```

```

1730     <xsd:complexContent>
1731       <xsd:extension base="CT_Perm">
1732         <xsd:attribute name="edGrp" type="ST_EdGrp" use="optional"/>
1733         <xsd:attribute name="ed" type="s:ST_String" use="optional"/>
1734         <xsd:attribute name="colFirst" type="ST_DecimalNumber" use="optional"/>
1735         <xsd:attribute name="colLast" type="ST_DecimalNumber" use="optional"/>
1736       </xsd:extension>
1737     </xsd:complexContent>
1738   </xsd:complexType>
1739   <xsd:complexType name="CT_Text">
1740     <xsd:simpleContent>
1741       <xsd:extension base="s:ST_String">
1742         <xsd:attribute ref="xml:space" use="optional"/>
1743       </xsd:extension>
1744     </xsd:simpleContent>
1745   </xsd:complexType>
1746   <xsd:group name="EG_RunInnerContent">
1747     <xsd:choice>
1748       <xsd:element name="br" type="CT_Br"/>
1749       <xsd:element name="t" type="CT_Text"/>
1750       <xsd:element name="contentPart" type="CT_Rel"/>
1751       <xsd:element name="delText" type="CT_Text"/>
1752       <xsd:element name="instrText" type="CT_Text"/>
1753       <xsd:element name="delInstrText" type="CT_Text"/>
1754       <xsd:element name="noBreakHyphen" type="CT_Empty"/>
1755       <xsd:element name="softHyphen" type="CT_Empty" minOccurs="0"/>
1756       <xsd:element name="dayShort" type="CT_Empty" minOccurs="0"/>
1757       <xsd:element name="monthShort" type="CT_Empty" minOccurs="0"/>
1758       <xsd:element name="yearShort" type="CT_Empty" minOccurs="0"/>
1759       <xsd:element name="dayLong" type="CT_Empty" minOccurs="0"/>
1760       <xsd:element name="monthLong" type="CT_Empty" minOccurs="0"/>
1761       <xsd:element name="yearLong" type="CT_Empty" minOccurs="0"/>
1762       <xsd:element name="annotationRef" type="CT_Empty" minOccurs="0"/>
1763       <xsd:element name="footnoteRef" type="CT_Empty" minOccurs="0"/>
1764       <xsd:element name="endnoteRef" type="CT_Empty" minOccurs="0"/>
1765       <xsd:element name="separator" type="CT_Empty" minOccurs="0"/>
1766       <xsd:element name="continuationSeparator" type="CT_Empty" minOccurs="0"/>
1767       <xsd:element name="sym" type="CT_Sym" minOccurs="0"/>
1768       <xsd:element name="pgNum" type="CT_Empty" minOccurs="0"/>
1769       <xsd:element name="cr" type="CT_Empty" minOccurs="0"/>
1770       <xsd:element name="tab" type="CT_Empty" minOccurs="0"/>
1771       <xsd:element name="object" type="CT_Object"/>
1772       <xsd:element name="pict" type="CT_Picture"/>
1773       <xsd:element name="fldChar" type="CT_FldChar"/>
1774       <xsd:element name="ruby" type="CT_Ruby"/>
1775       <xsd:element name="footnoteReference" type="CT_FtnEdnRef"/>
1776       <xsd:element name="endnoteReference" type="CT_FtnEdnRef"/>
1777       <xsd:element name="commentReference" type="CT_Markup"/>
1778       <xsd:element name="drawing" type="CT_Drawing"/>
1779       <xsd:element name="ptab" type="CT_PTab" minOccurs="0"/>
1780       <xsd:element name="lastRenderedPageBreak" type="CT_Empty" minOccurs="0" maxOccurs="1"/>
1781     </xsd:choice>
1782   </xsd:group>

```

```

1783 <xsd:complexType name="CT_R">
1784   <xsd:sequence>
1785     <xsd:group ref="EG_RPr" minOccurs="0"/>
1786     <xsd:group ref="EG_RunInnerContent" minOccurs="0" maxOccurs="unbounded"/>
1787   </xsd:sequence>
1788   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
1789   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
1790   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
1791 </xsd:complexType>
1792 <xsd:simpleType name="ST_Hint">
1793   <xsd:restriction base="xsd:string">
1794     <xsd:enumeration value="default"/>
1795     <xsd:enumeration value="eastAsia"/>
1796     <xsd:enumeration value="cs"/>
1797   </xsd:restriction>
1798 </xsd:simpleType>
1799 <xsd:simpleType name="ST_Theme">
1800   <xsd:restriction base="xsd:string">
1801     <xsd:enumeration value="majorEastAsia"/>
1802     <xsd:enumeration value="majorBidi"/>
1803     <xsd:enumeration value="majorAscii"/>
1804     <xsd:enumeration value="majorHAnsi"/>
1805     <xsd:enumeration value="minorEastAsia"/>
1806     <xsd:enumeration value="minorBidi"/>
1807     <xsd:enumeration value="minorAscii"/>
1808     <xsd:enumeration value="minorHAnsi"/>
1809   </xsd:restriction>
1810 </xsd:simpleType>
1811 <xsd:complexType name="CT_Fonts">
1812   <xsd:attribute name="hint" type="ST_Hint"/>
1813   <xsd:attribute name="ascii" type="s:ST_String"/>
1814   <xsd:attribute name="hAnsi" type="s:ST_String"/>
1815   <xsd:attribute name="eastAsia" type="s:ST_String"/>
1816   <xsd:attribute name="cs" type="s:ST_String"/>
1817   <xsd:attribute name="asciiTheme" type="ST_Theme"/>
1818   <xsd:attribute name="hAnsiTheme" type="ST_Theme"/>
1819   <xsd:attribute name="eastAsiaTheme" type="ST_Theme"/>
1820   <xsd:attribute name="cstheme" type="ST_Theme"/>
1821 </xsd:complexType>
1822 <xsd:group name="EG_RPrBase">
1823   <xsd:choice>
1824     <xsd:element name="rStyle" type="CT_String"/>
1825     <xsd:element name="rFonts" type="CT_Fonts"/>
1826     <xsd:element name="b" type="CT_OnOff"/>
1827     <xsd:element name="bCs" type="CT_OnOff"/>
1828     <xsd:element name="i" type="CT_OnOff"/>
1829     <xsd:element name="iCs" type="CT_OnOff"/>
1830     <xsd:element name="caps" type="CT_OnOff"/>
1831     <xsd:element name="smallCaps" type="CT_OnOff"/>
1832     <xsd:element name="strike" type="CT_OnOff"/>
1833     <xsd:element name="dstrike" type="CT_OnOff"/>
1834     <xsd:element name="outline" type="CT_OnOff"/>
1835     <xsd:element name="shadow" type="CT_OnOff"/>

```

```

1836     <xsd:element name="emboss" type="CT_OnOff"/>
1837     <xsd:element name="imprint" type="CT_OnOff"/>
1838     <xsd:element name="noProof" type="CT_OnOff"/>
1839     <xsd:element name="snapToGrid" type="CT_OnOff"/>
1840     <xsd:element name="vanish" type="CT_OnOff"/>
1841     <xsd:element name="webHidden" type="CT_OnOff"/>
1842     <xsd:element name="color" type="CT_Color"/>
1843     <xsd:element name="spacing" type="CT_SignedTwipsMeasure"/>
1844     <xsd:element name="w" type="CT_TextScale"/>
1845     <xsd:element name="kern" type="CT_HpsMeasure"/>
1846     <xsd:element name="position" type="CT_SignedHpsMeasure"/>
1847     <xsd:element name="sz" type="CT_HpsMeasure"/>
1848     <xsd:element name="szCs" type="CT_HpsMeasure"/>
1849     <xsd:element name="highlight" type="CT_Highlight"/>
1850     <xsd:element name="u" type="CT_Underline"/>
1851     <xsd:element name="effect" type="CT_TextEffect"/>
1852     <xsd:element name="bdr" type="CT_Border"/>
1853     <xsd:element name="shd" type="CT_Shadow"/>
1854     <xsd:element name="fitText" type="CT_FitText"/>
1855     <xsd:element name="vertAlign" type="CT_VerticalAlignRun"/>
1856     <xsd:element name="rtl" type="CT_OnOff"/>
1857     <xsd:element name="cs" type="CT_OnOff"/>
1858     <xsd:element name="em" type="CT_Em"/>
1859     <xsd:element name="lang" type="CT_Language"/>
1860     <xsd:element name="eastAsianLayout" type="CT_EastAsianLayout"/>
1861     <xsd:element name="specVanish" type="CT_OnOff"/>
1862     <xsd:element name="oMath" type="CT_OnOff"/>
1863   </xsd:choice>
1864 </xsd:group>
1865 <xsd:group name="EG_RPrContent">
1866   <xsd:sequence>
1867     <xsd:group ref="EG_RPrBase" minOccurs="0"/>
1868     <xsd:element name="rPrChange" type="CT_RPrChange" minOccurs="0"/>
1869   </xsd:sequence>
1870 </xsd:group>
1871 <xsd:complexType name="CT_RPr">
1872   <xsd:sequence>
1873     <xsd:group ref="EG_RPrContent" minOccurs="0"/>
1874   </xsd:sequence>
1875 </xsd:complexType>
1876 <xsd:group name="EG_RPr">
1877   <xsd:sequence>
1878     <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
1879   </xsd:sequence>
1880 </xsd:group>
1881 <xsd:group name="EG_RPrMath">
1882   <xsd:choice>
1883     <xsd:group ref="EG_RPr"/>
1884     <xsd:element name="ins" type="CT_MathCtrlIns"/>
1885     <xsd:element name="del" type="CT_MathCtrlDel"/>
1886   </xsd:choice>
1887 </xsd:group>
1888 <xsd:complexType name="CT_MathCtrlIns">

```



```

1889 <xsd:complexContent>
1890 <xsd:extension base="CT_TrackChange">
1891 <xsd:choice minOccurs="0">
1892 <xsd:element name="del" type="CT_RPrChange" minOccurs="1"/>
1893 <xsd:element name="rPr" type="CT_RPr" minOccurs="1"/>
1894 </xsd:choice>
1895 </xsd:extension>
1896 </xsd:complexContent>
1897 </xsd:complexType>
1898 <xsd:complexType name="CT_MathCtrlDel">
1899 <xsd:complexContent>
1900 <xsd:extension base="CT_TrackChange">
1901 <xsd:choice minOccurs="0">
1902 <xsd:element name="rPr" type="CT_RPr" minOccurs="1"/>
1903 </xsd:choice>
1904 </xsd:extension>
1905 </xsd:complexContent>
1906 </xsd:complexType>
1907 <xsd:complexType name="CT_RPrOriginal">
1908 <xsd:sequence>
1909 <xsd:group ref="EG_RPrBase" minOccurs="0" maxOccurs="unbounded"/>
1910 </xsd:sequence>
1911 </xsd:complexType>
1912 <xsd:complexType name="CT_ParaRPrOriginal">
1913 <xsd:sequence>
1914 <xsd:group ref="EG_ParaRPrTrackChanges" minOccurs="0"/>
1915 <xsd:group ref="EG_RPrBase" minOccurs="0" maxOccurs="unbounded"/>
1916 </xsd:sequence>
1917 </xsd:complexType>
1918 <xsd:complexType name="CT_ParaRPr">
1919 <xsd:sequence>
1920 <xsd:group ref="EG_ParaRPrTrackChanges" minOccurs="0"/>
1921 <xsd:group ref="EG_RPrBase" minOccurs="0" maxOccurs="unbounded"/>
1922 <xsd:element name="rPrChange" type="CT_ParaRPrChange" minOccurs="0"/>
1923 </xsd:sequence>
1924 </xsd:complexType>
1925 <xsd:group name="EG_ParaRPrTrackChanges">
1926 <xsd:sequence>
1927 <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
1928 <xsd:element name="del" type="CT_TrackChange" minOccurs="0"/>
1929 <xsd:element name="moveFrom" type="CT_TrackChange" minOccurs="0"/>
1930 <xsd:element name="moveTo" type="CT_TrackChange" minOccurs="0"/>
1931 </xsd:sequence>
1932 </xsd:group>
1933 <xsd:complexType name="CT_AltChunk">
1934 <xsd:sequence>
1935 <xsd:element name="altChunkPr" type="CT_AltChunkPr" minOccurs="0" maxOccurs="1"/>
1936 </xsd:sequence>
1937 <xsd:attribute ref="r:id" use="optional"/>
1938 </xsd:complexType>
1939 <xsd:complexType name="CT_AltChunkPr">
1940 <xsd:sequence>
1941 <xsd:element name="matchSrc" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>

```

```

1942     </xsd:sequence>
1943 </xsd:complexType>
1944 <xsd:simpleType name="ST_RubyAlign">
1945     <xsd:restriction base="xsd:string">
1946         <xsd:enumeration value="center"/>
1947         <xsd:enumeration value="distributeLetter"/>
1948         <xsd:enumeration value="distributeSpace"/>
1949         <xsd:enumeration value="left"/>
1950         <xsd:enumeration value="right"/>
1951         <xsd:enumeration value="rightVertical"/>
1952     </xsd:restriction>
1953 </xsd:simpleType>
1954 <xsd:complexType name="CT_RubyAlign">
1955     <xsd:attribute name="val" type="ST_RubyAlign" use="required"/>
1956 </xsd:complexType>
1957 <xsd:complexType name="CT_RubyPr">
1958     <xsd:sequence>
1959         <xsd:element name="rubyAlign" type="CT_RubyAlign"/>
1960         <xsd:element name="hps" type="CT_HpsMeasure"/>
1961         <xsd:element name="hpsRaise" type="CT_HpsMeasure"/>
1962         <xsd:element name="hpsBaseText" type="CT_HpsMeasure"/>
1963         <xsd:element name="lid" type="CT_Lang"/>
1964         <xsd:element name="dirty" type="CT_OnOff" minOccurs="0"/>
1965     </xsd:sequence>
1966 </xsd:complexType>
1967 <xsd:group name="EG_RubyContent">
1968     <xsd:choice>
1969         <xsd:element name="r" type="CT_R"/>
1970         <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
1971     </xsd:choice>
1972 </xsd:group>
1973 <xsd:complexType name="CT_RubyContent">
1974     <xsd:group ref="EG_RubyContent" minOccurs="0" maxOccurs="unbounded"/>
1975 </xsd:complexType>
1976 <xsd:complexType name="CT_Ruby">
1977     <xsd:sequence>
1978         <xsd:element name="rubyPr" type="CT_RubyPr"/>
1979         <xsd:element name="rt" type="CT_RubyContent"/>
1980         <xsd:element name="rubyBase" type="CT_RubyContent"/>
1981     </xsd:sequence>
1982 </xsd:complexType>
1983 <xsd:simpleType name="ST_Lock">
1984     <xsd:restriction base="xsd:string">
1985         <xsd:enumeration value="sdtLocked"/>
1986         <xsd:enumeration value="contentLocked"/>
1987         <xsd:enumeration value="unlocked"/>
1988         <xsd:enumeration value="sdtContentLocked"/>
1989     </xsd:restriction>
1990 </xsd:simpleType>
1991 <xsd:complexType name="CT_Lock">
1992     <xsd:attribute name="val" type="ST_Lock"/>
1993 </xsd:complexType>
1994 <xsd:complexType name="CT_SdtListItem">

```

```

1995     <xsd:attribute name="displayText" type="s:ST String"/>
1996     <xsd:attribute name="value" type="s:ST String"/>
1997 </xsd:complexType>
1998 <xsd:simpleType name="ST_SdtDateMappingType">
1999     <xsd:restriction base="xsd:string">
2000         <xsd:enumeration value="text"/>
2001         <xsd:enumeration value="date"/>
2002         <xsd:enumeration value="dateTime"/>
2003     </xsd:restriction>
2004 </xsd:simpleType>
2005 <xsd:complexType name="CT_SdtDateMappingType">
2006     <xsd:attribute name="val" type="ST_SdtDateMappingType"/>
2007 </xsd:complexType>
2008 <xsd:complexType name="CT_CalendarType">
2009     <xsd:attribute name="val" type="s:ST_CalendarType"/>
2010 </xsd:complexType>
2011 <xsd:complexType name="CT_SdtDate">
2012     <xsd:sequence>
2013         <xsd:element name="dateFormat" type="CT String" minOccurs="0"/>
2014         <xsd:element name="lid" type="CT Lang" minOccurs="0"/>
2015         <xsd:element name="storeMappedDataAs" type="CT_SdtDateMappingType" minOccurs="0"/>
2016         <xsd:element name="calendar" type="CT_CalendarType" minOccurs="0"/>
2017     </xsd:sequence>
2018     <xsd:attribute name="fullDate" type="ST DateTime" use="optional"/>
2019 </xsd:complexType>
2020 <xsd:complexType name="CT_SdtComboBox">
2021     <xsd:sequence>
2022         <xsd:element name="listItem" type="CT_SdtListItem" minOccurs="0" maxOccurs="unbounded"/>
2023     </xsd:sequence>
2024     <xsd:attribute name="lastValue" type="s:ST String" use="optional"/>
2025 </xsd:complexType>
2026 <xsd:complexType name="CT_SdtDocPart">
2027     <xsd:sequence>
2028         <xsd:element name="docPartGallery" type="CT String" minOccurs="0"/>
2029         <xsd:element name="docPartCategory" type="CT String" minOccurs="0"/>
2030         <xsd:element name="docPartUnique" type="CT OnOff" minOccurs="0"/>
2031     </xsd:sequence>
2032 </xsd:complexType>
2033 <xsd:complexType name="CT_SdtDropDownList">
2034     <xsd:sequence>
2035         <xsd:element name="listItem" type="CT_SdtListItem" minOccurs="0" maxOccurs="unbounded"/>
2036     </xsd:sequence>
2037     <xsd:attribute name="lastValue" type="s:ST String" use="optional"/>
2038 </xsd:complexType>
2039 <xsd:complexType name="CT_Placeholder">
2040     <xsd:sequence>
2041         <xsd:element name="docPart" type="CT String"/>
2042     </xsd:sequence>
2043 </xsd:complexType>
2044 <xsd:complexType name="CT_SdtText">
2045     <xsd:attribute name="multiLine" type="s:ST OnOff"/>
2046 </xsd:complexType>
2047 <xsd:complexType name="CT_DataBinding">

```

```

2048     <xsd:attribute name="prefixMappings" type="s:ST_String"/>
2049     <xsd:attribute name="xpath" type="s:ST_String" use="required"/>
2050     <xsd:attribute name="storeItemID" type="s:ST_String" use="required"/>
2051 </xsd:complexType>
2052 <xsd:complexType name="CT_SdtPr">
2053     <xsd:sequence>
2054         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
2055         <xsd:element name="alias" type="CT_String" minOccurs="0"/>
2056         <xsd:element name="tag" type="CT_String" minOccurs="0"/>
2057         <xsd:element name="id" type="CT_DecimalNumber" minOccurs="0"/>
2058         <xsd:element name="lock" type="CT_Lock" minOccurs="0"/>
2059         <xsd:element name="placeholder" type="CT_Placeholder" minOccurs="0"/>
2060         <xsd:element name="temporary" type="CT_OnOff" minOccurs="0"/>
2061         <xsd:element name="showingPlcHdr" type="CT_OnOff" minOccurs="0"/>
2062         <xsd:element name="dataBinding" type="CT_DataBinding" minOccurs="0"/>
2063         <xsd:element name="label" type="CT_DecimalNumber" minOccurs="0"/>
2064         <xsd:element name="tabIndex" type="CT_UnsignedDecimalNumber" minOccurs="0"/>
2065         <xsd:choice minOccurs="0" maxOccurs="1">
2066             <xsd:element name="equation" type="CT_Empty"/>
2067             <xsd:element name="comboBox" type="CT_SdtComboBox"/>
2068             <xsd:element name="date" type="CT_SdtDate"/>
2069             <xsd:element name="docPartObj" type="CT_SdtDocPart"/>
2070             <xsd:element name="docPartList" type="CT_SdtDocPart"/>
2071             <xsd:element name="dropDownList" type="CT_SdtDropDownList"/>
2072             <xsd:element name="picture" type="CT_Empty"/>
2073             <xsd:element name="richText" type="CT_Empty"/>
2074             <xsd:element name="text" type="CT_SdtText"/>
2075             <xsd:element name="citation" type="CT_Empty"/>
2076             <xsd:element name="group" type="CT_Empty"/>
2077             <xsd:element name="bibliography" type="CT_Empty"/>
2078         </xsd:choice>
2079     </xsd:sequence>
2080 </xsd:complexType>
2081 <xsd:complexType name="CT_SdtEndPr">
2082     <xsd:choice maxOccurs="unbounded">
2083         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
2084     </xsd:choice>
2085 </xsd:complexType>
2086 <xsd:group name="EG_ContentRunContent">
2087     <xsd:choice>
2088         <xsd:element name="customXml" type="CT_CustomXmlRun"/>
2089         <xsd:element name="smartTag" type="CT_SmartTagRun"/>
2090         <xsd:element name="sdt" type="CT_SdtRun"/>
2091         <xsd:element name="dir" type="CT_DirContentRun"/>
2092         <xsd:element name="bdo" type="CT_BdoContentRun"/>
2093         <xsd:element name="r" type="CT_R"/>
2094         <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2095     </xsd:choice>
2096 </xsd:group>
2097 <xsd:complexType name="CT_DirContentRun">
2098     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2099     <xsd:attribute name="val" type="ST_Direction" use="optional"/>
2100 </xsd:complexType>

```

```

2101 <xsd:complexType name="CT_BdoContentRun">
2102   <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2103   <xsd:attribute name="val" type="ST_Direction" use="optional"/>
2104 </xsd:complexType>
2105 <xsd:simpleType name="ST_Direction">
2106   <xsd:restriction base="xsd:string">
2107     <xsd:enumeration value="ltr"/>
2108     <xsd:enumeration value="rtl"/>
2109   </xsd:restriction>
2110 </xsd:simpleType>
2111 <xsd:complexType name="CT_SdtContentRun">
2112   <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2113 </xsd:complexType>
2114 <xsd:group name="EG_ContentBlockContent">
2115   <xsd:choice>
2116     <xsd:element name="customXml" type="CT_CustomXmlBlock"/>
2117     <xsd:element name="sdt" type="CT_SdtBlock"/>
2118     <xsd:element name="p" type="CT_P" minOccurs="0" maxOccurs="unbounded"/>
2119     <xsd:element name="tbl" type="CT_Tbl" minOccurs="0" maxOccurs="unbounded"/>
2120     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2121   </xsd:choice>
2122 </xsd:group>
2123 <xsd:complexType name="CT_SdtContentBlock">
2124   <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
2125 </xsd:complexType>
2126 <xsd:group name="EG_ContentRowContent">
2127   <xsd:choice>
2128     <xsd:element name="tr" type="CT_Row" minOccurs="0" maxOccurs="unbounded"/>
2129     <xsd:element name="customXml" type="CT_CustomXmlRow"/>
2130     <xsd:element name="sdt" type="CT_SdtRow"/>
2131     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2132   </xsd:choice>
2133 </xsd:group>
2134 <xsd:complexType name="CT_SdtContentRow">
2135   <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2136 </xsd:complexType>
2137 <xsd:group name="EG_ContentCellContent">
2138   <xsd:choice>
2139     <xsd:element name="tc" type="CT_Tc" minOccurs="0" maxOccurs="unbounded"/>
2140     <xsd:element name="customXml" type="CT_CustomXmlCell"/>
2141     <xsd:element name="sdt" type="CT_SdtCell"/>
2142     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2143   </xsd:choice>
2144 </xsd:group>
2145 <xsd:complexType name="CT_SdtContentCell">
2146   <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2147 </xsd:complexType>
2148 <xsd:complexType name="CT_SdtBlock">
2149   <xsd:sequence>
2150     <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2151     <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2152     <xsd:element name="sdtContent" type="CT_SdtContentBlock" minOccurs="0" maxOccurs="1"/>
2153   </xsd:sequence>

```

```

2154 </xsd:complexType>
2155 <xsd:complexType name="CT_SdtRun">
2156   <xsd:sequence>
2157     <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2158     <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2159     <xsd:element name="sdtContent" type="CT_SdtContentRun" minOccurs="0" maxOccurs="1"/>
2160   </xsd:sequence>
2161 </xsd:complexType>
2162 <xsd:complexType name="CT_SdtCell">
2163   <xsd:sequence>
2164     <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2165     <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2166     <xsd:element name="sdtContent" type="CT_SdtContentCell" minOccurs="0" maxOccurs="1"/>
2167   </xsd:sequence>
2168 </xsd:complexType>
2169 <xsd:complexType name="CT_SdtRow">
2170   <xsd:sequence>
2171     <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2172     <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2173     <xsd:element name="sdtContent" type="CT_SdtContentRow" minOccurs="0" maxOccurs="1"/>
2174   </xsd:sequence>
2175 </xsd:complexType>
2176 <xsd:complexType name="CT_Attr">
2177   <xsd:attribute name="uri" type="s:ST String"/>
2178   <xsd:attribute name="name" type="s:ST String" use="required"/>
2179   <xsd:attribute name="val" type="s:ST String" use="required"/>
2180 </xsd:complexType>
2181 <xsd:complexType name="CT_CustomXmlRun">
2182   <xsd:sequence>
2183     <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2184     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2185   </xsd:sequence>
2186   <xsd:attribute name="uri" type="s:ST String"/>
2187   <xsd:attribute name="element" type="s:ST XmlName" use="required"/>
2188 </xsd:complexType>
2189 <xsd:complexType name="CT_SmartTagRun">
2190   <xsd:sequence>
2191     <xsd:element name="smartTagPr" type="CT_SmartTagPr" minOccurs="0" maxOccurs="1"/>
2192     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2193   </xsd:sequence>
2194   <xsd:attribute name="uri" type="s:ST String"/>
2195   <xsd:attribute name="element" type="s:ST XmlName" use="required"/>
2196 </xsd:complexType>
2197 <xsd:complexType name="CT_CustomXmlBlock">
2198   <xsd:sequence>
2199     <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2200     <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
2201   </xsd:sequence>
2202   <xsd:attribute name="uri" type="s:ST String"/>
2203   <xsd:attribute name="element" type="s:ST XmlName" use="required"/>
2204 </xsd:complexType>
2205 <xsd:complexType name="CT_CustomXmlPr">
2206   <xsd:sequence>

```

```

2207     <xsd:element name="placeholder" type="CT_String" minOccurs="0"/>
2208     <xsd:element name="attr" type="CT_Attr" minOccurs="0" maxOccurs="unbounded"/>
2209   </xsd:sequence>
2210 </xsd:complexType>
2211 <xsd:complexType name="CT_CustomXmlRow">
2212   <xsd:sequence>
2213     <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2214     <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2215   </xsd:sequence>
2216   <xsd:attribute name="uri" type="s:ST_String"/>
2217   <xsd:attribute name="element" type="s:ST_XmlName" use="required"/>
2218 </xsd:complexType>
2219 <xsd:complexType name="CT_CustomXmlCell">
2220   <xsd:sequence>
2221     <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2222     <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2223   </xsd:sequence>
2224   <xsd:attribute name="uri" type="s:ST_String"/>
2225   <xsd:attribute name="element" type="s:ST_XmlName" use="required"/>
2226 </xsd:complexType>
2227 <xsd:complexType name="CT_SmartTagPr">
2228   <xsd:sequence>
2229     <xsd:element name="attr" type="CT_Attr" minOccurs="0" maxOccurs="unbounded"/>
2230   </xsd:sequence>
2231 </xsd:complexType>
2232 <xsd:group name="EG_PContent">
2233   <xsd:choice>
2234     <xsd:group ref="EG_ContentRunContent" minOccurs="0" maxOccurs="unbounded"/>
2235     <xsd:element name="fldSimple" type="CT_SimpleField" minOccurs="0" maxOccurs="unbounded"/>
2236     <xsd:element name="hyperlink" type="CT_Hyperlink"/>
2237     <xsd:element name="subDoc" type="CT_Rel"/>
2238   </xsd:choice>
2239 </xsd:group>
2240 <xsd:complexType name="CT_P">
2241   <xsd:sequence>
2242     <xsd:element name="pPr" type="CT_PPr" minOccurs="0"/>
2243     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2244   </xsd:sequence>
2245   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
2246   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
2247   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
2248   <xsd:attribute name="rsidP" type="ST_LongHexNumber"/>
2249   <xsd:attribute name="rsidRDefault" type="ST_LongHexNumber"/>
2250 </xsd:complexType>
2251 <xsd:simpleType name="ST_TblWidth">
2252   <xsd:restriction base="xsd:string">
2253     <xsd:enumeration value="nil"/>
2254     <xsd:enumeration value="pct"/>
2255     <xsd:enumeration value="dxa"/>
2256     <xsd:enumeration value="auto"/>
2257   </xsd:restriction>
2258 </xsd:simpleType>
2259 <xsd:complexType name="CT_Height">

```

```

2260     <xsd:attribute name="val" type="s:ST_TwipsMeasure"/>
2261     <xsd:attribute name="hRule" type="ST_HeightRule"/>
2262 </xsd:complexType>
2263 <xsd:simpleType name="ST_MeasurementOrPercent">
2264     <xsd:union memberTypes="ST_DecimalNumberOrPercent s:ST_UniversalMeasure"/>
2265 </xsd:simpleType>
2266 <xsd:complexType name="CT_TblWidth">
2267     <xsd:attribute name="w" type="ST_MeasurementOrPercent"/>
2268     <xsd:attribute name="type" type="ST_TblWidth"/>
2269 </xsd:complexType>
2270 <xsd:complexType name="CT_TblGridCol">
2271     <xsd:attribute name="w" type="s:ST_TwipsMeasure"/>
2272 </xsd:complexType>
2273 <xsd:complexType name="CT_TblGridBase">
2274     <xsd:sequence>
2275         <xsd:element name="gridCol" type="CT_TblGridCol" minOccurs="0" maxOccurs="unbounded"/>
2276     </xsd:sequence>
2277 </xsd:complexType>
2278 <xsd:complexType name="CT_TblGrid">
2279     <xsd:complexContent>
2280         <xsd:extension base="CT_TblGridBase">
2281             <xsd:sequence>
2282                 <xsd:element name="tblGridChange" type="CT_TblGridChange" minOccurs="0"/>
2283             </xsd:sequence>
2284         </xsd:extension>
2285     </xsd:complexContent>
2286 </xsd:complexType>
2287 <xsd:complexType name="CT_TcBorders">
2288     <xsd:sequence>
2289         <xsd:element name="top" type="CT_Border" minOccurs="0"/>
2290         <xsd:element name="start" type="CT_Border" minOccurs="0"/>
2291         <xsd:element name="left" type="CT_Border" minOccurs="0"/>
2292         <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
2293         <xsd:element name="end" type="CT_Border" minOccurs="0"/>
2294         <xsd:element name="right" type="CT_Border" minOccurs="0"/>
2295         <xsd:element name="insideH" type="CT_Border" minOccurs="0"/>
2296         <xsd:element name="insideV" type="CT_Border" minOccurs="0"/>
2297         <xsd:element name="tl2br" type="CT_Border" minOccurs="0"/>
2298         <xsd:element name="tr2bl" type="CT_Border" minOccurs="0"/>
2299     </xsd:sequence>
2300 </xsd:complexType>
2301 <xsd:complexType name="CT_TcMar">
2302     <xsd:sequence>
2303         <xsd:element name="top" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2304         <xsd:element name="start" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2305         <xsd:element name="left" type="CT_TblWidth" minOccurs="0"/>
2306         <xsd:element name="bottom" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2307         <xsd:element name="end" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2308         <xsd:element name="right" type="CT_TblWidth" minOccurs="0"/>
2309     </xsd:sequence>
2310 </xsd:complexType>
2311 <xsd:simpleType name="ST_Merge">
2312     <xsd:restriction base="xsd:string">

```



```

2313         <xsd:enumeration value="continue"/>
2314         <xsd:enumeration value="restart"/>
2315     </xsd:restriction>
2316 </xsd:simpleType>
2317 <xsd:complexType name="CT_VMerge">
2318     <xsd:attribute name="val" type="ST_Merge"/>
2319 </xsd:complexType>
2320 <xsd:complexType name="CT_HMerge">
2321     <xsd:attribute name="val" type="ST_Merge"/>
2322 </xsd:complexType>
2323 <xsd:complexType name="CT_TcPrBase">
2324     <xsd:sequence>
2325         <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
2326         <xsd:element name="tcW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2327         <xsd:element name="gridSpan" type="CT_DecimalNumber" minOccurs="0"/>
2328         <xsd:element name="hMerge" type="CT_HMerge" minOccurs="0"/>
2329         <xsd:element name="vMerge" type="CT_VMerge" minOccurs="0"/>
2330         <xsd:element name="tcBorders" type="CT_TcBorders" minOccurs="0" maxOccurs="1"/>
2331         <xsd:element name="shd" type="CT_Shdt" minOccurs="0"/>
2332         <xsd:element name="noWrap" type="CT_OnOff" minOccurs="0"/>
2333         <xsd:element name="tcMar" type="CT_TcMar" minOccurs="0" maxOccurs="1"/>
2334         <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0" maxOccurs="1"/>
2335         <xsd:element name="tcFitText" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
2336         <xsd:element name="vAlign" type="CT_VerticalJc" minOccurs="0"/>
2337         <xsd:element name="hideMark" type="CT_OnOff" minOccurs="0"/>
2338         <xsd:element name="headers" type="CT-Headers" minOccurs="0"/>
2339     </xsd:sequence>
2340 </xsd:complexType>
2341 <xsd:complexType name="CT_TcPr">
2342     <xsd:complexContent>
2343         <xsd:extension base="CT_TcPrInner">
2344             <xsd:sequence>
2345                 <xsd:element name="tcPrChange" type="CT_TcPrChange" minOccurs="0"/>
2346             </xsd:sequence>
2347         </xsd:extension>
2348     </xsd:complexContent>
2349 </xsd:complexType>
2350 <xsd:complexType name="CT_TcPrInner">
2351     <xsd:complexContent>
2352         <xsd:extension base="CT_TcPrBase">
2353             <xsd:sequence>
2354                 <xsd:group ref="EG_CellMarkupElements" minOccurs="0" maxOccurs="1"/>
2355             </xsd:sequence>
2356         </xsd:extension>
2357     </xsd:complexContent>
2358 </xsd:complexType>
2359 <xsd:complexType name="CT_Tc">
2360     <xsd:sequence>
2361         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
2362         <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
2363     </xsd:sequence>
2364     <xsd:attribute name="id" type="s:ST-String" use="optional"/>
2365 </xsd:complexType>

```

```

2366 <xsd:simpleType name="ST_Cnf">
2367   <xsd:restriction base="xsd:string">
2368     <xsd:length value="12"/>
2369     <xsd:pattern value="[01]*"/>
2370   </xsd:restriction>
2371 </xsd:simpleType>
2372 <xsd:complexType name="CT_Cnf">
2373   <xsd:attribute name="val" type="ST_Cnf"/>
2374   <xsd:attribute name="firstRow" type="s:ST_OnOff"/>
2375   <xsd:attribute name="lastRow" type="s:ST_OnOff"/>
2376   <xsd:attribute name="firstColumn" type="s:ST_OnOff"/>
2377   <xsd:attribute name="lastColumn" type="s:ST_OnOff"/>
2378   <xsd:attribute name="oddVBand" type="s:ST_OnOff"/>
2379   <xsd:attribute name="evenVBand" type="s:ST_OnOff"/>
2380   <xsd:attribute name="oddHBand" type="s:ST_OnOff"/>
2381   <xsd:attribute name="evenHBand" type="s:ST_OnOff"/>
2382   <xsd:attribute name="firstRowFirstColumn" type="s:ST_OnOff"/>
2383   <xsd:attribute name="firstRowLastColumn" type="s:ST_OnOff"/>
2384   <xsd:attribute name="lastRowFirstColumn" type="s:ST_OnOff"/>
2385   <xsd:attribute name="lastRowLastColumn" type="s:ST_OnOff"/>
2386 </xsd:complexType>
2387 <xsd:complexType name="CT_Headers">
2388   <xsd:sequence minOccurs="0" maxOccurs="unbounded">
2389     <xsd:element name="header" type="CT_String"/>
2390   </xsd:sequence>
2391 </xsd:complexType>
2392 <xsd:complexType name="CT_TrPrBase">
2393   <xsd:choice maxOccurs="unbounded">
2394     <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
2395     <xsd:element name="divId" type="CT_DecimalNumber" minOccurs="0"/>
2396     <xsd:element name="gridBefore" type="CT_DecimalNumber" minOccurs="0"/>
2397     <xsd:element name="gridAfter" type="CT_DecimalNumber" minOccurs="0"/>
2398     <xsd:element name="wBefore" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2399     <xsd:element name="wAfter" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2400     <xsd:element name="cantSplit" type="CT_OnOff" minOccurs="0"/>
2401     <xsd:element name="trHeight" type="CT_Height" minOccurs="0"/>
2402     <xsd:element name="tblHeader" type="CT_OnOff" minOccurs="0"/>
2403     <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2404     <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2405     <xsd:element name="hidden" type="CT_OnOff" minOccurs="0"/>
2406   </xsd:choice>
2407 </xsd:complexType>
2408 <xsd:complexType name="CT_TrPr">
2409   <xsd:complexContent>
2410     <xsd:extension base="CT_TrPrBase">
2411       <xsd:sequence>
2412         <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
2413         <xsd:element name="del" type="CT_TrackChange" minOccurs="0"/>
2414         <xsd:element name="trPrChange" type="CT_TrPrChange" minOccurs="0"/>
2415       </xsd:sequence>
2416     </xsd:extension>
2417   </xsd:complexContent>
2418 </xsd:complexType>

```

```

2419 <xsd:complexType name="CT_Row">
2420   <xsd:sequence>
2421     <xsd:element name="tblPrEx" type="CT_TblPrEx" minOccurs="0" maxOccurs="1"/>
2422     <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
2423     <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2424   </xsd:sequence>
2425   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
2426   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
2427   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
2428   <xsd:attribute name="rsidTr" type="ST_LongHexNumber"/>
2429 </xsd:complexType>
2430 <xsd:simpleType name="ST_TblLayoutType">
2431   <xsd:restriction base="xsd:string">
2432     <xsd:enumeration value="fixed"/>
2433     <xsd:enumeration value="autofit"/>
2434   </xsd:restriction>
2435 </xsd:simpleType>
2436 <xsd:complexType name="CT_TblLayoutType">
2437   <xsd:attribute name="type" type="ST_TblLayoutType"/>
2438 </xsd:complexType>
2439 <xsd:simpleType name="ST_TblOverlap">
2440   <xsd:restriction base="xsd:string">
2441     <xsd:enumeration value="never"/>
2442     <xsd:enumeration value="overlap"/>
2443   </xsd:restriction>
2444 </xsd:simpleType>
2445 <xsd:complexType name="CT_TblOverlap">
2446   <xsd:attribute name="val" type="ST_TblOverlap" use="required"/>
2447 </xsd:complexType>
2448 <xsd:complexType name="CT_TblPPr">
2449   <xsd:attribute name="leftFromText" type="s:ST_TwipsMeasure"/>
2450   <xsd:attribute name="rightFromText" type="s:ST_TwipsMeasure"/>
2451   <xsd:attribute name="topFromText" type="s:ST_TwipsMeasure"/>
2452   <xsd:attribute name="bottomFromText" type="s:ST_TwipsMeasure"/>
2453   <xsd:attribute name="vertAnchor" type="ST_VAnchor"/>
2454   <xsd:attribute name="horzAnchor" type="ST_HAnchor"/>
2455   <xsd:attribute name="tblpXSpec" type="s:ST_XAlign"/>
2456   <xsd:attribute name="tblpX" type="ST_SignedTwipsMeasure"/>
2457   <xsd:attribute name="tblpYSpec" type="s:ST_YAlign"/>
2458   <xsd:attribute name="tblpY" type="ST_SignedTwipsMeasure"/>
2459 </xsd:complexType>
2460 <xsd:complexType name="CT_TblCellMar">
2461   <xsd:sequence>
2462     <xsd:element name="top" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2463     <xsd:element name="start" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2464     <xsd:element name="left" type="CT_TblWidth" minOccurs="0"/>
2465     <xsd:element name="bottom" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2466     <xsd:element name="end" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2467     <xsd:element name="right" type="CT_TblWidth" minOccurs="0"/>
2468   </xsd:sequence>
2469 </xsd:complexType>
2470 <xsd:complexType name="CT_TblBorders">
2471   <xsd:sequence>

```

```

2472     <xsd:element name="top" type="CT_Border" minOccurs="0"/>
2473     <xsd:element name="start" type="CT_Border" minOccurs="0"/>
2474     <xsd:element name="left" type="CT_Border" minOccurs="0"/>
2475     <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
2476     <xsd:element name="end" type="CT_Border" minOccurs="0"/>
2477     <xsd:element name="right" type="CT_Border" minOccurs="0"/>
2478     <xsd:element name="insideH" type="CT_Border" minOccurs="0"/>
2479     <xsd:element name="insideV" type="CT_Border" minOccurs="0"/>
2480   </xsd:sequence>
2481 </xsd:complexType>
2482 <xsd:complexType name="CT_TblPrBase">
2483   <xsd:sequence>
2484     <xsd:element name="tblStyle" type="CT_String" minOccurs="0"/>
2485     <xsd:element name="tblpPr" type="CT_TblPPr" minOccurs="0" maxOccurs="1"/>
2486     <xsd:element name="tblOverlap" type="CT_TblOverlap" minOccurs="0" maxOccurs="1"/>
2487     <xsd:element name="bidiVisual" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
2488     <xsd:element name="tblStyleRowBandSize" type="CT_DecimalNumber" minOccurs="0"
2489       maxOccurs="1"/>
2490     <xsd:element name="tblStyleColBandSize" type="CT_DecimalNumber" minOccurs="0"
2491       maxOccurs="1"/>
2492     <xsd:element name="tblW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2493     <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2494     <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2495     <xsd:element name="tblInd" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2496     <xsd:element name="tblBorders" type="CT_TblBorders" minOccurs="0" maxOccurs="1"/>
2497     <xsd:element name="shd" type="CT_Shd" minOccurs="0" maxOccurs="1"/>
2498     <xsd:element name="tblLayout" type="CT_TblLayoutType" minOccurs="0" maxOccurs="1"/>
2499     <xsd:element name="tblCellMar" type="CT_TblCellMar" minOccurs="0" maxOccurs="1"/>
2500     <xsd:element name="tblLook" type="CT_TblLook" minOccurs="0" maxOccurs="1"/>
2501     <xsd:element name="tblCaption" type="CT_String" minOccurs="0" maxOccurs="1"/>
2502     <xsd:element name="tblDescription" type="CT_String" minOccurs="0" maxOccurs="1"/>
2503   </xsd:sequence>
2504 </xsd:complexType>
2505 <xsd:complexType name="CT_TblPr">
2506   <xsd:complexContent>
2507     <xsd:extension base="CT_TblPrBase">
2508       <xsd:sequence>
2509         <xsd:element name="tblPrChange" type="CT_TblPrChange" minOccurs="0"/>
2510       </xsd:sequence>
2511     </xsd:extension>
2512   </xsd:complexContent>
2513 </xsd:complexType>
2514 <xsd:complexType name="CT_TblPrExBase">
2515   <xsd:sequence>
2516     <xsd:element name="tblW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2517     <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2518     <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2519     <xsd:element name="tblInd" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2520     <xsd:element name="tblBorders" type="CT_TblBorders" minOccurs="0" maxOccurs="1"/>
2521     <xsd:element name="shd" type="CT_Shd" minOccurs="0" maxOccurs="1"/>
2522     <xsd:element name="tblLayout" type="CT_TblLayoutType" minOccurs="0" maxOccurs="1"/>
2523     <xsd:element name="tblCellMar" type="CT_TblCellMar" minOccurs="0" maxOccurs="1"/>
2524     <xsd:element name="tblLook" type="CT_TblLook" minOccurs="0" maxOccurs="1"/>

```

```

2525     </xsd:sequence>
2526 </xsd:complexType>
2527 <xsd:complexType name="CT_TblPrEx">
2528     <xsd:complexContent>
2529         <xsd:extension base="CT_TblPrExBase">
2530             <xsd:sequence>
2531                 <xsd:element name="tblPrExChange" type="CT_TblPrExChange" minOccurs="0"/>
2532             </xsd:sequence>
2533         </xsd:extension>
2534     </xsd:complexContent>
2535 </xsd:complexType>
2536 <xsd:complexType name="CT_Tbl">
2537     <xsd:sequence>
2538         <xsd:group ref="EG_RangeMarkupElements" minOccurs="0" maxOccurs="unbounded"/>
2539         <xsd:element name="tblPr" type="CT_TblPr"/>
2540         <xsd:element name="tblGrid" type="CT_TblGrid"/>
2541         <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2542     </xsd:sequence>
2543 </xsd:complexType>
2544 <xsd:complexType name="CT_TblLook">
2545     <xsd:attribute name="firstRow" type="s:ST_OnOff"/>
2546     <xsd:attribute name="lastRow" type="s:ST_OnOff"/>
2547     <xsd:attribute name="firstColumn" type="s:ST_OnOff"/>
2548     <xsd:attribute name="lastColumn" type="s:ST_OnOff"/>
2549     <xsd:attribute name="noHBand" type="s:ST_OnOff"/>
2550     <xsd:attribute name="noVBand" type="s:ST_OnOff"/>
2551     <xsd:attribute name="val" type="ST_ShortHexNumber"/>
2552 </xsd:complexType>
2553 <xsd:simpleType name="ST_FtnPos">
2554     <xsd:restriction base="xsd:string">
2555         <xsd:enumeration value="pageBottom"/>
2556         <xsd:enumeration value="beneathText"/>
2557         <xsd:enumeration value="sectEnd"/>
2558         <xsd:enumeration value="docEnd"/>
2559     </xsd:restriction>
2560 </xsd:simpleType>
2561 <xsd:complexType name="CT_FtnPos">
2562     <xsd:attribute name="val" type="ST_FtnPos" use="required"/>
2563 </xsd:complexType>
2564 <xsd:simpleType name="ST_EdnPos">
2565     <xsd:restriction base="xsd:string">
2566         <xsd:enumeration value="sectEnd"/>
2567         <xsd:enumeration value="docEnd"/>
2568     </xsd:restriction>
2569 </xsd:simpleType>
2570 <xsd:complexType name="CT_EdnPos">
2571     <xsd:attribute name="val" type="ST_EdnPos" use="required"/>
2572 </xsd:complexType>
2573 <xsd:complexType name="CT_NumFmt">
2574     <xsd:attribute name="val" type="ST_NumberFormat" use="required"/>
2575     <xsd:attribute name="format" type="s:ST_String" use="optional"/>
2576 </xsd:complexType>
2577 <xsd:simpleType name="ST_RestartNumber">

```

```

2578     <xsd:restriction base="xsd:string">
2579         <xsd:enumeration value="continuous"/>
2580         <xsd:enumeration value="eachSect"/>
2581         <xsd:enumeration value="eachPage"/>
2582     </xsd:restriction>
2583 </xsd:simpleType>
2584 <xsd:complexType name="CT_NumRestart">
2585     <xsd:attribute name="val" type="ST_RestartNumber" use="required"/>
2586 </xsd:complexType>
2587 <xsd:complexType name="CT_FtnEdnRef">
2588     <xsd:attribute name="customMarkFollows" type="s:ST_OnOff" use="optional"/>
2589     <xsd:attribute name="id" use="required" type="ST_DecimalNumber"/>
2590 </xsd:complexType>
2591 <xsd:complexType name="CT_FtnEdnSepRef">
2592     <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
2593 </xsd:complexType>
2594 <xsd:complexType name="CT_FtnEdn">
2595     <xsd:sequence>
2596         <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
2597     </xsd:sequence>
2598     <xsd:attribute name="type" type="ST_FtnEdn" use="optional"/>
2599     <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
2600 </xsd:complexType>
2601 <xsd:group name="EG_FtnEdnNumProps">
2602     <xsd:sequence>
2603         <xsd:element name="numStart" type="CT_DecimalNumber" minOccurs="0"/>
2604         <xsd:element name="numRestart" type="CT_NumRestart" minOccurs="0"/>
2605     </xsd:sequence>
2606 </xsd:group>
2607 <xsd:complexType name="CT_FtnProps">
2608     <xsd:sequence>
2609         <xsd:element name="pos" type="CT_FtnPos" minOccurs="0"/>
2610         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0"/>
2611         <xsd:group ref="EG_FtnEdnNumProps" minOccurs="0"/>
2612     </xsd:sequence>
2613 </xsd:complexType>
2614 <xsd:complexType name="CT_EdnProps">
2615     <xsd:sequence>
2616         <xsd:element name="pos" type="CT_EdnPos" minOccurs="0"/>
2617         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0"/>
2618         <xsd:group ref="EG_FtnEdnNumProps" minOccurs="0"/>
2619     </xsd:sequence>
2620 </xsd:complexType>
2621 <xsd:complexType name="CT_FtnDocProps">
2622     <xsd:complexContent>
2623         <xsd:extension base="CT_FtnProps">
2624             <xsd:sequence>
2625                 <xsd:element name="footnote" type="CT_FtnEdnSepRef" minOccurs="0" maxOccurs="3"/>
2626             </xsd:sequence>
2627         </xsd:extension>
2628     </xsd:complexContent>
2629 </xsd:complexType>
2630 <xsd:complexType name="CT_EdnDocProps">

```

```

2631     <xsd:complexContent>
2632         <xsd:extension base="CT_EdnProps">
2633             <xsd:sequence>
2634                 <xsd:element name="endnote" type="CT_FtnEdnSepRef" minOccurs="0" maxOccurs="3"/>
2635             </xsd:sequence>
2636         </xsd:extension>
2637     </xsd:complexContent>
2638 </xsd:complexType>
2639 <xsd:complexType name="CT_RecipientData">
2640     <xsd:sequence>
2641         <xsd:element name="active" type="CT_OnOff" minOccurs="0"/>
2642         <xsd:element name="column" type="CT_DecimalNumber" minOccurs="1"/>
2643         <xsd:element name="uniqueTag" type="CT_Base64Binary" minOccurs="1"/>
2644     </xsd:sequence>
2645 </xsd:complexType>
2646 <xsd:complexType name="CT_Base64Binary">
2647     <xsd:attribute name="val" type="xsd:base64Binary" use="required">
2648     </xsd:attribute>
2649 </xsd:complexType>
2650 <xsd:complexType name="CT_Recipients">
2651     <xsd:sequence>
2652         <xsd:element name="recipientData" type="CT_RecipientData" minOccurs="1"
2653             maxOccurs="unbounded"/>
2654     </xsd:sequence>
2655 </xsd:complexType>
2656 <xsd:element name="recipients" type="CT_Recipients"/>
2657 <xsd:complexType name="CT_OdsoFieldMapData">
2658     <xsd:sequence>
2659         <xsd:element name="type" type="CT_MailMergeOdsoFMDFieldType" minOccurs="0"/>
2660         <xsd:element name="name" type="CT_String" minOccurs="0"/>
2661         <xsd:element name="mappedName" type="CT_String" minOccurs="0"/>
2662         <xsd:element name="column" type="CT_DecimalNumber" minOccurs="0"/>
2663         <xsd:element name="lid" type="CT_Lang" minOccurs="0"/>
2664         <xsd:element name="dynamicAddress" type="CT_OnOff" minOccurs="0"/>
2665     </xsd:sequence>
2666 </xsd:complexType>
2667 <xsd:simpleType name="ST_MailMergeSourceType">
2668     <xsd:restriction base="xsd:string">
2669         <xsd:enumeration value="database"/>
2670         <xsd:enumeration value="addressBook"/>
2671         <xsd:enumeration value="document1"/>
2672         <xsd:enumeration value="document2"/>
2673         <xsd:enumeration value="text"/>
2674         <xsd:enumeration value="email"/>
2675         <xsd:enumeration value="native"/>
2676         <xsd:enumeration value="legacy"/>
2677         <xsd:enumeration value="master"/>
2678     </xsd:restriction>
2679 </xsd:simpleType>
2680 <xsd:complexType name="CT_MailMergeSourceType">
2681     <xsd:attribute name="val" use="required" type="ST_MailMergeSourceType"/>
2682 </xsd:complexType>
2683 <xsd:complexType name="CT_Odso">

```

```

2684     <xsd:sequence>
2685         <xsd:element name="udl" type="CT_String" minOccurs="0"/>
2686         <xsd:element name="table" type="CT_String" minOccurs="0"/>
2687         <xsd:element name="src" type="CT_Rel" minOccurs="0"/>
2688         <xsd:element name="colDelim" type="CT_DecimalNumber" minOccurs="0"/>
2689         <xsd:element name="type" type="CT_MailMergeSourceType" minOccurs="0"/>
2690         <xsd:element name="fHdr" type="CT_OnOff" minOccurs="0"/>
2691         <xsd:element name="fieldMapData" type="CT_OdsoFieldMapData" minOccurs="0"
2692             maxOccurs="unbounded"/>
2693         <xsd:element name="recipientData" type="CT_Rel" minOccurs="0" maxOccurs="unbounded"/>
2694     </xsd:sequence>
2695 </xsd:complexType>
2696 <xsd:complexType name="CT_MailMerge">
2697     <xsd:sequence>
2698         <xsd:element name="mainDocumentType" type="CT_MailMergeDocType" minOccurs="1"/>
2699         <xsd:element name="linkToQuery" type="CT_OnOff" minOccurs="0"/>
2700         <xsd:element name="dataType" type="CT_MailMergeDataType" minOccurs="1"/>
2701         <xsd:element name="connectString" type="CT_String" minOccurs="0"/>
2702         <xsd:element name="query" type="CT_String" minOccurs="0"/>
2703         <xsd:element name="dataSource" type="CT_Rel" minOccurs="0"/>
2704         <xsd:element name="headerSource" type="CT_Rel" minOccurs="0"/>
2705         <xsd:element name="doNotSuppressBlankLines" type="CT_OnOff" minOccurs="0"/>
2706         <xsd:element name="destination" type="CT_MailMergeDest" minOccurs="0"/>
2707         <xsd:element name="addressFieldName" type="CT_String" minOccurs="0"/>
2708         <xsd:element name="mailSubject" type="CT_String" minOccurs="0"/>
2709         <xsd:element name="mailAsAttachment" type="CT_OnOff" minOccurs="0"/>
2710         <xsd:element name="viewMergedData" type="CT_OnOff" minOccurs="0"/>
2711         <xsd:element name="activeRecord" type="CT_DecimalNumber" minOccurs="0"/>
2712         <xsd:element name="checkErrors" type="CT_DecimalNumber" minOccurs="0"/>
2713         <xsd:element name="odso" type="CT_Odso" minOccurs="0"/>
2714     </xsd:sequence>
2715 </xsd:complexType>
2716 <xsd:simpleType name="ST_TargetScreenSz">
2717     <xsd:restriction base="xsd:string">
2718         <xsd:enumeration value="544x376"/>
2719         <xsd:enumeration value="640x480"/>
2720         <xsd:enumeration value="720x512"/>
2721         <xsd:enumeration value="800x600"/>
2722         <xsd:enumeration value="1024x768"/>
2723         <xsd:enumeration value="1152x882"/>
2724         <xsd:enumeration value="1152x900"/>
2725         <xsd:enumeration value="1280x1024"/>
2726         <xsd:enumeration value="1600x1200"/>
2727         <xsd:enumeration value="1800x1440"/>
2728         <xsd:enumeration value="1920x1200"/>
2729     </xsd:restriction>
2730 </xsd:simpleType>
2731 <xsd:complexType name="CT_TargetScreenSz">
2732     <xsd:attribute name="val" type="ST_TargetScreenSz" use="required"/>
2733 </xsd:complexType>
2734 <xsd:complexType name="CT_Compat">
2735     <xsd:sequence>
2736         <xsd:element name="useSingleBorderforContiguousCells" type="CT_OnOff" minOccurs="0"/>

```



```

2737 <xsd:element name="wpJustification" type="CT_OnOff" minOccurs="0"/>
2738 <xsd:element name="noTabHangInd" type="CT_OnOff" minOccurs="0"/>
2739 <xsd:element name="noLeading" type="CT_OnOff" minOccurs="0"/>
2740 <xsd:element name="spaceForUL" type="CT_OnOff" minOccurs="0"/>
2741 <xsd:element name="noColumnBalance" type="CT_OnOff" minOccurs="0"/>
2742 <xsd:element name="balanceSingleByteDoubleByteWidth" type="CT_OnOff" minOccurs="0"/>
2743 <xsd:element name="noExtraLineSpacing" type="CT_OnOff" minOccurs="0"/>
2744 <xsd:element name="doNotLeaveBackslashAlone" type="CT_OnOff" minOccurs="0"/>
2745 <xsd:element name="ulTrailSpace" type="CT_OnOff" minOccurs="0"/>
2746 <xsd:element name="doNotExpandShiftReturn" type="CT_OnOff" minOccurs="0"/>
2747 <xsd:element name="spacingInWholePoints" type="CT_OnOff" minOccurs="0"/>
2748 <xsd:element name="lineWrapLikeWord6" type="CT_OnOff" minOccurs="0"/>
2749 <xsd:element name="printBodyTextBeforeHeader" type="CT_OnOff" minOccurs="0"/>
2750 <xsd:element name="printColBlack" type="CT_OnOff" minOccurs="0"/>
2751 <xsd:element name="wpSpaceWidth" type="CT_OnOff" minOccurs="0"/>
2752 <xsd:element name="showBreaksInFrames" type="CT_OnOff" minOccurs="0"/>
2753 <xsd:element name="subFontBySize" type="CT_OnOff" minOccurs="0"/>
2754 <xsd:element name="suppressBottomSpacing" type="CT_OnOff" minOccurs="0"/>
2755 <xsd:element name="suppressTopSpacing" type="CT_OnOff" minOccurs="0"/>
2756 <xsd:element name="suppressSpacingAtTopOfPage" type="CT_OnOff" minOccurs="0"/>
2757 <xsd:element name="suppressTopSpacingWP" type="CT_OnOff" minOccurs="0"/>
2758 <xsd:element name="suppressSpBfAAfterPgBrk" type="CT_OnOff" minOccurs="0"/>
2759 <xsd:element name="swapBordersFacingPages" type="CT_OnOff" minOccurs="0"/>
2760 <xsd:element name="convMailMergeEsc" type="CT_OnOff" minOccurs="0"/>
2761 <xsd:element name="truncateFontHeightsLikeWP6" type="CT_OnOff" minOccurs="0"/>
2762 <xsd:element name="mwSmallCaps" type="CT_OnOff" minOccurs="0"/>
2763 <xsd:element name="usePrinterMetrics" type="CT_OnOff" minOccurs="0"/>
2764 <xsd:element name="doNotSuppressParagraphBorders" type="CT_OnOff" minOccurs="0"/>
2765 <xsd:element name="wrapTrailSpaces" type="CT_OnOff" minOccurs="0"/>
2766 <xsd:element name="footnoteLayoutLikeWW8" type="CT_OnOff" minOccurs="0"/>
2767 <xsd:element name="shapeLayoutLikeWW8" type="CT_OnOff" minOccurs="0"/>
2768 <xsd:element name="alignTablesRowByRow" type="CT_OnOff" minOccurs="0"/>
2769 <xsd:element name="forgetLastTabAlignment" type="CT_OnOff" minOccurs="0"/>
2770 <xsd:element name="adjustLineHeightInTable" type="CT_OnOff" minOccurs="0"/>
2771 <xsd:element name="autoSpaceLikeWord95" type="CT_OnOff" minOccurs="0"/>
2772 <xsd:element name="noSpaceRaiseLower" type="CT_OnOff" minOccurs="0"/>
2773 <xsd:element name="doNotUseHTMLParagraphAutoSpacing" type="CT_OnOff" minOccurs="0"/>
2774 <xsd:element name="layoutRawTableWidth" type="CT_OnOff" minOccurs="0"/>
2775 <xsd:element name="layoutTableRowsApart" type="CT_OnOff" minOccurs="0"/>
2776 <xsd:element name="useWord97LineBreakRules" type="CT_OnOff" minOccurs="0"/>
2777 <xsd:element name="doNotBreakWrappedTables" type="CT_OnOff" minOccurs="0"/>
2778 <xsd:element name="doNotSnapToGridInCell" type="CT_OnOff" minOccurs="0"/>
2779 <xsd:element name="selectFldWithFirstOrLastChar" type="CT_OnOff" minOccurs="0"/>
2780 <xsd:element name="applyBreakingRules" type="CT_OnOff" minOccurs="0"/>
2781 <xsd:element name="doNotWrapTextWithPunct" type="CT_OnOff" minOccurs="0"/>
2782 <xsd:element name="doNotUseEastAsianBreakRules" type="CT_OnOff" minOccurs="0"/>
2783 <xsd:element name="useWord2002TableStyleRules" type="CT_OnOff" minOccurs="0"/>
2784 <xsd:element name="growAutofit" type="CT_OnOff" minOccurs="0"/>
2785 <xsd:element name="useFELayout" type="CT_OnOff" minOccurs="0"/>
2786 <xsd:element name="useNormalStyleForList" type="CT_OnOff" minOccurs="0"/>
2787 <xsd:element name="doNotUseIndentAsNumberingTabStop" type="CT_OnOff" minOccurs="0"/>
2788 <xsd:element name="useAltKinsokuLineBreakRules" type="CT_OnOff" minOccurs="0"/>
2789 <xsd:element name="allowSpaceOfSameStyleInTable" type="CT_OnOff" minOccurs="0"/>

```

```

2790     <xsd:element name="doNotSuppressIndentation" type="CT_OnOff" minOccurs="0"/>
2791     <xsd:element name="doNotAutofitConstrainedTables" type="CT_OnOff" minOccurs="0"/>
2792     <xsd:element name="autofitToFirstFixedWidthCell" type="CT_OnOff" minOccurs="0"/>
2793     <xsd:element name="underlineTabInNumList" type="CT_OnOff" minOccurs="0"/>
2794     <xsd:element name="displayHangulFixedWidth" type="CT_OnOff" minOccurs="0"/>
2795     <xsd:element name="splitPgBreakAndParaMark" type="CT_OnOff" minOccurs="0"/>
2796     <xsd:element name="doNotVertAlignCellWithSp" type="CT_OnOff" minOccurs="0"/>
2797     <xsd:element name="doNotBreakConstrainedForcedTable" type="CT_OnOff" minOccurs="0"/>
2798     <xsd:element name="doNotVertAlignInTxbx" type="CT_OnOff" minOccurs="0"/>
2799     <xsd:element name="useAnsiKerningPairs" type="CT_OnOff" minOccurs="0"/>
2800     <xsd:element name="cachedColBalance" type="CT_OnOff" minOccurs="0"/>
2801     <xsd:element name="compatSetting" type="CT_CompatSetting" minOccurs="0"
2802         maxOccurs="unbounded"/>
2803 </xsd:sequence>
2804 </xsd:complexType>
2805 <xsd:complexType name="CT_CompatSetting">
2806     <xsd:attribute name="name" type="s:ST_String"/>
2807     <xsd:attribute name="uri" type="s:ST_String"/>
2808     <xsd:attribute name="val" type="s:ST_String"/>
2809 </xsd:complexType>
2810 <xsd:complexType name="CT_DocVar">
2811     <xsd:attribute name="name" type="s:ST_String" use="required"/>
2812     <xsd:attribute name="val" type="s:ST_String" use="required"/>
2813 </xsd:complexType>
2814 <xsd:complexType name="CT_DocVars">
2815     <xsd:sequence>
2816         <xsd:element name="docVar" type="CT_DocVar" minOccurs="0" maxOccurs="unbounded"/>
2817     </xsd:sequence>
2818 </xsd:complexType>
2819 <xsd:complexType name="CT_DocRsids">
2820     <xsd:sequence>
2821         <xsd:element name="rsidRoot" type="CT_LongHexNumber" minOccurs="0" maxOccurs="1"/>
2822         <xsd:element name="rsid" type="CT_LongHexNumber" minOccurs="0" maxOccurs="unbounded"/>
2823     </xsd:sequence>
2824 </xsd:complexType>
2825 <xsd:simpleType name="ST_CharacterSpacing">
2826     <xsd:restriction base="xsd:string">
2827         <xsd:enumeration value="doNotCompress"/>
2828         <xsd:enumeration value="compressPunctuation"/>
2829         <xsd:enumeration value="compressPunctuationAndJapaneseKana"/>
2830     </xsd:restriction>
2831 </xsd:simpleType>
2832 <xsd:complexType name="CT_CharacterSpacing">
2833     <xsd:attribute name="val" type="ST_CharacterSpacing" use="required"/>
2834 </xsd:complexType>
2835 <xsd:complexType name="CT_SaveThroughXslt">
2836     <xsd:attribute ref="r:id" use="optional"/>
2837     <xsd:attribute name="solutionID" type="s:ST_String" use="optional"/>
2838 </xsd:complexType>
2839 <xsd:complexType name="CT_RPrDefault">
2840     <xsd:sequence>
2841         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
2842     </xsd:sequence>

```

```

2843 </xsd:complexType>
2844 <xsd:complexType name="CT_PPrDefault">
2845   <xsd:sequence>
2846     <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0"/>
2847   </xsd:sequence>
2848 </xsd:complexType>
2849 <xsd:complexType name="CT_DocDefaults">
2850   <xsd:sequence>
2851     <xsd:element name="rPrDefault" type="CT_RPrDefault" minOccurs="0"/>
2852     <xsd:element name="pPrDefault" type="CT_PPrDefault" minOccurs="0"/>
2853   </xsd:sequence>
2854 </xsd:complexType>
2855 <xsd:simpleType name="ST_WmlColorSchemeIndex">
2856   <xsd:restriction base="xsd:string">
2857     <xsd:enumeration value="dark1"/>
2858     <xsd:enumeration value="light1"/>
2859     <xsd:enumeration value="dark2"/>
2860     <xsd:enumeration value="light2"/>
2861     <xsd:enumeration value="accent1"/>
2862     <xsd:enumeration value="accent2"/>
2863     <xsd:enumeration value="accent3"/>
2864     <xsd:enumeration value="accent4"/>
2865     <xsd:enumeration value="accent5"/>
2866     <xsd:enumeration value="accent6"/>
2867     <xsd:enumeration value="hyperlink"/>
2868     <xsd:enumeration value="followedHyperlink"/>
2869   </xsd:restriction>
2870 </xsd:simpleType>
2871 <xsd:complexType name="CT_ColorSchemeMapping">
2872   <xsd:attribute name="bg1" type="ST_WmlColorSchemeIndex"/>
2873   <xsd:attribute name="t1" type="ST_WmlColorSchemeIndex"/>
2874   <xsd:attribute name="bg2" type="ST_WmlColorSchemeIndex"/>
2875   <xsd:attribute name="t2" type="ST_WmlColorSchemeIndex"/>
2876   <xsd:attribute name="accent1" type="ST_WmlColorSchemeIndex"/>
2877   <xsd:attribute name="accent2" type="ST_WmlColorSchemeIndex"/>
2878   <xsd:attribute name="accent3" type="ST_WmlColorSchemeIndex"/>
2879   <xsd:attribute name="accent4" type="ST_WmlColorSchemeIndex"/>
2880   <xsd:attribute name="accent5" type="ST_WmlColorSchemeIndex"/>
2881   <xsd:attribute name="accent6" type="ST_WmlColorSchemeIndex"/>
2882   <xsd:attribute name="hyperlink" type="ST_WmlColorSchemeIndex"/>
2883   <xsd:attribute name="followedHyperlink" type="ST_WmlColorSchemeIndex"/>
2884 </xsd:complexType>
2885 <xsd:complexType name="CT_ReadingModeInkLockDown">
2886   <xsd:attribute name="actualPg" type="s:ST_OnOff" use="required"/>
2887   <xsd:attribute name="w" type="ST_PixelsMeasure" use="required"/>
2888   <xsd:attribute name="h" type="ST_PixelsMeasure" use="required"/>
2889   <xsd:attribute name="fontSz" type="ST_DecimalNumberOrPercent" use="required"/>
2890 </xsd:complexType>
2891 <xsd:complexType name="CT_WriteProtection">
2892   <xsd:attribute name="recommended" type="s:ST_OnOff" use="optional"/>
2893   <xsd:attributeGroup ref="AG_Password"/>
2894   <xsd:attributeGroup ref="AG_TransitionalPassword"/>
2895 </xsd:complexType>

```

```

2896 <xsd:complexType name="CT_Settings">
2897   <xsd:sequence>
2898     <xsd:element name="writeProtection" type="CT_WriteProtection" minOccurs="0"/>
2899     <xsd:element name="view" type="CT_View" minOccurs="0"/>
2900     <xsd:element name="zoom" type="CT_Zoom" minOccurs="0"/>
2901     <xsd:element name="removePersonalInformation" type="CT_OnOff" minOccurs="0"/>
2902     <xsd:element name="removeDateAndTime" type="CT_OnOff" minOccurs="0"/>
2903     <xsd:element name="doNotDisplayPageBoundaries" type="CT_OnOff" minOccurs="0"/>
2904     <xsd:element name="displayBackgroundShape" type="CT_OnOff" minOccurs="0"/>
2905     <xsd:element name="printPostScriptOverText" type="CT_OnOff" minOccurs="0"/>
2906     <xsd:element name="printFractionalCharacterWidth" type="CT_OnOff" minOccurs="0"/>
2907     <xsd:element name="printFormsData" type="CT_OnOff" minOccurs="0"/>
2908     <xsd:element name="embedTrueTypeFonts" type="CT_OnOff" minOccurs="0"/>
2909     <xsd:element name="embedSystemFonts" type="CT_OnOff" minOccurs="0"/>
2910     <xsd:element name="saveSubsetFonts" type="CT_OnOff" minOccurs="0"/>
2911     <xsd:element name="saveFormsData" type="CT_OnOff" minOccurs="0"/>
2912     <xsd:element name="mirrorMargins" type="CT_OnOff" minOccurs="0"/>
2913     <xsd:element name="alignBordersAndEdges" type="CT_OnOff" minOccurs="0"/>
2914     <xsd:element name="bordersDoNotSurroundHeader" type="CT_OnOff" minOccurs="0"/>
2915     <xsd:element name="bordersDoNotSurroundFooter" type="CT_OnOff" minOccurs="0"/>
2916     <xsd:element name="gutterAtTop" type="CT_OnOff" minOccurs="0"/>
2917     <xsd:element name="hideSpellingErrors" type="CT_OnOff" minOccurs="0"/>
2918     <xsd:element name="hideGrammaticalErrors" type="CT_OnOff" minOccurs="0"/>
2919     <xsd:element name="activeWritingStyle" type="CT_WritingStyle" minOccurs="0"
2920       maxOccurs="unbounded"/>
2921     <xsd:element name="proofState" type="CT_Proof" minOccurs="0"/>
2922     <xsd:element name="formsDesign" type="CT_OnOff" minOccurs="0"/>
2923     <xsd:element name="attachedTemplate" type="CT_Rel" minOccurs="0"/>
2924     <xsd:element name="linkStyles" type="CT_OnOff" minOccurs="0"/>
2925     <xsd:element name="stylePaneFormatFilter" type="CT_StylePaneFilter" minOccurs="0"/>
2926     <xsd:element name="stylePaneSortMethod" type="CT_StyleSort" minOccurs="0"/>
2927     <xsd:element name="documentType" type="CT_DocType" minOccurs="0"/>
2928     <xsd:element name="mailMerge" type="CT_MailMerge" minOccurs="0"/>
2929     <xsd:element name="revisionView" type="CT_TrackChangesView" minOccurs="0"/>
2930     <xsd:element name="trackRevisions" type="CT_OnOff" minOccurs="0"/>
2931     <xsd:element name="doNotTrackMoves" type="CT_OnOff" minOccurs="0"/>
2932     <xsd:element name="doNotTrackFormatting" type="CT_OnOff" minOccurs="0"/>
2933     <xsd:element name="documentProtection" type="CT_DocProtect" minOccurs="0"/>
2934     <xsd:element name="autoFormatOverride" type="CT_OnOff" minOccurs="0"/>
2935     <xsd:element name="styleLockTheme" type="CT_OnOff" minOccurs="0"/>
2936     <xsd:element name="styleLockQFSet" type="CT_OnOff" minOccurs="0"/>
2937     <xsd:element name="defaultTabStop" type="CT_TwipsMeasure" minOccurs="0"/>
2938     <xsd:element name="autoHyphenation" type="CT_OnOff" minOccurs="0"/>
2939     <xsd:element name="consecutiveHyphenLimit" type="CT_DecimalNumber" minOccurs="0"/>
2940     <xsd:element name="hyphenationZone" type="CT_TwipsMeasure" minOccurs="0"/>
2941     <xsd:element name="doNotHyphenateCaps" type="CT_OnOff" minOccurs="0"/>
2942     <xsd:element name="showEnvelope" type="CT_OnOff" minOccurs="0"/>
2943     <xsd:element name="summaryLength" type="CT_DecimalNumberOrPrecent" minOccurs="0"/>
2944     <xsd:element name="clickAndTypeStyle" type="CT_String" minOccurs="0"/>
2945     <xsd:element name="defaultTableStyle" type="CT_String" minOccurs="0"/>
2946     <xsd:element name="evenAndOddHeaders" type="CT_OnOff" minOccurs="0"/>
2947     <xsd:element name="bookFoldRevPrinting" type="CT_OnOff" minOccurs="0"/>
2948     <xsd:element name="bookFoldPrinting" type="CT_OnOff" minOccurs="0"/>

```

```

2949 <xsd:element name="bookFoldPrintingSheets" type="CT DecimalNumber" minOccurs="0"/>
2950 <xsd:element name="drawingGridHorizontalSpacing" type="CT TwipsMeasure" minOccurs="0"/>
2951 <xsd:element name="drawingGridVerticalSpacing" type="CT TwipsMeasure" minOccurs="0"/>
2952 <xsd:element name="displayHorizontalDrawingGridEvery" type="CT DecimalNumber"
2953     minOccurs="0"/>
2954 <xsd:element name="displayVerticalDrawingGridEvery" type="CT DecimalNumber"
2955     minOccurs="0"/>
2956 <xsd:element name="doNotUseMarginsForDrawingGridOrigin" type="CT OnOff" minOccurs="0"/>
2957 <xsd:element name="drawingGridHorizontalOrigin" type="CT TwipsMeasure" minOccurs="0"/>
2958 <xsd:element name="drawingGridVerticalOrigin" type="CT TwipsMeasure" minOccurs="0"/>
2959 <xsd:element name="doNotShadeFormData" type="CT OnOff" minOccurs="0"/>
2960 <xsd:element name="noPunctuationKerning" type="CT OnOff" minOccurs="0"/>
2961 <xsd:element name="characterSpacingControl" type="CT CharacterSpacing" minOccurs="0"/>
2962 <xsd:element name="printTwoOnOne" type="CT OnOff" minOccurs="0"/>
2963 <xsd:element name="strictFirstAndLastChars" type="CT OnOff" minOccurs="0"/>
2964 <xsd:element name="noLineBreaksAfter" type="CT Kinsoku" minOccurs="0"/>
2965 <xsd:element name="noLineBreaksBefore" type="CT Kinsoku" minOccurs="0"/>
2966 <xsd:element name="savePreviewPicture" type="CT OnOff" minOccurs="0"/>
2967 <xsd:element name="doNotValidateAgainstSchema" type="CT OnOff" minOccurs="0"/>
2968 <xsd:element name="saveInvalidXml" type="CT OnOff" minOccurs="0"/>
2969 <xsd:element name="ignoreMixedContent" type="CT OnOff" minOccurs="0"/>
2970 <xsd:element name="alwaysShowPlaceholderText" type="CT OnOff" minOccurs="0"/>
2971 <xsd:element name="doNotDemarcateInvalidXml" type="CT OnOff" minOccurs="0"/>
2972 <xsd:element name="saveXmlDataOnly" type="CT OnOff" minOccurs="0"/>
2973 <xsd:element name="useXSLTWhenSaving" type="CT OnOff" minOccurs="0"/>
2974 <xsd:element name="saveThroughXslt" type="CT SaveThroughXslt" minOccurs="0"/>
2975 <xsd:element name="showXMLTags" type="CT OnOff" minOccurs="0"/>
2976 <xsd:element name="alwaysMergeEmptyNamespaces" type="CT OnOff" minOccurs="0"/>
2977 <xsd:element name="updateFields" type="CT OnOff" minOccurs="0"/>
2978 <xsd:element name="hdrShapeDefaults" type="CT ShapeDefaults" minOccurs="0"/>
2979 <xsd:element name="footnotePr" type="CT FtnDocProps" minOccurs="0"/>
2980 <xsd:element name="endnotePr" type="CT EdnDocProps" minOccurs="0"/>
2981 <xsd:element name="compat" type="CT Compat" minOccurs="0"/>
2982 <xsd:element name="docVars" type="CT DocVars" minOccurs="0"/>
2983 <xsd:element name="rsids" type="CT DocRsids" minOccurs="0"/>
2984 <xsd:element ref="m:mathPr" minOccurs="0" maxOccurs="1"/>
2985 <xsd:element name="attachedSchema" type="CT String" minOccurs="0" maxOccurs="unbounded"/>
2986 <xsd:element name="themeFontLang" type="CT Language" minOccurs="0" maxOccurs="1"/>
2987 <xsd:element name="clrSchemeMapping" type="CT ColorSchemeMapping" minOccurs="0"/>
2988 <xsd:element name="doNotIncludeSubdocsInStats" type="CT OnOff" minOccurs="0"/>
2989 <xsd:element name="doNotAutoCompressPictures" type="CT OnOff" minOccurs="0"/>
2990 <xsd:element name="forceUpgrade" type="CT Empty" minOccurs="0" maxOccurs="1"/>
2991 <xsd:element name="captions" type="CT Captions" minOccurs="0" maxOccurs="1"/>
2992 <xsd:element name="readModeInkLockDown" type="CT ReadingModeInkLockDown" minOccurs="0"/>
2993 <xsd:element name="smartTagType" type="CT SmartTagType" minOccurs="0"
2994     maxOccurs="unbounded"/>
2995 <xsd:element ref="sl:schemaLibrary" minOccurs="0" maxOccurs="1"/>
2996 <xsd:element name="shapeDefaults" type="CT ShapeDefaults" minOccurs="0"/>
2997 <xsd:element name="doNotEmbedSmartTags" type="CT OnOff" minOccurs="0"/>
2998 <xsd:element name="decimalSymbol" type="CT String" minOccurs="0" maxOccurs="1"/>
2999 <xsd:element name="listSeparator" type="CT String" minOccurs="0" maxOccurs="1"/>
3000 </xsd:sequence>
3001 </xsd:complexType>

```

```

3002 <xsd:complexType name="CT_StyleSort">
3003   <xsd:attribute name="val" type="ST_StyleSort" use="required"/>
3004 </xsd:complexType>
3005 <xsd:complexType name="CT_StylePaneFilter">
3006   <xsd:attribute name="allStyles" type="s:ST_OnOff"/>
3007   <xsd:attribute name="customStyles" type="s:ST_OnOff"/>
3008   <xsd:attribute name="latentStyles" type="s:ST_OnOff"/>
3009   <xsd:attribute name="stylesInUse" type="s:ST_OnOff"/>
3010   <xsd:attribute name="headingStyles" type="s:ST_OnOff"/>
3011   <xsd:attribute name="numberingStyles" type="s:ST_OnOff"/>
3012   <xsd:attribute name="tableStyles" type="s:ST_OnOff"/>
3013   <xsd:attribute name="directFormattingOnRuns" type="s:ST_OnOff"/>
3014   <xsd:attribute name="directFormattingOnParagraphs" type="s:ST_OnOff"/>
3015   <xsd:attribute name="directFormattingOnNumbering" type="s:ST_OnOff"/>
3016   <xsd:attribute name="directFormattingOnTables" type="s:ST_OnOff"/>
3017   <xsd:attribute name="clearFormatting" type="s:ST_OnOff"/>
3018   <xsd:attribute name="top3HeadingStyles" type="s:ST_OnOff"/>
3019   <xsd:attribute name="visibleStyles" type="s:ST_OnOff"/>
3020   <xsd:attribute name="alternateStyleNames" type="s:ST_OnOff"/>
3021   <xsd:attribute name="val" type="ST_ShortHexNumber"/>
3022 </xsd:complexType>
3023 <xsd:simpleType name="ST_StyleSort">
3024   <xsd:restriction base="xsd:string">
3025     <xsd:enumeration value="name"/>
3026     <xsd:enumeration value="priority"/>
3027     <xsd:enumeration value="default"/>
3028     <xsd:enumeration value="font"/>
3029     <xsd:enumeration value="basedOn"/>
3030     <xsd:enumeration value="type"/>
3031     <xsd:enumeration value="0000"/>
3032     <xsd:enumeration value="0001"/>
3033     <xsd:enumeration value="0002"/>
3034     <xsd:enumeration value="0003"/>
3035     <xsd:enumeration value="0004"/>
3036     <xsd:enumeration value="0005"/>
3037   </xsd:restriction>
3038 </xsd:simpleType>
3039 <xsd:complexType name="CT_WebSettings">
3040   <xsd:sequence>
3041     <xsd:element name="frameset" type="CT_Frameset" minOccurs="0"/>
3042     <xsd:element name="divs" type="CT_Divs" minOccurs="0"/>
3043     <xsd:element name="encoding" type="CT_String" minOccurs="0"/>
3044     <xsd:element name="optimizeForBrowser" type="CT_OptimizeForBrowser" minOccurs="0"/>
3045     <xsd:element name="relyOnVML" type="CT_OnOff" minOccurs="0"/>
3046     <xsd:element name="allowPNG" type="CT_OnOff" minOccurs="0"/>
3047     <xsd:element name="doNotRelyOnCSS" type="CT_OnOff" minOccurs="0"/>
3048     <xsd:element name="doNotSaveAsSingleFile" type="CT_OnOff" minOccurs="0"/>
3049     <xsd:element name="doNotOrganizeInFolder" type="CT_OnOff" minOccurs="0"/>
3050     <xsd:element name="doNotUseLongFileNames" type="CT_OnOff" minOccurs="0"/>
3051     <xsd:element name="pixelsPerInch" type="CT_DecimalNumber" minOccurs="0"/>
3052     <xsd:element name="targetScreenSz" type="CT_TargetScreenSz" minOccurs="0"/>
3053     <xsd:element name="saveSmartTagsAsXml" type="CT_OnOff" minOccurs="0"/>
3054   </xsd:sequence>

```

```

3055 </xsd:complexType>
3056 <xsd:simpleType name="ST_FrameScrollbar">
3057   <xsd:restriction base="xsd:string">
3058     <xsd:enumeration value="on"/>
3059     <xsd:enumeration value="off"/>
3060     <xsd:enumeration value="auto"/>
3061   </xsd:restriction>
3062 </xsd:simpleType>
3063 <xsd:complexType name="CT_FrameScrollbar">
3064   <xsd:attribute name="val" type="ST_FrameScrollbar" use="required"/>
3065 </xsd:complexType>
3066 <xsd:complexType name="CT_OptimizeForBrowser">
3067   <xsd:complexContent>
3068     <xsd:extension base="CT_OnOff">
3069       <xsd:attribute name="target" type="s:ST_String" use="optional"/>
3070     </xsd:extension>
3071   </xsd:complexContent>
3072 </xsd:complexType>
3073 <xsd:complexType name="CT_Frame">
3074   <xsd:sequence>
3075     <xsd:element name="sz" type="CT_String" minOccurs="0"/>
3076     <xsd:element name="name" type="CT_String" minOccurs="0"/>
3077     <xsd:element name="title" type="CT_String" minOccurs="0"/>
3078     <xsd:element name="longDesc" type="CT_Rel" minOccurs="0"/>
3079     <xsd:element name="sourceFileName" type="CT_Rel" minOccurs="0"/>
3080     <xsd:element name="marW" type="CT_PixelsMeasure" minOccurs="0"/>
3081     <xsd:element name="marH" type="CT_PixelsMeasure" minOccurs="0"/>
3082     <xsd:element name="scrollbar" type="CT_FrameScrollbar" minOccurs="0"/>
3083     <xsd:element name="noResizeAllowed" type="CT_OnOff" minOccurs="0"/>
3084     <xsd:element name="linkedToFile" type="CT_OnOff" minOccurs="0"/>
3085   </xsd:sequence>
3086 </xsd:complexType>
3087 <xsd:simpleType name="ST_FrameLayout">
3088   <xsd:restriction base="xsd:string">
3089     <xsd:enumeration value="rows"/>
3090     <xsd:enumeration value="cols"/>
3091     <xsd:enumeration value="none"/>
3092   </xsd:restriction>
3093 </xsd:simpleType>
3094 <xsd:complexType name="CT_FrameLayout">
3095   <xsd:attribute name="val" type="ST_FrameLayout" use="required"/>
3096 </xsd:complexType>
3097 <xsd:complexType name="CT_FramesetSplitbar">
3098   <xsd:sequence>
3099     <xsd:element name="w" type="CT_TwipsMeasure" minOccurs="0"/>
3100     <xsd:element name="color" type="CT_Color" minOccurs="0"/>
3101     <xsd:element name="noBorder" type="CT_OnOff" minOccurs="0"/>
3102     <xsd:element name="flatBorders" type="CT_OnOff" minOccurs="0"/>
3103   </xsd:sequence>
3104 </xsd:complexType>
3105 <xsd:complexType name="CT_Frameset">
3106   <xsd:sequence>
3107     <xsd:element name="sz" type="CT_String" minOccurs="0"/>

```

```

3108     <xsd:element name="framesetSplitbar" type="CT FramesetSplitbar" minOccurs="0"/>
3109     <xsd:element name="frameLayout" type="CT FrameLayout" minOccurs="0"/>
3110     <xsd:element name="title" type="CT String" minOccurs="0"/>
3111     <xsd:choice minOccurs="0" maxOccurs="unbounded">
3112         <xsd:element name="frameset" type="CT Frameset" minOccurs="0" maxOccurs="unbounded"/>
3113         <xsd:element name="frame" type="CT Frame" minOccurs="0" maxOccurs="unbounded"/>
3114     </xsd:choice>
3115 </xsd:sequence>
3116 </xsd:complexType>
3117 <xsd:complexType name="CT_NumPicBullet">
3118     <xsd:choice>
3119         <xsd:element name="pict" type="CT Picture"/>
3120         <xsd:element name="drawing" type="CT Drawing"/>
3121     </xsd:choice>
3122     <xsd:attribute name="numPicBulletId" type="ST DecimalNumber" use="required"/>
3123 </xsd:complexType>
3124 <xsd:simpleType name="ST_LevelSuffix">
3125     <xsd:restriction base="xsd:string">
3126         <xsd:enumeration value="tab"/>
3127         <xsd:enumeration value="space"/>
3128         <xsd:enumeration value="nothing"/>
3129     </xsd:restriction>
3130 </xsd:simpleType>
3131 <xsd:complexType name="CT_LevelSuffix">
3132     <xsd:attribute name="val" type="ST LevelSuffix" use="required"/>
3133 </xsd:complexType>
3134 <xsd:complexType name="CT_LevelText">
3135     <xsd:attribute name="val" type="s:ST String" use="optional"/>
3136     <xsd:attribute name="null" type="s:ST OnOff" use="optional"/>
3137 </xsd:complexType>
3138 <xsd:complexType name="CT_LvlLegacy">
3139     <xsd:attribute name="legacy" type="s:ST OnOff" use="optional"/>
3140     <xsd:attribute name="legacySpace" type="s:ST TwipsMeasure" use="optional"/>
3141     <xsd:attribute name="legacyIndent" type="ST SignedTwipsMeasure" use="optional"/>
3142 </xsd:complexType>
3143 <xsd:complexType name="CT_Lvl">
3144     <xsd:sequence>
3145         <xsd:element name="start" type="CT DecimalNumber" minOccurs="0"/>
3146         <xsd:element name="numFmt" type="CT NumFmt" minOccurs="0"/>
3147         <xsd:element name="lvlRestart" type="CT DecimalNumber" minOccurs="0"/>
3148         <xsd:element name="pStyle" type="CT String" minOccurs="0"/>
3149         <xsd:element name="isLgl" type="CT OnOff" minOccurs="0"/>
3150         <xsd:element name="suff" type="CT LevelSuffix" minOccurs="0"/>
3151         <xsd:element name="lvlText" type="CT LevelText" minOccurs="0"/>
3152         <xsd:element name="lvlPicBulletId" type="CT DecimalNumber" minOccurs="0"/>
3153         <xsd:element name="legacy" type="CT LvlLegacy" minOccurs="0"/>
3154         <xsd:element name="lvlJc" type="CT Jc" minOccurs="0"/>
3155         <xsd:element name="pPr" type="CT PPrGeneral" minOccurs="0"/>
3156         <xsd:element name="rPr" type="CT RPr" minOccurs="0"/>
3157     </xsd:sequence>
3158     <xsd:attribute name="ilvl" type="ST DecimalNumber" use="required"/>
3159     <xsd:attribute name="tplc" type="ST LongHexNumber" use="optional"/>
3160     <xsd:attribute name="tentative" type="s:ST OnOff" use="optional"/>

```



```

3161 </xsd:complexType>
3162 <xsd:simpleType name="ST_MultiLevelType">
3163   <xsd:restriction base="xsd:string">
3164     <xsd:enumeration value="singleLevel"/>
3165     <xsd:enumeration value="multilevel"/>
3166     <xsd:enumeration value="hybridMultilevel"/>
3167   </xsd:restriction>
3168 </xsd:simpleType>
3169 <xsd:complexType name="CT_MultiLevelType">
3170   <xsd:attribute name="val" type="ST_MultiLevelType" use="required"/>
3171 </xsd:complexType>
3172 <xsd:complexType name="CT_AbstractNum">
3173   <xsd:sequence>
3174     <xsd:element name="nsid" type="CT_LongHexNumber" minOccurs="0"/>
3175     <xsd:element name="multiLevelType" type="CT_MultiLevelType" minOccurs="0"/>
3176     <xsd:element name="tmpl" type="CT_LongHexNumber" minOccurs="0"/>
3177     <xsd:element name="name" type="CT_String" minOccurs="0"/>
3178     <xsd:element name="styleLink" type="CT_String" minOccurs="0"/>
3179     <xsd:element name="numStyleLink" type="CT_String" minOccurs="0"/>
3180     <xsd:element name="lvl" type="CT_Lvl" minOccurs="0" maxOccurs="9"/>
3181   </xsd:sequence>
3182   <xsd:attribute name="abstractNumId" type="ST_DecimalNumber" use="required"/>
3183 </xsd:complexType>
3184 <xsd:complexType name="CT_NumLvl">
3185   <xsd:sequence>
3186     <xsd:element name="startOverride" type="CT_DecimalNumber" minOccurs="0"/>
3187     <xsd:element name="lvl" type="CT_Lvl" minOccurs="0" maxOccurs="1"/>
3188   </xsd:sequence>
3189   <xsd:attribute name="ilvl" type="ST_DecimalNumber" use="required"/>
3190 </xsd:complexType>
3191 <xsd:complexType name="CT_Num">
3192   <xsd:sequence>
3193     <xsd:element name="abstractNumId" type="CT_DecimalNumber" minOccurs="1"/>
3194     <xsd:element name="lvlOverride" type="CT_NumLvl" minOccurs="0" maxOccurs="9"/>
3195   </xsd:sequence>
3196   <xsd:attribute name="numId" type="ST_DecimalNumber" use="required"/>
3197 </xsd:complexType>
3198 <xsd:complexType name="CT_Numbering">
3199   <xsd:sequence>
3200     <xsd:element name="numPicBullet" type="CT_NumPicBullet" minOccurs="0"
3201       maxOccurs="unbounded"/>
3202     <xsd:element name="abstractNum" type="CT_AbstractNum" minOccurs="0"
3203       maxOccurs="unbounded"/>
3204     <xsd:element name="num" type="CT_Num" minOccurs="0" maxOccurs="unbounded"/>
3205     <xsd:element name="numIdMacAtCleanup" type="CT_DecimalNumber" minOccurs="0"/>
3206   </xsd:sequence>
3207 </xsd:complexType>
3208 <xsd:simpleType name="ST_TblStyleOverrideType">
3209   <xsd:restriction base="xsd:string">
3210     <xsd:enumeration value="wholeTable"/>
3211     <xsd:enumeration value="firstRow"/>
3212     <xsd:enumeration value="lastRow"/>
3213     <xsd:enumeration value="firstCol"/>

```

```

3214         <xsd:enumeration value="lastCol"/>
3215         <xsd:enumeration value="band1Vert"/>
3216         <xsd:enumeration value="band2Vert"/>
3217         <xsd:enumeration value="band1Horz"/>
3218         <xsd:enumeration value="band2Horz"/>
3219         <xsd:enumeration value="neCell"/>
3220         <xsd:enumeration value="nwCell"/>
3221         <xsd:enumeration value="seCell"/>
3222         <xsd:enumeration value="swCell"/>
3223     </xsd:restriction>
3224 </xsd:simpleType>
3225 <xsd:complexType name="CT_TblStylePr">
3226     <xsd:sequence>
3227         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0"/>
3228         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
3229         <xsd:element name="tblPr" type="CT_TblPrBase" minOccurs="0"/>
3230         <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
3231         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
3232     </xsd:sequence>
3233     <xsd:attribute name="type" type="ST_TblStyleOverrideType" use="required"/>
3234 </xsd:complexType>
3235 <xsd:simpleType name="ST_StyleType">
3236     <xsd:restriction base="xsd:string">
3237         <xsd:enumeration value="paragraph"/>
3238         <xsd:enumeration value="character"/>
3239         <xsd:enumeration value="table"/>
3240         <xsd:enumeration value="numbering"/>
3241     </xsd:restriction>
3242 </xsd:simpleType>
3243 <xsd:complexType name="CT_Style">
3244     <xsd:sequence>
3245         <xsd:element name="name" type="CT_String" minOccurs="0" maxOccurs="1"/>
3246         <xsd:element name="aliases" type="CT_String" minOccurs="0"/>
3247         <xsd:element name="basedOn" type="CT_String" minOccurs="0"/>
3248         <xsd:element name="next" type="CT_String" minOccurs="0"/>
3249         <xsd:element name="link" type="CT_String" minOccurs="0"/>
3250         <xsd:element name="autoRedefine" type="CT_OnOff" minOccurs="0"/>
3251         <xsd:element name="hidden" type="CT_OnOff" minOccurs="0"/>
3252         <xsd:element name="uiPriority" type="CT_DecimalNumber" minOccurs="0"/>
3253         <xsd:element name="semiHidden" type="CT_OnOff" minOccurs="0"/>
3254         <xsd:element name="unhideWhenUsed" type="CT_OnOff" minOccurs="0"/>
3255         <xsd:element name="qFormat" type="CT_OnOff" minOccurs="0"/>
3256         <xsd:element name="locked" type="CT_OnOff" minOccurs="0"/>
3257         <xsd:element name="personal" type="CT_OnOff" minOccurs="0"/>
3258         <xsd:element name="personalCompose" type="CT_OnOff" minOccurs="0"/>
3259         <xsd:element name="personalReply" type="CT_OnOff" minOccurs="0"/>
3260         <xsd:element name="rsid" type="CT_LongHexNumber" minOccurs="0"/>
3261         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0" maxOccurs="1"/>
3262         <xsd:element name="rPr" type="CT_RPr" minOccurs="0" maxOccurs="1"/>
3263         <xsd:element name="tblPr" type="CT_TblPrBase" minOccurs="0" maxOccurs="1"/>
3264         <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
3265         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
3266         <xsd:element name="tblStylePr" type="CT_TblStylePr" minOccurs="0" maxOccurs="unbounded"/>

```

```

3267     </xsd:sequence>
3268     <xsd:attribute name="type" type="ST_StyleType" use="optional"/>
3269     <xsd:attribute name="styleId" type="s:ST_String" use="optional"/>
3270     <xsd:attribute name="default" type="s:ST_OnOff" use="optional"/>
3271     <xsd:attribute name="customStyle" type="s:ST_OnOff" use="optional"/>
3272 </xsd:complexType>
3273 <xsd:complexType name="CT_LsdException">
3274     <xsd:attribute name="name" type="s:ST_String" use="required"/>
3275     <xsd:attribute name="locked" type="s:ST_OnOff"/>
3276     <xsd:attribute name="uiPriority" type="ST_DecimalNumber"/>
3277     <xsd:attribute name="semiHidden" type="s:ST_OnOff"/>
3278     <xsd:attribute name="unhideWhenUsed" type="s:ST_OnOff"/>
3279     <xsd:attribute name="qFormat" type="s:ST_OnOff"/>
3280 </xsd:complexType>
3281 <xsd:complexType name="CT_LatentStyles">
3282     <xsd:sequence>
3283         <xsd:element name="lsdException" type="CT_LsdException" minOccurs="0"
3284             maxOccurs="unbounded"/>
3285     </xsd:sequence>
3286     <xsd:attribute name="defLockedState" type="s:ST_OnOff"/>
3287     <xsd:attribute name="defUIPriority" type="ST_DecimalNumber"/>
3288     <xsd:attribute name="defSemiHidden" type="s:ST_OnOff"/>
3289     <xsd:attribute name="defUnhideWhenUsed" type="s:ST_OnOff"/>
3290     <xsd:attribute name="defQFormat" type="s:ST_OnOff"/>
3291     <xsd:attribute name="count" type="ST_DecimalNumber"/>
3292 </xsd:complexType>
3293 <xsd:complexType name="CT_Styles">
3294     <xsd:sequence>
3295         <xsd:element name="docDefaults" type="CT_DocDefaults" minOccurs="0"/>
3296         <xsd:element name="latentStyles" type="CT_LatentStyles" minOccurs="0" maxOccurs="1"/>
3297         <xsd:element name="style" type="CT_Style" minOccurs="0" maxOccurs="unbounded"/>
3298     </xsd:sequence>
3299 </xsd:complexType>
3300 <xsd:complexType name="CT_Panose">
3301     <xsd:attribute name="val" type="s:ST_Panose" use="required"/>
3302 </xsd:complexType>
3303 <xsd:simpleType name="ST_FontFamily">
3304     <xsd:restriction base="xsd:string">
3305         <xsd:enumeration value="decorative"/>
3306         <xsd:enumeration value="modern"/>
3307         <xsd:enumeration value="roman"/>
3308         <xsd:enumeration value="script"/>
3309         <xsd:enumeration value="swiss"/>
3310         <xsd:enumeration value="auto"/>
3311     </xsd:restriction>
3312 </xsd:simpleType>
3313 <xsd:complexType name="CT_FontFamily">
3314     <xsd:attribute name="val" type="ST_FontFamily" use="required"/>
3315 </xsd:complexType>
3316 <xsd:simpleType name="ST_Pitch">
3317     <xsd:restriction base="xsd:string">
3318         <xsd:enumeration value="fixed"/>
3319         <xsd:enumeration value="variable"/>

```

```

3320     <xsd:enumeration value="default"/>
3321   </xsd:restriction>
3322 </xsd:simpleType>
3323 <xsd:complexType name="CT_Pitch">
3324   <xsd:attribute name="val" type="ST_Pitch" use="required"/>
3325 </xsd:complexType>
3326 <xsd:complexType name="CT_FontSig">
3327   <xsd:attribute name="usb0" use="required" type="ST_LongHexNumber"/>
3328   <xsd:attribute name="usb1" use="required" type="ST_LongHexNumber"/>
3329   <xsd:attribute name="usb2" use="required" type="ST_LongHexNumber"/>
3330   <xsd:attribute name="usb3" use="required" type="ST_LongHexNumber"/>
3331   <xsd:attribute name="csb0" use="required" type="ST_LongHexNumber"/>
3332   <xsd:attribute name="csb1" use="required" type="ST_LongHexNumber"/>
3333 </xsd:complexType>
3334 <xsd:complexType name="CT_FontRel">
3335   <xsd:complexContent>
3336     <xsd:extension base="CT_Rel">
3337       <xsd:attribute name="fontKey" type="s:ST_Guid"/>
3338       <xsd:attribute name="subsetting" type="s:ST_OnOff"/>
3339     </xsd:extension>
3340   </xsd:complexContent>
3341 </xsd:complexType>
3342 <xsd:complexType name="CT_Font">
3343   <xsd:sequence>
3344     <xsd:element name="altName" type="CT_String" minOccurs="0" maxOccurs="1"/>
3345     <xsd:element name="panose1" type="CT_Panose" minOccurs="0" maxOccurs="1"/>
3346     <xsd:element name="charset" type="CT_Charset" minOccurs="0" maxOccurs="1"/>
3347     <xsd:element name="family" type="CT_FontFamily" minOccurs="0" maxOccurs="1"/>
3348     <xsd:element name="notTrueType" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
3349     <xsd:element name="pitch" type="CT_Pitch" minOccurs="0" maxOccurs="1"/>
3350     <xsd:element name="sig" type="CT_FontSig" minOccurs="0" maxOccurs="1"/>
3351     <xsd:element name="embedRegular" type="CT_FontRel" minOccurs="0" maxOccurs="1"/>
3352     <xsd:element name="embedBold" type="CT_FontRel" minOccurs="0" maxOccurs="1"/>
3353     <xsd:element name="embedItalic" type="CT_FontRel" minOccurs="0" maxOccurs="1"/>
3354     <xsd:element name="embedBoldItalic" type="CT_FontRel" minOccurs="0" maxOccurs="1"/>
3355   </xsd:sequence>
3356   <xsd:attribute name="name" type="s:ST_String" use="required"/>
3357 </xsd:complexType>
3358 <xsd:complexType name="CT_FontsList">
3359   <xsd:sequence>
3360     <xsd:element name="font" type="CT_Font" minOccurs="0" maxOccurs="unbounded"/>
3361   </xsd:sequence>
3362 </xsd:complexType>
3363 <xsd:complexType name="CT_DivBdr">
3364   <xsd:sequence>
3365     <xsd:element name="top" type="CT_Border" minOccurs="0"/>
3366     <xsd:element name="left" type="CT_Border" minOccurs="0"/>
3367     <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
3368     <xsd:element name="right" type="CT_Border" minOccurs="0"/>
3369   </xsd:sequence>
3370 </xsd:complexType>
3371 <xsd:complexType name="CT_Div">
3372   <xsd:sequence>

```

```

3373     <xsd:element name="blockquote" type="CT_OnOff" minOccurs="0"/>
3374     <xsd:element name="bodyDiv" type="CT_OnOff" minOccurs="0"/>
3375     <xsd:element name="marLeft" type="CT_SignedTwipsMeasure"/>
3376     <xsd:element name="marRight" type="CT_SignedTwipsMeasure"/>
3377     <xsd:element name="marTop" type="CT_SignedTwipsMeasure"/>
3378     <xsd:element name="marBottom" type="CT_SignedTwipsMeasure"/>
3379     <xsd:element name="divBdr" type="CT_DivBdr" minOccurs="0"/>
3380     <xsd:element name="divsChild" type="CT_Divs" minOccurs="0" maxOccurs="unbounded"/>
3381   </xsd:sequence>
3382   <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
3383 </xsd:complexType>
3384 <xsd:complexType name="CT_Divs">
3385   <xsd:sequence minOccurs="1" maxOccurs="unbounded">
3386     <xsd:element name="div" type="CT_Div"/>
3387   </xsd:sequence>
3388 </xsd:complexType>
3389 <xsd:complexType name="CT_TxbxContent">
3390   <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
3391 </xsd:complexType>
3392 <xsd:element name="txbxContent" type="CT_TxbxContent"/>
3393 <xsd:group name="EG_MathContent">
3394   <xsd:choice>
3395     <xsd:element ref="m:oMathPara"/>
3396     <xsd:element ref="m:oMath"/>
3397   </xsd:choice>
3398 </xsd:group>
3399 <xsd:group name="EG_BlockLevelChunkElts">
3400   <xsd:choice>
3401     <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
3402   </xsd:choice>
3403 </xsd:group>
3404 <xsd:group name="EG_BlockLevelElts">
3405   <xsd:choice>
3406     <xsd:group ref="EG_BlockLevelChunkElts" minOccurs="0" maxOccurs="unbounded"/>
3407     <xsd:element name="altChunk" type="CT_AltChunk" minOccurs="0" maxOccurs="unbounded"/>
3408   </xsd:choice>
3409 </xsd:group>
3410 <xsd:group name="EG_RunLevelElts">
3411   <xsd:choice>
3412     <xsd:element name="proofErr" minOccurs="0" type="CT_ProofErr"/>
3413     <xsd:element name="permStart" minOccurs="0" type="CT_PermStart"/>
3414     <xsd:element name="permEnd" minOccurs="0" type="CT_Perm"/>
3415     <xsd:group ref="EG_RangeMarkupElements" minOccurs="0" maxOccurs="unbounded"/>
3416     <xsd:element name="ins" type="CT_RunTrackChange" minOccurs="0"/>
3417     <xsd:element name="del" type="CT_RunTrackChange" minOccurs="0"/>
3418     <xsd:element name="moveFrom" type="CT_RunTrackChange"/>
3419     <xsd:element name="moveTo" type="CT_RunTrackChange"/>
3420     <xsd:group ref="EG_MathContent" minOccurs="0" maxOccurs="unbounded"/>
3421   </xsd:choice>
3422 </xsd:group>
3423 <xsd:complexType name="CT_Body">
3424   <xsd:sequence>
3425     <xsd:group ref="EG_BlockLevelElts" minOccurs="0" maxOccurs="unbounded"/>

```

```

3426     <xsd:element name="sectPr" minOccurs="0" maxOccurs="1" type="CT_SectPr"/>
3427   </xsd:sequence>
3428 </xsd:complexType>
3429 <xsd:complexType name="CT_ShapeDefaults">
3430   <xsd:choice maxOccurs="unbounded">
3431     <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
3432       minOccurs="0" maxOccurs="unbounded"/>
3433   </xsd:choice>
3434 </xsd:complexType>
3435 <xsd:complexType name="CT_Comments">
3436   <xsd:sequence>
3437     <xsd:element name="comment" type="CT_Comment" minOccurs="0" maxOccurs="unbounded"/>
3438   </xsd:sequence>
3439 </xsd:complexType>
3440 <xsd:element name="comments" type="CT_Comments"/>
3441 <xsd:complexType name="CT_Footnotes">
3442   <xsd:sequence maxOccurs="unbounded">
3443     <xsd:element name="footnote" type="CT_FtnEdn" minOccurs="0"/>
3444   </xsd:sequence>
3445 </xsd:complexType>
3446 <xsd:element name="footnotes" type="CT_Footnotes"/>
3447 <xsd:complexType name="CT_Endnotes">
3448   <xsd:sequence maxOccurs="unbounded">
3449     <xsd:element name="endnote" type="CT_FtnEdn" minOccurs="0"/>
3450   </xsd:sequence>
3451 </xsd:complexType>
3452 <xsd:element name="endnotes" type="CT_Endnotes"/>
3453 <xsd:element name="hdr" type="CT_HdrFtr"/>
3454 <xsd:element name="ftr" type="CT_HdrFtr"/>
3455 <xsd:complexType name="CT_SmartTagType">
3456   <xsd:attribute name="namespaceuri" type="s:ST_String"/>
3457   <xsd:attribute name="name" type="s:ST_String"/>
3458   <xsd:attribute name="url" type="s:ST_String"/>
3459 </xsd:complexType>
3460 <xsd:simpleType name="ST_ThemeColor">
3461   <xsd:restriction base="xsd:string">
3462     <xsd:enumeration value="dark1"/>
3463     <xsd:enumeration value="light1"/>
3464     <xsd:enumeration value="dark2"/>
3465     <xsd:enumeration value="light2"/>
3466     <xsd:enumeration value="accent1"/>
3467     <xsd:enumeration value="accent2"/>
3468     <xsd:enumeration value="accent3"/>
3469     <xsd:enumeration value="accent4"/>
3470     <xsd:enumeration value="accent5"/>
3471     <xsd:enumeration value="accent6"/>
3472     <xsd:enumeration value="hyperlink"/>
3473     <xsd:enumeration value="followedHyperlink"/>
3474     <xsd:enumeration value="none"/>
3475     <xsd:enumeration value="background1"/>
3476     <xsd:enumeration value="text1"/>
3477     <xsd:enumeration value="background2"/>
3478     <xsd:enumeration value="text2"/>

```

```

3479     </xsd:restriction>
3480 </xsd:simpleType>
3481 <xsd:simpleType name="ST_DocPartBehavior">
3482     <xsd:restriction base="xsd:string">
3483         <xsd:enumeration value="content"/>
3484         <xsd:enumeration value="p"/>
3485         <xsd:enumeration value="pg"/>
3486     </xsd:restriction>
3487 </xsd:simpleType>
3488 <xsd:complexType name="CT_DocPartBehavior">
3489     <xsd:attribute name="val" use="required" type="ST_DocPartBehavior"/>
3490 </xsd:complexType>
3491 <xsd:complexType name="CT_DocPartBehaviors">
3492     <xsd:choice>
3493         <xsd:element name="behavior" type="CT_DocPartBehavior" maxOccurs="unbounded"/>
3494     </xsd:choice>
3495 </xsd:complexType>
3496 <xsd:simpleType name="ST_DocPartType">
3497     <xsd:restriction base="xsd:string">
3498         <xsd:enumeration value="none"/>
3499         <xsd:enumeration value="normal"/>
3500         <xsd:enumeration value="autoExp"/>
3501         <xsd:enumeration value="toolbar"/>
3502         <xsd:enumeration value="speller"/>
3503         <xsd:enumeration value="formFld"/>
3504         <xsd:enumeration value="bbPlcHdr"/>
3505     </xsd:restriction>
3506 </xsd:simpleType>
3507 <xsd:complexType name="CT_DocPartType">
3508     <xsd:attribute name="val" use="required" type="ST_DocPartType"/>
3509 </xsd:complexType>
3510 <xsd:complexType name="CT_DocPartTypes">
3511     <xsd:choice>
3512         <xsd:element name="type" type="CT_DocPartType" maxOccurs="unbounded"/>
3513     </xsd:choice>
3514     <xsd:attribute name="all" type="s:ST_OnOff" use="optional"/>
3515 </xsd:complexType>
3516 <xsd:simpleType name="ST_DocPartGallery">
3517     <xsd:restriction base="xsd:string">
3518         <xsd:enumeration value="placeholder"/>
3519         <xsd:enumeration value="any"/>
3520         <xsd:enumeration value="default"/>
3521         <xsd:enumeration value="docParts"/>
3522         <xsd:enumeration value="coverPg"/>
3523         <xsd:enumeration value="eq"/>
3524         <xsd:enumeration value="ftrs"/>
3525         <xsd:enumeration value="hdrs"/>
3526         <xsd:enumeration value="pgNum"/>
3527         <xsd:enumeration value="tbls"/>
3528         <xsd:enumeration value="watermarks"/>
3529         <xsd:enumeration value="autoTxt"/>
3530         <xsd:enumeration value="txtBox"/>
3531         <xsd:enumeration value="pgNumT"/>

```

```

3532     <xsd:enumeration value="pgNumB"/>
3533     <xsd:enumeration value="pgNumMargins"/>
3534     <xsd:enumeration value="tblOfContents"/>
3535     <xsd:enumeration value="bib"/>
3536     <xsd:enumeration value="custQuickParts"/>
3537     <xsd:enumeration value="custCoverPg"/>
3538     <xsd:enumeration value="custEq"/>
3539     <xsd:enumeration value="custFtrs"/>
3540     <xsd:enumeration value="custHdrs"/>
3541     <xsd:enumeration value="custPgNum"/>
3542     <xsd:enumeration value="custTbls"/>
3543     <xsd:enumeration value="custWatermarks"/>
3544     <xsd:enumeration value="custAutoTxt"/>
3545     <xsd:enumeration value="custTxtBox"/>
3546     <xsd:enumeration value="custPgNumT"/>
3547     <xsd:enumeration value="custPgNumB"/>
3548     <xsd:enumeration value="custPgNumMargins"/>
3549     <xsd:enumeration value="custTblOfContents"/>
3550     <xsd:enumeration value="custBib"/>
3551     <xsd:enumeration value="custom1"/>
3552     <xsd:enumeration value="custom2"/>
3553     <xsd:enumeration value="custom3"/>
3554     <xsd:enumeration value="custom4"/>
3555     <xsd:enumeration value="custom5"/>
3556   </xsd:restriction>
3557 </xsd:simpleType>
3558 <xsd:complexType name="CT_DocPartGallery">
3559   <xsd:attribute name="val" type="ST_DocPartGallery" use="required"/>
3560 </xsd:complexType>
3561 <xsd:complexType name="CT_DocPartCategory">
3562   <xsd:sequence>
3563     <xsd:element name="name" type="CT_String" minOccurs="1" maxOccurs="1"/>
3564     <xsd:element name="gallery" type="CT_DocPartGallery" minOccurs="1" maxOccurs="1"/>
3565   </xsd:sequence>
3566 </xsd:complexType>
3567 <xsd:complexType name="CT_DocPartName">
3568   <xsd:attribute name="val" type="s:ST_String" use="required"/>
3569   <xsd:attribute name="decorated" type="s:ST_OnOff" use="optional"/>
3570 </xsd:complexType>
3571 <xsd:complexType name="CT_DocPartPr">
3572   <xsd:all>
3573     <xsd:element name="style" type="CT_String" minOccurs="0"/>
3574     <xsd:element name="category" type="CT_DocPartCategory" minOccurs="0"/>
3575     <xsd:element name="types" type="CT_DocPartTypes" minOccurs="0"/>
3576     <xsd:element name="behaviors" type="CT_DocPartBehaviors" minOccurs="0"/>
3577     <xsd:element name="description" type="CT_String" minOccurs="0"/>
3578     <xsd:element name="guid" type="CT_Guid" minOccurs="0"/>
3579   </xsd:all>
3580 </xsd:complexType>
3581 <xsd:complexType name="CT_DocPart">
3582   <xsd:sequence>
3583     <xsd:element name="docPartPr" type="CT_DocPartPr" minOccurs="0"/>
3584     <xsd:element name="docPartBody" type="CT_Body" minOccurs="0"/>

```



```

3585     </xsd:sequence>
3586 </xsd:complexType>
3587 <xsd:complexType name="CT_DocParts">
3588     <xsd:choice>
3589         <xsd:element name="docPart" type="CT_DocPart" minOccurs="1" maxOccurs="unbounded"/>
3590     </xsd:choice>
3591 </xsd:complexType>
3592 <xsd:element name="settings" type="CT_Settings"/>
3593 <xsd:element name="webSettings" type="CT_WebSettings"/>
3594 <xsd:element name="fonts" type="CT_FontsList"/>
3595 <xsd:element name="numbering" type="CT_Numbering"/>
3596 <xsd:element name="styles" type="CT_Styles"/>
3597 <xsd:simpleType name="ST_CaptionPos">
3598     <xsd:restriction base="xsd:string">
3599         <xsd:enumeration value="above"/>
3600         <xsd:enumeration value="below"/>
3601         <xsd:enumeration value="left"/>
3602         <xsd:enumeration value="right"/>
3603     </xsd:restriction>
3604 </xsd:simpleType>
3605 <xsd:complexType name="CT_Caption">
3606     <xsd:attribute name="name" type="s:ST_String" use="required"/>
3607     <xsd:attribute name="pos" type="ST_CaptionPos" use="optional"/>
3608     <xsd:attribute name="chapNum" type="s:ST_OnOff" use="optional"/>
3609     <xsd:attribute name="heading" type="ST_DecimalNumber" use="optional"/>
3610     <xsd:attribute name="noLabel" type="s:ST_OnOff" use="optional"/>
3611     <xsd:attribute name="numFmt" type="ST_NumberFormat" use="optional"/>
3612     <xsd:attribute name="sep" type="ST_ChapterSep" use="optional"/>
3613 </xsd:complexType>
3614 <xsd:complexType name="CT_AutoCaption">
3615     <xsd:attribute name="name" type="s:ST_String" use="required"/>
3616     <xsd:attribute name="caption" type="s:ST_String" use="required"/>
3617 </xsd:complexType>
3618 <xsd:complexType name="CT_AutoCaptions">
3619     <xsd:sequence>
3620         <xsd:element name="autoCaption" type="CT_AutoCaption" minOccurs="1"
3621             maxOccurs="unbounded"/>
3622     </xsd:sequence>
3623 </xsd:complexType>
3624 <xsd:complexType name="CT_Captions">
3625     <xsd:sequence>
3626         <xsd:element name="caption" type="CT_Caption" minOccurs="1" maxOccurs="unbounded"/>
3627         <xsd:element name="autoCaptions" type="CT_AutoCaptions" minOccurs="0" maxOccurs="1"/>
3628     </xsd:sequence>
3629 </xsd:complexType>
3630 <xsd:complexType name="CT_DocumentBase">
3631     <xsd:sequence>
3632         <xsd:element name="background" type="CT_Background" minOccurs="0"/>
3633     </xsd:sequence>
3634 </xsd:complexType>
3635 <xsd:complexType name="CT_Document">
3636     <xsd:complexContent>
3637         <xsd:extension base="CT_DocumentBase">

```

```

3638         <xsd:sequence>
3639             <xsd:element name="body" type="CT_Body" minOccurs="0" maxOccurs="1"/>
3640         </xsd:sequence>
3641         <xsd:attribute name="conformance" type="s:ST_ConformanceClass"/>
3642     </xsd:extension>
3643 </xsd:complexContent>
3644 </xsd:complexType>
3645 <xsd:complexType name="CT_GlossaryDocument">
3646     <xsd:complexContent>
3647         <xsd:extension base="CT_DocumentBase">
3648             <xsd:sequence>
3649                 <xsd:element name="docParts" type="CT_DocParts" minOccurs="0"/>
3650             </xsd:sequence>
3651         </xsd:extension>
3652     </xsd:complexContent>
3653 </xsd:complexType>
3654 <xsd:element name="document" type="CT_Document"/>
3655 <xsd:element name="glossaryDocument" type="CT_GlossaryDocument"/>
3656 </xsd:schema>

```

A.2 SpreadsheetML

This schema is available in the file sml.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/spreadsheetml/2006/main"
3   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4   xmlns:xdr="http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
5   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6   targetNamespace="http://schemas.openxmlformats.org/spreadsheetml/2006/main"
7   elementFormDefault="qualified">
8     <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9       schemaLocation="shared-relationshipReference.xsd"/>
10    <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11      schemaLocation="shared-commonSimpleTypes.xsd"/>
12    <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
13      schemaLocation="dml-spreadsheetDrawing.xsd"/>
14    <xsd:complexType name="CT_AutoFilter">
15      <xsd:sequence>
16        <xsd:element name="filterColumn" minOccurs="0" maxOccurs="unbounded"
17          type="CT_FilterColumn"/>
18        <xsd:element name="sortState" minOccurs="0" maxOccurs="1" type="CT_SortState"/>
19        <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
20      </xsd:sequence>
21      <xsd:attribute name="ref" type="ST_Ref"/>
22    </xsd:complexType>
23    <xsd:complexType name="CT_FilterColumn">
24      <xsd:choice minOccurs="0" maxOccurs="1">
25        <xsd:element name="filters" type="CT_Filters" minOccurs="0" maxOccurs="1"/>
26        <xsd:element name="top10" type="CT_Top10" minOccurs="0" maxOccurs="1"/>
27        <xsd:element name="customFilters" type="CT_CustomFilters" minOccurs="0" maxOccurs="1"/>
28        <xsd:element name="dynamicFilter" type="CT_DynamicFilter" minOccurs="0" maxOccurs="1"/>
29        <xsd:element name="colorFilter" type="CT_ColorFilter" minOccurs="0" maxOccurs="1"/>

```

```

30     <xsd:element name="iconFilter" minOccurs="0" maxOccurs="1" type="CT_IconFilter"/>
31     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
32 </xsd:choice>
33 <xsd:attribute name="colId" type="xsd:unsignedInt" use="required"/>
34 <xsd:attribute name="hiddenButton" type="xsd:boolean" use="optional" default="false"/>
35 <xsd:attribute name="showButton" type="xsd:boolean" use="optional" default="true"/>
36 </xsd:complexType>
37 <xsd:complexType name="CT_Filters">
38     <xsd:sequence>
39         <xsd:element name="filter" type="CT_Filter" minOccurs="0" maxOccurs="unbounded"/>
40         <xsd:element name="dateGroupItem" type="CT_DateGroupItem" minOccurs="0"
41             maxOccurs="unbounded"/>
42     </xsd:sequence>
43     <xsd:attribute name="blank" type="xsd:boolean" use="optional" default="false"/>
44     <xsd:attribute name="calendarType" type="s:ST_CalendarType" use="optional" default="none"/>
45 </xsd:complexType>
46 <xsd:complexType name="CT_Filter">
47     <xsd:attribute name="val" type="s:ST_Xstring"/>
48 </xsd:complexType>
49 <xsd:complexType name="CT_CustomFilters">
50     <xsd:sequence>
51         <xsd:element name="customFilter" type="CT_CustomFilter" minOccurs="1" maxOccurs="2"/>
52     </xsd:sequence>
53     <xsd:attribute name="and" type="xsd:boolean" use="optional" default="false"/>
54 </xsd:complexType>
55 <xsd:complexType name="CT_CustomFilter">
56     <xsd:attribute name="operator" type="ST_FilterOperator" default="equal" use="optional"/>
57     <xsd:attribute name="val" type="s:ST_Xstring"/>
58 </xsd:complexType>
59 <xsd:complexType name="CT_Top10">
60     <xsd:attribute name="top" type="xsd:boolean" use="optional" default="true"/>
61     <xsd:attribute name="percent" type="xsd:boolean" use="optional" default="false"/>
62     <xsd:attribute name="val" type="xsd:double" use="required"/>
63     <xsd:attribute name="filterVal" type="xsd:double" use="optional"/>
64 </xsd:complexType>
65 <xsd:complexType name="CT_ColorFilter">
66     <xsd:attribute name="dxId" type="ST_DxfId" use="optional"/>
67     <xsd:attribute name="cellColor" type="xsd:boolean" use="optional" default="true"/>
68 </xsd:complexType>
69 <xsd:complexType name="CT_IconFilter">
70     <xsd:attribute name="iconSet" type="ST_IconSetType" use="required"/>
71     <xsd:attribute name="iconId" type="xsd:unsignedInt" use="optional"/>
72 </xsd:complexType>
73 <xsd:simpleType name="ST_FilterOperator">
74     <xsd:restriction base="xsd:string">
75         <xsd:enumeration value="equal"/>
76         <xsd:enumeration value="lessThan"/>
77         <xsd:enumeration value="lessThanOrEqual"/>
78         <xsd:enumeration value="notEqual"/>
79         <xsd:enumeration value="greaterThanOrEqual"/>
80         <xsd:enumeration value="greaterThan"/>
81     </xsd:restriction>
82 </xsd:simpleType>

```

```

83 <xsd:complexType name="CT_DynamicFilter">
84   <xsd:attribute name="type" type="ST_DynamicFilterType" use="required"/>
85   <xsd:attribute name="val" type="xsd:double" use="optional"/>
86   <xsd:attribute name="valIso" type="xsd:dateTime" use="optional"/>
87   <xsd:attribute name="maxVal" type="xsd:double" use="optional"/>
88   <xsd:attribute name="maxValIso" type="xsd:dateTime" use="optional"/>
89 </xsd:complexType>
90 <xsd:simpleType name="ST_DynamicFilterType">
91   <xsd:restriction base="xsd:string">
92     <xsd:enumeration value="null"/>
93     <xsd:enumeration value="aboveAverage"/>
94     <xsd:enumeration value="belowAverage"/>
95     <xsd:enumeration value="tomorrow"/>
96     <xsd:enumeration value="today"/>
97     <xsd:enumeration value="yesterday"/>
98     <xsd:enumeration value="nextWeek"/>
99     <xsd:enumeration value="thisWeek"/>
100    <xsd:enumeration value="lastWeek"/>
101    <xsd:enumeration value="nextMonth"/>
102    <xsd:enumeration value="thisMonth"/>
103    <xsd:enumeration value="lastMonth"/>
104    <xsd:enumeration value="nextQuarter"/>
105    <xsd:enumeration value="thisQuarter"/>
106    <xsd:enumeration value="lastQuarter"/>
107    <xsd:enumeration value="nextYear"/>
108    <xsd:enumeration value="thisYear"/>
109    <xsd:enumeration value="lastYear"/>
110    <xsd:enumeration value="yearToDate"/>
111    <xsd:enumeration value="Q1"/>
112    <xsd:enumeration value="Q2"/>
113    <xsd:enumeration value="Q3"/>
114    <xsd:enumeration value="Q4"/>
115    <xsd:enumeration value="M1"/>
116    <xsd:enumeration value="M2"/>
117    <xsd:enumeration value="M3"/>
118    <xsd:enumeration value="M4"/>
119    <xsd:enumeration value="M5"/>
120    <xsd:enumeration value="M6"/>
121    <xsd:enumeration value="M7"/>
122    <xsd:enumeration value="M8"/>
123    <xsd:enumeration value="M9"/>
124    <xsd:enumeration value="M10"/>
125    <xsd:enumeration value="M11"/>
126    <xsd:enumeration value="M12"/>
127   </xsd:restriction>
128 </xsd:simpleType>
129 <xsd:simpleType name="ST_IconSetType">
130   <xsd:restriction base="xsd:string">
131     <xsd:enumeration value="3Arrows"/>
132     <xsd:enumeration value="3ArrowsGray"/>
133     <xsd:enumeration value="3Flags"/>
134     <xsd:enumeration value="3TrafficLights1"/>
135     <xsd:enumeration value="3TrafficLights2"/>

```

```

136     <xsd:enumeration value="3Signs"/>
137     <xsd:enumeration value="3Symbols"/>
138     <xsd:enumeration value="3Symbols2"/>
139     <xsd:enumeration value="4Arrows"/>
140     <xsd:enumeration value="4ArrowsGray"/>
141     <xsd:enumeration value="4RedToBlack"/>
142     <xsd:enumeration value="4Rating"/>
143     <xsd:enumeration value="4TrafficLights"/>
144     <xsd:enumeration value="5Arrows"/>
145     <xsd:enumeration value="5ArrowsGray"/>
146     <xsd:enumeration value="5Rating"/>
147     <xsd:enumeration value="5Quarters"/>
148 </xsd:restriction>
149 </xsd:simpleType>
150 <xsd:complexType name="CT_SortState">
151     <xsd:sequence>
152         <xsd:element name="sortCondition" minOccurs="0" maxOccurs="64" type="CT_SortCondition"/>
153         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
154     </xsd:sequence>
155     <xsd:attribute name="columnSort" type="xsd:boolean" use="optional" default="false"/>
156     <xsd:attribute name="caseSensitive" type="xsd:boolean" use="optional" default="false"/>
157     <xsd:attribute name="sortMethod" type="ST_SortMethod" use="optional" default="none"/>
158     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
159 </xsd:complexType>
160 <xsd:complexType name="CT_SortCondition">
161     <xsd:attribute name="descending" type="xsd:boolean" use="optional" default="false"/>
162     <xsd:attribute name="sortBy" type="ST_SortBy" use="optional" default="value"/>
163     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
164     <xsd:attribute name="customList" type="s:ST_Xstring" use="optional"/>
165     <xsd:attribute name="dxfId" type="ST_DxfId" use="optional"/>
166     <xsd:attribute name="iconSet" type="ST_IconSetType" use="optional" default="3Arrows"/>
167     <xsd:attribute name="iconId" type="xsd:unsignedInt" use="optional"/>
168 </xsd:complexType>
169 <xsd:simpleType name="ST_SortBy">
170     <xsd:restriction base="xsd:string">
171         <xsd:enumeration value="value"/>
172         <xsd:enumeration value="cellColor"/>
173         <xsd:enumeration value="fontColor"/>
174         <xsd:enumeration value="icon"/>
175     </xsd:restriction>
176 </xsd:simpleType>
177 <xsd:simpleType name="ST_SortMethod">
178     <xsd:restriction base="xsd:string">
179         <xsd:enumeration value="stroke"/>
180         <xsd:enumeration value="pinYin"/>
181         <xsd:enumeration value="none"/>
182     </xsd:restriction>
183 </xsd:simpleType>
184 <xsd:complexType name="CT_DateGroupItem">
185     <xsd:attribute name="year" type="xsd:unsignedShort" use="required"/>
186     <xsd:attribute name="month" type="xsd:unsignedShort" use="optional"/>
187     <xsd:attribute name="day" type="xsd:unsignedShort" use="optional"/>
188     <xsd:attribute name="hour" type="xsd:unsignedShort" use="optional"/>

```

```

189     <xsd:attribute name="minute" type="xsd:unsignedShort" use="optional"/>
190     <xsd:attribute name="second" type="xsd:unsignedShort" use="optional"/>
191     <xsd:attribute name="dateTimeGrouping" type="ST_DateTimeGrouping" use="required"/>
192 </xsd:complexType>
193 <xsd:simpleType name="ST_DateTimeGrouping">
194     <xsd:restriction base="xsd:string">
195         <xsd:enumeration value="year"/>
196         <xsd:enumeration value="month"/>
197         <xsd:enumeration value="day"/>
198         <xsd:enumeration value="hour"/>
199         <xsd:enumeration value="minute"/>
200         <xsd:enumeration value="second"/>
201     </xsd:restriction>
202 </xsd:simpleType>
203 <xsd:simpleType name="ST_CellRef">
204     <xsd:restriction base="xsd:string"/>
205 </xsd:simpleType>
206 <xsd:simpleType name="ST_Ref">
207     <xsd:restriction base="xsd:string"/>
208 </xsd:simpleType>
209 <xsd:simpleType name="ST_RefA">
210     <xsd:restriction base="xsd:string"/>
211 </xsd:simpleType>
212 <xsd:simpleType name="ST_Sqref">
213     <xsd:list itemType="ST_Ref"/>
214 </xsd:simpleType>
215 <xsd:simpleType name="ST_Formula">
216     <xsd:restriction base="s:ST_Xstring"/>
217 </xsd:simpleType>
218 <xsd:simpleType name="ST_UnsignedIntHex">
219     <xsd:restriction base="xsd:hexBinary">
220         <xsd:length value="4"/>
221     </xsd:restriction>
222 </xsd:simpleType>
223 <xsd:simpleType name="ST_UnsignedShortHex">
224     <xsd:restriction base="xsd:hexBinary">
225         <xsd:length value="2"/>
226     </xsd:restriction>
227 </xsd:simpleType>
228 <xsd:complexType name="CT_XStringElement">
229     <xsd:attribute name="v" type="s:ST_Xstring" use="required"/>
230 </xsd:complexType>
231 <xsd:complexType name="CT_Extension">
232     <xsd:sequence>
233         <xsd:any processContents="lax"/>
234     </xsd:sequence>
235     <xsd:attribute name="uri" type="xsd:token"/>
236 </xsd:complexType>
237 <xsd:complexType name="CT_ObjectAnchor">
238     <xsd:sequence>
239         <xsd:element ref="xdr:from" minOccurs="1" maxOccurs="1"/>
240         <xsd:element ref="xdr:to" minOccurs="1" maxOccurs="1"/>
241     </xsd:sequence>

```

```

242     <xsd:attribute name="moveWithCells" type="xsd:boolean" use="optional" default="false"/>
243     <xsd:attribute name="sizeWithCells" type="xsd:boolean" use="optional" default="false"/>
244 </xsd:complexType>
245 <xsd:group name="EG_ExtensionList">
246     <xsd:sequence>
247         <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
248     </xsd:sequence>
249 </xsd:group>
250 <xsd:complexType name="CT_ExtensionList">
251     <xsd:sequence>
252         <xsd:group ref="EG_ExtensionList" minOccurs="0"/>
253     </xsd:sequence>
254 </xsd:complexType>
255 <xsd:element name="calcChain" type="CT_CalcChain"/>
256 <xsd:complexType name="CT_CalcChain">
257     <xsd:sequence>
258         <xsd:element name="c" type="CT_CalcCell" minOccurs="1" maxOccurs="unbounded"/>
259         <xsd:element name="extList" minOccurs="0" type="CT_ExtensionList"/>
260     </xsd:sequence>
261 </xsd:complexType>
262 <xsd:complexType name="CT_CalcCell">
263     <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
264     <xsd:attribute name="ref" type="ST_CellRef" use="optional"/>
265     <xsd:attribute name="i" type="xsd:int" use="optional" default="0"/>
266     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
267     <xsd:attribute name="l" type="xsd:boolean" use="optional" default="false"/>
268     <xsd:attribute name="t" type="xsd:boolean" use="optional" default="false"/>
269     <xsd:attribute name="a" type="xsd:boolean" use="optional" default="false"/>
270 </xsd:complexType>
271 <xsd:element name="comments" type="CT_Comments"/>
272 <xsd:complexType name="CT_Comments">
273     <xsd:sequence>
274         <xsd:element name="authors" type="CT_Authors" minOccurs="1" maxOccurs="1"/>
275         <xsd:element name="commentList" type="CT_CommentList" minOccurs="1" maxOccurs="1"/>
276         <xsd:element name="extList" minOccurs="0" type="CT_ExtensionList"/>
277     </xsd:sequence>
278 </xsd:complexType>
279 <xsd:complexType name="CT_Authors">
280     <xsd:sequence>
281         <xsd:element name="author" type="s:ST_Xstring" minOccurs="0" maxOccurs="unbounded"/>
282     </xsd:sequence>
283 </xsd:complexType>
284 <xsd:complexType name="CT_CommentList">
285     <xsd:sequence>
286         <xsd:element name="comment" type="CT_Comment" minOccurs="0" maxOccurs="unbounded"/>
287     </xsd:sequence>
288 </xsd:complexType>
289 <xsd:complexType name="CT_Comment">
290     <xsd:sequence>
291         <xsd:element name="text" type="CT_Rst" minOccurs="1" maxOccurs="1"/>
292         <xsd:element name="commentPr" type="CT_CommentPr" minOccurs="0" maxOccurs="1"/>
293     </xsd:sequence>
294     <xsd:attribute name="ref" type="ST_Ref" use="required"/>

```

```

295     <xsd:attribute name="authorId" type="xsd:unsignedInt" use="required"/>
296     <xsd:attribute name="guid" type="s:ST_Guid" use="optional"/>
297     <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="optional"/>
298 </xsd:complexType>
299 <xsd:complexType name="CT_CommentPr">
300     <xsd:sequence>
301         <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
302     </xsd:sequence>
303     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
304     <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
305     <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
306     <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
307     <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
308     <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
309     <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
310     <xsd:attribute name="textHAlign" type="ST_TextHAlign" use="optional" default="left"/>
311     <xsd:attribute name="textVAlign" type="ST_TextVAlign" use="optional" default="top"/>
312     <xsd:attribute name="lockText" type="xsd:boolean" use="optional" default="true"/>
313     <xsd:attribute name="justLastX" type="xsd:boolean" use="optional" default="false"/>
314     <xsd:attribute name="autoScale" type="xsd:boolean" use="optional" default="false"/>
315 </xsd:complexType>
316 <xsd:simpleType name="ST_TextHAlign">
317     <xsd:restriction base="xsd:string">
318         <xsd:enumeration value="left"/>
319         <xsd:enumeration value="center"/>
320         <xsd:enumeration value="right"/>
321         <xsd:enumeration value="justify"/>
322         <xsd:enumeration value="distributed"/>
323     </xsd:restriction>
324 </xsd:simpleType>
325 <xsd:simpleType name="ST_TextVAlign">
326     <xsd:restriction base="xsd:string">
327         <xsd:enumeration value="top"/>
328         <xsd:enumeration value="center"/>
329         <xsd:enumeration value="bottom"/>
330         <xsd:enumeration value="justify"/>
331         <xsd:enumeration value="distributed"/>
332     </xsd:restriction>
333 </xsd:simpleType>
334 <xsd:element name="MapInfo" type="CT_MapInfo"/>
335 <xsd:complexType name="CT_MapInfo">
336     <xsd:sequence>
337         <xsd:element name="Schema" type="CT_Schema" minOccurs="1" maxOccurs="unbounded"/>
338         <xsd:element name="Map" type="CT_Map" minOccurs="1" maxOccurs="unbounded"/>
339     </xsd:sequence>
340     <xsd:attribute name="SelectionNamespaces" type="xsd:string" use="required"/>
341 </xsd:complexType>
342 <xsd:complexType name="CT_Schema" mixed="true">
343     <xsd:sequence>
344         <xsd:any/>
345     </xsd:sequence>
346     <xsd:attribute name="ID" type="xsd:string" use="required"/>
347     <xsd:attribute name="SchemaRef" type="xsd:string" use="optional"/>

```



```

348     <xsd:attribute name="Namespace" type="xsd:string" use="optional"/>
349     <xsd:attribute name="SchemaLanguage" type="xsd:token" use="optional"/>
350 </xsd:complexType>
351 <xsd:complexType name="CT_Map">
352     <xsd:sequence>
353         <xsd:element name="DataBinding" type="CT_DataBinding" minOccurs="0" maxOccurs="1"/>
354     </xsd:sequence>
355     <xsd:attribute name="ID" type="xsd:unsignedInt" use="required"/>
356     <xsd:attribute name="Name" type="xsd:string" use="required"/>
357     <xsd:attribute name="RootElement" type="xsd:string" use="required"/>
358     <xsd:attribute name="SchemaID" type="xsd:string" use="required"/>
359     <xsd:attribute name="ShowImportExportValidationErrors" type="xsd:boolean" use="required"/>
360     <xsd:attribute name="AutoFit" type="xsd:boolean" use="required"/>
361     <xsd:attribute name="Append" type="xsd:boolean" use="required"/>
362     <xsd:attribute name="PreserveSortAFLayout" type="xsd:boolean" use="required"/>
363     <xsd:attribute name="PreserveFormat" type="xsd:boolean" use="required"/>
364 </xsd:complexType>
365 <xsd:complexType name="CT_DataBinding">
366     <xsd:sequence>
367         <xsd:any/>
368     </xsd:sequence>
369     <xsd:attribute name="DataBindingName" type="xsd:string" use="optional"/>
370     <xsd:attribute name="FileBinding" type="xsd:boolean" use="optional"/>
371     <xsd:attribute name="ConnectionID" type="xsd:unsignedInt" use="optional"/>
372     <xsd:attribute name="FileBindingName" type="xsd:string" use="optional"/>
373     <xsd:attribute name="DataBindingLoadMode" type="xsd:unsignedInt" use="required"/>
374 </xsd:complexType>
375 <xsd:element name="connections" type="CT_Connections"/>
376 <xsd:complexType name="CT_Connections">
377     <xsd:sequence>
378         <xsd:element name="connection" minOccurs="1" maxOccurs="unbounded" type="CT_Connection"/>
379     </xsd:sequence>
380 </xsd:complexType>
381 <xsd:complexType name="CT_Connection">
382     <xsd:sequence>
383         <xsd:element name="dbPr" minOccurs="0" maxOccurs="1" type="CT_DbPr"/>
384         <xsd:element name="olapPr" minOccurs="0" maxOccurs="1" type="CT_OlapPr"/>
385         <xsd:element name="webPr" minOccurs="0" maxOccurs="1" type="CT_WebPr"/>
386         <xsd:element name="textPr" minOccurs="0" maxOccurs="1" type="CT_TextPr"/>
387         <xsd:element name="parameters" minOccurs="0" maxOccurs="1" type="CT_Parameters"/>
388         <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
389     </xsd:sequence>
390     <xsd:attribute name="id" use="required" type="xsd:unsignedInt"/>
391     <xsd:attribute name="sourceFile" use="optional" type="s:ST_Xstring"/>
392     <xsd:attribute name="odcFile" use="optional" type="s:ST_Xstring"/>
393     <xsd:attribute name="keepAlive" use="optional" type="xsd:boolean" default="false"/>
394     <xsd:attribute name="interval" use="optional" type="xsd:unsignedInt" default="0"/>
395     <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
396     <xsd:attribute name="description" use="optional" type="s:ST_Xstring"/>
397     <xsd:attribute name="type" use="optional" type="xsd:unsignedInt"/>
398     <xsd:attribute name="reconnectionMethod" use="optional" type="xsd:unsignedInt" default="1"/>
399     <xsd:attribute name="refreshedVersion" use="required" type="xsd:unsignedByte"/>

```

```

400     <xsd:attribute name="minRefreshableVersion" use="optional" type="xsd:unsignedByte"
401       default="0"/>
402     <xsd:attribute name="savePassword" use="optional" type="xsd:boolean" default="false"/>
403     <xsd:attribute name="new" use="optional" type="xsd:boolean" default="false"/>
404     <xsd:attribute name="deleted" use="optional" type="xsd:boolean" default="false"/>
405     <xsd:attribute name="onlyUseConnectionFile" use="optional" type="xsd:boolean"
406       default="false"/>
407     <xsd:attribute name="background" use="optional" type="xsd:boolean" default="false"/>
408     <xsd:attribute name="refreshOnLoad" use="optional" type="xsd:boolean" default="false"/>
409     <xsd:attribute name="saveData" use="optional" type="xsd:boolean" default="false"/>
410     <xsd:attribute name="credentials" use="optional" type="ST_CredMethod" default="integrated"/>
411     <xsd:attribute name="singleSignOnId" use="optional" type="s:ST_Xstring"/>
412   </xsd:complexType>
413   <xsd:simpleType name="ST_CredMethod">
414     <xsd:restriction base="xsd:string">
415       <xsd:enumeration value="integrated"/>
416       <xsd:enumeration value="none"/>
417       <xsd:enumeration value="stored"/>
418       <xsd:enumeration value="prompt"/>
419     </xsd:restriction>
420   </xsd:simpleType>
421   <xsd:complexType name="CT_DbPr">
422     <xsd:attribute name="connection" use="required" type="s:ST_Xstring"/>
423     <xsd:attribute name="command" use="optional" type="s:ST_Xstring"/>
424     <xsd:attribute name="serverCommand" use="optional" type="s:ST_Xstring"/>
425     <xsd:attribute name="commandType" use="optional" type="xsd:unsignedInt" default="2"/>
426   </xsd:complexType>
427   <xsd:complexType name="CT_OlapPr">
428     <xsd:attribute name="local" use="optional" type="xsd:boolean" default="false"/>
429     <xsd:attribute name="localConnection" use="optional" type="s:ST_Xstring"/>
430     <xsd:attribute name="localRefresh" use="optional" type="xsd:boolean" default="true"/>
431     <xsd:attribute name="sendLocale" use="optional" type="xsd:boolean" default="false"/>
432     <xsd:attribute name="rowDrillCount" use="optional" type="xsd:unsignedInt"/>
433     <xsd:attribute name="serverFill" use="optional" type="xsd:boolean" default="true"/>
434     <xsd:attribute name="serverNumberFormat" use="optional" type="xsd:boolean" default="true"/>
435     <xsd:attribute name="serverFont" use="optional" type="xsd:boolean" default="true"/>
436     <xsd:attribute name="serverFontColor" use="optional" type="xsd:boolean" default="true"/>
437   </xsd:complexType>
438   <xsd:complexType name="CT_WebPr">
439     <xsd:sequence>
440       <xsd:element name="tables" minOccurs="0" maxOccurs="1" type="CT_Tables"/>
441     </xsd:sequence>
442     <xsd:attribute name="xml" use="optional" type="xsd:boolean" default="false"/>
443     <xsd:attribute name="sourceData" use="optional" type="xsd:boolean" default="false"/>
444     <xsd:attribute name="parsePre" use="optional" type="xsd:boolean" default="false"/>
445     <xsd:attribute name="consecutive" use="optional" type="xsd:boolean" default="false"/>
446     <xsd:attribute name="firstRow" use="optional" type="xsd:boolean" default="false"/>
447     <xsd:attribute name="xl97" use="optional" type="xsd:boolean" default="false"/>
448     <xsd:attribute name="textDates" use="optional" type="xsd:boolean" default="false"/>
449     <xsd:attribute name="xl2000" use="optional" type="xsd:boolean" default="false"/>
450     <xsd:attribute name="url" use="optional" type="s:ST_Xstring"/>
451     <xsd:attribute name="post" use="optional" type="s:ST_Xstring"/>
452     <xsd:attribute name="htmlTables" use="optional" type="xsd:boolean" default="false"/>

```

```

453     <xsd:attribute name="htmlFormat" use="optional" type="ST_HtmlFmt" default="none"/>
454     <xsd:attribute name="editPage" use="optional" type="s:ST_Xstring"/>
455 </xsd:complexType>
456 <xsd:simpleType name="ST_HtmlFmt">
457     <xsd:restriction base="xsd:string">
458         <xsd:enumeration value="none"/>
459         <xsd:enumeration value="rtf"/>
460         <xsd:enumeration value="all"/>
461     </xsd:restriction>
462 </xsd:simpleType>
463 <xsd:complexType name="CT_Parameters">
464     <xsd:sequence>
465         <xsd:element name="parameter" minOccurs="1" maxOccurs="unbounded" type="CT_Parameter"/>
466     </xsd:sequence>
467     <xsd:attribute name="count" use="optional" type="xsd:unsignedInt"/>
468 </xsd:complexType>
469 <xsd:complexType name="CT_Parameter">
470     <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
471     <xsd:attribute name="sqlType" use="optional" type="xsd:int" default="0"/>
472     <xsd:attribute name="parameterType" use="optional" type="ST_ParameterType" default="prompt"/>
473     <xsd:attribute name="refreshOnChange" use="optional" type="xsd:boolean" default="false"/>
474     <xsd:attribute name="prompt" use="optional" type="s:ST_Xstring"/>
475     <xsd:attribute name="boolean" use="optional" type="xsd:boolean"/>
476     <xsd:attribute name="double" use="optional" type="xsd:double"/>
477     <xsd:attribute name="integer" use="optional" type="xsd:int"/>
478     <xsd:attribute name="string" use="optional" type="s:ST_Xstring"/>
479     <xsd:attribute name="cell" use="optional" type="s:ST_Xstring"/>
480 </xsd:complexType>
481 <xsd:simpleType name="ST_ParameterType">
482     <xsd:restriction base="xsd:string">
483         <xsd:enumeration value="prompt"/>
484         <xsd:enumeration value="value"/>
485         <xsd:enumeration value="cell"/>
486     </xsd:restriction>
487 </xsd:simpleType>
488 <xsd:complexType name="CT_Tables">
489     <xsd:choice minOccurs="1" maxOccurs="unbounded">
490         <xsd:element name="m" type="CT_TableMissing"/>
491         <xsd:element name="s" type="CT_XStringElement"/>
492         <xsd:element name="x" type="CT_Index"/>
493     </xsd:choice>
494     <xsd:attribute name="count" use="optional" type="xsd:unsignedInt"/>
495 </xsd:complexType>
496 <xsd:complexType name="CT_TableMissing"/>
497 <xsd:complexType name="CT_TextPr">
498     <xsd:sequence>
499         <xsd:element name="textFields" minOccurs="0" maxOccurs="1" type="CT_TextFields"/>
500     </xsd:sequence>
501     <xsd:attribute name="prompt" use="optional" type="xsd:boolean" default="true"/>
502     <xsd:attribute name="fileType" use="optional" type="ST_FileType" default="win"/>
503     <xsd:attribute name="codePage" use="optional" type="xsd:unsignedInt" default="1252"/>
504     <xsd:attribute name="characterSet" use="optional" type="xsd:string"/>
505     <xsd:attribute name="firstRow" use="optional" type="xsd:unsignedInt" default="1"/>

```

```

506 <xsd:attribute name="sourceFile" use="optional" type="s:ST Xstring" default=""/>
507 <xsd:attribute name="delimited" use="optional" type="xsd:boolean" default="true"/>
508 <xsd:attribute name="decimal" use="optional" type="s:ST Xstring" default="."/>
509 <xsd:attribute name="thousands" use="optional" type="s:ST Xstring" default=","/>
510 <xsd:attribute name="tab" use="optional" type="xsd:boolean" default="true"/>
511 <xsd:attribute name="space" use="optional" type="xsd:boolean" default="false"/>
512 <xsd:attribute name="comma" use="optional" type="xsd:boolean" default="false"/>
513 <xsd:attribute name="semicolon" use="optional" type="xsd:boolean" default="false"/>
514 <xsd:attribute name="consecutive" use="optional" type="xsd:boolean" default="false"/>
515 <xsd:attribute name="qualifier" use="optional" type="ST Qualifier" default="doubleQuote"/>
516 <xsd:attribute name="delimiter" use="optional" type="s:ST Xstring"/>
517 </xsd:complexType>
518 <xsd:simpleType name="ST_FileType">
519   <xsd:restriction base="xsd:string">
520     <xsd:enumeration value="mac"/>
521     <xsd:enumeration value="win"/>
522     <xsd:enumeration value="dos"/>
523     <xsd:enumeration value="lin"/>
524     <xsd:enumeration value="other"/>
525   </xsd:restriction>
526 </xsd:simpleType>
527 <xsd:simpleType name="ST_Qualifier">
528   <xsd:restriction base="xsd:string">
529     <xsd:enumeration value="doubleQuote"/>
530     <xsd:enumeration value="singleQuote"/>
531     <xsd:enumeration value="none"/>
532   </xsd:restriction>
533 </xsd:simpleType>
534 <xsd:complexType name="CT_TextFields">
535   <xsd:sequence>
536     <xsd:element name="textField" minOccurs="1" maxOccurs="unbounded" type="CT_TextField"/>
537   </xsd:sequence>
538   <xsd:attribute name="count" use="optional" type="xsd:unsignedInt" default="1"/>
539 </xsd:complexType>
540 <xsd:complexType name="CT_TextField">
541   <xsd:attribute name="type" use="optional" type="ST_ExternalConnectionType" default="general"/>
542   <xsd:attribute name="position" use="optional" type="xsd:unsignedInt" default="0"/>
543 </xsd:complexType>
544 <xsd:simpleType name="ST_ExternalConnectionType">
545   <xsd:restriction base="xsd:string">
546     <xsd:enumeration value="general"/>
547     <xsd:enumeration value="text"/>
548     <xsd:enumeration value="MDY"/>
549     <xsd:enumeration value="DMY"/>
550     <xsd:enumeration value="YMD"/>
551     <xsd:enumeration value="MYD"/>
552     <xsd:enumeration value="DYM"/>
553     <xsd:enumeration value="YDM"/>
554     <xsd:enumeration value="skip"/>
555     <xsd:enumeration value="EMD"/>
556   </xsd:restriction>
557 </xsd:simpleType>
558 <xsd:element name="pivotCacheDefinition" type="CT_PivotCacheDefinition"/>

```

```

559 <xsd:element name="pivotCacheRecords" type="CT_PivotCacheRecords"/>
560 <xsd:element name="pivotTableDefinition" type="CT_pivotTableDefinition"/>
561 <xsd:complexType name="CT_PivotCacheDefinition">
562     <xsd:sequence>
563         <xsd:element name="cacheSource" type="CT_CacheSource" minOccurs="1" maxOccurs="1"/>
564         <xsd:element name="cacheFields" type="CT_CacheFields" minOccurs="1" maxOccurs="1"/>
565         <xsd:element name="cacheHierarchies" minOccurs="0" type="CT_CacheHierarchies"/>
566         <xsd:element name="kpis" minOccurs="0" type="CT_PCDKPIs"/>
567         <xsd:element name="tupleCache" minOccurs="0" type="CT_TupleCache"/>
568         <xsd:element name="calculatedItems" minOccurs="0" type="CT_CalculatedItems"/>
569         <xsd:element name="calculatedMembers" type="CT_CalculatedMembers" minOccurs="0"/>
570         <xsd:element name="dimensions" type="CT_Dimensions" minOccurs="0"/>
571         <xsd:element name="measureGroups" type="CT_MeasureGroups" minOccurs="0"/>
572         <xsd:element name="maps" type="CT_MeasureDimensionMaps" minOccurs="0"/>
573         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
574     </xsd:sequence>
575     <xsd:attribute ref="r:id" use="optional"/>
576     <xsd:attribute name="invalid" type="xsd:boolean" use="optional" default="false"/>
577     <xsd:attribute name="saveData" type="xsd:boolean" use="optional" default="true"/>
578     <xsd:attribute name="refreshOnLoad" type="xsd:boolean" use="optional" default="false"/>
579     <xsd:attribute name="optimizeMemory" type="xsd:boolean" use="optional" default="false"/>
580     <xsd:attribute name="enableRefresh" type="xsd:boolean" use="optional" default="true"/>
581     <xsd:attribute name="refreshedBy" type="s:ST_Xstring" use="optional"/>
582     <xsd:attribute name="refreshedDate" type="xsd:double" use="optional"/>
583     <xsd:attribute name="refreshedDateIso" type="xsd:dateTime" use="optional"/>
584     <xsd:attribute name="backgroundQuery" type="xsd:boolean" default="false"/>
585     <xsd:attribute name="missingItemsLimit" type="xsd:unsignedInt" use="optional"/>
586     <xsd:attribute name="createdVersion" type="xsd:unsignedByte" use="optional" default="0"/>
587     <xsd:attribute name="refreshedVersion" type="xsd:unsignedByte" use="optional" default="0"/>
588     <xsd:attribute name="minRefreshableVersion" type="xsd:unsignedByte" use="optional"
589         default="0"/>
590     <xsd:attribute name="recordCount" type="xsd:unsignedInt" use="optional"/>
591     <xsd:attribute name="upgradeOnRefresh" type="xsd:boolean" use="optional" default="false"/>
592     <xsd:attribute name="tupleCache" type="xsd:boolean" use="optional" default="false"/>
593     <xsd:attribute name="supportSubquery" type="xsd:boolean" use="optional" default="false"/>
594     <xsd:attribute name="supportAdvancedDrill" type="xsd:boolean" use="optional" default="false"/>
595 </xsd:complexType>
596 <xsd:complexType name="CT_CacheFields">
597     <xsd:sequence>
598         <xsd:element name="cacheField" type="CT_CacheField" minOccurs="0" maxOccurs="unbounded"/>
599     </xsd:sequence>
600     <xsd:attribute name="count" type="xsd:unsignedInt"/>
601 </xsd:complexType>
602 <xsd:complexType name="CT_CacheField">
603     <xsd:sequence>
604         <xsd:element name="sharedItems" type="CT_SharedItems" minOccurs="0" maxOccurs="1"/>
605         <xsd:element name="fieldGroup" minOccurs="0" type="CT_FieldGroup"/>
606         <xsd:element name="mpMap" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
607         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
608     </xsd:sequence>
609     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
610     <xsd:attribute name="caption" type="s:ST_Xstring" use="optional"/>
611     <xsd:attribute name="propertyName" type="s:ST_Xstring" use="optional"/>

```

```

612     <xsd:attribute name="serverField" type="xsd:boolean" use="optional" default="false"/>
613     <xsd:attribute name="uniqueList" type="xsd:boolean" use="optional" default="true"/>
614     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
615     <xsd:attribute name="formula" type="s:ST_Xstring" use="optional"/>
616     <xsd:attribute name="sqlType" type="xsd:int" use="optional" default="0"/>
617     <xsd:attribute name="hierarchy" type="xsd:int" use="optional" default="0"/>
618     <xsd:attribute name="level" type="xsd:unsignedInt" use="optional" default="0"/>
619     <xsd:attribute name="databaseField" type="xsd:boolean" default="true"/>
620     <xsd:attribute name="mappingCount" type="xsd:unsignedInt" use="optional"/>
621     <xsd:attribute name="memberPropertyField" type="xsd:boolean" use="optional" default="false"/>
622 </xsd:complexType>
623 <xsd:complexType name="CT_CacheSource">
624     <xsd:choice minOccurs="0" maxOccurs="1">
625         <xsd:element name="worksheetSource" type="CT_WorksheetSource" minOccurs="1"
626             maxOccurs="1"/>
627         <xsd:element name="consolidation" type="CT_Consolidation" minOccurs="1" maxOccurs="1"/>
628         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0"/>
629     </xsd:choice>
630     <xsd:attribute name="type" type="ST_SourceType" use="required"/>
631     <xsd:attribute name="connectionId" type="xsd:unsignedInt" default="0" use="optional"/>
632 </xsd:complexType>
633 <xsd:simpleType name="ST_SourceType">
634     <xsd:restriction base="xsd:string">
635         <xsd:enumeration value="worksheet"/>
636         <xsd:enumeration value="external"/>
637         <xsd:enumeration value="consolidation"/>
638         <xsd:enumeration value="scenario"/>
639     </xsd:restriction>
640 </xsd:simpleType>
641 <xsd:complexType name="CT_WorksheetSource">
642     <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
643     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
644     <xsd:attribute name="sheet" type="s:ST_Xstring" use="optional"/>
645     <xsd:attribute ref="r:id" use="optional"/>
646 </xsd:complexType>
647 <xsd:complexType name="CT_Consolidation">
648     <xsd:sequence>
649         <xsd:element name="pages" type="CT_Pages" minOccurs="0" maxOccurs="1"/>
650         <xsd:element name="rangeSets" type="CT_RangeSets" minOccurs="1" maxOccurs="1"/>
651     </xsd:sequence>
652     <xsd:attribute name="autoPage" type="xsd:boolean" default="true" use="optional"/>
653 </xsd:complexType>
654 <xsd:complexType name="CT_Pages">
655     <xsd:sequence>
656         <xsd:element name="page" type="CT_PCDSCPage" minOccurs="1" maxOccurs="4"/>
657     </xsd:sequence>
658     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
659 </xsd:complexType>
660 <xsd:complexType name="CT_PCDSCPage">
661     <xsd:sequence>
662         <xsd:element name="pageItem" type="CT_PageItem" minOccurs="0" maxOccurs="unbounded"/>
663     </xsd:sequence>
664     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>

```

```

665 </xsd:complexType>
666 <xsd:complexType name="CT_PageItem">
667   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
668 </xsd:complexType>
669 <xsd:complexType name="CT_RangeSets">
670   <xsd:sequence>
671     <xsd:element name="rangeSet" type="CT_RangeSet" minOccurs="1" maxOccurs="unbounded"/>
672   </xsd:sequence>
673   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
674 </xsd:complexType>
675 <xsd:complexType name="CT_RangeSet">
676   <xsd:attribute name="i1" type="xsd:unsignedInt" use="optional"/>
677   <xsd:attribute name="i2" type="xsd:unsignedInt" use="optional"/>
678   <xsd:attribute name="i3" type="xsd:unsignedInt" use="optional"/>
679   <xsd:attribute name="i4" type="xsd:unsignedInt" use="optional"/>
680   <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
681   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
682   <xsd:attribute name="sheet" type="s:ST_Xstring" use="optional"/>
683   <xsd:attribute ref="r:id" use="optional"/>
684 </xsd:complexType>
685 <xsd:complexType name="CT_SharedItems">
686   <xsd:choice minOccurs="0" maxOccurs="unbounded">
687     <xsd:element name="m" type="CT_Missing" minOccurs="1" maxOccurs="1"/>
688     <xsd:element name="n" type="CT_Number" minOccurs="1" maxOccurs="1"/>
689     <xsd:element name="b" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
690     <xsd:element name="e" type="CT_Error" minOccurs="1" maxOccurs="1"/>
691     <xsd:element name="s" type="CT_String" minOccurs="1" maxOccurs="1"/>
692     <xsd:element name="d" type="CT_DateTime" minOccurs="1" maxOccurs="1"/>
693   </xsd:choice>
694   <xsd:attribute name="containsSemiMixedTypes" type="xsd:boolean" use="optional"
695     default="true"/>
696   <xsd:attribute name="containsNonDate" type="xsd:boolean" use="optional" default="true"/>
697   <xsd:attribute name="containsDate" type="xsd:boolean" use="optional" default="false"/>
698   <xsd:attribute name="containsString" type="xsd:boolean" use="optional" default="true"/>
699   <xsd:attribute name="containsBlank" type="xsd:boolean" use="optional" default="false"/>
700   <xsd:attribute name="containsMixedTypes" type="xsd:boolean" use="optional" default="false"/>
701   <xsd:attribute name="containsNumber" type="xsd:boolean" use="optional" default="false"/>
702   <xsd:attribute name="containsInteger" type="xsd:boolean" use="optional" default="false"/>
703   <xsd:attribute name="minValue" type="xsd:double" use="optional"/>
704   <xsd:attribute name="maxValue" type="xsd:double" use="optional"/>
705   <xsd:attribute name="minDate" type="xsd:dateTime" use="optional"/>
706   <xsd:attribute name="maxDate" type="xsd:dateTime" use="optional"/>
707   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
708   <xsd:attribute name="longText" type="xsd:boolean" use="optional" default="false"/>
709 </xsd:complexType>
710 <xsd:complexType name="CT_Missing">
711   <xsd:sequence>
712     <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
713     <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
714   </xsd:sequence>
715   <xsd:attribute name="u" type="xsd:boolean"/>
716   <xsd:attribute name="f" type="xsd:boolean"/>
717   <xsd:attribute name="c" type="s:ST_Xstring"/>

```

```

718     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
719     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
720     <xsd:attribute name="bc" type="ST UnsignedIntHex" use="optional"/>
721     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
722     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
723     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
724     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
725     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
726 </xsd:complexType>
727 <xsd:complexType name="CT_Number">
728     <xsd:sequence>
729         <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
730         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
731     </xsd:sequence>
732     <xsd:attribute name="v" use="required" type="xsd:double"/>
733     <xsd:attribute name="u" type="xsd:boolean"/>
734     <xsd:attribute name="f" type="xsd:boolean"/>
735     <xsd:attribute name="c" type="s:ST Xstring"/>
736     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
737     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
738     <xsd:attribute name="bc" type="ST UnsignedIntHex" use="optional"/>
739     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
740     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
741     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
742     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
743     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
744 </xsd:complexType>
745 <xsd:complexType name="CT_Boolean">
746     <xsd:sequence>
747         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
748     </xsd:sequence>
749     <xsd:attribute name="v" use="required" type="xsd:boolean"/>
750     <xsd:attribute name="u" type="xsd:boolean"/>
751     <xsd:attribute name="f" type="xsd:boolean"/>
752     <xsd:attribute name="c" type="s:ST Xstring"/>
753     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
754 </xsd:complexType>
755 <xsd:complexType name="CT_Error">
756     <xsd:sequence>
757         <xsd:element name="tpls" minOccurs="0" type="CT_Tuples"/>
758         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
759     </xsd:sequence>
760     <xsd:attribute name="v" use="required" type="s:ST Xstring"/>
761     <xsd:attribute name="u" type="xsd:boolean"/>
762     <xsd:attribute name="f" type="xsd:boolean"/>
763     <xsd:attribute name="c" type="s:ST Xstring"/>
764     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
765     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
766     <xsd:attribute name="bc" type="ST UnsignedIntHex" use="optional"/>
767     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
768     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
769     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
770     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>

```



```

771     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
772 </xsd:complexType>
773 <xsd:complexType name="CT_String">
774     <xsd:sequence>
775         <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
776         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
777     </xsd:sequence>
778     <xsd:attribute name="v" use="required" type="s:ST_Xstring"/>
779     <xsd:attribute name="u" type="xsd:boolean"/>
780     <xsd:attribute name="f" type="xsd:boolean"/>
781     <xsd:attribute name="c" type="s:ST_Xstring"/>
782     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
783     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
784     <xsd:attribute name="bc" type="ST_UnsignedIntHex" use="optional"/>
785     <xsd:attribute name="fc" type="ST_UnsignedIntHex" use="optional"/>
786     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
787     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
788     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
789     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
790 </xsd:complexType>
791 <xsd:complexType name="CT_DateTime">
792     <xsd:sequence>
793         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
794     </xsd:sequence>
795     <xsd:attribute name="v" use="required" type="xsd:dateTime"/>
796     <xsd:attribute name="u" type="xsd:boolean"/>
797     <xsd:attribute name="f" type="xsd:boolean"/>
798     <xsd:attribute name="c" type="s:ST_Xstring"/>
799     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
800 </xsd:complexType>
801 <xsd:complexType name="CT_FieldGroup">
802     <xsd:sequence>
803         <xsd:element name="rangePr" minOccurs="0" type="CT_RangePr"/>
804         <xsd:element name="discretePr" minOccurs="0" type="CT_DiscretePr"/>
805         <xsd:element name="groupItems" minOccurs="0" type="CT_GroupItems"/>
806     </xsd:sequence>
807     <xsd:attribute name="par" type="xsd:unsignedInt" use="optional"/>
808     <xsd:attribute name="base" type="xsd:unsignedInt" use="optional"/>
809 </xsd:complexType>
810 <xsd:complexType name="CT_RangePr">
811     <xsd:attribute name="autoStart" type="xsd:boolean" default="true"/>
812     <xsd:attribute name="autoEnd" type="xsd:boolean" default="true"/>
813     <xsd:attribute name="groupBy" type="ST_GroupBy" default="range"/>
814     <xsd:attribute name="startNum" type="xsd:double"/>
815     <xsd:attribute name="endNum" type="xsd:double"/>
816     <xsd:attribute name="startDate" type="xsd:dateTime"/>
817     <xsd:attribute name="endDate" type="xsd:dateTime"/>
818     <xsd:attribute name="groupInterval" type="xsd:double" default="1"/>
819 </xsd:complexType>
820 <xsd:simpleType name="ST_GroupBy">
821     <xsd:restriction base="xsd:string">
822         <xsd:enumeration value="range"/>
823         <xsd:enumeration value="seconds"/>

```

```

824         <xsd:enumeration value="minutes"/>
825         <xsd:enumeration value="hours"/>
826         <xsd:enumeration value="days"/>
827         <xsd:enumeration value="months"/>
828         <xsd:enumeration value="quarters"/>
829         <xsd:enumeration value="years"/>
830     </xsd:restriction>
831 </xsd:simpleType>
832 <xsd:complexType name="CT_DiscretePr">
833     <xsd:sequence>
834         <xsd:element name="x" maxOccurs="unbounded" type="CT_Index"/>
835     </xsd:sequence>
836     <xsd:attribute name="count" type="xsd:unsignedInt"/>
837 </xsd:complexType>
838 <xsd:complexType name="CT_GroupItems">
839     <xsd:choice maxOccurs="unbounded">
840         <xsd:element name="m" type="CT_Missing"/>
841         <xsd:element name="n" type="CT_Number"/>
842         <xsd:element name="b" type="CT_Boolean"/>
843         <xsd:element name="e" type="CT_Error"/>
844         <xsd:element name="s" type="CT_String"/>
845         <xsd:element name="d" type="CT_DateTime"/>
846     </xsd:choice>
847     <xsd:attribute name="count" type="xsd:unsignedInt"/>
848 </xsd:complexType>
849 <xsd:complexType name="CT_PivotCacheRecords">
850     <xsd:sequence>
851         <xsd:element name="r" minOccurs="0" maxOccurs="unbounded" type="CT_Record"/>
852         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
853     </xsd:sequence>
854     <xsd:attribute name="count" type="xsd:unsignedInt"/>
855 </xsd:complexType>
856 <xsd:complexType name="CT_Record">
857     <xsd:choice maxOccurs="unbounded">
858         <xsd:element name="m" type="CT_Missing"/>
859         <xsd:element name="n" type="CT_Number"/>
860         <xsd:element name="b" type="CT_Boolean"/>
861         <xsd:element name="e" type="CT_Error"/>
862         <xsd:element name="s" type="CT_String"/>
863         <xsd:element name="d" type="CT_DateTime"/>
864         <xsd:element name="x" type="CT_Index"/>
865     </xsd:choice>
866 </xsd:complexType>
867 <xsd:complexType name="CT_PCDKPIs">
868     <xsd:sequence>
869         <xsd:element name="kpi" minOccurs="0" maxOccurs="unbounded" type="CT_PCDKPI"/>
870     </xsd:sequence>
871     <xsd:attribute name="count" type="xsd:unsignedInt"/>
872 </xsd:complexType>
873 <xsd:complexType name="CT_PCDKPI">
874     <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
875     <xsd:attribute name="caption" use="optional" type="s:ST_Xstring"/>
876     <xsd:attribute name="displayFolder" type="s:ST_Xstring"/>

```

```

877     <xsd:attribute name="measureGroup" type="s:ST Xstring"/>
878     <xsd:attribute name="parent" type="s:ST Xstring"/>
879     <xsd:attribute name="value" use="required" type="s:ST Xstring"/>
880     <xsd:attribute name="goal" type="s:ST Xstring"/>
881     <xsd:attribute name="status" type="s:ST Xstring"/>
882     <xsd:attribute name="trend" type="s:ST Xstring"/>
883     <xsd:attribute name="weight" type="s:ST Xstring"/>
884     <xsd:attribute name="time" type="s:ST Xstring"/>
885 </xsd:complexType>
886 <xsd:complexType name="CT_CacheHierarchies">
887     <xsd:sequence>
888         <xsd:element name="cacheHierarchy" minOccurs="0" maxOccurs="unbounded"
889             type="CT_CacheHierarchy"/>
890     </xsd:sequence>
891     <xsd:attribute name="count" type="xsd:unsignedInt"/>
892 </xsd:complexType>
893 <xsd:complexType name="CT_CacheHierarchy">
894     <xsd:sequence>
895         <xsd:element name="fieldsUsage" minOccurs="0" type="CT_FieldsUsage"/>
896         <xsd:element name="groupLevels" minOccurs="0" type="CT_GroupLevels"/>
897         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
898     </xsd:sequence>
899     <xsd:attribute name="uniqueName" use="required" type="s:ST Xstring"/>
900     <xsd:attribute name="caption" use="optional" type="s:ST Xstring"/>
901     <xsd:attribute name="measure" type="xsd:boolean" default="false"/>
902     <xsd:attribute name="set" type="xsd:boolean" default="false"/>
903     <xsd:attribute name="parentSet" type="xsd:unsignedInt" use="optional"/>
904     <xsd:attribute name="iconSet" type="xsd:int" default="0"/>
905     <xsd:attribute name="attribute" type="xsd:boolean" default="false"/>
906     <xsd:attribute name="time" type="xsd:boolean" default="false"/>
907     <xsd:attribute name="keyAttribute" type="xsd:boolean" default="false"/>
908     <xsd:attribute name="defaultMemberUniqueName" type="s:ST Xstring"/>
909     <xsd:attribute name="allUniqueName" type="s:ST Xstring"/>
910     <xsd:attribute name="allCaption" type="s:ST Xstring"/>
911     <xsd:attribute name="dimensionUniqueName" type="s:ST Xstring"/>
912     <xsd:attribute name="displayFolder" type="s:ST Xstring"/>
913     <xsd:attribute name="measureGroup" type="s:ST Xstring"/>
914     <xsd:attribute name="measures" type="xsd:boolean" default="false"/>
915     <xsd:attribute name="count" use="required" type="xsd:unsignedInt"/>
916     <xsd:attribute name="oneField" type="xsd:boolean" default="false"/>
917     <xsd:attribute name="memberValueDatatype" use="optional" type="xsd:unsignedShort"/>
918     <xsd:attribute name="unbalanced" use="optional" type="xsd:boolean"/>
919     <xsd:attribute name="unbalancedGroup" use="optional" type="xsd:boolean"/>
920     <xsd:attribute name="hidden" type="xsd:boolean" default="false"/>
921 </xsd:complexType>
922 <xsd:complexType name="CT_FieldsUsage">
923     <xsd:sequence>
924         <xsd:element name="fieldUsage" minOccurs="0" maxOccurs="unbounded" type="CT_FieldUsage"/>
925     </xsd:sequence>
926     <xsd:attribute name="count" type="xsd:unsignedInt"/>
927 </xsd:complexType>
928 <xsd:complexType name="CT_FieldUsage">
929     <xsd:attribute name="x" use="required" type="xsd:int"/>

```

```

930 </xsd:complexType>
931 <xsd:complexType name="CT_GroupLevels">
932   <xsd:sequence>
933     <xsd:element name="groupLevel" maxOccurs="unbounded" type="CT_GroupLevel1"/>
934   </xsd:sequence>
935   <xsd:attribute name="count" type="xsd:unsignedInt"/>
936 </xsd:complexType>
937 <xsd:complexType name="CT_GroupLevel">
938   <xsd:sequence>
939     <xsd:element name="groups" minOccurs="0" type="CT_Groups"/>
940     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
941   </xsd:sequence>
942   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
943   <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
944   <xsd:attribute name="user" type="xsd:boolean" default="false"/>
945   <xsd:attribute name="customRollUp" type="xsd:boolean" default="false"/>
946 </xsd:complexType>
947 <xsd:complexType name="CT_Groups">
948   <xsd:sequence>
949     <xsd:element name="group" maxOccurs="unbounded" type="CT_LevelGroup"/>
950   </xsd:sequence>
951   <xsd:attribute name="count" type="xsd:unsignedInt"/>
952 </xsd:complexType>
953 <xsd:complexType name="CT_LevelGroup">
954   <xsd:sequence>
955     <xsd:element name="groupMembers" type="CT_GroupMembers"/>
956   </xsd:sequence>
957   <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
958   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
959   <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
960   <xsd:attribute name="uniqueParent" type="s:ST_Xstring"/>
961   <xsd:attribute name="id" type="xsd:int"/>
962 </xsd:complexType>
963 <xsd:complexType name="CT_GroupMembers">
964   <xsd:sequence>
965     <xsd:element name="groupMember" maxOccurs="unbounded" type="CT_GroupMember"/>
966   </xsd:sequence>
967   <xsd:attribute name="count" type="xsd:unsignedInt"/>
968 </xsd:complexType>
969 <xsd:complexType name="CT_GroupMember">
970   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
971   <xsd:attribute name="group" type="xsd:boolean" default="false"/>
972 </xsd:complexType>
973 <xsd:complexType name="CT_TupleCache">
974   <xsd:sequence>
975     <xsd:element name="entries" minOccurs="0" type="CT_PCDSDTCEntries"/>
976     <xsd:element name="sets" minOccurs="0" type="CT_Sets"/>
977     <xsd:element name="queryCache" minOccurs="0" type="CT_QueryCache"/>
978     <xsd:element name="serverFormats" minOccurs="0" maxOccurs="1" type="CT_ServerFormats"/>
979     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
980   </xsd:sequence>
981 </xsd:complexType>
982 <xsd:complexType name="CT_ServerFormat">

```

```

983     <xsd:attribute name="culture" use="optional" type="s:ST Xstring"/>
984     <xsd:attribute name="format" use="optional" type="s:ST Xstring"/>
985 </xsd:complexType>
986 <xsd:complexType name="CT_ServerFormats">
987     <xsd:sequence>
988         <xsd:element name="serverFormat" type="CT_ServerFormat" minOccurs="0"
989             maxOccurs="unbounded"/>
990     </xsd:sequence>
991     <xsd:attribute name="count" type="xsd:unsignedInt"/>
992 </xsd:complexType>
993 <xsd:complexType name="CT_PCSDTCEntries">
994     <xsd:choice maxOccurs="unbounded">
995         <xsd:element name="m" type="CT_Missing"/>
996         <xsd:element name="n" type="CT_Number"/>
997         <xsd:element name="e" type="CT_Error"/>
998         <xsd:element name="s" type="CT_String"/>
999     </xsd:choice>
1000     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1001 </xsd:complexType>
1002 <xsd:complexType name="CT_Tuples">
1003     <xsd:sequence>
1004         <xsd:element name="tpl" type="CT_Tuple" minOccurs="1" maxOccurs="unbounded"/>
1005     </xsd:sequence>
1006     <xsd:attribute name="c" type="xsd:unsignedInt" use="optional"/>
1007 </xsd:complexType>
1008 <xsd:complexType name="CT_Tuple">
1009     <xsd:attribute name="fld" type="xsd:unsignedInt"/>
1010     <xsd:attribute name="hier" type="xsd:unsignedInt"/>
1011     <xsd:attribute name="item" type="xsd:unsignedInt" use="required"/>
1012 </xsd:complexType>
1013 <xsd:complexType name="CT_Sets">
1014     <xsd:sequence>
1015         <xsd:element name="set" maxOccurs="unbounded" type="CT_Set"/>
1016     </xsd:sequence>
1017     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1018 </xsd:complexType>
1019 <xsd:complexType name="CT_Set">
1020     <xsd:sequence>
1021         <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
1022         <xsd:element name="sortByTuple" minOccurs="0" type="CT_Tuples"/>
1023     </xsd:sequence>
1024     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1025     <xsd:attribute name="maxRank" use="required" type="xsd:int"/>
1026     <xsd:attribute name="setDefinition" use="required" type="s:ST Xstring"/>
1027     <xsd:attribute name="sortType" type="ST_SortType" default="none"/>
1028     <xsd:attribute name="queryFailed" type="xsd:boolean" default="false"/>
1029 </xsd:complexType>
1030 <xsd:simpleType name="ST_SortType">
1031     <xsd:restriction base="xsd:string">
1032         <xsd:enumeration value="none"/>
1033         <xsd:enumeration value="ascending"/>
1034         <xsd:enumeration value="descending"/>
1035         <xsd:enumeration value="ascendingAlpha"/>

```

```

1036         <xsd:enumeration value="descendingAlpha"/>
1037         <xsd:enumeration value="ascendingNatural"/>
1038         <xsd:enumeration value="descendingNatural"/>
1039     </xsd:restriction>
1040 </xsd:simpleType>
1041 <xsd:complexType name="CT_QueryCache">
1042     <xsd:sequence>
1043         <xsd:element name="query" maxOccurs="unbounded" type="CT_Query"/>
1044     </xsd:sequence>
1045     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1046 </xsd:complexType>
1047 <xsd:complexType name="CT_Query">
1048     <xsd:sequence>
1049         <xsd:element name="tpls" minOccurs="0" type="CT_Tuples"/>
1050     </xsd:sequence>
1051     <xsd:attribute name="mdx" use="required" type="s:ST_Xstring"/>
1052 </xsd:complexType>
1053 <xsd:complexType name="CT_CalculatedItems">
1054     <xsd:sequence>
1055         <xsd:element name="calculatedItem" maxOccurs="unbounded" type="CT_CalculatedItem"/>
1056     </xsd:sequence>
1057     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1058 </xsd:complexType>
1059 <xsd:complexType name="CT_CalculatedItem">
1060     <xsd:sequence>
1061         <xsd:element name="pivotArea" type="CT_PivotArea"/>
1062         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1063     </xsd:sequence>
1064     <xsd:attribute name="field" type="xsd:unsignedInt" use="optional"/>
1065     <xsd:attribute name="formula" type="s:ST_Xstring"/>
1066 </xsd:complexType>
1067 <xsd:complexType name="CT_CalculatedMembers">
1068     <xsd:sequence>
1069         <xsd:element name="calculatedMember" maxOccurs="unbounded" type="CT_CalculatedMember"/>
1070     </xsd:sequence>
1071     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1072 </xsd:complexType>
1073 <xsd:complexType name="CT_CalculatedMember">
1074     <xsd:sequence minOccurs="0">
1075         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1076     </xsd:sequence>
1077     <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1078     <xsd:attribute name="mdx" use="required" type="s:ST_Xstring"/>
1079     <xsd:attribute name="memberName" type="s:ST_Xstring"/>
1080     <xsd:attribute name="hierarchy" type="s:ST_Xstring"/>
1081     <xsd:attribute name="parent" type="s:ST_Xstring"/>
1082     <xsd:attribute name="solveOrder" type="xsd:int" default="0"/>
1083     <xsd:attribute name="set" type="xsd:boolean" default="false"/>
1084 </xsd:complexType>
1085 <xsd:complexType name="CT_pivotTableDefinition">
1086     <xsd:sequence>
1087         <xsd:element name="location" type="CT_Location"/>
1088         <xsd:element name="pivotFields" type="CT_PivotFields" minOccurs="0"/>

```

```

1089     <xsd:element name="rowFields" type="CT RowFields" minOccurs="0"/>
1090     <xsd:element name="rowItems" type="CT rowItems" minOccurs="0"/>
1091     <xsd:element name="colFields" type="CT ColFields" minOccurs="0"/>
1092     <xsd:element name="colItems" type="CT colItems" minOccurs="0"/>
1093     <xsd:element name="pageFields" type="CT PageFields" minOccurs="0"/>
1094     <xsd:element name="dataFields" type="CT DataFields" minOccurs="0"/>
1095     <xsd:element name="formats" type="CT Formats" minOccurs="0"/>
1096     <xsd:element name="conditionalFormats" type="CT ConditionalFormats" minOccurs="0"/>
1097     <xsd:element name="chartFormats" type="CT ChartFormats" minOccurs="0"/>
1098     <xsd:element name="pivotHierarchies" type="CT PivotHierarchies" minOccurs="0"/>
1099     <xsd:element name="pivotTableStyleInfo" minOccurs="0" maxOccurs="1"
1100         type="CT PivotTableStyle"/>
1101     <xsd:element name="filters" minOccurs="0" maxOccurs="1" type="CT PivotFilters"/>
1102     <xsd:element name="rowHierarchiesUsage" type="CT RowHierarchiesUsage" minOccurs="0"
1103         maxOccurs="1"/>
1104     <xsd:element name="colHierarchiesUsage" type="CT ColHierarchiesUsage" minOccurs="0"
1105         maxOccurs="1"/>
1106     <xsd:element name="extLst" minOccurs="0" type="CT ExtensionList"/>
1107 </xsd:sequence>
1108 <xsd:attribute name="name" use="required" type="s:ST Xstring"/>
1109 <xsd:attribute name="cacheId" use="required" type="xsd:unsignedInt"/>
1110 <xsd:attribute name="dataOnRows" type="xsd:boolean" default="false"/>
1111 <xsd:attribute name="dataPosition" type="xsd:unsignedInt" use="optional"/>
1112 <xsd:attributeGroup ref="AG AutoFormat"/>
1113 <xsd:attribute name="dataCaption" use="required" type="s:ST Xstring"/>
1114 <xsd:attribute name="grandTotalCaption" type="s:ST Xstring"/>
1115 <xsd:attribute name="errorCaption" type="s:ST Xstring"/>
1116 <xsd:attribute name="showError" type="xsd:boolean" default="false"/>
1117 <xsd:attribute name="missingCaption" type="s:ST Xstring"/>
1118 <xsd:attribute name="showMissing" type="xsd:boolean" default="true"/>
1119 <xsd:attribute name="pageStyle" type="s:ST Xstring"/>
1120 <xsd:attribute name="pivotTableStyle" type="s:ST Xstring"/>
1121 <xsd:attribute name="vacatedStyle" type="s:ST Xstring"/>
1122 <xsd:attribute name="tag" type="s:ST Xstring"/>
1123 <xsd:attribute name="updatedVersion" type="xsd:unsignedByte" default="0"/>
1124 <xsd:attribute name="minRefreshableVersion" type="xsd:unsignedByte" default="0"/>
1125 <xsd:attribute name="asteriskTotals" type="xsd:boolean" default="false"/>
1126 <xsd:attribute name="showItems" type="xsd:boolean" default="true"/>
1127 <xsd:attribute name="editData" type="xsd:boolean" default="false"/>
1128 <xsd:attribute name="disableFieldList" type="xsd:boolean" default="false"/>
1129 <xsd:attribute name="showCalcMbrs" type="xsd:boolean" default="true"/>
1130 <xsd:attribute name="visualTotals" type="xsd:boolean" default="true"/>
1131 <xsd:attribute name="showMultipleLabel" type="xsd:boolean" default="true"/>
1132 <xsd:attribute name="showDataDropDown" type="xsd:boolean" default="true"/>
1133 <xsd:attribute name="showDrill" type="xsd:boolean" default="true"/>
1134 <xsd:attribute name="printDrill" type="xsd:boolean" default="false"/>
1135 <xsd:attribute name="showMemberPropertyTips" type="xsd:boolean" default="true"/>
1136 <xsd:attribute name="showDataTips" type="xsd:boolean" default="true"/>
1137 <xsd:attribute name="enableWizard" type="xsd:boolean" default="true"/>
1138 <xsd:attribute name="enableDrill" type="xsd:boolean" default="true"/>
1139 <xsd:attribute name="enableFieldProperties" type="xsd:boolean" default="true"/>
1140 <xsd:attribute name="preserveFormatting" type="xsd:boolean" default="true"/>
1141 <xsd:attribute name="useAutoFormatting" type="xsd:boolean" default="false"/>

```

```

1142 <xsd:attribute name="pageWrap" type="xsd:unsignedInt" default="0"/>
1143 <xsd:attribute name="pageOverThenDown" type="xsd:boolean" default="false"/>
1144 <xsd:attribute name="subtotalHiddenItems" type="xsd:boolean" default="false"/>
1145 <xsd:attribute name="rowGrandTotals" type="xsd:boolean" default="true"/>
1146 <xsd:attribute name="colGrandTotals" type="xsd:boolean" default="true"/>
1147 <xsd:attribute name="fieldPrintTitles" type="xsd:boolean" default="false"/>
1148 <xsd:attribute name="itemPrintTitles" type="xsd:boolean" default="false"/>
1149 <xsd:attribute name="mergeItem" type="xsd:boolean" default="false"/>
1150 <xsd:attribute name="showDropZones" type="xsd:boolean" default="true"/>
1151 <xsd:attribute name="createdVersion" type="xsd:unsignedByte" default="0"/>
1152 <xsd:attribute name="indent" type="xsd:unsignedInt" default="1"/>
1153 <xsd:attribute name="showEmptyRow" type="xsd:boolean" default="false"/>
1154 <xsd:attribute name="showEmptyCol" type="xsd:boolean" default="false"/>
1155 <xsd:attribute name="showHeaders" type="xsd:boolean" default="true"/>
1156 <xsd:attribute name="compact" type="xsd:boolean" default="true"/>
1157 <xsd:attribute name="outline" type="xsd:boolean" default="false"/>
1158 <xsd:attribute name="outlineData" type="xsd:boolean" default="false"/>
1159 <xsd:attribute name="compactData" type="xsd:boolean" default="true"/>
1160 <xsd:attribute name="published" type="xsd:boolean" default="false"/>
1161 <xsd:attribute name="gridDropZones" type="xsd:boolean" default="false"/>
1162 <xsd:attribute name="immersive" type="xsd:boolean" default="true"/>
1163 <xsd:attribute name="multipleFieldFilters" type="xsd:boolean" default="true"/>
1164 <xsd:attribute name="chartFormat" type="xsd:unsignedInt" default="0"/>
1165 <xsd:attribute name="rowHeaderCaption" type="s:ST Xstring"/>
1166 <xsd:attribute name="colHeaderCaption" type="s:ST Xstring"/>
1167 <xsd:attribute name="fieldListSortAscending" type="xsd:boolean" default="false"/>
1168 <xsd:attribute name="mdxSubqueries" type="xsd:boolean" default="false"/>
1169 <xsd:attribute name="customListSort" type="xsd:boolean" use="optional" default="true"/>
1170 </xsd:complexType>
1171 <xsd:complexType name="CT_Location">
1172 <xsd:attribute name="ref" use="required" type="ST Ref"/>
1173 <xsd:attribute name="firstHeaderRow" use="required" type="xsd:unsignedInt"/>
1174 <xsd:attribute name="firstDataRow" use="required" type="xsd:unsignedInt"/>
1175 <xsd:attribute name="firstDataCol" use="required" type="xsd:unsignedInt"/>
1176 <xsd:attribute name="rowPageCount" type="xsd:unsignedInt" default="0"/>
1177 <xsd:attribute name="colPageCount" type="xsd:unsignedInt" default="0"/>
1178 </xsd:complexType>
1179 <xsd:complexType name="CT_PivotFields">
1180 <xsd:sequence>
1181 <xsd:element name="pivotField" maxOccurs="unbounded" type="CT_PivotField"/>
1182 </xsd:sequence>
1183 <xsd:attribute name="count" type="xsd:unsignedInt"/>
1184 </xsd:complexType>
1185 <xsd:complexType name="CT_PivotField">
1186 <xsd:sequence>
1187 <xsd:element name="items" minOccurs="0" type="CT_Items"/>
1188 <xsd:element name="autoSortScope" minOccurs="0" type="CT_AutoSortScope"/>
1189 <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1190 </xsd:sequence>
1191 <xsd:attribute name="name" type="s:ST Xstring"/>
1192 <xsd:attribute name="axis" use="optional" type="ST Axis"/>
1193 <xsd:attribute name="dataField" type="xsd:boolean" default="false"/>
1194 <xsd:attribute name="subtotalCaption" type="s:ST Xstring"/>

```



```

1195 <xsd:attribute name="showDropDowns" type="xsd:boolean" default="true"/>
1196 <xsd:attribute name="hiddenLevel" type="xsd:boolean" default="false"/>
1197 <xsd:attribute name="uniqueMemberProperty" type="s:ST_Xstring"/>
1198 <xsd:attribute name="compact" type="xsd:boolean" default="true"/>
1199 <xsd:attribute name="allDrilled" type="xsd:boolean" default="false"/>
1200 <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
1201 <xsd:attribute name="outline" type="xsd:boolean" default="true"/>
1202 <xsd:attribute name="subtotalTop" type="xsd:boolean" default="true"/>
1203 <xsd:attribute name="dragToRow" type="xsd:boolean" default="true"/>
1204 <xsd:attribute name="dragToCol" type="xsd:boolean" default="true"/>
1205 <xsd:attribute name="multipleItemSelectionAllowed" type="xsd:boolean" default="false"/>
1206 <xsd:attribute name="dragToPage" type="xsd:boolean" default="true"/>
1207 <xsd:attribute name="dragToData" type="xsd:boolean" default="true"/>
1208 <xsd:attribute name="dragOff" type="xsd:boolean" default="true"/>
1209 <xsd:attribute name="showAll" type="xsd:boolean" default="true"/>
1210 <xsd:attribute name="insertBlankRow" type="xsd:boolean" default="false"/>
1211 <xsd:attribute name="serverField" type="xsd:boolean" default="false"/>
1212 <xsd:attribute name="insertPageBreak" type="xsd:boolean" default="false"/>
1213 <xsd:attribute name="autoShow" type="xsd:boolean" default="false"/>
1214 <xsd:attribute name="topAutoShow" type="xsd:boolean" default="true"/>
1215 <xsd:attribute name="hideNewItem" type="xsd:boolean" default="false"/>
1216 <xsd:attribute name="measureFilter" type="xsd:boolean" default="false"/>
1217 <xsd:attribute name="includeNewItemInFilter" type="xsd:boolean" default="false"/>
1218 <xsd:attribute name="itemPageCount" type="xsd:unsignedInt" default="10"/>
1219 <xsd:attribute name="sortType" type="ST_FieldSortType" default="manual"/>
1220 <xsd:attribute name="dataSourceSort" type="xsd:boolean" use="optional"/>
1221 <xsd:attribute name="nonAutoSortDefault" type="xsd:boolean" default="false"/>
1222 <xsd:attribute name="rankBy" type="xsd:unsignedInt" use="optional"/>
1223 <xsd:attribute name="defaultSubtotal" type="xsd:boolean" default="true"/>
1224 <xsd:attribute name="sumSubtotal" type="xsd:boolean" default="false"/>
1225 <xsd:attribute name="countASubtotal" type="xsd:boolean" default="false"/>
1226 <xsd:attribute name="avgSubtotal" type="xsd:boolean" default="false"/>
1227 <xsd:attribute name="maxSubtotal" type="xsd:boolean" default="false"/>
1228 <xsd:attribute name="minSubtotal" type="xsd:boolean" default="false"/>
1229 <xsd:attribute name="productSubtotal" type="xsd:boolean" default="false"/>
1230 <xsd:attribute name="countSubtotal" type="xsd:boolean" default="false"/>
1231 <xsd:attribute name="stdDevSubtotal" type="xsd:boolean" default="false"/>
1232 <xsd:attribute name="stdDevPSubtotal" type="xsd:boolean" default="false"/>
1233 <xsd:attribute name="varSubtotal" type="xsd:boolean" default="false"/>
1234 <xsd:attribute name="varPSubtotal" type="xsd:boolean" default="false"/>
1235 <xsd:attribute name="showPropCell" type="xsd:boolean" use="optional" default="false"/>
1236 <xsd:attribute name="showPropTip" type="xsd:boolean" use="optional" default="false"/>
1237 <xsd:attribute name="showPropAsCaption" type="xsd:boolean" use="optional" default="false"/>
1238 <xsd:attribute name="defaultAttributeDrillState" type="xsd:boolean" use="optional"
1239     default="false"/>
1240 </xsd:complexType>
1241 <xsd:complexType name="CT_AutoSortScope">
1242     <xsd:sequence>
1243         <xsd:element name="pivotArea" type="CT_PivotArea"/>
1244     </xsd:sequence>
1245 </xsd:complexType>
1246 <xsd:complexType name="CT_Items">
1247     <xsd:sequence>

```

```

1248     <xsd:element name="item" maxOccurs="unbounded" type="CT_Item"/>
1249   </xsd:sequence>
1250   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1251 </xsd:complexType>
1252 <xsd:complexType name="CT_Item">
1253   <xsd:attribute name="n" type="s:ST_Xstring"/>
1254   <xsd:attribute name="t" type="ST_ItemType" default="data"/>
1255   <xsd:attribute name="h" type="xsd:boolean" default="false"/>
1256   <xsd:attribute name="s" type="xsd:boolean" default="false"/>
1257   <xsd:attribute name="sd" type="xsd:boolean" default="true"/>
1258   <xsd:attribute name="f" type="xsd:boolean" default="false"/>
1259   <xsd:attribute name="m" type="xsd:boolean" default="false"/>
1260   <xsd:attribute name="c" type="xsd:boolean" default="false"/>
1261   <xsd:attribute name="x" type="xsd:unsignedInt" use="optional"/>
1262   <xsd:attribute name="d" type="xsd:boolean" default="false"/>
1263   <xsd:attribute name="e" type="xsd:boolean" default="true"/>
1264 </xsd:complexType>
1265 <xsd:complexType name="CT_PageFields">
1266   <xsd:sequence>
1267     <xsd:element name="pageField" maxOccurs="unbounded" type="CT_PageField"/>
1268   </xsd:sequence>
1269   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1270 </xsd:complexType>
1271 <xsd:complexType name="CT_PageField">
1272   <xsd:sequence minOccurs="0">
1273     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1274   </xsd:sequence>
1275   <xsd:attribute name="fld" use="required" type="xsd:int"/>
1276   <xsd:attribute name="item" use="optional" type="xsd:unsignedInt"/>
1277   <xsd:attribute name="hier" type="xsd:int"/>
1278   <xsd:attribute name="name" type="s:ST_Xstring"/>
1279   <xsd:attribute name="cap" type="s:ST_Xstring"/>
1280 </xsd:complexType>
1281 <xsd:complexType name="CT_DataFields">
1282   <xsd:sequence>
1283     <xsd:element name="dataField" maxOccurs="unbounded" type="CT_DataField"/>
1284   </xsd:sequence>
1285   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1286 </xsd:complexType>
1287 <xsd:complexType name="CT_DataField">
1288   <xsd:sequence>
1289     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1290   </xsd:sequence>
1291   <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
1292   <xsd:attribute name="fld" type="xsd:unsignedInt" use="required"/>
1293   <xsd:attribute name="subtotal" type="ST_DataConsolidateFunction" default="sum"/>
1294   <xsd:attribute name="showDataAs" type="ST_ShowDataAs" default="normal"/>
1295   <xsd:attribute name="baseField" type="xsd:int" default="-1"/>
1296   <xsd:attribute name="baseItem" type="xsd:unsignedInt" default="1048832"/>
1297   <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
1298 </xsd:complexType>
1299 <xsd:complexType name="CT_rowItems">
1300   <xsd:sequence>

```

```

1301     <xsd:element name="i" maxOccurs="unbounded" type="CT_I"/>
1302   </xsd:sequence>
1303   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1304 </xsd:complexType>
1305 <xsd:complexType name="CT_colItems">
1306   <xsd:sequence>
1307     <xsd:element name="i" maxOccurs="unbounded" type="CT_I"/>
1308   </xsd:sequence>
1309   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1310 </xsd:complexType>
1311 <xsd:complexType name="CT_I">
1312   <xsd:sequence>
1313     <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
1314   </xsd:sequence>
1315   <xsd:attribute name="t" type="ST_ItemType" default="data"/>
1316   <xsd:attribute name="r" type="xsd:unsignedInt" default="0"/>
1317   <xsd:attribute name="i" type="xsd:unsignedInt" default="0"/>
1318 </xsd:complexType>
1319 <xsd:complexType name="CT_X">
1320   <xsd:attribute name="v" type="xsd:int" default="0"/>
1321 </xsd:complexType>
1322 <xsd:complexType name="CT_RowFields">
1323   <xsd:sequence>
1324     <xsd:element name="field" maxOccurs="unbounded" type="CT_Field"/>
1325   </xsd:sequence>
1326   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1327 </xsd:complexType>
1328 <xsd:complexType name="CT_ColFields">
1329   <xsd:sequence>
1330     <xsd:element name="field" maxOccurs="unbounded" type="CT_Field"/>
1331   </xsd:sequence>
1332   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1333 </xsd:complexType>
1334 <xsd:complexType name="CT_Field">
1335   <xsd:attribute name="x" type="xsd:int" use="required"/>
1336 </xsd:complexType>
1337 <xsd:complexType name="CT_Formats">
1338   <xsd:sequence>
1339     <xsd:element name="format" maxOccurs="unbounded" type="CT_Format"/>
1340   </xsd:sequence>
1341   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1342 </xsd:complexType>
1343 <xsd:complexType name="CT_Format">
1344   <xsd:sequence>
1345     <xsd:element name="pivotArea" type="CT_PivotArea"/>
1346     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1347   </xsd:sequence>
1348   <xsd:attribute name="action" type="ST_FormatAction" default="formatting"/>
1349   <xsd:attribute name="dxId" type="ST_DxfId" use="optional"/>
1350 </xsd:complexType>
1351 <xsd:complexType name="CT_ConditionalFormats">
1352   <xsd:sequence>
1353     <xsd:element name="conditionalFormat" maxOccurs="unbounded" type="CT_ConditionalFormat"/>

```

```

1354     </xsd:sequence>
1355     <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1356 </xsd:complexType>
1357 <xsd:complexType name="CT_ConditionalFormat">
1358     <xsd:sequence>
1359         <xsd:element name="pivotAreas" type="CT_PivotAreas"/>
1360         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1361     </xsd:sequence>
1362     <xsd:attribute name="scope" type="ST_Scope" default="selection"/>
1363     <xsd:attribute name="type" type="ST_Type" default="none"/>
1364     <xsd:attribute name="priority" use="required" type="xsd:unsignedInt"/>
1365 </xsd:complexType>
1366 <xsd:complexType name="CT_PivotAreas">
1367     <xsd:sequence>
1368         <xsd:element name="pivotArea" minOccurs="0" maxOccurs="unbounded" type="CT_PivotArea"/>
1369     </xsd:sequence>
1370     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1371 </xsd:complexType>
1372 <xsd:simpleType name="ST_Scope">
1373     <xsd:restriction base="xsd:string">
1374         <xsd:enumeration value="selection"/>
1375         <xsd:enumeration value="data"/>
1376         <xsd:enumeration value="field"/>
1377     </xsd:restriction>
1378 </xsd:simpleType>
1379 <xsd:simpleType name="ST_Type">
1380     <xsd:restriction base="xsd:string">
1381         <xsd:enumeration value="none"/>
1382         <xsd:enumeration value="all"/>
1383         <xsd:enumeration value="row"/>
1384         <xsd:enumeration value="column"/>
1385     </xsd:restriction>
1386 </xsd:simpleType>
1387 <xsd:complexType name="CT_ChartFormats">
1388     <xsd:sequence>
1389         <xsd:element name="chartFormat" maxOccurs="unbounded" type="CT_ChartFormat"/>
1390     </xsd:sequence>
1391     <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1392 </xsd:complexType>
1393 <xsd:complexType name="CT_ChartFormat">
1394     <xsd:sequence>
1395         <xsd:element name="pivotArea" type="CT_PivotArea"/>
1396     </xsd:sequence>
1397     <xsd:attribute name="chart" use="required" type="xsd:unsignedInt"/>
1398     <xsd:attribute name="format" use="required" type="xsd:unsignedInt"/>
1399     <xsd:attribute name="series" type="xsd:boolean" default="false"/>
1400 </xsd:complexType>
1401 <xsd:complexType name="CT_PivotHierarchies">
1402     <xsd:sequence>
1403         <xsd:element name="pivotHierarchy" maxOccurs="unbounded" type="CT_PivotHierarchy"/>
1404     </xsd:sequence>
1405     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1406 </xsd:complexType>

```

```

1407 <xsd:complexType name="CT_PivotHierarchy">
1408     <xsd:sequence>
1409         <xsd:element name="mps" minOccurs="0" type="CT_MemberProperties"/>
1410         <xsd:element name="members" minOccurs="0" maxOccurs="unbounded" type="CT_Members"/>
1411         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1412     </xsd:sequence>
1413     <xsd:attribute name="outline" type="xsd:boolean" default="false"/>
1414     <xsd:attribute name="multipleItemSelectionAllowed" type="xsd:boolean" default="false"/>
1415     <xsd:attribute name="subtotalTop" type="xsd:boolean" default="false"/>
1416     <xsd:attribute name="showInFieldList" type="xsd:boolean" default="true"/>
1417     <xsd:attribute name="dragToRow" type="xsd:boolean" default="true"/>
1418     <xsd:attribute name="dragToCol" type="xsd:boolean" default="true"/>
1419     <xsd:attribute name="dragToPage" type="xsd:boolean" default="true"/>
1420     <xsd:attribute name="dragToData" type="xsd:boolean" default="false"/>
1421     <xsd:attribute name="dragOff" type="xsd:boolean" default="true"/>
1422     <xsd:attribute name="includeNewItemInFilter" type="xsd:boolean" default="false"/>
1423     <xsd:attribute name="caption" type="s:ST_Xstring" use="optional"/>
1424 </xsd:complexType>
1425 <xsd:complexType name="CT_RowHierarchiesUsage">
1426     <xsd:sequence>
1427         <xsd:element name="rowHierarchyUsage" minOccurs="1" maxOccurs="unbounded"
1428             type="CT_HierarchyUsage"/>
1429     </xsd:sequence>
1430     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1431 </xsd:complexType>
1432 <xsd:complexType name="CT_ColHierarchiesUsage">
1433     <xsd:sequence>
1434         <xsd:element name="colHierarchyUsage" minOccurs="1" maxOccurs="unbounded"
1435             type="CT_HierarchyUsage"/>
1436     </xsd:sequence>
1437     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1438 </xsd:complexType>
1439 <xsd:complexType name="CT_HierarchyUsage">
1440     <xsd:attribute name="hierarchyUsage" type="xsd:int" use="required"/>
1441 </xsd:complexType>
1442 <xsd:complexType name="CT_MemberProperties">
1443     <xsd:sequence>
1444         <xsd:element name="mp" maxOccurs="unbounded" type="CT_MemberProperty"/>
1445     </xsd:sequence>
1446     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1447 </xsd:complexType>
1448 <xsd:complexType name="CT_MemberProperty">
1449     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
1450     <xsd:attribute name="showCell" type="xsd:boolean" use="optional" default="false"/>
1451     <xsd:attribute name="showTip" type="xsd:boolean" use="optional" default="false"/>
1452     <xsd:attribute name="showAsCaption" type="xsd:boolean" use="optional" default="false"/>
1453     <xsd:attribute name="nameLen" type="xsd:unsignedInt" use="optional"/>
1454     <xsd:attribute name="pPos" type="xsd:unsignedInt" use="optional"/>
1455     <xsd:attribute name="pLen" type="xsd:unsignedInt" use="optional"/>
1456     <xsd:attribute name="level" type="xsd:unsignedInt" use="optional"/>
1457     <xsd:attribute name="field" use="required" type="xsd:unsignedInt"/>
1458 </xsd:complexType>
1459 <xsd:complexType name="CT_Members">

```

```

1460     <xsd:sequence>
1461         <xsd:element name="member" maxOccurs="unbounded" type="CT_Member"/>
1462     </xsd:sequence>
1463     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1464     <xsd:attribute name="level" use="optional" type="xsd:unsignedInt"/>
1465 </xsd:complexType>
1466 <xsd:complexType name="CT_Member">
1467     <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1468 </xsd:complexType>
1469 <xsd:complexType name="CT_Dimensions">
1470     <xsd:sequence>
1471         <xsd:element name="dimension" minOccurs="0" maxOccurs="unbounded"
1472             type="CT_PivotDimension"/>
1473     </xsd:sequence>
1474     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1475 </xsd:complexType>
1476 <xsd:complexType name="CT_PivotDimension">
1477     <xsd:attribute name="measure" type="xsd:boolean" default="false"/>
1478     <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1479     <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
1480     <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
1481 </xsd:complexType>
1482 <xsd:complexType name="CT_MeasureGroups">
1483     <xsd:sequence>
1484         <xsd:element name="measureGroup" minOccurs="0" maxOccurs="unbounded"
1485             type="CT_MeasureGroup"/>
1486     </xsd:sequence>
1487     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1488 </xsd:complexType>
1489 <xsd:complexType name="CT_MeasureDimensionMaps">
1490     <xsd:sequence>
1491         <xsd:element name="map" minOccurs="0" maxOccurs="unbounded"
1492             type="CT_MeasureDimensionMap"/>
1493     </xsd:sequence>
1494     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1495 </xsd:complexType>
1496 <xsd:complexType name="CT_MeasureGroup">
1497     <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1498     <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
1499 </xsd:complexType>
1500 <xsd:complexType name="CT_MeasureDimensionMap">
1501     <xsd:attribute name="measureGroup" use="optional" type="xsd:unsignedInt"/>
1502     <xsd:attribute name="dimension" use="optional" type="xsd:unsignedInt"/>
1503 </xsd:complexType>
1504 <xsd:complexType name="CT_PivotTableStyle">
1505     <xsd:attribute name="name" type="xsd:string"/>
1506     <xsd:attribute name="showRowHeaders" type="xsd:boolean"/>
1507     <xsd:attribute name="showColHeaders" type="xsd:boolean"/>
1508     <xsd:attribute name="showRowStripes" type="xsd:boolean"/>
1509     <xsd:attribute name="showColStripes" type="xsd:boolean"/>
1510     <xsd:attribute name="showLastColumn" type="xsd:boolean" use="optional"/>
1511 </xsd:complexType>
1512 <xsd:complexType name="CT_PivotFilters">

```

```

1513     <xsd:sequence>
1514         <xsd:element name="filter" minOccurs="0" maxOccurs="unbounded" type="CT_PivotFilter"/>
1515     </xsd:sequence>
1516     <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1517 </xsd:complexType>
1518 <xsd:complexType name="CT_PivotFilter">
1519     <xsd:sequence>
1520         <xsd:element name="autoFilter" minOccurs="1" maxOccurs="1" type="CT_AutoFilter"/>
1521         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1522     </xsd:sequence>
1523     <xsd:attribute name="fld" use="required" type="xsd:unsignedInt"/>
1524     <xsd:attribute name="mpFld" type="xsd:unsignedInt" use="optional"/>
1525     <xsd:attribute name="type" use="required" type="ST_PivotFilterType"/>
1526     <xsd:attribute name="evalOrder" use="optional" type="xsd:int" default="0"/>
1527     <xsd:attribute name="id" use="required" type="xsd:unsignedInt"/>
1528     <xsd:attribute name="iMeasureHier" use="optional" type="xsd:unsignedInt"/>
1529     <xsd:attribute name="iMeasureFld" use="optional" type="xsd:unsignedInt"/>
1530     <xsd:attribute name="name" type="s:ST_Xstring"/>
1531     <xsd:attribute name="description" type="s:ST_Xstring"/>
1532     <xsd:attribute name="stringValue1" type="s:ST_Xstring"/>
1533     <xsd:attribute name="stringValue2" type="s:ST_Xstring"/>
1534 </xsd:complexType>
1535 <xsd:simpleType name="ST_ShowDataAs">
1536     <xsd:restriction base="xsd:string">
1537         <xsd:enumeration value="normal"/>
1538         <xsd:enumeration value="difference"/>
1539         <xsd:enumeration value="percent"/>
1540         <xsd:enumeration value="percentDiff"/>
1541         <xsd:enumeration value="runTotal"/>
1542         <xsd:enumeration value="percentOfRow"/>
1543         <xsd:enumeration value="percentOfCol"/>
1544         <xsd:enumeration value="percentOfTotal"/>
1545         <xsd:enumeration value="index"/>
1546     </xsd:restriction>
1547 </xsd:simpleType>
1548 <xsd:simpleType name="ST_ItemType">
1549     <xsd:restriction base="xsd:string">
1550         <xsd:enumeration value="data"/>
1551         <xsd:enumeration value="default"/>
1552         <xsd:enumeration value="sum"/>
1553         <xsd:enumeration value="countA"/>
1554         <xsd:enumeration value="avg"/>
1555         <xsd:enumeration value="max"/>
1556         <xsd:enumeration value="min"/>
1557         <xsd:enumeration value="product"/>
1558         <xsd:enumeration value="count"/>
1559         <xsd:enumeration value="stdDev"/>
1560         <xsd:enumeration value="stdDevP"/>
1561         <xsd:enumeration value="var"/>
1562         <xsd:enumeration value="varP"/>
1563         <xsd:enumeration value="grand"/>
1564         <xsd:enumeration value="blank"/>
1565     </xsd:restriction>

```

```

1566 </xsd:simpleType>
1567 <xsd:simpleType name="ST_FormatAction">
1568   <xsd:restriction base="xsd:string">
1569     <xsd:enumeration value="blank"/>
1570     <xsd:enumeration value="formatting"/>
1571     <xsd:enumeration value="drill"/>
1572     <xsd:enumeration value="formula"/>
1573   </xsd:restriction>
1574 </xsd:simpleType>
1575 <xsd:simpleType name="ST_FieldSortType">
1576   <xsd:restriction base="xsd:string">
1577     <xsd:enumeration value="manual"/>
1578     <xsd:enumeration value="ascending"/>
1579     <xsd:enumeration value="descending"/>
1580   </xsd:restriction>
1581 </xsd:simpleType>
1582 <xsd:simpleType name="ST_PivotFilterType">
1583   <xsd:restriction base="xsd:string">
1584     <xsd:enumeration value="unknown"/>
1585     <xsd:enumeration value="count"/>
1586     <xsd:enumeration value="percent"/>
1587     <xsd:enumeration value="sum"/>
1588     <xsd:enumeration value="captionEqual"/>
1589     <xsd:enumeration value="captionNotEqual"/>
1590     <xsd:enumeration value="captionBeginsWith"/>
1591     <xsd:enumeration value="captionNotBeginsWith"/>
1592     <xsd:enumeration value="captionEndsWith"/>
1593     <xsd:enumeration value="captionNotEndsWith"/>
1594     <xsd:enumeration value="captionContains"/>
1595     <xsd:enumeration value="captionNotContains"/>
1596     <xsd:enumeration value="captionGreaterThan"/>
1597     <xsd:enumeration value="captionGreaterThanOrEqual"/>
1598     <xsd:enumeration value="captionLessThan"/>
1599     <xsd:enumeration value="captionLessThanOrEqual"/>
1600     <xsd:enumeration value="captionBetween"/>
1601     <xsd:enumeration value="captionNotBetween"/>
1602     <xsd:enumeration value="valueEqual"/>
1603     <xsd:enumeration value="valueNotEqual"/>
1604     <xsd:enumeration value="valueGreaterThan"/>
1605     <xsd:enumeration value="valueGreaterThanOrEqual"/>
1606     <xsd:enumeration value="valueLessThan"/>
1607     <xsd:enumeration value="valueLessThanOrEqual"/>
1608     <xsd:enumeration value="valueBetween"/>
1609     <xsd:enumeration value="valueNotBetween"/>
1610     <xsd:enumeration value="dateEqual"/>
1611     <xsd:enumeration value="dateNotEqual"/>
1612     <xsd:enumeration value="dateOlderThan"/>
1613     <xsd:enumeration value="dateOlderThanOrEqual"/>
1614     <xsd:enumeration value="dateNewerThan"/>
1615     <xsd:enumeration value="dateNewerThanOrEqual"/>
1616     <xsd:enumeration value="dateBetween"/>
1617     <xsd:enumeration value="dateNotBetween"/>
1618     <xsd:enumeration value="tomorrow"/>

```



```

1619     <xsd:enumeration value="today"/>
1620     <xsd:enumeration value="yesterday"/>
1621     <xsd:enumeration value="nextWeek"/>
1622     <xsd:enumeration value="thisWeek"/>
1623     <xsd:enumeration value="lastWeek"/>
1624     <xsd:enumeration value="nextMonth"/>
1625     <xsd:enumeration value="thisMonth"/>
1626     <xsd:enumeration value="lastMonth"/>
1627     <xsd:enumeration value="nextQuarter"/>
1628     <xsd:enumeration value="thisQuarter"/>
1629     <xsd:enumeration value="lastQuarter"/>
1630     <xsd:enumeration value="nextYear"/>
1631     <xsd:enumeration value="thisYear"/>
1632     <xsd:enumeration value="lastYear"/>
1633     <xsd:enumeration value="yearToDate"/>
1634     <xsd:enumeration value="Q1"/>
1635     <xsd:enumeration value="Q2"/>
1636     <xsd:enumeration value="Q3"/>
1637     <xsd:enumeration value="Q4"/>
1638     <xsd:enumeration value="M1"/>
1639     <xsd:enumeration value="M2"/>
1640     <xsd:enumeration value="M3"/>
1641     <xsd:enumeration value="M4"/>
1642     <xsd:enumeration value="M5"/>
1643     <xsd:enumeration value="M6"/>
1644     <xsd:enumeration value="M7"/>
1645     <xsd:enumeration value="M8"/>
1646     <xsd:enumeration value="M9"/>
1647     <xsd:enumeration value="M10"/>
1648     <xsd:enumeration value="M11"/>
1649     <xsd:enumeration value="M12"/>
1650 </xsd:restriction>
1651 </xsd:simpleType>
1652 <xsd:complexType name="CT_PivotArea">
1653     <xsd:sequence>
1654         <xsd:element name="references" minOccurs="0" type="CT_PivotAreaReferences"/>
1655         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1656     </xsd:sequence>
1657     <xsd:attribute name="field" use="optional" type="xsd:int"/>
1658     <xsd:attribute name="type" type="ST_PivotAreaType" default="normal"/>
1659     <xsd:attribute name="dataOnly" type="xsd:boolean" default="true"/>
1660     <xsd:attribute name="labelOnly" type="xsd:boolean" default="false"/>
1661     <xsd:attribute name="grandRow" type="xsd:boolean" default="false"/>
1662     <xsd:attribute name="grandCol" type="xsd:boolean" default="false"/>
1663     <xsd:attribute name="cacheIndex" type="xsd:boolean" default="false"/>
1664     <xsd:attribute name="outline" type="xsd:boolean" default="true"/>
1665     <xsd:attribute name="offset" type="ST_Ref"/>
1666     <xsd:attribute name="collapsedLevelsAreSubtotals" type="xsd:boolean" default="false"/>
1667     <xsd:attribute name="axis" type="ST_Axis" use="optional"/>
1668     <xsd:attribute name="fieldPosition" type="xsd:unsignedInt" use="optional"/>
1669 </xsd:complexType>
1670 <xsd:simpleType name="ST_PivotAreaType">
1671     <xsd:restriction base="xsd:string">

```

```

1672         <xsd:enumeration value="none"/>
1673         <xsd:enumeration value="normal"/>
1674         <xsd:enumeration value="data"/>
1675         <xsd:enumeration value="all"/>
1676         <xsd:enumeration value="origin"/>
1677         <xsd:enumeration value="button"/>
1678         <xsd:enumeration value="topEnd"/>
1679         <xsd:enumeration value="topRight"/>
1680     </xsd:restriction>
1681 </xsd:simpleType>
1682 <xsd:complexType name="CT_PivotAreaReferences">
1683     <xsd:sequence>
1684         <xsd:element name="reference" maxOccurs="unbounded" type="CT_PivotAreaReference"/>
1685     </xsd:sequence>
1686     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1687 </xsd:complexType>
1688 <xsd:complexType name="CT_PivotAreaReference">
1689     <xsd:sequence>
1690         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_Index"/>
1691         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1692     </xsd:sequence>
1693     <xsd:attribute name="field" use="optional" type="xsd:unsignedInt"/>
1694     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1695     <xsd:attribute name="selected" type="xsd:boolean" default="true"/>
1696     <xsd:attribute name="byPosition" type="xsd:boolean" default="false"/>
1697     <xsd:attribute name="relative" type="xsd:boolean" default="false"/>
1698     <xsd:attribute name="defaultSubtotal" type="xsd:boolean" default="false"/>
1699     <xsd:attribute name="sumSubtotal" type="xsd:boolean" default="false"/>
1700     <xsd:attribute name="countASubtotal" type="xsd:boolean" default="false"/>
1701     <xsd:attribute name="avgSubtotal" type="xsd:boolean" default="false"/>
1702     <xsd:attribute name="maxSubtotal" type="xsd:boolean" default="false"/>
1703     <xsd:attribute name="minSubtotal" type="xsd:boolean" default="false"/>
1704     <xsd:attribute name="productSubtotal" type="xsd:boolean" default="false"/>
1705     <xsd:attribute name="countSubtotal" type="xsd:boolean" default="false"/>
1706     <xsd:attribute name="stdDevSubtotal" type="xsd:boolean" default="false"/>
1707     <xsd:attribute name="stdDevPSubtotal" type="xsd:boolean" default="false"/>
1708     <xsd:attribute name="varSubtotal" type="xsd:boolean" default="false"/>
1709     <xsd:attribute name="varPSubtotal" type="xsd:boolean" default="false"/>
1710 </xsd:complexType>
1711 <xsd:complexType name="CT_Index">
1712     <xsd:attribute name="v" use="required" type="xsd:unsignedInt"/>
1713 </xsd:complexType>
1714 <xsd:simpleType name="ST_Axis">
1715     <xsd:restriction base="xsd:string">
1716         <xsd:enumeration value="axisRow"/>
1717         <xsd:enumeration value="axisCol"/>
1718         <xsd:enumeration value="axisPage"/>
1719         <xsd:enumeration value="axisValues"/>
1720     </xsd:restriction>
1721 </xsd:simpleType>
1722 <xsd:element name="queryTable" type="CT_QueryTable"/>
1723 <xsd:complexType name="CT_QueryTable">
1724     <xsd:sequence>

```

```

1725     <xsd:element name="queryTableRefresh" type="CT_QueryTableRefresh" minOccurs="0"
1726         maxOccurs="1"/>
1727     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1728 </xsd:sequence>
1729 <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
1730 <xsd:attribute name="headers" type="xsd:boolean" use="optional" default="true"/>
1731 <xsd:attribute name="rowNumbers" type="xsd:boolean" use="optional" default="false"/>
1732 <xsd:attribute name="disableRefresh" type="xsd:boolean" use="optional" default="false"/>
1733 <xsd:attribute name="backgroundRefresh" type="xsd:boolean" use="optional" default="true"/>
1734 <xsd:attribute name="firstBackgroundRefresh" type="xsd:boolean" use="optional"
1735     default="false"/>
1736 <xsd:attribute name="refreshOnLoad" type="xsd:boolean" use="optional" default="false"/>
1737 <xsd:attribute name="growShrinkType" type="ST_GrowShrinkType" use="optional"
1738     default="insertDelete"/>
1739 <xsd:attribute name="fillFormulas" type="xsd:boolean" use="optional" default="false"/>
1740 <xsd:attribute name="removeDataOnSave" type="xsd:boolean" use="optional" default="false"/>
1741 <xsd:attribute name="disableEdit" type="xsd:boolean" use="optional" default="false"/>
1742 <xsd:attribute name="preserveFormatting" type="xsd:boolean" use="optional" default="true"/>
1743 <xsd:attribute name="adjustColumnWidth" type="xsd:boolean" use="optional" default="true"/>
1744 <xsd:attribute name="intermediate" type="xsd:boolean" use="optional" default="false"/>
1745 <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="required"/>
1746 <xsd:attributeGroup ref="AG_AutoFormat"/>
1747 </xsd:complexType>
1748 <xsd:complexType name="CT_QueryTableRefresh">
1749     <xsd:sequence>
1750         <xsd:element name="queryTableFields" type="CT_QueryTableFields" minOccurs="1"
1751             maxOccurs="1"/>
1752         <xsd:element name="queryTableDeletedFields" type="CT_QueryTableDeletedFields"
1753             minOccurs="0" maxOccurs="1"/>
1754         <xsd:element name="sortState" minOccurs="0" maxOccurs="1" type="CT_SortState"/>
1755         <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
1756     </xsd:sequence>
1757     <xsd:attribute name="preserveSortFilterLayout" type="xsd:boolean" use="optional"
1758         default="true"/>
1759     <xsd:attribute name="fieldIdWrapped" type="xsd:boolean" use="optional" default="false"/>
1760     <xsd:attribute name="headersInLastRefresh" type="xsd:boolean" use="optional" default="true"/>
1761     <xsd:attribute name="minimumVersion" type="xsd:unsignedByte" use="optional" default="0"/>
1762     <xsd:attribute name="nextId" type="xsd:unsignedInt" use="optional" default="1"/>
1763     <xsd:attribute name="unboundColumnsLeft" type="xsd:unsignedInt" use="optional" default="0"/>
1764     <xsd:attribute name="unboundColumnsRight" type="xsd:unsignedInt" use="optional" default="0"/>
1765 </xsd:complexType>
1766 <xsd:complexType name="CT_QueryTableDeletedFields">
1767     <xsd:sequence>
1768         <xsd:element name="deletedField" type="CT_DeletedField" minOccurs="1"
1769             maxOccurs="unbounded"/>
1770     </xsd:sequence>
1771     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1772 </xsd:complexType>
1773 <xsd:complexType name="CT_DeletedField">
1774     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
1775 </xsd:complexType>
1776 <xsd:complexType name="CT_QueryTableFields">
1777     <xsd:sequence>

```

```

1778     <xsd:element name="queryTableField" type="CT_QueryTableField" minOccurs="0"
1779         maxOccurs="unbounded"/>
1780 </xsd:sequence>
1781     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
1782 </xsd:complexType>
1783 <xsd:complexType name="CT_QueryTableField">
1784     <xsd:sequence minOccurs="0">
1785         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1786     </xsd:sequence>
1787     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
1788     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
1789     <xsd:attribute name="dataBound" type="xsd:boolean" use="optional" default="true"/>
1790     <xsd:attribute name="rowNumbers" type="xsd:boolean" use="optional" default="false"/>
1791     <xsd:attribute name="fillFormulas" type="xsd:boolean" use="optional" default="false"/>
1792     <xsd:attribute name="clipped" type="xsd:boolean" use="optional" default="false"/>
1793     <xsd:attribute name="tableColumnId" type="xsd:unsignedInt" default="0"/>
1794 </xsd:complexType>
1795 <xsd:simpleType name="ST_GrowShrinkType">
1796     <xsd:restriction base="xsd:string">
1797         <xsd:enumeration value="insertDelete"/>
1798         <xsd:enumeration value="insertClear"/>
1799         <xsd:enumeration value="overwriteClear"/>
1800     </xsd:restriction>
1801 </xsd:simpleType>
1802 <xsd:element name="sst" type="CT_Sst"/>
1803 <xsd:complexType name="CT_Sst">
1804     <xsd:sequence>
1805         <xsd:element name="si" type="CT_Rst" minOccurs="0" maxOccurs="unbounded"/>
1806         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1807     </xsd:sequence>
1808     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1809     <xsd:attribute name="uniqueCount" type="xsd:unsignedInt" use="optional"/>
1810 </xsd:complexType>
1811 <xsd:simpleType name="ST_PhoneticType">
1812     <xsd:restriction base="xsd:string">
1813         <xsd:enumeration value="halfwidthKatakana"/>
1814         <xsd:enumeration value="fullwidthKatakana"/>
1815         <xsd:enumeration value="Hiragana"/>
1816         <xsd:enumeration value="noConversion"/>
1817     </xsd:restriction>
1818 </xsd:simpleType>
1819 <xsd:simpleType name="ST_PhoneticAlignment">
1820     <xsd:restriction base="xsd:string">
1821         <xsd:enumeration value="noControl"/>
1822         <xsd:enumeration value="left"/>
1823         <xsd:enumeration value="center"/>
1824         <xsd:enumeration value="distributed"/>
1825     </xsd:restriction>
1826 </xsd:simpleType>
1827 <xsd:complexType name="CT_PhoneticRun">
1828     <xsd:sequence>
1829         <xsd:element name="t" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
1830     </xsd:sequence>

```

```

1831     <xsd:attribute name="sb" type="xsd:unsignedInt" use="required"/>
1832     <xsd:attribute name="eb" type="xsd:unsignedInt" use="required"/>
1833 </xsd:complexType>
1834 <xsd:complexType name="CT_RElt">
1835     <xsd:sequence>
1836         <xsd:element name="rPr" type="CT_RPrElt" minOccurs="0" maxOccurs="1"/>
1837         <xsd:element name="t" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
1838     </xsd:sequence>
1839 </xsd:complexType>
1840 <xsd:complexType name="CT_RPrElt">
1841     <xsd:choice maxOccurs="unbounded">
1842         <xsd:element name="rFont" type="CT_FontName" minOccurs="0" maxOccurs="1"/>
1843         <xsd:element name="charset" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
1844         <xsd:element name="family" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
1845         <xsd:element name="b" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1846         <xsd:element name="i" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1847         <xsd:element name="strike" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1848         <xsd:element name="outline" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1849         <xsd:element name="shadow" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1850         <xsd:element name="condense" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1851         <xsd:element name="extend" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1852         <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1853         <xsd:element name="sz" type="CT_FontSize" minOccurs="0" maxOccurs="1"/>
1854         <xsd:element name="u" type="CT_UnderlineProperty" minOccurs="0" maxOccurs="1"/>
1855         <xsd:element name="vertAlign" type="CT_VerticalAlignFontProperty" minOccurs="0"
1856             maxOccurs="1"/>
1857         <xsd:element name="scheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
1858     </xsd:choice>
1859 </xsd:complexType>
1860 <xsd:complexType name="CT_Rst">
1861     <xsd:sequence>
1862         <xsd:element name="t" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1863         <xsd:element name="r" type="CT_RElt" minOccurs="0" maxOccurs="unbounded"/>
1864         <xsd:element name="rPh" type="CT_PhoneticRun" minOccurs="0" maxOccurs="unbounded"/>
1865         <xsd:element name="phoneticPr" minOccurs="0" maxOccurs="1" type="CT_PhoneticPr"/>
1866     </xsd:sequence>
1867 </xsd:complexType>
1868 <xsd:complexType name="CT_PhoneticPr">
1869     <xsd:attribute name="fontId" type="ST_FontId" use="required"/>
1870     <xsd:attribute name="type" type="ST_PhoneticType" use="optional" default="fullwidthKatakana"/>
1871     <xsd:attribute name="alignment" type="ST_PhoneticAlignment" use="optional" default="left"/>
1872 </xsd:complexType>
1873 <xsd:element name="headers" type="CT_RevisionHeaders"/>
1874 <xsd:element name="revisions" type="CT_Revisions"/>
1875 <xsd:complexType name="CT_RevisionHeaders">
1876     <xsd:sequence>
1877         <xsd:element name="header" type="CT_RevisionHeader" minOccurs="1" maxOccurs="unbounded"/>
1878     </xsd:sequence>
1879     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
1880     <xsd:attribute name="lastGuid" type="s:ST_Guid" use="optional"/>
1881     <xsd:attribute name="shared" type="xsd:boolean" default="true"/>
1882     <xsd:attribute name="diskRevisions" type="xsd:boolean" default="false"/>
1883     <xsd:attribute name="history" type="xsd:boolean" default="true"/>

```

```

1884 <xsd:attribute name="trackRevisions" type="xsd:boolean" default="true"/>
1885 <xsd:attribute name="exclusive" type="xsd:boolean" default="false"/>
1886 <xsd:attribute name="revisionId" type="xsd:unsignedInt" default="0"/>
1887 <xsd:attribute name="version" type="xsd:int" default="1"/>
1888 <xsd:attribute name="keepChangeHistory" type="xsd:boolean" use="optional" default="true"/>
1889 <xsd:attribute name="protected" type="xsd:boolean" use="optional" default="false"/>
1890 <xsd:attribute name="preserveHistory" type="xsd:unsignedInt" default="30"/>
1891 </xsd:complexType>
1892 <xsd:complexType name="CT_Revisions">
1893   <xsd:choice maxOccurs="unbounded">
1894     <xsd:element name="rrc" type="CT_RevisionRowColumn" minOccurs="0" maxOccurs="unbounded"/>
1895     <xsd:element name="rm" type="CT_RevisionMove" minOccurs="0" maxOccurs="unbounded"/>
1896     <xsd:element name="rcv" type="CT_RevisionCustomView" minOccurs="0" maxOccurs="unbounded"/>
1897     <xsd:element name="rsnm" type="CT_RevisionSheetRename" minOccurs="0"
1898       maxOccurs="unbounded"/>
1899     <xsd:element name="ris" type="CT_RevisionInsertSheet" minOccurs="0"
1900       maxOccurs="unbounded"/>
1901     <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1902     <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1903       maxOccurs="unbounded"/>
1904     <xsd:element name="raf" type="CT_RevisionAutoFormatting" minOccurs="0"
1905       maxOccurs="unbounded"/>
1906     <xsd:element name="rdn" type="CT_RevisionDefinedName" minOccurs="0"
1907       maxOccurs="unbounded"/>
1908     <xsd:element name="rcmt" type="CT_RevisionComment" minOccurs="0" maxOccurs="unbounded"/>
1909     <xsd:element name="rqt" type="CT_RevisionQueryTableField" minOccurs="0"
1910       maxOccurs="unbounded"/>
1911     <xsd:element name="rcft" type="CT_RevisionConflict" minOccurs="0" maxOccurs="unbounded"/>
1912   </xsd:choice>
1913 </xsd:complexType>
1914 <xsd:attributeGroup name="AG_RevData">
1915   <xsd:attribute name="rId" type="xsd:unsignedInt" use="required"/>
1916   <xsd:attribute name="ua" type="xsd:boolean" use="optional" default="false"/>
1917   <xsd:attribute name="ra" type="xsd:boolean" use="optional" default="false"/>
1918 </xsd:attributeGroup>
1919 <xsd:complexType name="CT_RevisionHeader">
1920   <xsd:sequence>
1921     <xsd:element name="sheetIdMap" minOccurs="1" maxOccurs="1" type="CT_SheetIdMap"/>
1922     <xsd:element name="reviewedList" minOccurs="0" maxOccurs="1" type="CT_ReviewedRevisions"/>
1923     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1924   </xsd:sequence>
1925   <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
1926   <xsd:attribute name="dateTime" type="xsd:dateTime" use="required"/>
1927   <xsd:attribute name="maxSheetId" type="xsd:unsignedInt" use="required"/>
1928   <xsd:attribute name="userName" type="s:ST_Xstring" use="required"/>
1929   <xsd:attribute ref="r:id" use="required"/>
1930   <xsd:attribute name="minRId" type="xsd:unsignedInt" use="optional"/>
1931   <xsd:attribute name="maxRId" type="xsd:unsignedInt" use="optional"/>
1932 </xsd:complexType>
1933 <xsd:complexType name="CT_SheetIdMap">
1934   <xsd:sequence>
1935     <xsd:element name="sheetId" type="CT_SheetId" minOccurs="1" maxOccurs="unbounded"/>
1936   </xsd:sequence>

```

```

1937     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1938 </xsd:complexType>
1939 <xsd:complexType name="CT_SheetId">
1940     <xsd:attribute name="val" type="xsd:unsignedInt" use="required"/>
1941 </xsd:complexType>
1942 <xsd:complexType name="CT_ReviewedRevisions">
1943     <xsd:sequence>
1944         <xsd:element name="reviewed" type="CT_Reviewed" minOccurs="1" maxOccurs="unbounded"/>
1945     </xsd:sequence>
1946     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1947 </xsd:complexType>
1948 <xsd:complexType name="CT_Reviewed">
1949     <xsd:attribute name="rId" type="xsd:unsignedInt" use="required"/>
1950 </xsd:complexType>
1951 <xsd:complexType name="CT_UndoInfo">
1952     <xsd:attribute name="index" type="xsd:unsignedInt" use="required"/>
1953     <xsd:attribute name="exp" type="ST_FormulaExpression" use="required"/>
1954     <xsd:attribute name="ref3D" type="xsd:boolean" use="optional" default="false"/>
1955     <xsd:attribute name="array" type="xsd:boolean" use="optional" default="false"/>
1956     <xsd:attribute name="v" type="xsd:boolean" use="optional" default="false"/>
1957     <xsd:attribute name="nf" type="xsd:boolean" use="optional" default="false"/>
1958     <xsd:attribute name="cs" type="xsd:boolean" use="optional" default="false"/>
1959     <xsd:attribute name="dr" type="ST_RefA" use="required"/>
1960     <xsd:attribute name="dn" type="s:ST_Xstring" use="optional"/>
1961     <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
1962     <xsd:attribute name="sId" type="xsd:unsignedInt" use="optional"/>
1963 </xsd:complexType>
1964 <xsd:complexType name="CT_RevisionRowColumn">
1965     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1966         <xsd:element name="undo" type="CT_UndoInfo" minOccurs="0" maxOccurs="unbounded"/>
1967         <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1968         <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1969             maxOccurs="unbounded"/>
1970     </xsd:choice>
1971     <xsd:attributeGroup ref="AG_RevData"/>
1972     <xsd:attribute name="sId" type="xsd:unsignedInt" use="required"/>
1973     <xsd:attribute name="eol" type="xsd:boolean" use="optional" default="false"/>
1974     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
1975     <xsd:attribute name="action" type="ST_rwColActionType" use="required"/>
1976     <xsd:attribute name="edge" type="xsd:boolean" use="optional" default="false"/>
1977 </xsd:complexType>
1978 <xsd:complexType name="CT_RevisionMove">
1979     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1980         <xsd:element name="undo" type="CT_UndoInfo" minOccurs="0" maxOccurs="unbounded"/>
1981         <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1982         <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1983             maxOccurs="unbounded"/>
1984     </xsd:choice>
1985     <xsd:attributeGroup ref="AG_RevData"/>
1986     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
1987     <xsd:attribute name="source" type="ST_Ref" use="required"/>
1988     <xsd:attribute name="destination" type="ST_Ref" use="required"/>
1989     <xsd:attribute name="sourceSheetId" type="xsd:unsignedInt" use="optional" default="0"/>

```

```

1990 </xsd:complexType>
1991 <xsd:complexType name="CT_RevisionCustomView">
1992   <xsd:attribute name="guid" type="s:ST Guid" use="required"/>
1993   <xsd:attribute name="action" type="ST RevisionAction" use="required"/>
1994 </xsd:complexType>
1995 <xsd:complexType name="CT_RevisionSheetRename">
1996   <xsd:sequence>
1997     <xsd:element name="extLst" minOccurs="0" type="CT ExtensionList"/>
1998   </xsd:sequence>
1999   <xsd:attributeGroup ref="AG RevData"/>
2000   <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2001   <xsd:attribute name="oldName" type="s:ST Xstring" use="required"/>
2002   <xsd:attribute name="newName" type="s:ST Xstring" use="required"/>
2003 </xsd:complexType>
2004 <xsd:complexType name="CT_RevisionInsertSheet">
2005   <xsd:attributeGroup ref="AG RevData"/>
2006   <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2007   <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
2008   <xsd:attribute name="sheetPosition" type="xsd:unsignedInt" use="required"/>
2009 </xsd:complexType>
2010 <xsd:complexType name="CT_RevisionCellChange">
2011   <xsd:sequence>
2012     <xsd:element name="oc" type="CT Cell" minOccurs="0" maxOccurs="1"/>
2013     <xsd:element name="nc" type="CT Cell" minOccurs="1" maxOccurs="1"/>
2014     <xsd:element name="odxf" type="CT Dxf" minOccurs="0" maxOccurs="1"/>
2015     <xsd:element name="ndxf" type="CT Dxf" minOccurs="0" maxOccurs="1"/>
2016     <xsd:element name="extLst" minOccurs="0" type="CT ExtensionList"/>
2017   </xsd:sequence>
2018   <xsd:attributeGroup ref="AG RevData"/>
2019   <xsd:attribute name="sId" type="xsd:unsignedInt" use="required"/>
2020   <xsd:attribute name="odxf" type="xsd:boolean" default="false"/>
2021   <xsd:attribute name="xfDxf" type="xsd:boolean" use="optional" default="false"/>
2022   <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
2023   <xsd:attribute name="dxf" type="xsd:boolean" default="false"/>
2024   <xsd:attribute name="numFmtId" type="ST NumFmtId" use="optional"/>
2025   <xsd:attribute name="quotePrefix" type="xsd:boolean" use="optional" default="false"/>
2026   <xsd:attribute name="oldQuotePrefix" type="xsd:boolean" use="optional" default="false"/>
2027   <xsd:attribute name="ph" type="xsd:boolean" default="false"/>
2028   <xsd:attribute name="oldPh" type="xsd:boolean" default="false"/>
2029   <xsd:attribute name="endOfListFormulaUpdate" type="xsd:boolean" default="false"/>
2030 </xsd:complexType>
2031 <xsd:complexType name="CT_RevisionFormatting">
2032   <xsd:sequence>
2033     <xsd:element name="dxf" type="CT Dxf" minOccurs="0" maxOccurs="1"/>
2034     <xsd:element name="extLst" minOccurs="0" type="CT ExtensionList"/>
2035   </xsd:sequence>
2036   <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2037   <xsd:attribute name="xfDxf" type="xsd:boolean" use="optional" default="false"/>
2038   <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
2039   <xsd:attribute name="sqref" type="ST Sqref" use="required"/>
2040   <xsd:attribute name="start" type="xsd:unsignedInt" use="optional"/>
2041   <xsd:attribute name="length" type="xsd:unsignedInt" use="optional"/>
2042 </xsd:complexType>

```



```

2043 <xsd:complexType name="CT_RevisionAutoFormatting">
2044     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2045     <xsd:attributeGroup ref="AG_AutoFormat"/>
2046     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2047 </xsd:complexType>
2048 <xsd:complexType name="CT_RevisionComment">
2049     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2050     <xsd:attribute name="cell" type="ST_CellRef" use="required"/>
2051     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
2052     <xsd:attribute name="action" type="ST_RevisionAction" default="add"/>
2053     <xsd:attribute name="alwaysShow" type="xsd:boolean" use="optional" default="false"/>
2054     <xsd:attribute name="old" type="xsd:boolean" use="optional" default="false"/>
2055     <xsd:attribute name="hiddenRow" type="xsd:boolean" use="optional" default="false"/>
2056     <xsd:attribute name="hiddenColumn" type="xsd:boolean" use="optional" default="false"/>
2057     <xsd:attribute name="author" type="s:ST_Xstring" use="required"/>
2058     <xsd:attribute name="oldLength" type="xsd:unsignedInt" default="0"/>
2059     <xsd:attribute name="newLength" type="xsd:unsignedInt" default="0"/>
2060 </xsd:complexType>
2061 <xsd:complexType name="CT_RevisionDefinedName">
2062     <xsd:sequence>
2063         <xsd:element name="formula" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2064         <xsd:element name="oldFormula" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2065         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2066     </xsd:sequence>
2067     <xsd:attributeGroup ref="AG_RevData"/>
2068     <xsd:attribute name="localSheetId" type="xsd:unsignedInt" use="optional"/>
2069     <xsd:attribute name="customView" type="xsd:boolean" use="optional" default="false"/>
2070     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2071     <xsd:attribute name="function" type="xsd:boolean" use="optional" default="false"/>
2072     <xsd:attribute name="oldFunction" type="xsd:boolean" default="false"/>
2073     <xsd:attribute name="functionGroupId" type="xsd:unsignedByte" use="optional"/>
2074     <xsd:attribute name="oldFunctionGroupId" type="xsd:unsignedByte" use="optional"/>
2075     <xsd:attribute name="shortcutKey" type="xsd:unsignedByte" use="optional"/>
2076     <xsd:attribute name="oldShortcutKey" type="xsd:unsignedByte" use="optional"/>
2077     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2078     <xsd:attribute name="oldHidden" type="xsd:boolean" use="optional" default="false"/>
2079     <xsd:attribute name="customMenu" type="s:ST_Xstring" use="optional"/>
2080     <xsd:attribute name="oldCustomMenu" type="s:ST_Xstring" use="optional"/>
2081     <xsd:attribute name="description" type="s:ST_Xstring" use="optional"/>
2082     <xsd:attribute name="oldDescription" type="s:ST_Xstring" use="optional"/>
2083     <xsd:attribute name="help" type="s:ST_Xstring" use="optional"/>
2084     <xsd:attribute name="oldHelp" type="s:ST_Xstring" use="optional"/>
2085     <xsd:attribute name="statusBar" type="s:ST_Xstring" use="optional"/>
2086     <xsd:attribute name="oldStatusBar" type="s:ST_Xstring" use="optional"/>
2087     <xsd:attribute name="comment" type="s:ST_Xstring" use="optional"/>
2088     <xsd:attribute name="oldComment" type="s:ST_Xstring" use="optional"/>
2089 </xsd:complexType>
2090 <xsd:complexType name="CT_RevisionConflict">
2091     <xsd:attributeGroup ref="AG_RevData"/>
2092     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="optional"/>
2093 </xsd:complexType>
2094 <xsd:complexType name="CT_RevisionQueryTableField">
2095     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>

```

```

2096     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2097     <xsd:attribute name="fieldId" type="xsd:unsignedInt" use="required"/>
2098 </xsd:complexType>
2099 <xsd:simpleType name="ST_rwColActionType">
2100     <xsd:restriction base="xsd:string">
2101         <xsd:enumeration value="insertRow"/>
2102         <xsd:enumeration value="deleteRow"/>
2103         <xsd:enumeration value="insertCol"/>
2104         <xsd:enumeration value="deleteCol"/>
2105     </xsd:restriction>
2106 </xsd:simpleType>
2107 <xsd:simpleType name="ST_RevisionAction">
2108     <xsd:restriction base="xsd:string">
2109         <xsd:enumeration value="add"/>
2110         <xsd:enumeration value="delete"/>
2111     </xsd:restriction>
2112 </xsd:simpleType>
2113 <xsd:simpleType name="ST_FormulaExpression">
2114     <xsd:restriction base="xsd:string">
2115         <xsd:enumeration value="ref"/>
2116         <xsd:enumeration value="refError"/>
2117         <xsd:enumeration value="area"/>
2118         <xsd:enumeration value="areaError"/>
2119         <xsd:enumeration value="computedArea"/>
2120     </xsd:restriction>
2121 </xsd:simpleType>
2122 <xsd:element name="users" type="CT_Users"/>
2123 <xsd:complexType name="CT_Users">
2124     <xsd:sequence>
2125         <xsd:element name="userInfo" minOccurs="0" maxOccurs="256" type="CT_SharedUser"/>
2126     </xsd:sequence>
2127     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2128 </xsd:complexType>
2129 <xsd:complexType name="CT_SharedUser">
2130     <xsd:sequence>
2131         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2132     </xsd:sequence>
2133     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
2134     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2135     <xsd:attribute name="id" type="xsd:int" use="required"/>
2136     <xsd:attribute name="dateTime" type="xsd:dateTime" use="required"/>
2137 </xsd:complexType>
2138 <xsd:element name="worksheet" type="CT_Worksheet"/>
2139 <xsd:element name="chartsheet" type="CT_Chartsheet"/>
2140 <xsd:element name="dialogsheet" type="CT_Dialogsheet"/>
2141 <xsd:complexType name="CT_Macrosheet">
2142     <xsd:sequence>
2143         <xsd:element name="sheetPr" type="CT_SheetPr" minOccurs="0" maxOccurs="1"/>
2144         <xsd:element name="dimension" type="CT_SheetDimension" minOccurs="0" maxOccurs="1"/>
2145         <xsd:element name="sheetViews" type="CT_SheetViews" minOccurs="0" maxOccurs="1"/>
2146         <xsd:element name="sheetFormatPr" type="CT_SheetFormatPr" minOccurs="0" maxOccurs="1"/>
2147         <xsd:element name="cols" type="CT_Cols" minOccurs="0" maxOccurs="unbounded"/>
2148         <xsd:element name="sheetData" type="CT_SheetData" minOccurs="1" maxOccurs="1"/>

```

```

2149     <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2150         maxOccurs="1"/>
2151     <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
2152     <xsd:element name="sortState" type="CT_SortState" minOccurs="0" maxOccurs="1"/>
2153     <xsd:element name="dataConsolidate" type="CT_DataConsolidate" minOccurs="0"
2154         maxOccurs="1"/>
2155     <xsd:element name="customSheetViews" type="CT_CustomSheetViews" minOccurs="0"
2156         maxOccurs="1"/>
2157     <xsd:element name="phoneticPr" type="CT_PhoneticPr" minOccurs="0" maxOccurs="1"/>
2158     <xsd:element name="conditionalFormatting" type="CT_ConditionalFormatting" minOccurs="0"
2159         maxOccurs="unbounded"/>
2160     <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2161     <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2162     <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2163     <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2164     <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2165     <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2166     <xsd:element name="customProperties" type="CT_CustomProperties" minOccurs="0"
2167         maxOccurs="1"/>
2168     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0" maxOccurs="1"/>
2169     <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2170     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2171     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2172     <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
2173     <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2174     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2175 </xsd:sequence>
2176 </xsd:complexType>
2177 <xsd:complexType name="CT_Dialogsheet">
2178     <xsd:sequence>
2179         <xsd:element name="sheetPr" minOccurs="0" type="CT_SheetPr"/>
2180         <xsd:element name="sheetViews" minOccurs="0" type="CT_SheetViews"/>
2181         <xsd:element name="sheetFormatPr" minOccurs="0" type="CT_SheetFormatPr"/>
2182         <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2183             maxOccurs="1"/>
2184         <xsd:element name="customSheetViews" minOccurs="0" type="CT_CustomSheetViews"/>
2185         <xsd:element name="printOptions" minOccurs="0" type="CT_PrintOptions"/>
2186         <xsd:element name="pageMargins" minOccurs="0" type="CT_PageMargins"/>
2187         <xsd:element name="pageSetup" minOccurs="0" type="CT_PageSetup"/>
2188         <xsd:element name="headerFooter" minOccurs="0" type="CT_HeaderFooter"/>
2189         <xsd:element name="drawing" minOccurs="0" type="CT_Drawing"/>
2190         <xsd:element name="legacyDrawing" minOccurs="0" type="CT_LegacyDrawing"/>
2191         <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2192         <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2193         <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2194         <xsd:element name="controls" type="CT_Controls" minOccurs="0" maxOccurs="1"/>
2195         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2196     </xsd:sequence>
2197 </xsd:complexType>
2198 <xsd:complexType name="CT_Worksheet">
2199     <xsd:sequence>
2200         <xsd:element name="sheetPr" type="CT_SheetPr" minOccurs="0" maxOccurs="1"/>
2201         <xsd:element name="dimension" type="CT_SheetDimension" minOccurs="0" maxOccurs="1"/>

```

```

2202 <xsd:element name="sheetViews" type="CT_SheetViews" minOccurs="0" maxOccurs="1"/>
2203 <xsd:element name="sheetFormatPr" type="CT_SheetFormatPr" minOccurs="0" maxOccurs="1"/>
2204 <xsd:element name="cols" type="CT_Cols" minOccurs="0" maxOccurs="unbounded"/>
2205 <xsd:element name="sheetData" type="CT_SheetData" minOccurs="1" maxOccurs="1"/>
2206 <xsd:element name="sheetCalcPr" type="CT_SheetCalcPr" minOccurs="0" maxOccurs="1"/>
2207 <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2208     maxOccurs="1"/>
2209 <xsd:element name="protectedRanges" type="CT_ProtectedRanges" minOccurs="0"
2210     maxOccurs="1"/>
2211 <xsd:element name="scenarios" type="CT_Scenarios" minOccurs="0" maxOccurs="1"/>
2212 <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
2213 <xsd:element name="sortState" type="CT_SortState" minOccurs="0" maxOccurs="1"/>
2214 <xsd:element name="dataConsolidate" type="CT_DataConsolidate" minOccurs="0"
2215     maxOccurs="1"/>
2216 <xsd:element name="customSheetViews" type="CT_CustomSheetViews" minOccurs="0"
2217     maxOccurs="1"/>
2218 <xsd:element name="mergeCells" type="CT_MergeCells" minOccurs="0" maxOccurs="1"/>
2219 <xsd:element name="phoneticPr" type="CT_PhoneticPr" minOccurs="0" maxOccurs="1"/>
2220 <xsd:element name="conditionalFormatting" type="CT_ConditionalFormatting" minOccurs="0"
2221     maxOccurs="unbounded"/>
2222 <xsd:element name="dataValidations" type="CT_DataValidations" minOccurs="0"
2223     maxOccurs="1"/>
2224 <xsd:element name="hyperlinks" type="CT_Hyperlinks" minOccurs="0" maxOccurs="1"/>
2225 <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2226 <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2227 <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2228 <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2229 <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2230 <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2231 <xsd:element name="customProperties" type="CT_CustomProperties" minOccurs="0"
2232     maxOccurs="1"/>
2233 <xsd:element name="cellWatches" type="CT_CellWatches" minOccurs="0" maxOccurs="1"/>
2234 <xsd:element name="ignoredErrors" type="CT_IgnoredErrors" minOccurs="0" maxOccurs="1"/>
2235 <xsd:element name="smartTags" type="CT_SmartTags" minOccurs="0" maxOccurs="1"/>
2236 <xsd:element name="drawing" type="CT_Drawing" minOccurs="0" maxOccurs="1"/>
2237 <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2238 <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2239 <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2240 <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
2241 <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2242 <xsd:element name="controls" type="CT_Controls" minOccurs="0" maxOccurs="1"/>
2243 <xsd:element name="webPublishItems" type="CT_WebPublishItems" minOccurs="0"
2244     maxOccurs="1"/>
2245 <xsd:element name="tableParts" type="CT_TableParts" minOccurs="0" maxOccurs="1"/>
2246 <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
2247 </xsd:sequence>
2248 </xsd:complexType>
2249 <xsd:complexType name="CT_SheetData">
2250     <xsd:sequence>
2251         <xsd:element name="row" type="CT_Row" minOccurs="0" maxOccurs="unbounded"/>
2252     </xsd:sequence>
2253 </xsd:complexType>
2254 <xsd:complexType name="CT_SheetCalcPr">

```

```

2255     <xsd:attribute name="fullCalcOnLoad" type="xsd:boolean" use="optional" default="false"/>
2256 </xsd:complexType>
2257 <xsd:complexType name="CT_SheetFormatPr">
2258     <xsd:attribute name="baseColWidth" type="xsd:unsignedInt" use="optional" default="8"/>
2259     <xsd:attribute name="defaultColWidth" type="xsd:double" use="optional"/>
2260     <xsd:attribute name="defaultRowHeight" type="xsd:double" use="required"/>
2261     <xsd:attribute name="customHeight" type="xsd:boolean" use="optional" default="false"/>
2262     <xsd:attribute name="zeroHeight" type="xsd:boolean" use="optional" default="false"/>
2263     <xsd:attribute name="thickTop" type="xsd:boolean" use="optional" default="false"/>
2264     <xsd:attribute name="thickBottom" type="xsd:boolean" use="optional" default="false"/>
2265     <xsd:attribute name="outlineLevelRow" type="xsd:unsignedByte" use="optional" default="0"/>
2266     <xsd:attribute name="outlineLevelCol" type="xsd:unsignedByte" use="optional" default="0"/>
2267 </xsd:complexType>
2268 <xsd:complexType name="CT_Cols">
2269     <xsd:sequence>
2270         <xsd:element name="col" type="CT_Col" minOccurs="1" maxOccurs="unbounded"/>
2271     </xsd:sequence>
2272 </xsd:complexType>
2273 <xsd:complexType name="CT_Col">
2274     <xsd:attribute name="min" type="xsd:unsignedInt" use="required"/>
2275     <xsd:attribute name="max" type="xsd:unsignedInt" use="required"/>
2276     <xsd:attribute name="width" type="xsd:double" use="optional"/>
2277     <xsd:attribute name="style" type="xsd:unsignedInt" use="optional" default="0"/>
2278     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2279     <xsd:attribute name="bestFit" type="xsd:boolean" use="optional" default="false"/>
2280     <xsd:attribute name="customWidth" type="xsd:boolean" use="optional" default="false"/>
2281     <xsd:attribute name="phonetic" type="xsd:boolean" use="optional" default="false"/>
2282     <xsd:attribute name="outlineLevel" type="xsd:unsignedByte" use="optional" default="0"/>
2283     <xsd:attribute name="collapsed" type="xsd:boolean" use="optional" default="false"/>
2284 </xsd:complexType>
2285 <xsd:simpleType name="ST_CellSpan">
2286     <xsd:restriction base="xsd:string"/>
2287 </xsd:simpleType>
2288 <xsd:simpleType name="ST_CellSpans">
2289     <xsd:list itemType="ST_CellSpan"/>
2290 </xsd:simpleType>
2291 <xsd:complexType name="CT_Row">
2292     <xsd:sequence>
2293         <xsd:element name="c" type="CT_Cell" minOccurs="0" maxOccurs="unbounded"/>
2294         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2295     </xsd:sequence>
2296     <xsd:attribute name="r" type="xsd:unsignedInt" use="optional"/>
2297     <xsd:attribute name="spans" type="ST_CellSpans" use="optional"/>
2298     <xsd:attribute name="s" type="xsd:unsignedInt" use="optional" default="0"/>
2299     <xsd:attribute name="customFormat" type="xsd:boolean" use="optional" default="false"/>
2300     <xsd:attribute name="ht" type="xsd:double" use="optional"/>
2301     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2302     <xsd:attribute name="customHeight" type="xsd:boolean" use="optional" default="false"/>
2303     <xsd:attribute name="outlineLevel" type="xsd:unsignedByte" use="optional" default="0"/>
2304     <xsd:attribute name="collapsed" type="xsd:boolean" use="optional" default="false"/>
2305     <xsd:attribute name="thickTop" type="xsd:boolean" use="optional" default="false"/>
2306     <xsd:attribute name="thickBot" type="xsd:boolean" use="optional" default="false"/>
2307     <xsd:attribute name="ph" type="xsd:boolean" use="optional" default="false"/>

```

```

2308 </xsd:complexType>
2309 <xsd:complexType name="CT_Cell">
2310   <xsd:sequence>
2311     <xsd:element name="f" type="CT_CellFormula" minOccurs="0" maxOccurs="1"/>
2312     <xsd:element name="v" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2313     <xsd:element name="is" type="CT_Rst" minOccurs="0" maxOccurs="1"/>
2314     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2315   </xsd:sequence>
2316   <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
2317   <xsd:attribute name="s" type="xsd:unsignedInt" use="optional" default="0"/>
2318   <xsd:attribute name="t" type="ST_CellType" use="optional" default="n"/>
2319   <xsd:attribute name="cm" type="xsd:unsignedInt" use="optional" default="0"/>
2320   <xsd:attribute name="vm" type="xsd:unsignedInt" use="optional" default="0"/>
2321   <xsd:attribute name="ph" type="xsd:boolean" use="optional" default="false"/>
2322 </xsd:complexType>
2323 <xsd:simpleType name="ST_CellType">
2324   <xsd:restriction base="xsd:string">
2325     <xsd:enumeration value="b"/>
2326     <xsd:enumeration value="n"/>
2327     <xsd:enumeration value="e"/>
2328     <xsd:enumeration value="s"/>
2329     <xsd:enumeration value="str"/>
2330     <xsd:enumeration value="inlineStr"/>
2331   </xsd:restriction>
2332 </xsd:simpleType>
2333 <xsd:simpleType name="ST_CellFormulaType">
2334   <xsd:restriction base="xsd:string">
2335     <xsd:enumeration value="normal"/>
2336     <xsd:enumeration value="array"/>
2337     <xsd:enumeration value="dataTable"/>
2338     <xsd:enumeration value="shared"/>
2339   </xsd:restriction>
2340 </xsd:simpleType>
2341 <xsd:complexType name="CT_SheetPr">
2342   <xsd:sequence>
2343     <xsd:element name="tabColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
2344     <xsd:element name="outlinePr" type="CT_OutlinePr" minOccurs="0" maxOccurs="1"/>
2345     <xsd:element name="pageSetUpPr" type="CT_PageSetUpPr" minOccurs="0" maxOccurs="1"/>
2346   </xsd:sequence>
2347   <xsd:attribute name="syncHorizontal" type="xsd:boolean" use="optional" default="false"/>
2348   <xsd:attribute name="syncVertical" type="xsd:boolean" use="optional" default="false"/>
2349   <xsd:attribute name="syncRef" type="ST_Ref" use="optional"/>
2350   <xsd:attribute name="transitionEvaluation" type="xsd:boolean" use="optional" default="false"/>
2351   <xsd:attribute name="transitionEntry" type="xsd:boolean" use="optional" default="false"/>
2352   <xsd:attribute name="published" type="xsd:boolean" use="optional" default="true"/>
2353   <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
2354   <xsd:attribute name="filterMode" type="xsd:boolean" use="optional" default="false"/>
2355   <xsd:attribute name="enableFormatConditionsCalculation" type="xsd:boolean" use="optional"
2356     default="true"/>
2357 </xsd:complexType>
2358 <xsd:complexType name="CT_SheetDimension">
2359   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2360 </xsd:complexType>

```

```

2361 <xsd:complexType name="CT_SheetViews">
2362   <xsd:sequence>
2363     <xsd:element name="sheetView" type="CT_SheetView" minOccurs="1" maxOccurs="unbounded"/>
2364     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
2365   </xsd:sequence>
2366 </xsd:complexType>
2367 <xsd:complexType name="CT_SheetView">
2368   <xsd:sequence>
2369     <xsd:element name="pane" type="CT_Pane" minOccurs="0" maxOccurs="1"/>
2370     <xsd:element name="selection" type="CT_Selection" minOccurs="0" maxOccurs="4"/>
2371     <xsd:element name="pivotSelection" type="CT_PivotSelection" minOccurs="0" maxOccurs="4"/>
2372     <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
2373   </xsd:sequence>
2374   <xsd:attribute name="windowProtection" type="xsd:boolean" use="optional" default="false"/>
2375   <xsd:attribute name="showFormulas" type="xsd:boolean" use="optional" default="false"/>
2376   <xsd:attribute name="showGridLines" type="xsd:boolean" use="optional" default="true"/>
2377   <xsd:attribute name="showRowColHeaders" type="xsd:boolean" use="optional" default="true"/>
2378   <xsd:attribute name="showZeros" type="xsd:boolean" use="optional" default="true"/>
2379   <xsd:attribute name="rightToLeft" type="xsd:boolean" use="optional" default="false"/>
2380   <xsd:attribute name="tabSelected" type="xsd:boolean" use="optional" default="false"/>
2381   <xsd:attribute name="showRuler" type="xsd:boolean" use="optional" default="true"/>
2382   <xsd:attribute name="showOutlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2383   <xsd:attribute name="defaultGridColor" type="xsd:boolean" use="optional" default="true"/>
2384   <xsd:attribute name="showWhiteSpace" type="xsd:boolean" use="optional" default="true"/>
2385   <xsd:attribute name="view" type="ST_SheetViewType" use="optional" default="normal"/>
2386   <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2387   <xsd:attribute name="colorId" type="xsd:unsignedInt" use="optional" default="64"/>
2388   <xsd:attribute name="zoomScale" type="xsd:unsignedInt" use="optional" default="100"/>
2389   <xsd:attribute name="zoomScaleNormal" type="xsd:unsignedInt" use="optional" default="0"/>
2390   <xsd:attribute name="zoomScaleSheetLayoutView" type="xsd:unsignedInt" use="optional"
2391     default="0"/>
2392   <xsd:attribute name="zoomScalePageLayoutView" type="xsd:unsignedInt" use="optional"
2393     default="0"/>
2394   <xsd:attribute name="workbookViewId" type="xsd:unsignedInt" use="required"/>
2395 </xsd:complexType>
2396 <xsd:complexType name="CT_Pane">
2397   <xsd:attribute name="xSplit" type="xsd:double" use="optional" default="0"/>
2398   <xsd:attribute name="ySplit" type="xsd:double" use="optional" default="0"/>
2399   <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2400   <xsd:attribute name="activePane" type="ST_Pane" use="optional" default="topLeft"/>
2401   <xsd:attribute name="state" type="ST_PaneState" use="optional" default="split"/>
2402 </xsd:complexType>
2403 <xsd:complexType name="CT_PivotSelection">
2404   <xsd:sequence>
2405     <xsd:element name="pivotArea" type="CT_PivotArea"/>
2406   </xsd:sequence>
2407   <xsd:attribute name="pane" type="ST_Pane" use="optional" default="topLeft"/>
2408   <xsd:attribute name="showHeader" type="xsd:boolean" default="false"/>
2409   <xsd:attribute name="label" type="xsd:boolean" default="false"/>
2410   <xsd:attribute name="data" type="xsd:boolean" default="false"/>
2411   <xsd:attribute name="extendable" type="xsd:boolean" default="false"/>
2412   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
2413   <xsd:attribute name="axis" type="ST_Axis" use="optional"/>

```

```

2414     <xsd:attribute name="dimension" type="xsd:unsignedInt" default="0"/>
2415     <xsd:attribute name="start" type="xsd:unsignedInt" default="0"/>
2416     <xsd:attribute name="min" type="xsd:unsignedInt" default="0"/>
2417     <xsd:attribute name="max" type="xsd:unsignedInt" default="0"/>
2418     <xsd:attribute name="activeRow" type="xsd:unsignedInt" default="0"/>
2419     <xsd:attribute name="activeCol" type="xsd:unsignedInt" default="0"/>
2420     <xsd:attribute name="previousRow" type="xsd:unsignedInt" default="0"/>
2421     <xsd:attribute name="previousCol" type="xsd:unsignedInt" default="0"/>
2422     <xsd:attribute name="click" type="xsd:unsignedInt" default="0"/>
2423     <xsd:attribute ref="r:id" use="optional"/>
2424 </xsd:complexType>
2425 <xsd:complexType name="CT_Selection">
2426     <xsd:attribute name="pane" type="ST_Pane" use="optional" default="topLeft"/>
2427     <xsd:attribute name="activeCell" type="ST_CellRef" use="optional"/>
2428     <xsd:attribute name="activeCellId" type="xsd:unsignedInt" use="optional" default="0"/>
2429     <xsd:attribute name="sqref" type="ST_Sqref" use="optional" default="A1"/>
2430 </xsd:complexType>
2431 <xsd:simpleType name="ST_Pane">
2432     <xsd:restriction base="xsd:string">
2433         <xsd:enumeration value="bottomRight"/>
2434         <xsd:enumeration value="topRight"/>
2435         <xsd:enumeration value="bottomLeft"/>
2436         <xsd:enumeration value="topLeft"/>
2437     </xsd:restriction>
2438 </xsd:simpleType>
2439 <xsd:complexType name="CT_PageBreak">
2440     <xsd:sequence>
2441         <xsd:element name="brk" type="CT_Break" minOccurs="0" maxOccurs="unbounded"/>
2442     </xsd:sequence>
2443     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
2444     <xsd:attribute name="manualBreakCount" type="xsd:unsignedInt" use="optional" default="0"/>
2445 </xsd:complexType>
2446 <xsd:complexType name="CT_Break">
2447     <xsd:attribute name="id" type="xsd:unsignedInt" use="optional" default="0"/>
2448     <xsd:attribute name="min" type="xsd:unsignedInt" use="optional" default="0"/>
2449     <xsd:attribute name="max" type="xsd:unsignedInt" use="optional" default="0"/>
2450     <xsd:attribute name="man" type="xsd:boolean" use="optional" default="false"/>
2451     <xsd:attribute name="pt" type="xsd:boolean" use="optional" default="false"/>
2452 </xsd:complexType>
2453 <xsd:simpleType name="ST_SheetViewType">
2454     <xsd:restriction base="xsd:string">
2455         <xsd:enumeration value="normal"/>
2456         <xsd:enumeration value="pageBreakPreview"/>
2457         <xsd:enumeration value="pageLayout"/>
2458     </xsd:restriction>
2459 </xsd:simpleType>
2460 <xsd:complexType name="CT_OutlinePr">
2461     <xsd:attribute name="applyStyles" type="xsd:boolean" use="optional" default="false"/>
2462     <xsd:attribute name="summaryBelow" type="xsd:boolean" use="optional" default="true"/>
2463     <xsd:attribute name="summaryRight" type="xsd:boolean" use="optional" default="true"/>
2464     <xsd:attribute name="showOutlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2465 </xsd:complexType>
2466 <xsd:complexType name="CT_PageSetUpPr">

```



```

2467     <xsd:attribute name="autoPageBreaks" type="xsd:boolean" use="optional" default="true"/>
2468     <xsd:attribute name="fitToPage" type="xsd:boolean" use="optional" default="false"/>
2469 </xsd:complexType>
2470 <xsd:complexType name="CT_DataConsolidate">
2471     <xsd:sequence>
2472         <xsd:element name="dataRefs" type="CT_DataRefs" minOccurs="0" maxOccurs="1"/>
2473     </xsd:sequence>
2474     <xsd:attribute name="function" type="ST_DataConsolidateFunction" use="optional"
2475         default="sum"/>
2476     <xsd:attribute name="startLabels" type="xsd:boolean" use="optional" default="false"/>
2477     <xsd:attribute name="leftLabels" type="xsd:boolean" use="optional" default="false"/>
2478     <xsd:attribute name="topLabels" type="xsd:boolean" use="optional" default="false"/>
2479     <xsd:attribute name="link" type="xsd:boolean" use="optional" default="false"/>
2480 </xsd:complexType>
2481 <xsd:simpleType name="ST_DataConsolidateFunction">
2482     <xsd:restriction base="xsd:string">
2483         <xsd:enumeration value="average"/>
2484         <xsd:enumeration value="count"/>
2485         <xsd:enumeration value="countNums"/>
2486         <xsd:enumeration value="max"/>
2487         <xsd:enumeration value="min"/>
2488         <xsd:enumeration value="product"/>
2489         <xsd:enumeration value="stdDev"/>
2490         <xsd:enumeration value="stdDevp"/>
2491         <xsd:enumeration value="sum"/>
2492         <xsd:enumeration value="var"/>
2493         <xsd:enumeration value="varp"/>
2494     </xsd:restriction>
2495 </xsd:simpleType>
2496 <xsd:complexType name="CT_DataRefs">
2497     <xsd:sequence>
2498         <xsd:element name="dataRef" type="CT_DataRef" minOccurs="0" maxOccurs="unbounded"/>
2499     </xsd:sequence>
2500     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2501 </xsd:complexType>
2502 <xsd:complexType name="CT_DataRef">
2503     <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
2504     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
2505     <xsd:attribute name="sheet" type="s:ST_Xstring" use="optional"/>
2506     <xsd:attribute ref="r:id" use="optional"/>
2507 </xsd:complexType>
2508 <xsd:complexType name="CT_MergeCells">
2509     <xsd:sequence>
2510         <xsd:element name="mergeCell" type="CT_MergeCell" minOccurs="1" maxOccurs="unbounded"/>
2511     </xsd:sequence>
2512     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2513 </xsd:complexType>
2514 <xsd:complexType name="CT_MergeCell">
2515     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2516 </xsd:complexType>
2517 <xsd:complexType name="CT_SmartTags">
2518     <xsd:sequence>

```

```

2519     <xsd:element name="cellSmartTags" type="CT_CellSmartTags" minOccurs="1"
2520       maxOccurs="unbounded"/>
2521   </xsd:sequence>
2522 </xsd:complexType>
2523 <xsd:complexType name="CT_CellSmartTags">
2524   <xsd:sequence>
2525     <xsd:element name="cellSmartTag" type="CT_CellSmartTag" minOccurs="1"
2526       maxOccurs="unbounded"/>
2527   </xsd:sequence>
2528   <xsd:attribute name="r" type="ST_CellRef" use="required"/>
2529 </xsd:complexType>
2530 <xsd:complexType name="CT_CellSmartTag">
2531   <xsd:sequence>
2532     <xsd:element name="cellSmartTagPr" minOccurs="0" maxOccurs="unbounded"
2533       type="CT_CellSmartTagPr"/>
2534   </xsd:sequence>
2535   <xsd:attribute name="type" type="xsd:unsignedInt" use="required"/>
2536   <xsd:attribute name="deleted" type="xsd:boolean" use="optional" default="false"/>
2537   <xsd:attribute name="xmlBased" type="xsd:boolean" use="optional" default="false"/>
2538 </xsd:complexType>
2539 <xsd:complexType name="CT_CellSmartTagPr">
2540   <xsd:attribute name="key" type="s:ST_Xstring" use="required"/>
2541   <xsd:attribute name="val" type="s:ST_Xstring" use="required"/>
2542 </xsd:complexType>
2543 <xsd:complexType name="CT_Drawing">
2544   <xsd:attribute ref="r:id" use="required"/>
2545 </xsd:complexType>
2546 <xsd:complexType name="CT_LegacyDrawing">
2547   <xsd:attribute ref="r:id" use="required"/>
2548 </xsd:complexType>
2549 <xsd:complexType name="CT_DrawingHF">
2550   <xsd:attribute ref="r:id" use="required"/>
2551   <xsd:attribute name="lho" type="xsd:unsignedInt" use="optional"/>
2552   <xsd:attribute name="lhe" type="xsd:unsignedInt" use="optional"/>
2553   <xsd:attribute name="lhf" type="xsd:unsignedInt" use="optional"/>
2554   <xsd:attribute name="cho" type="xsd:unsignedInt" use="optional"/>
2555   <xsd:attribute name="che" type="xsd:unsignedInt" use="optional"/>
2556   <xsd:attribute name="chf" type="xsd:unsignedInt" use="optional"/>
2557   <xsd:attribute name="rho" type="xsd:unsignedInt" use="optional"/>
2558   <xsd:attribute name="rhe" type="xsd:unsignedInt" use="optional"/>
2559   <xsd:attribute name="rhf" type="xsd:unsignedInt" use="optional"/>
2560   <xsd:attribute name="lfo" type="xsd:unsignedInt" use="optional"/>
2561   <xsd:attribute name="lfe" type="xsd:unsignedInt" use="optional"/>
2562   <xsd:attribute name="lff" type="xsd:unsignedInt" use="optional"/>
2563   <xsd:attribute name="cfo" type="xsd:unsignedInt" use="optional"/>
2564   <xsd:attribute name="cfe" type="xsd:unsignedInt" use="optional"/>
2565   <xsd:attribute name="cff" type="xsd:unsignedInt" use="optional"/>
2566   <xsd:attribute name="rfo" type="xsd:unsignedInt" use="optional"/>
2567   <xsd:attribute name="rfe" type="xsd:unsignedInt" use="optional"/>
2568   <xsd:attribute name="rff" type="xsd:unsignedInt" use="optional"/>
2569 </xsd:complexType>
2570 <xsd:complexType name="CT_CustomSheetViews">
2571   <xsd:sequence>

```

```

2572     <xsd:element name="customSheetView" minOccurs="1" maxOccurs="unbounded"
2573         type="CT_CustomSheetView"/>
2574 </xsd:sequence>
2575 </xsd:complexType>
2576 <xsd:complexType name="CT_CustomSheetView">
2577     <xsd:sequence>
2578         <xsd:element name="pane" type="CT_Pane" minOccurs="0" maxOccurs="1"/>
2579         <xsd:element name="selection" type="CT_Selection" minOccurs="0" maxOccurs="1"/>
2580         <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2581         <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2582         <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2583         <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2584         <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2585         <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2586         <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
2587         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2588     </xsd:sequence>
2589     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
2590     <xsd:attribute name="scale" type="xsd:unsignedInt" default="100"/>
2591     <xsd:attribute name="colorId" type="xsd:unsignedInt" default="64"/>
2592     <xsd:attribute name="showPageBreaks" type="xsd:boolean" use="optional" default="false"/>
2593     <xsd:attribute name="showFormulas" type="xsd:boolean" use="optional" default="false"/>
2594     <xsd:attribute name="showGridLines" type="xsd:boolean" use="optional" default="true"/>
2595     <xsd:attribute name="showRowCol" type="xsd:boolean" use="optional" default="true"/>
2596     <xsd:attribute name="outlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2597     <xsd:attribute name="zeroValues" type="xsd:boolean" use="optional" default="true"/>
2598     <xsd:attribute name="fitToPage" type="xsd:boolean" use="optional" default="false"/>
2599     <xsd:attribute name="printArea" type="xsd:boolean" use="optional" default="false"/>
2600     <xsd:attribute name="filter" type="xsd:boolean" use="optional" default="false"/>
2601     <xsd:attribute name="showAutoFilter" type="xsd:boolean" use="optional" default="false"/>
2602     <xsd:attribute name="hiddenRows" type="xsd:boolean" use="optional" default="false"/>
2603     <xsd:attribute name="hiddenColumns" type="xsd:boolean" use="optional" default="false"/>
2604     <xsd:attribute name="state" type="ST_SheetState" default="visible"/>
2605     <xsd:attribute name="filterUnique" type="xsd:boolean" use="optional" default="false"/>
2606     <xsd:attribute name="view" type="ST_SheetViewType" default="normal"/>
2607     <xsd:attribute name="showRuler" type="xsd:boolean" use="optional" default="true"/>
2608     <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2609 </xsd:complexType>
2610 <xsd:complexType name="CT_DataValidations">
2611     <xsd:sequence>
2612         <xsd:element name="dataValidation" type="CT_DataValidation" minOccurs="1"
2613             maxOccurs="unbounded"/>
2614     </xsd:sequence>
2615     <xsd:attribute name="disablePrompts" type="xsd:boolean" use="optional" default="false"/>
2616     <xsd:attribute name="xWindow" type="xsd:unsignedInt" use="optional"/>
2617     <xsd:attribute name="yWindow" type="xsd:unsignedInt" use="optional"/>
2618     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2619 </xsd:complexType>
2620 <xsd:complexType name="CT_DataValidation">
2621     <xsd:sequence>
2622         <xsd:element name="formula1" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2623         <xsd:element name="formula2" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2624     </xsd:sequence>

```

```

2625 <xsd:attribute name="type" type="ST_DataValidationType" use="optional" default="none"/>
2626 <xsd:attribute name="errorStyle" type="ST_DataValidationErrorStyle" use="optional"
2627     default="stop"/>
2628 <xsd:attribute name="imeMode" type="ST_DataValidationImeMode" use="optional"
2629     default="noControl"/>
2630 <xsd:attribute name="operator" type="ST_DataValidationOperator" use="optional"
2631     default="between"/>
2632 <xsd:attribute name="allowBlank" type="xsd:boolean" use="optional" default="false"/>
2633 <xsd:attribute name="showDropDown" type="xsd:boolean" use="optional" default="false"/>
2634 <xsd:attribute name="showInputMessage" type="xsd:boolean" use="optional" default="false"/>
2635 <xsd:attribute name="showErrorMessage" type="xsd:boolean" use="optional" default="false"/>
2636 <xsd:attribute name="errorTitle" type="s:ST_Xstring" use="optional"/>
2637 <xsd:attribute name="error" type="s:ST_Xstring" use="optional"/>
2638 <xsd:attribute name="promptTitle" type="s:ST_Xstring" use="optional"/>
2639 <xsd:attribute name="prompt" type="s:ST_Xstring" use="optional"/>
2640 <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
2641 </xsd:complexType>
2642 <xsd:simpleType name="ST_DataValidationType">
2643     <xsd:restriction base="xsd:string">
2644         <xsd:enumeration value="none"/>
2645         <xsd:enumeration value="whole"/>
2646         <xsd:enumeration value="decimal"/>
2647         <xsd:enumeration value="list"/>
2648         <xsd:enumeration value="date"/>
2649         <xsd:enumeration value="time"/>
2650         <xsd:enumeration value="textLength"/>
2651         <xsd:enumeration value="custom"/>
2652     </xsd:restriction>
2653 </xsd:simpleType>
2654 <xsd:simpleType name="ST_DataValidationOperator">
2655     <xsd:restriction base="xsd:string">
2656         <xsd:enumeration value="between"/>
2657         <xsd:enumeration value="notBetween"/>
2658         <xsd:enumeration value="equal"/>
2659         <xsd:enumeration value="notEqual"/>
2660         <xsd:enumeration value="lessThan"/>
2661         <xsd:enumeration value="lessThanOrEqual"/>
2662         <xsd:enumeration value="greaterThan"/>
2663         <xsd:enumeration value="greaterThanOrEqual"/>
2664     </xsd:restriction>
2665 </xsd:simpleType>
2666 <xsd:simpleType name="ST_DataValidationErrorStyle">
2667     <xsd:restriction base="xsd:string">
2668         <xsd:enumeration value="stop"/>
2669         <xsd:enumeration value="warning"/>
2670         <xsd:enumeration value="information"/>
2671     </xsd:restriction>
2672 </xsd:simpleType>
2673 <xsd:simpleType name="ST_DataValidationImeMode">
2674     <xsd:restriction base="xsd:string">
2675         <xsd:enumeration value="noControl"/>
2676         <xsd:enumeration value="off"/>
2677         <xsd:enumeration value="on"/>

```

```

2678         <xsd:enumeration value="disabled"/>
2679         <xsd:enumeration value="hiragana"/>
2680         <xsd:enumeration value="fullKatakana"/>
2681         <xsd:enumeration value="halfKatakana"/>
2682         <xsd:enumeration value="fullAlpha"/>
2683         <xsd:enumeration value="halfAlpha"/>
2684         <xsd:enumeration value="fullHangul"/>
2685         <xsd:enumeration value="halfHangul"/>
2686     </xsd:restriction>
2687 </xsd:simpleType>
2688 <xsd:simpleType name="ST_CfType">
2689     <xsd:restriction base="xsd:string">
2690         <xsd:enumeration value="expression"/>
2691         <xsd:enumeration value="cellIs"/>
2692         <xsd:enumeration value="colorScale"/>
2693         <xsd:enumeration value="dataBar"/>
2694         <xsd:enumeration value="iconSet"/>
2695         <xsd:enumeration value="top10"/>
2696         <xsd:enumeration value="uniqueValues"/>
2697         <xsd:enumeration value="duplicateValues"/>
2698         <xsd:enumeration value="containsText"/>
2699         <xsd:enumeration value="notContainsText"/>
2700         <xsd:enumeration value="beginsWith"/>
2701         <xsd:enumeration value="endsWith"/>
2702         <xsd:enumeration value="containsBlanks"/>
2703         <xsd:enumeration value="notContainsBlanks"/>
2704         <xsd:enumeration value="containsErrors"/>
2705         <xsd:enumeration value="notContainsErrors"/>
2706         <xsd:enumeration value="timePeriod"/>
2707         <xsd:enumeration value="aboveAverage"/>
2708     </xsd:restriction>
2709 </xsd:simpleType>
2710 <xsd:simpleType name="ST_TimePeriod">
2711     <xsd:restriction base="xsd:string">
2712         <xsd:enumeration value="today"/>
2713         <xsd:enumeration value="yesterday"/>
2714         <xsd:enumeration value="tomorrow"/>
2715         <xsd:enumeration value="last7Days"/>
2716         <xsd:enumeration value="thisMonth"/>
2717         <xsd:enumeration value="lastMonth"/>
2718         <xsd:enumeration value="nextMonth"/>
2719         <xsd:enumeration value="thisWeek"/>
2720         <xsd:enumeration value="lastWeek"/>
2721         <xsd:enumeration value="nextWeek"/>
2722     </xsd:restriction>
2723 </xsd:simpleType>
2724 <xsd:simpleType name="ST_ConditionalFormattingOperator">
2725     <xsd:restriction base="xsd:string">
2726         <xsd:enumeration value="lessThan"/>
2727         <xsd:enumeration value="lessThanOrEqual"/>
2728         <xsd:enumeration value="equal"/>
2729         <xsd:enumeration value="notEqual"/>
2730         <xsd:enumeration value="greaterThanOrEqual"/>

```

```

2731         <xsd:enumeration value="greaterThan"/>
2732         <xsd:enumeration value="between"/>
2733         <xsd:enumeration value="notBetween"/>
2734         <xsd:enumeration value="containsText"/>
2735         <xsd:enumeration value="notContains"/>
2736         <xsd:enumeration value="beginsWith"/>
2737         <xsd:enumeration value="endsWith"/>
2738     </xsd:restriction>
2739 </xsd:simpleType>
2740 <xsd:simpleType name="ST_CfvoType">
2741     <xsd:restriction base="xsd:string">
2742         <xsd:enumeration value="num"/>
2743         <xsd:enumeration value="percent"/>
2744         <xsd:enumeration value="max"/>
2745         <xsd:enumeration value="min"/>
2746         <xsd:enumeration value="formula"/>
2747         <xsd:enumeration value="percentile"/>
2748     </xsd:restriction>
2749 </xsd:simpleType>
2750 <xsd:complexType name="CT_ConditionalFormatting">
2751     <xsd:sequence>
2752         <xsd:element name="cfRule" type="CT_CfRule" minOccurs="1" maxOccurs="unbounded"/>
2753         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2754     </xsd:sequence>
2755     <xsd:attribute name="pivot" type="xsd:boolean" default="false"/>
2756     <xsd:attribute name="sqref" type="ST_Sqref"/>
2757 </xsd:complexType>
2758 <xsd:complexType name="CT_CfRule">
2759     <xsd:sequence>
2760         <xsd:element name="formula" type="ST_Formula" minOccurs="0" maxOccurs="3"/>
2761         <xsd:element name="colorScale" type="CT_ColorScale" minOccurs="0" maxOccurs="1"/>
2762         <xsd:element name="dataBar" type="CT_DataBar" minOccurs="0" maxOccurs="1"/>
2763         <xsd:element name="iconSet" type="CT_IconSet" minOccurs="0" maxOccurs="1"/>
2764         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2765     </xsd:sequence>
2766     <xsd:attribute name="type" type="ST_CfType"/>
2767     <xsd:attribute name="dxfId" type="ST_DxfId" use="optional"/>
2768     <xsd:attribute name="priority" type="xsd:int" use="required"/>
2769     <xsd:attribute name="stopIfTrue" type="xsd:boolean" use="optional" default="false"/>
2770     <xsd:attribute name="aboveAverage" type="xsd:boolean" use="optional" default="true"/>
2771     <xsd:attribute name="percent" type="xsd:boolean" use="optional" default="false"/>
2772     <xsd:attribute name="bottom" type="xsd:boolean" use="optional" default="false"/>
2773     <xsd:attribute name="operator" type="ST_ConditionalFormattingOperator" use="optional"/>
2774     <xsd:attribute name="text" type="xsd:string" use="optional"/>
2775     <xsd:attribute name="timePeriod" type="ST_TimePeriod" use="optional"/>
2776     <xsd:attribute name="rank" type="xsd:unsignedInt" use="optional"/>
2777     <xsd:attribute name="stdDev" type="xsd:int" use="optional"/>
2778     <xsd:attribute name="equalAverage" type="xsd:boolean" use="optional" default="false"/>
2779 </xsd:complexType>
2780 <xsd:complexType name="CT_Hyperlinks">
2781     <xsd:sequence>
2782         <xsd:element name="hyperlink" type="CT_Hyperlink" minOccurs="1" maxOccurs="unbounded"/>
2783     </xsd:sequence>

```

```

2784 </xsd:complexType>
2785 <xsd:complexType name="CT_Hyperlink">
2786   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2787   <xsd:attribute ref="r:id" use="optional"/>
2788   <xsd:attribute name="location" type="s:ST_Xstring" use="optional"/>
2789   <xsd:attribute name="tooltip" type="s:ST_Xstring" use="optional"/>
2790   <xsd:attribute name="display" type="s:ST_Xstring" use="optional"/>
2791 </xsd:complexType>
2792 <xsd:complexType name="CT_CellFormula">
2793   <xsd:simpleContent>
2794     <xsd:extension base="ST_Formula">
2795       <xsd:attribute name="t" type="ST_CellFormulaType" use="optional" default="normal"/>
2796       <xsd:attribute name="aca" type="xsd:boolean" use="optional" default="false"/>
2797       <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
2798       <xsd:attribute name="dt2D" type="xsd:boolean" use="optional" default="false"/>
2799       <xsd:attribute name="dtr" type="xsd:boolean" use="optional" default="false"/>
2800       <xsd:attribute name="del1" type="xsd:boolean" use="optional" default="false"/>
2801       <xsd:attribute name="del2" type="xsd:boolean" use="optional" default="false"/>
2802       <xsd:attribute name="r1" type="ST_CellRef" use="optional"/>
2803       <xsd:attribute name="r2" type="ST_CellRef" use="optional"/>
2804       <xsd:attribute name="ca" type="xsd:boolean" use="optional" default="false"/>
2805       <xsd:attribute name="si" type="xsd:unsignedInt" use="optional"/>
2806       <xsd:attribute name="bx" type="xsd:boolean" use="optional" default="false"/>
2807     </xsd:extension>
2808   </xsd:simpleContent>
2809 </xsd:complexType>
2810 <xsd:complexType name="CT_ColorScale">
2811   <xsd:sequence>
2812     <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="unbounded"/>
2813     <xsd:element name="color" type="CT_Color" minOccurs="2" maxOccurs="unbounded"/>
2814   </xsd:sequence>
2815 </xsd:complexType>
2816 <xsd:complexType name="CT_DataBar">
2817   <xsd:sequence>
2818     <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="2"/>
2819     <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="1"/>
2820   </xsd:sequence>
2821   <xsd:attribute name="minLength" type="xsd:unsignedInt" use="optional" default="10"/>
2822   <xsd:attribute name="maxLength" type="xsd:unsignedInt" use="optional" default="90"/>
2823   <xsd:attribute name="showValue" type="xsd:boolean" use="optional" default="true"/>
2824 </xsd:complexType>
2825 <xsd:complexType name="CT_IconSet">
2826   <xsd:sequence>
2827     <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="unbounded"/>
2828   </xsd:sequence>
2829   <xsd:attribute name="iconSet" type="ST_IconSetType" use="optional" default="3TrafficLights1"/>
2830   <xsd:attribute name="showValue" type="xsd:boolean" use="optional" default="true"/>
2831   <xsd:attribute name="percent" type="xsd:boolean" default="true"/>
2832   <xsd:attribute name="reverse" type="xsd:boolean" use="optional" default="false"/>
2833 </xsd:complexType>
2834 <xsd:complexType name="CT_Cfvo">
2835   <xsd:sequence>
2836     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

2837     </xsd:sequence>
2838     <xsd:attribute name="type" type="ST_CfvoType" use="required"/>
2839     <xsd:attribute name="val" type="s:ST_Xstring" use="optional"/>
2840     <xsd:attribute name="gte" type="xsd:boolean" use="optional" default="true"/>
2841 </xsd:complexType>
2842 <xsd:complexType name="CT_PageMargins">
2843     <xsd:attribute name="left" type="xsd:double" use="required"/>
2844     <xsd:attribute name="right" type="xsd:double" use="required"/>
2845     <xsd:attribute name="top" type="xsd:double" use="required"/>
2846     <xsd:attribute name="bottom" type="xsd:double" use="required"/>
2847     <xsd:attribute name="header" type="xsd:double" use="required"/>
2848     <xsd:attribute name="footer" type="xsd:double" use="required"/>
2849 </xsd:complexType>
2850 <xsd:complexType name="CT_PrintOptions">
2851     <xsd:attribute name="horizontalCentered" type="xsd:boolean" use="optional" default="false"/>
2852     <xsd:attribute name="verticalCentered" type="xsd:boolean" use="optional" default="false"/>
2853     <xsd:attribute name="headings" type="xsd:boolean" use="optional" default="false"/>
2854     <xsd:attribute name="gridLines" type="xsd:boolean" use="optional" default="false"/>
2855     <xsd:attribute name="gridLinesSet" type="xsd:boolean" use="optional" default="true"/>
2856 </xsd:complexType>
2857 <xsd:complexType name="CT_PageSetup">
2858     <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>
2859     <xsd:attribute name="paperHeight" type="s:ST_PositiveUniversalMeasure" use="optional"/>
2860     <xsd:attribute name="paperWidth" type="s:ST_PositiveUniversalMeasure" use="optional"/>
2861     <xsd:attribute name="scale" type="xsd:unsignedInt" use="optional" default="100"/>
2862     <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
2863     <xsd:attribute name="fitToWidth" type="xsd:unsignedInt" use="optional" default="1"/>
2864     <xsd:attribute name="fitToHeight" type="xsd:unsignedInt" use="optional" default="1"/>
2865     <xsd:attribute name="pageOrder" type="ST_PageOrder" use="optional" default="downThenOver"/>
2866     <xsd:attribute name="orientation" type="ST_Orientation" use="optional" default="default"/>
2867     <xsd:attribute name="usePrinterDefaults" type="xsd:boolean" use="optional" default="true"/>
2868     <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
2869     <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
2870     <xsd:attribute name="cellComments" type="ST_CellComments" use="optional" default="none"/>
2871     <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
2872     <xsd:attribute name="errors" type="ST_PrintError" use="optional" default="displayed"/>
2873     <xsd:attribute name="horizontalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
2874     <xsd:attribute name="verticalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
2875     <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
2876     <xsd:attribute ref="r:id" use="optional"/>
2877 </xsd:complexType>
2878 <xsd:simpleType name="ST_PageOrder">
2879     <xsd:restriction base="xsd:string">
2880         <xsd:enumeration value="downThenOver"/>
2881         <xsd:enumeration value="overThenDown"/>
2882     </xsd:restriction>
2883 </xsd:simpleType>
2884 <xsd:simpleType name="ST_Orientation">
2885     <xsd:restriction base="xsd:string">
2886         <xsd:enumeration value="default"/>
2887         <xsd:enumeration value="portrait"/>
2888         <xsd:enumeration value="landscape"/>
2889     </xsd:restriction>

```



```

2890 </xsd:simpleType>
2891 <xsd:simpleType name="ST_CellComments">
2892   <xsd:restriction base="xsd:string">
2893     <xsd:enumeration value="none"/>
2894     <xsd:enumeration value="asDisplayed"/>
2895     <xsd:enumeration value="atEnd"/>
2896   </xsd:restriction>
2897 </xsd:simpleType>
2898 <xsd:complexType name="CT_HeaderFooter">
2899   <xsd:sequence>
2900     <xsd:element name="oddHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2901     <xsd:element name="oddFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2902     <xsd:element name="evenHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2903     <xsd:element name="evenFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2904     <xsd:element name="firstHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2905     <xsd:element name="firstFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2906   </xsd:sequence>
2907   <xsd:attribute name="differentOddEven" type="xsd:boolean" default="false"/>
2908   <xsd:attribute name="differentFirst" type="xsd:boolean" default="false"/>
2909   <xsd:attribute name="scaleWithDoc" type="xsd:boolean" default="true"/>
2910   <xsd:attribute name="alignWithMargins" type="xsd:boolean" default="true"/>
2911 </xsd:complexType>
2912 <xsd:simpleType name="ST_PrintError">
2913   <xsd:restriction base="xsd:string">
2914     <xsd:enumeration value="displayed"/>
2915     <xsd:enumeration value="blank"/>
2916     <xsd:enumeration value="dash"/>
2917     <xsd:enumeration value="NA"/>
2918   </xsd:restriction>
2919 </xsd:simpleType>
2920 <xsd:complexType name="CT_Scenarios">
2921   <xsd:sequence>
2922     <xsd:element name="scenario" type="CT_Scenario" minOccurs="1" maxOccurs="unbounded"/>
2923   </xsd:sequence>
2924   <xsd:attribute name="current" type="xsd:unsignedInt" use="optional"/>
2925   <xsd:attribute name="show" type="xsd:unsignedInt" use="optional"/>
2926   <xsd:attribute name="sqref" type="ST_Sqref" use="optional"/>
2927 </xsd:complexType>
2928 <xsd:complexType name="CT_SheetProtection">
2929   <xsd:attribute name="password" type="ST_UnsignedShortHex" use="optional"/>
2930   <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
2931   <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
2932   <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
2933   <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
2934   <xsd:attribute name="sheet" type="xsd:boolean" use="optional" default="false"/>
2935   <xsd:attribute name="objects" type="xsd:boolean" use="optional" default="false"/>
2936   <xsd:attribute name="scenarios" type="xsd:boolean" use="optional" default="false"/>
2937   <xsd:attribute name="formatCells" type="xsd:boolean" use="optional" default="true"/>
2938   <xsd:attribute name="formatColumns" type="xsd:boolean" use="optional" default="true"/>
2939   <xsd:attribute name="formatRows" type="xsd:boolean" use="optional" default="true"/>
2940   <xsd:attribute name="insertColumns" type="xsd:boolean" use="optional" default="true"/>
2941   <xsd:attribute name="insertRows" type="xsd:boolean" use="optional" default="true"/>
2942   <xsd:attribute name="insertHyperlinks" type="xsd:boolean" use="optional" default="true"/>

```

```

2943     <xsd:attribute name="deleteColumns" type="xsd:boolean" use="optional" default="true"/>
2944     <xsd:attribute name="deleteRows" type="xsd:boolean" use="optional" default="true"/>
2945     <xsd:attribute name="selectLockedCells" type="xsd:boolean" use="optional" default="false"/>
2946     <xsd:attribute name="sort" type="xsd:boolean" use="optional" default="true"/>
2947     <xsd:attribute name="autoFilter" type="xsd:boolean" use="optional" default="true"/>
2948     <xsd:attribute name="pivotTables" type="xsd:boolean" use="optional" default="true"/>
2949     <xsd:attribute name="selectUnlockedCells" type="xsd:boolean" use="optional" default="false"/>
2950 </xsd:complexType>
2951 <xsd:complexType name="CT_ProtectedRanges">
2952     <xsd:sequence>
2953         <xsd:element name="protectedRange" type="CT_ProtectedRange" minOccurs="1"
2954             maxOccurs="unbounded"/>
2955     </xsd:sequence>
2956 </xsd:complexType>
2957 <xsd:complexType name="CT_ProtectedRange">
2958     <xsd:sequence>
2959         <xsd:element name="securityDescriptor" type="xsd:string" minOccurs="0"
2960             maxOccurs="unbounded"/>
2961     </xsd:sequence>
2962     <xsd:attribute name="password" type="ST UnsignedShortHex" use="optional"/>
2963     <xsd:attribute name="sqref" type="ST Sqref" use="required"/>
2964     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
2965     <xsd:attribute name="securityDescriptor" type="xsd:string" use="optional"/>
2966     <xsd:attribute name="algorithmName" type="s:ST Xstring" use="optional"/>
2967     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
2968     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
2969     <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
2970 </xsd:complexType>
2971 <xsd:complexType name="CT_Scenario">
2972     <xsd:sequence>
2973         <xsd:element name="inputCells" type="CT_InputCells" minOccurs="1" maxOccurs="unbounded"/>
2974     </xsd:sequence>
2975     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
2976     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="false"/>
2977     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2978     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2979     <xsd:attribute name="user" type="s:ST Xstring" use="optional"/>
2980     <xsd:attribute name="comment" type="s:ST Xstring" use="optional"/>
2981 </xsd:complexType>
2982 <xsd:complexType name="CT_InputCells">
2983     <xsd:attribute name="r" type="ST CellRef" use="required"/>
2984     <xsd:attribute name="deleted" type="xsd:boolean" use="optional" default="false"/>
2985     <xsd:attribute name="undone" type="xsd:boolean" use="optional" default="false"/>
2986     <xsd:attribute name="val" type="s:ST Xstring" use="required"/>
2987     <xsd:attribute name="numFmtId" type="ST NumFmtId" use="optional"/>
2988 </xsd:complexType>
2989 <xsd:complexType name="CT_CellWatches">
2990     <xsd:sequence>
2991         <xsd:element name="cellWatch" type="CT_CellWatch" minOccurs="1" maxOccurs="unbounded"/>
2992     </xsd:sequence>
2993 </xsd:complexType>
2994 <xsd:complexType name="CT_CellWatch">
2995     <xsd:attribute name="r" type="ST CellRef" use="required"/>

```

```

2996 </xsd:complexType>
2997 <xsd:complexType name="CT_Chartsheet">
2998   <xsd:sequence>
2999     <xsd:element name="sheetPr" type="CT_ChartsheetPr" minOccurs="0" maxOccurs="1"/>
3000     <xsd:element name="sheetViews" type="CT_ChartsheetViews" minOccurs="1" maxOccurs="1"/>
3001     <xsd:element name="sheetProtection" type="CT_ChartsheetProtection" minOccurs="0"
3002       maxOccurs="1"/>
3003     <xsd:element name="customSheetViews" type="CT_CustomChartsheetViews" minOccurs="0"
3004       maxOccurs="1"/>
3005     <xsd:element name="pageMargins" minOccurs="0" type="CT_PageMargins"/>
3006     <xsd:element name="pageSetup" type="CT_CsPageSetup" minOccurs="0" maxOccurs="1"/>
3007     <xsd:element name="headerFooter" minOccurs="0" type="CT_HeaderFooter"/>
3008     <xsd:element name="drawing" type="CT_Drawing" minOccurs="1" maxOccurs="1"/>
3009     <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
3010     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
3011     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
3012     <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
3013     <xsd:element name="webPublishItems" type="CT_WebPublishItems" minOccurs="0"
3014       maxOccurs="1"/>
3015     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3016   </xsd:sequence>
3017 </xsd:complexType>
3018 <xsd:complexType name="CT_ChartsheetPr">
3019   <xsd:sequence>
3020     <xsd:element name="tabColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3021   </xsd:sequence>
3022   <xsd:attribute name="published" type="xsd:boolean" use="optional" default="true"/>
3023   <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
3024 </xsd:complexType>
3025 <xsd:complexType name="CT_ChartsheetViews">
3026   <xsd:sequence>
3027     <xsd:element name="sheetView" type="CT_ChartsheetView" minOccurs="1"
3028       maxOccurs="unbounded"/>
3029     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3030   </xsd:sequence>
3031 </xsd:complexType>
3032 <xsd:complexType name="CT_ChartsheetView">
3033   <xsd:sequence>
3034     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3035   </xsd:sequence>
3036   <xsd:attribute name="tabSelected" type="xsd:boolean" use="optional" default="false"/>
3037   <xsd:attribute name="zoomScale" type="xsd:unsignedInt" default="100" use="optional"/>
3038   <xsd:attribute name="workbookViewId" type="xsd:unsignedInt" use="required"/>
3039   <xsd:attribute name="zoomToFit" type="xsd:boolean" use="optional" default="false"/>
3040 </xsd:complexType>
3041 <xsd:complexType name="CT_ChartsheetProtection">
3042   <xsd:attribute name="password" type="ST_UnsignedShortHex" use="optional"/>
3043   <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
3044   <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
3045   <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
3046   <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
3047   <xsd:attribute name="content" type="xsd:boolean" use="optional" default="false"/>
3048   <xsd:attribute name="objects" type="xsd:boolean" use="optional" default="false"/>

```

```

3049 </xsd:complexType>
3050 <xsd:complexType name="CT_CsPageSetup">
3051     <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>
3052     <xsd:attribute name="paperHeight" type="s:ST_PositiveUniversalMeasure" use="optional"/>
3053     <xsd:attribute name="paperWidth" type="s:ST_PositiveUniversalMeasure" use="optional"/>
3054     <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
3055     <xsd:attribute name="orientation" type="ST_Orientation" use="optional" default="default"/>
3056     <xsd:attribute name="usePrinterDefaults" type="xsd:boolean" use="optional" default="true"/>
3057     <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
3058     <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
3059     <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
3060     <xsd:attribute name="horizontalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
3061     <xsd:attribute name="verticalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
3062     <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
3063     <xsd:attribute ref="r:id" use="optional"/>
3064 </xsd:complexType>
3065 <xsd:complexType name="CT_CustomChartsheetViews">
3066     <xsd:sequence>
3067         <xsd:element name="customSheetView" minOccurs="0" maxOccurs="unbounded"
3068             type="CT_CustomChartsheetView"/>
3069     </xsd:sequence>
3070 </xsd:complexType>
3071 <xsd:complexType name="CT_CustomChartsheetView">
3072     <xsd:sequence>
3073         <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
3074         <xsd:element name="pageSetup" type="CT_CsPageSetup" minOccurs="0" maxOccurs="1"/>
3075         <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
3076     </xsd:sequence>
3077     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
3078     <xsd:attribute name="scale" type="xsd:unsignedInt" default="100"/>
3079     <xsd:attribute name="state" type="ST_SheetState" default="visible"/>
3080     <xsd:attribute name="zoomToFit" type="xsd:boolean" use="optional" default="false"/>
3081 </xsd:complexType>
3082 <xsd:complexType name="CT_CustomProperties">
3083     <xsd:sequence>
3084         <xsd:element name="customPr" type="CT_CustomProperty" minOccurs="1"
3085             maxOccurs="unbounded"/>
3086     </xsd:sequence>
3087 </xsd:complexType>
3088 <xsd:complexType name="CT_CustomProperty">
3089     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3090     <xsd:attribute ref="r:id" use="required"/>
3091 </xsd:complexType>
3092 <xsd:complexType name="CT_OleObjects">
3093     <xsd:sequence>
3094         <xsd:element name="oleObject" type="CT_OleObject" minOccurs="1" maxOccurs="unbounded"/>
3095     </xsd:sequence>
3096 </xsd:complexType>
3097 <xsd:complexType name="CT_OleObject">
3098     <xsd:sequence>
3099         <xsd:element name="objectPr" type="CT_ObjectPr" minOccurs="0" maxOccurs="1"/>
3100     </xsd:sequence>
3101     <xsd:attribute name="progId" type="xsd:string" use="optional"/>

```

```

3102     <xsd:attribute name="dvAspect" type="ST_DvAspect" use="optional" default="DVASPECT_CONTENT"/>
3103     <xsd:attribute name="link" type="s:ST_Xstring" use="optional"/>
3104     <xsd:attribute name="oleUpdate" type="ST_OleUpdate" use="optional"/>
3105     <xsd:attribute name="autoLoad" type="xsd:boolean" use="optional" default="false"/>
3106     <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="required"/>
3107     <xsd:attribute ref="r:id" use="optional"/>
3108 </xsd:complexType>
3109 <xsd:complexType name="CT_ObjectPr">
3110     <xsd:sequence>
3111         <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
3112     </xsd:sequence>
3113     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
3114     <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
3115     <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
3116     <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
3117     <xsd:attribute name="uiObject" type="xsd:boolean" use="optional" default="false"/>
3118     <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
3119     <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
3120     <xsd:attribute name="autoPict" type="xsd:boolean" use="optional" default="true"/>
3121     <xsd:attribute name="macro" type="ST_Formula" use="optional"/>
3122     <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
3123     <xsd:attribute name="dde" type="xsd:boolean" use="optional" default="false"/>
3124     <xsd:attribute ref="r:id" use="optional"/>
3125 </xsd:complexType>
3126 <xsd:simpleType name="ST_DvAspect">
3127     <xsd:restriction base="xsd:string">
3128         <xsd:enumeration value="DVASPECT_CONTENT"/>
3129         <xsd:enumeration value="DVASPECT_ICON"/>
3130     </xsd:restriction>
3131 </xsd:simpleType>
3132 <xsd:simpleType name="ST_OleUpdate">
3133     <xsd:restriction base="xsd:string">
3134         <xsd:enumeration value="OLEUPDATE_ALWAYS"/>
3135         <xsd:enumeration value="OLEUPDATE_ONCALL"/>
3136     </xsd:restriction>
3137 </xsd:simpleType>
3138 <xsd:complexType name="CT_WebPublishItems">
3139     <xsd:sequence>
3140         <xsd:element name="webPublishItem" type="CT_WebPublishItem" minOccurs="1"
3141             maxOccurs="unbounded"/>
3142     </xsd:sequence>
3143     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3144 </xsd:complexType>
3145 <xsd:complexType name="CT_WebPublishItem">
3146     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3147     <xsd:attribute name="divId" type="s:ST_Xstring" use="required"/>
3148     <xsd:attribute name="sourceType" type="ST_WebSourceType" use="required"/>
3149     <xsd:attribute name="sourceRef" type="ST_Ref" use="optional"/>
3150     <xsd:attribute name="sourceObject" type="s:ST_Xstring" use="optional"/>
3151     <xsd:attribute name="destinationFile" type="s:ST_Xstring" use="required"/>
3152     <xsd:attribute name="title" type="s:ST_Xstring" use="optional"/>
3153     <xsd:attribute name="autoRepublish" type="xsd:boolean" use="optional" default="false"/>
3154 </xsd:complexType>

```

```

3155 <xsd:complexType name="CT_Controls">
3156   <xsd:sequence>
3157     <xsd:element name="control" type="CT_Control" minOccurs="1" maxOccurs="unbounded"/>
3158   </xsd:sequence>
3159 </xsd:complexType>
3160 <xsd:complexType name="CT_Control">
3161   <xsd:sequence>
3162     <xsd:element name="controlPr" type="CT_ControlPr" minOccurs="0" maxOccurs="1"/>
3163   </xsd:sequence>
3164   <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="required"/>
3165   <xsd:attribute ref="r:id" use="required"/>
3166   <xsd:attribute name="name" type="xsd:string" use="optional"/>
3167 </xsd:complexType>
3168 <xsd:complexType name="CT_ControlPr">
3169   <xsd:sequence>
3170     <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
3171   </xsd:sequence>
3172   <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
3173   <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
3174   <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
3175   <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
3176   <xsd:attribute name="recalcAlways" type="xsd:boolean" use="optional" default="false"/>
3177   <xsd:attribute name="uiObject" type="xsd:boolean" use="optional" default="false"/>
3178   <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
3179   <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
3180   <xsd:attribute name="autoPict" type="xsd:boolean" use="optional" default="true"/>
3181   <xsd:attribute name="macro" type="ST_Formula" use="optional"/>
3182   <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
3183   <xsd:attribute name="linkedCell" type="ST_Formula" use="optional"/>
3184   <xsd:attribute name="listFillRange" type="ST_Formula" use="optional"/>
3185   <xsd:attribute name="cf" type="s:ST_Xstring" use="optional" default="pict"/>
3186   <xsd:attribute ref="r:id" use="optional"/>
3187 </xsd:complexType>
3188 <xsd:simpleType name="ST_WebSourceType">
3189   <xsd:restriction base="xsd:string">
3190     <xsd:enumeration value="sheet"/>
3191     <xsd:enumeration value="printArea"/>
3192     <xsd:enumeration value="autoFilter"/>
3193     <xsd:enumeration value="range"/>
3194     <xsd:enumeration value="chart"/>
3195     <xsd:enumeration value="pivotTable"/>
3196     <xsd:enumeration value="query"/>
3197     <xsd:enumeration value="label"/>
3198   </xsd:restriction>
3199 </xsd:simpleType>
3200 <xsd:complexType name="CT_IgnoredErrors">
3201   <xsd:sequence>
3202     <xsd:element name="ignoredError" type="CT_IgnoredError" minOccurs="1"
3203       maxOccurs="unbounded"/>
3204     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3205   </xsd:sequence>
3206 </xsd:complexType>
3207 <xsd:complexType name="CT_IgnoredError">

```

```

3208     <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
3209     <xsd:attribute name="evalError" type="xsd:boolean" use="optional" default="false"/>
3210     <xsd:attribute name="twoDigitTextYear" type="xsd:boolean" use="optional" default="false"/>
3211     <xsd:attribute name="numberStoredAsText" type="xsd:boolean" use="optional" default="false"/>
3212     <xsd:attribute name="formula" type="xsd:boolean" use="optional" default="false"/>
3213     <xsd:attribute name="formulaRange" type="xsd:boolean" use="optional" default="false"/>
3214     <xsd:attribute name="unlockedFormula" type="xsd:boolean" use="optional" default="false"/>
3215     <xsd:attribute name="emptyCellReference" type="xsd:boolean" use="optional" default="false"/>
3216     <xsd:attribute name="listDataValidation" type="xsd:boolean" use="optional" default="false"/>
3217     <xsd:attribute name="calculatedColumn" type="xsd:boolean" use="optional" default="false"/>
3218 </xsd:complexType>
3219 <xsd:simpleType name="ST_PaneState">
3220     <xsd:restriction base="xsd:string">
3221         <xsd:enumeration value="split"/>
3222         <xsd:enumeration value="frozen"/>
3223         <xsd:enumeration value="frozenSplit"/>
3224     </xsd:restriction>
3225 </xsd:simpleType>
3226 <xsd:complexType name="CT_TableParts">
3227     <xsd:sequence>
3228         <xsd:element name="tablePart" type="CT_TablePart" minOccurs="0" maxOccurs="unbounded"/>
3229     </xsd:sequence>
3230     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3231 </xsd:complexType>
3232 <xsd:complexType name="CT_TablePart">
3233     <xsd:attribute ref="r:id" use="required"/>
3234 </xsd:complexType>
3235 <xsd:element name="metadata" type="CT_Metadata"/>
3236 <xsd:complexType name="CT_Metadata">
3237     <xsd:sequence>
3238         <xsd:element name="metadataTypes" type="CT_MetadataTypes" minOccurs="0" maxOccurs="1"/>
3239         <xsd:element name="metadataStrings" type="CT_MetadataStrings" minOccurs="0"
3240             maxOccurs="1"/>
3241         <xsd:element name="mdxMetadata" type="CT_MdxMetadata" minOccurs="0" maxOccurs="1"/>
3242         <xsd:element name="futureMetadata" type="CT_FutureMetadata" minOccurs="0"
3243             maxOccurs="unbounded"/>
3244         <xsd:element name="cellMetadata" type="CT_MetadataBlocks" minOccurs="0" maxOccurs="1"/>
3245         <xsd:element name="valueMetadata" type="CT_MetadataBlocks" minOccurs="0" maxOccurs="1"/>
3246         <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3247     </xsd:sequence>
3248 </xsd:complexType>
3249 <xsd:complexType name="CT_MetadataTypes">
3250     <xsd:sequence>
3251         <xsd:element name="metadataType" type="CT_MetadataType" minOccurs="1"
3252             maxOccurs="unbounded"/>
3253     </xsd:sequence>
3254     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3255 </xsd:complexType>
3256 <xsd:complexType name="CT_MetadataType">
3257     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3258     <xsd:attribute name="minSupportedVersion" type="xsd:unsignedInt" use="required"/>
3259     <xsd:attribute name="ghostRow" type="xsd:boolean" use="optional" default="false"/>
3260     <xsd:attribute name="ghostCol" type="xsd:boolean" use="optional" default="false"/>

```

```

3261 <xsd:attribute name="edit" type="xsd:boolean" use="optional" default="false"/>
3262 <xsd:attribute name="delete" type="xsd:boolean" use="optional" default="false"/>
3263 <xsd:attribute name="copy" type="xsd:boolean" use="optional" default="false"/>
3264 <xsd:attribute name="pasteAll" type="xsd:boolean" use="optional" default="false"/>
3265 <xsd:attribute name="pasteFormulas" type="xsd:boolean" use="optional" default="false"/>
3266 <xsd:attribute name="pasteValues" type="xsd:boolean" use="optional" default="false"/>
3267 <xsd:attribute name="pasteFormats" type="xsd:boolean" use="optional" default="false"/>
3268 <xsd:attribute name="pasteComments" type="xsd:boolean" use="optional" default="false"/>
3269 <xsd:attribute name="pasteDataValidation" type="xsd:boolean" use="optional" default="false"/>
3270 <xsd:attribute name="pasteBorders" type="xsd:boolean" use="optional" default="false"/>
3271 <xsd:attribute name="pasteColWidths" type="xsd:boolean" use="optional" default="false"/>
3272 <xsd:attribute name="pasteNumberFormats" type="xsd:boolean" use="optional" default="false"/>
3273 <xsd:attribute name="merge" type="xsd:boolean" use="optional" default="false"/>
3274 <xsd:attribute name="splitFirst" type="xsd:boolean" use="optional" default="false"/>
3275 <xsd:attribute name="splitAll" type="xsd:boolean" use="optional" default="false"/>
3276 <xsd:attribute name="rowColShift" type="xsd:boolean" use="optional" default="false"/>
3277 <xsd:attribute name="clearAll" type="xsd:boolean" default="false"/>
3278 <xsd:attribute name="clearFormats" type="xsd:boolean" use="optional" default="false"/>
3279 <xsd:attribute name="clearContents" type="xsd:boolean" use="optional" default="false"/>
3280 <xsd:attribute name="clearComments" type="xsd:boolean" use="optional" default="false"/>
3281 <xsd:attribute name="assign" type="xsd:boolean" use="optional" default="false"/>
3282 <xsd:attribute name="coerce" type="xsd:boolean" use="optional" default="false"/>
3283 <xsd:attribute name="adjust" type="xsd:boolean" use="optional" default="false"/>
3284 <xsd:attribute name="cellMeta" type="xsd:boolean" use="optional" default="false"/>
3285 </xsd:complexType>
3286 <xsd:complexType name="CT_MetadataBlocks">
3287   <xsd:sequence>
3288     <xsd:element name="bk" type="CT_MetadataBlock" minOccurs="1" maxOccurs="unbounded"/>
3289   </xsd:sequence>
3290   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3291 </xsd:complexType>
3292 <xsd:complexType name="CT_MetadataBlock">
3293   <xsd:sequence>
3294     <xsd:element name="rc" type="CT_MetadataRecord" minOccurs="1" maxOccurs="unbounded"/>
3295   </xsd:sequence>
3296 </xsd:complexType>
3297 <xsd:complexType name="CT_MetadataRecord">
3298   <xsd:attribute name="t" type="xsd:unsignedInt" use="required"/>
3299   <xsd:attribute name="v" type="xsd:unsignedInt" use="required"/>
3300 </xsd:complexType>
3301 <xsd:complexType name="CT_FutureMetadata">
3302   <xsd:sequence>
3303     <xsd:element name="bk" type="CT_FutureMetadataBlock" minOccurs="0" maxOccurs="unbounded"/>
3304     <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3305   </xsd:sequence>
3306   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3307   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3308 </xsd:complexType>
3309 <xsd:complexType name="CT_FutureMetadataBlock">
3310   <xsd:sequence>
3311     <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3312   </xsd:sequence>
3313 </xsd:complexType>

```



```

3314 <xsd:complexType name="CT_MdxMetadata">
3315   <xsd:sequence>
3316     <xsd:element name="mdx" type="CT_Mdx" minOccurs="1" maxOccurs="unbounded"/>
3317   </xsd:sequence>
3318   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3319 </xsd:complexType>
3320 <xsd:complexType name="CT_Mdx">
3321   <xsd:choice minOccurs="1" maxOccurs="1">
3322     <xsd:element name="t" type="CT_MdxTuple"/>
3323     <xsd:element name="ms" type="CT_MdxSet"/>
3324     <xsd:element name="p" type="CT_MdxMemberProp"/>
3325     <xsd:element name="k" type="CT_MdxKPI"/>
3326   </xsd:choice>
3327   <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3328   <xsd:attribute name="f" type="ST_MdxFunctionType" use="required"/>
3329 </xsd:complexType>
3330 <xsd:simpleType name="ST_MdxFunctionType">
3331   <xsd:restriction base="xsd:string">
3332     <xsd:enumeration value="m"/>
3333     <xsd:enumeration value="v"/>
3334     <xsd:enumeration value="s"/>
3335     <xsd:enumeration value="c"/>
3336     <xsd:enumeration value="r"/>
3337     <xsd:enumeration value="p"/>
3338     <xsd:enumeration value="k"/>
3339   </xsd:restriction>
3340 </xsd:simpleType>
3341 <xsd:complexType name="CT_MdxTuple">
3342   <xsd:sequence>
3343     <xsd:element name="n" type="CT_MetadataStringIndex" minOccurs="0" maxOccurs="unbounded"/>
3344   </xsd:sequence>
3345   <xsd:attribute name="c" type="xsd:unsignedInt" use="optional" default="0"/>
3346   <xsd:attribute name="ct" type="s:ST_Xstring" use="optional"/>
3347   <xsd:attribute name="si" type="xsd:unsignedInt" use="optional"/>
3348   <xsd:attribute name="fi" type="xsd:unsignedInt" use="optional"/>
3349   <xsd:attribute name="bc" type="ST_UnsignedIntHex" use="optional"/>
3350   <xsd:attribute name="fc" type="ST_UnsignedIntHex" use="optional"/>
3351   <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
3352   <xsd:attribute name="u" type="xsd:boolean" use="optional" default="false"/>
3353   <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
3354   <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
3355 </xsd:complexType>
3356 <xsd:complexType name="CT_MdxSet">
3357   <xsd:sequence>
3358     <xsd:element name="n" type="CT_MetadataStringIndex" minOccurs="0" maxOccurs="unbounded"/>
3359   </xsd:sequence>
3360   <xsd:attribute name="ns" type="xsd:unsignedInt" use="required"/>
3361   <xsd:attribute name="c" type="xsd:unsignedInt" use="optional" default="0"/>
3362   <xsd:attribute name="o" type="ST_MdxSetOrder" use="optional" default="u"/>
3363 </xsd:complexType>
3364 <xsd:simpleType name="ST_MdxSetOrder">
3365   <xsd:restriction base="xsd:string">
3366     <xsd:enumeration value="u"/>

```

```

3367         <xsd:enumeration value="a"/>
3368         <xsd:enumeration value="d"/>
3369         <xsd:enumeration value="aa"/>
3370         <xsd:enumeration value="ad"/>
3371         <xsd:enumeration value="na"/>
3372         <xsd:enumeration value="nd"/>
3373     </xsd:restriction>
3374 </xsd:simpleType>
3375 <xsd:complexType name="CT_MdxMemberProp">
3376     <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3377     <xsd:attribute name="np" type="xsd:unsignedInt" use="required"/>
3378 </xsd:complexType>
3379 <xsd:complexType name="CT_MdxKPI">
3380     <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3381     <xsd:attribute name="np" type="xsd:unsignedInt" use="required"/>
3382     <xsd:attribute name="p" type="ST_MdxKPIProperty" use="required"/>
3383 </xsd:complexType>
3384 <xsd:simpleType name="ST_MdxKPIProperty">
3385     <xsd:restriction base="xsd:string">
3386         <xsd:enumeration value="v"/>
3387         <xsd:enumeration value="g"/>
3388         <xsd:enumeration value="s"/>
3389         <xsd:enumeration value="t"/>
3390         <xsd:enumeration value="w"/>
3391         <xsd:enumeration value="m"/>
3392     </xsd:restriction>
3393 </xsd:simpleType>
3394 <xsd:complexType name="CT_MetadataStringIndex">
3395     <xsd:attribute name="x" type="xsd:unsignedInt" use="required"/>
3396     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
3397 </xsd:complexType>
3398 <xsd:complexType name="CT_MetadataStrings">
3399     <xsd:sequence>
3400         <xsd:element name="s" type="CT_XStringElement" minOccurs="1" maxOccurs="unbounded"/>
3401     </xsd:sequence>
3402     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3403 </xsd:complexType>
3404 <xsd:element name="singleXmlCells" type="CT_SingleXmlCells"/>
3405 <xsd:complexType name="CT_SingleXmlCells">
3406     <xsd:sequence>
3407         <xsd:element name="singleXmlCell" type="CT_SingleXmlCell" maxOccurs="unbounded"/>
3408     </xsd:sequence>
3409 </xsd:complexType>
3410 <xsd:complexType name="CT_SingleXmlCell">
3411     <xsd:sequence>
3412         <xsd:element name="xmlCellPr" type="CT_XmlCellPr" minOccurs="1" maxOccurs="1"/>
3413         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3414     </xsd:sequence>
3415     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3416     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
3417     <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="required"/>
3418 </xsd:complexType>
3419 <xsd:complexType name="CT_XmlCellPr">

```

```

3420     <xsd:sequence>
3421         <xsd:element name="xmlPr" type="CT_XmlPr" minOccurs="1" maxOccurs="1"/>
3422         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3423     </xsd:sequence>
3424     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3425     <xsd:attribute name="uniqueName" type="s:ST_Xstring" use="optional"/>
3426 </xsd:complexType>
3427 <xsd:complexType name="CT_XmlPr">
3428     <xsd:sequence>
3429         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3430     </xsd:sequence>
3431     <xsd:attribute name="mapId" type="xsd:unsignedInt" use="required"/>
3432     <xsd:attribute name="xpath" type="s:ST_Xstring" use="required"/>
3433     <xsd:attribute name="xmlDataType" type="ST_XmlDataType" use="required"/>
3434 </xsd:complexType>
3435 <xsd:element name="styleSheet" type="CT_Stylesheet"/>
3436 <xsd:complexType name="CT_Stylesheet">
3437     <xsd:sequence>
3438         <xsd:element name="numFmts" type="CT_NumFmts" minOccurs="0" maxOccurs="1"/>
3439         <xsd:element name="fonts" type="CT_Fonts" minOccurs="0" maxOccurs="1"/>
3440         <xsd:element name="fills" type="CT_Fills" minOccurs="0" maxOccurs="1"/>
3441         <xsd:element name="borders" type="CT_Borders" minOccurs="0" maxOccurs="1"/>
3442         <xsd:element name="cellStyleXfs" type="CT_CellStyleXfs" minOccurs="0" maxOccurs="1"/>
3443         <xsd:element name="cellXfs" type="CT_CellXfs" minOccurs="0" maxOccurs="1"/>
3444         <xsd:element name="cellStyles" type="CT_CellStyles" minOccurs="0" maxOccurs="1"/>
3445         <xsd:element name="dxfs" type="CT_Dxfs" minOccurs="0" maxOccurs="1"/>
3446         <xsd:element name="tableStyles" type="CT_TableStyles" minOccurs="0" maxOccurs="1"/>
3447         <xsd:element name="colors" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
3448         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3449     </xsd:sequence>
3450 </xsd:complexType>
3451 <xsd:complexType name="CT_CellAlignment">
3452     <xsd:attribute name="horizontal" type="ST_HorizontalAlignment" use="optional"/>
3453     <xsd:attribute name="vertical" type="ST_VerticalAlignment" use="optional"/>
3454     <xsd:attribute name="textRotation" type="xsd:unsignedInt" use="optional"/>
3455     <xsd:attribute name="wrapText" type="xsd:boolean" use="optional"/>
3456     <xsd:attribute name="indent" type="xsd:unsignedInt" use="optional"/>
3457     <xsd:attribute name="relativeIndent" type="xsd:int" use="optional"/>
3458     <xsd:attribute name="justifyLastLine" type="xsd:boolean" use="optional"/>
3459     <xsd:attribute name="shrinkToFit" type="xsd:boolean" use="optional"/>
3460     <xsd:attribute name="readingOrder" type="xsd:unsignedInt" use="optional"/>
3461 </xsd:complexType>
3462 <xsd:simpleType name="ST_BorderStyle">
3463     <xsd:restriction base="xsd:string">
3464         <xsd:enumeration value="none"/>
3465         <xsd:enumeration value="thin"/>
3466         <xsd:enumeration value="medium"/>
3467         <xsd:enumeration value="dashed"/>
3468         <xsd:enumeration value="dotted"/>
3469         <xsd:enumeration value="thick"/>
3470         <xsd:enumeration value="double"/>
3471         <xsd:enumeration value="hair"/>
3472         <xsd:enumeration value="mediumDashed"/>

```

```

3473     <xsd:enumeration value="dashDot"/>
3474     <xsd:enumeration value="mediumDashDot"/>
3475     <xsd:enumeration value="dashDotDot"/>
3476     <xsd:enumeration value="mediumDashDotDot"/>
3477     <xsd:enumeration value="slantDashDot"/>
3478   </xsd:restriction>
3479 </xsd:simpleType>
3480 <xsd:complexType name="CT_Borders">
3481   <xsd:sequence>
3482     <xsd:element name="border" type="CT_Border" minOccurs="0" maxOccurs="unbounded"/>
3483   </xsd:sequence>
3484   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3485 </xsd:complexType>
3486 <xsd:complexType name="CT_Border">
3487   <xsd:sequence>
3488     <xsd:element name="start" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3489     <xsd:element name="end" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3490     <xsd:element name="left" type="CT_BorderPr" minOccurs="0"/>
3491     <xsd:element name="right" type="CT_BorderPr" minOccurs="0"/>
3492     <xsd:element name="top" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3493     <xsd:element name="bottom" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3494     <xsd:element name="diagonal" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3495     <xsd:element name="vertical" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3496     <xsd:element name="horizontal" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3497   </xsd:sequence>
3498   <xsd:attribute name="diagonalUp" type="xsd:boolean" use="optional"/>
3499   <xsd:attribute name="diagonalDown" type="xsd:boolean" use="optional"/>
3500   <xsd:attribute name="outline" type="xsd:boolean" use="optional" default="true"/>
3501 </xsd:complexType>
3502 <xsd:complexType name="CT_BorderPr">
3503   <xsd:sequence>
3504     <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3505   </xsd:sequence>
3506   <xsd:attribute name="style" type="ST_BorderStyle" use="optional" default="none"/>
3507 </xsd:complexType>
3508 <xsd:complexType name="CT_CellProtection">
3509   <xsd:attribute name="locked" type="xsd:boolean" use="optional"/>
3510   <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
3511 </xsd:complexType>
3512 <xsd:complexType name="CT_Fonts">
3513   <xsd:sequence>
3514     <xsd:element name="font" type="CT_Font" minOccurs="0" maxOccurs="unbounded"/>
3515   </xsd:sequence>
3516   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3517 </xsd:complexType>
3518 <xsd:complexType name="CT_Fills">
3519   <xsd:sequence>
3520     <xsd:element name="fill" type="CT_Fill" minOccurs="0" maxOccurs="unbounded"/>
3521   </xsd:sequence>
3522   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3523 </xsd:complexType>
3524 <xsd:complexType name="CT_Fill">
3525   <xsd:choice minOccurs="1" maxOccurs="1">

```

```

3526     <xsd:element name="patternFill" type="CT PatternFill" minOccurs="0" maxOccurs="1"/>
3527     <xsd:element name="gradientFill" type="CT GradientFill" minOccurs="0" maxOccurs="1"/>
3528   </xsd:choice>
3529 </xsd:complexType>
3530 <xsd:complexType name="CT_PatternFill">
3531   <xsd:sequence>
3532     <xsd:element name="fgColor" type="CT Color" minOccurs="0" maxOccurs="1"/>
3533     <xsd:element name="bgColor" type="CT Color" minOccurs="0" maxOccurs="1"/>
3534   </xsd:sequence>
3535   <xsd:attribute name="patternType" type="ST PatternType" use="optional"/>
3536 </xsd:complexType>
3537 <xsd:complexType name="CT_Color">
3538   <xsd:attribute name="auto" type="xsd:boolean" use="optional"/>
3539   <xsd:attribute name="indexed" type="xsd:unsignedInt" use="optional"/>
3540   <xsd:attribute name="rgb" type="ST UnsignedIntHex" use="optional"/>
3541   <xsd:attribute name="theme" type="xsd:unsignedInt" use="optional"/>
3542   <xsd:attribute name="tint" type="xsd:double" use="optional" default="0.0"/>
3543 </xsd:complexType>
3544 <xsd:simpleType name="ST_PatternType">
3545   <xsd:restriction base="xsd:string">
3546     <xsd:enumeration value="none"/>
3547     <xsd:enumeration value="solid"/>
3548     <xsd:enumeration value="mediumGray"/>
3549     <xsd:enumeration value="darkGray"/>
3550     <xsd:enumeration value="lightGray"/>
3551     <xsd:enumeration value="darkHorizontal"/>
3552     <xsd:enumeration value="darkVertical"/>
3553     <xsd:enumeration value="darkDown"/>
3554     <xsd:enumeration value="darkUp"/>
3555     <xsd:enumeration value="darkGrid"/>
3556     <xsd:enumeration value="darkTrellis"/>
3557     <xsd:enumeration value="lightHorizontal"/>
3558     <xsd:enumeration value="lightVertical"/>
3559     <xsd:enumeration value="lightDown"/>
3560     <xsd:enumeration value="lightUp"/>
3561     <xsd:enumeration value="lightGrid"/>
3562     <xsd:enumeration value="lightTrellis"/>
3563     <xsd:enumeration value="gray125"/>
3564     <xsd:enumeration value="gray0625"/>
3565   </xsd:restriction>
3566 </xsd:simpleType>
3567 <xsd:complexType name="CT_GradientFill">
3568   <xsd:sequence>
3569     <xsd:element name="stop" type="CT GradientStop" minOccurs="0" maxOccurs="unbounded"/>
3570   </xsd:sequence>
3571   <xsd:attribute name="type" type="ST GradientType" use="optional" default="linear"/>
3572   <xsd:attribute name="degree" type="xsd:double" use="optional" default="0"/>
3573   <xsd:attribute name="left" type="xsd:double" use="optional" default="0"/>
3574   <xsd:attribute name="right" type="xsd:double" use="optional" default="0"/>
3575   <xsd:attribute name="top" type="xsd:double" use="optional" default="0"/>
3576   <xsd:attribute name="bottom" type="xsd:double" use="optional" default="0"/>
3577 </xsd:complexType>
3578 <xsd:complexType name="CT_GradientStop">

```

```

3579     <xsd:sequence>
3580       <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="1"/>
3581     </xsd:sequence>
3582     <xsd:attribute name="position" type="xsd:double" use="required"/>
3583   </xsd:complexType>
3584   <xsd:simpleType name="ST_GradientType">
3585     <xsd:restriction base="xsd:string">
3586       <xsd:enumeration value="linear"/>
3587       <xsd:enumeration value="path"/>
3588     </xsd:restriction>
3589   </xsd:simpleType>
3590   <xsd:simpleType name="ST_HorizontalAlignment">
3591     <xsd:restriction base="xsd:string">
3592       <xsd:enumeration value="general"/>
3593       <xsd:enumeration value="left"/>
3594       <xsd:enumeration value="center"/>
3595       <xsd:enumeration value="right"/>
3596       <xsd:enumeration value="fill"/>
3597       <xsd:enumeration value="justify"/>
3598       <xsd:enumeration value="centerContinuous"/>
3599       <xsd:enumeration value="distributed"/>
3600     </xsd:restriction>
3601   </xsd:simpleType>
3602   <xsd:simpleType name="ST_VerticalAlignment">
3603     <xsd:restriction base="xsd:string">
3604       <xsd:enumeration value="top"/>
3605       <xsd:enumeration value="center"/>
3606       <xsd:enumeration value="bottom"/>
3607       <xsd:enumeration value="justify"/>
3608       <xsd:enumeration value="distributed"/>
3609     </xsd:restriction>
3610   </xsd:simpleType>
3611   <xsd:complexType name="CT_NumFmts">
3612     <xsd:sequence>
3613       <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="unbounded"/>
3614     </xsd:sequence>
3615     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3616   </xsd:complexType>
3617   <xsd:complexType name="CT_NumFmt">
3618     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="required"/>
3619     <xsd:attribute name="formatCode" type="s:ST_Xstring" use="required"/>
3620   </xsd:complexType>
3621   <xsd:complexType name="CT_CellStyleXfs">
3622     <xsd:sequence>
3623       <xsd:element name="xf" type="CT_Xf" minOccurs="1" maxOccurs="unbounded"/>
3624     </xsd:sequence>
3625     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3626   </xsd:complexType>
3627   <xsd:complexType name="CT_CellXfs">
3628     <xsd:sequence>
3629       <xsd:element name="xf" type="CT_Xf" minOccurs="1" maxOccurs="unbounded"/>
3630     </xsd:sequence>
3631     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>

```

```

3632 </xsd:complexType>
3633 <xsd:complexType name="CT_Xf">
3634   <xsd:sequence>
3635     <xsd:element name="alignment" type="CT_CellAlignment" minOccurs="0" maxOccurs="1"/>
3636     <xsd:element name="protection" type="CT_CellProtection" minOccurs="0" maxOccurs="1"/>
3637     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3638   </xsd:sequence>
3639   <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
3640   <xsd:attribute name="fontId" type="ST_FontId" use="optional"/>
3641   <xsd:attribute name="fillId" type="ST_FillId" use="optional"/>
3642   <xsd:attribute name="borderId" type="ST_BorderId" use="optional"/>
3643   <xsd:attribute name="xfId" type="ST_CellStyleXfId" use="optional"/>
3644   <xsd:attribute name="quotePrefix" type="xsd:boolean" use="optional" default="false"/>
3645   <xsd:attribute name="pivotButton" type="xsd:boolean" use="optional" default="false"/>
3646   <xsd:attribute name="applyNumberFormat" type="xsd:boolean" use="optional"/>
3647   <xsd:attribute name="applyFont" type="xsd:boolean" use="optional"/>
3648   <xsd:attribute name="applyFill" type="xsd:boolean" use="optional"/>
3649   <xsd:attribute name="applyBorder" type="xsd:boolean" use="optional"/>
3650   <xsd:attribute name="applyAlignment" type="xsd:boolean" use="optional"/>
3651   <xsd:attribute name="applyProtection" type="xsd:boolean" use="optional"/>
3652 </xsd:complexType>
3653 <xsd:complexType name="CT_CellStyles">
3654   <xsd:sequence>
3655     <xsd:element name="cellStyle" type="CT_CellStyle" minOccurs="1" maxOccurs="unbounded"/>
3656   </xsd:sequence>
3657   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3658 </xsd:complexType>
3659 <xsd:complexType name="CT_CellStyle">
3660   <xsd:sequence>
3661     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3662   </xsd:sequence>
3663   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
3664   <xsd:attribute name="xfId" type="ST_CellStyleXfId" use="required"/>
3665   <xsd:attribute name="builtinId" type="xsd:unsignedInt" use="optional"/>
3666   <xsd:attribute name="iLevel" type="xsd:unsignedInt" use="optional"/>
3667   <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
3668   <xsd:attribute name="customBuiltin" type="xsd:boolean" use="optional"/>
3669 </xsd:complexType>
3670 <xsd:complexType name="CT_Dxfs">
3671   <xsd:sequence>
3672     <xsd:element name="dxf" type="CT_Dxf" minOccurs="0" maxOccurs="unbounded"/>
3673   </xsd:sequence>
3674   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3675 </xsd:complexType>
3676 <xsd:complexType name="CT_Dxf">
3677   <xsd:sequence>
3678     <xsd:element name="font" type="CT_Font" minOccurs="0" maxOccurs="1"/>
3679     <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
3680     <xsd:element name="fill" type="CT_Fill" minOccurs="0" maxOccurs="1"/>
3681     <xsd:element name="alignment" type="CT_CellAlignment" minOccurs="0" maxOccurs="1"/>
3682     <xsd:element name="border" type="CT_Border" minOccurs="0" maxOccurs="1"/>
3683     <xsd:element name="protection" type="CT_CellProtection" minOccurs="0" maxOccurs="1"/>
3684     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

3685     </xsd:sequence>
3686 </xsd:complexType>
3687 <xsd:simpleType name="ST_NumFmtId">
3688     <xsd:restriction base="xsd:unsignedInt"/>
3689 </xsd:simpleType>
3690 <xsd:simpleType name="ST_FontId">
3691     <xsd:restriction base="xsd:unsignedInt"/>
3692 </xsd:simpleType>
3693 <xsd:simpleType name="ST_FillId">
3694     <xsd:restriction base="xsd:unsignedInt"/>
3695 </xsd:simpleType>
3696 <xsd:simpleType name="ST_BorderId">
3697     <xsd:restriction base="xsd:unsignedInt"/>
3698 </xsd:simpleType>
3699 <xsd:simpleType name="ST_CellStyleXfId">
3700     <xsd:restriction base="xsd:unsignedInt"/>
3701 </xsd:simpleType>
3702 <xsd:simpleType name="ST_DxfId">
3703     <xsd:restriction base="xsd:unsignedInt"/>
3704 </xsd:simpleType>
3705 <xsd:complexType name="CT_Colors">
3706     <xsd:sequence>
3707         <xsd:element name="indexedColors" type="CT_IndexedColors" minOccurs="0" maxOccurs="1"/>
3708         <xsd:element name="mruColors" type="CT_MRUColors" minOccurs="0" maxOccurs="1"/>
3709     </xsd:sequence>
3710 </xsd:complexType>
3711 <xsd:complexType name="CT_IndexedColors">
3712     <xsd:sequence>
3713         <xsd:element name="rgbColor" type="CT_RgbColor" minOccurs="1" maxOccurs="unbounded"/>
3714     </xsd:sequence>
3715 </xsd:complexType>
3716 <xsd:complexType name="CT_MRUColors">
3717     <xsd:sequence>
3718         <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="unbounded"/>
3719     </xsd:sequence>
3720 </xsd:complexType>
3721 <xsd:complexType name="CT_RgbColor">
3722     <xsd:attribute name="rgb" type="ST_UnsignedIntHex" use="optional"/>
3723 </xsd:complexType>
3724 <xsd:complexType name="CT_TableStyles">
3725     <xsd:sequence>
3726         <xsd:element name="tableStyle" type="CT_TableStyle" minOccurs="0" maxOccurs="unbounded"/>
3727     </xsd:sequence>
3728     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3729     <xsd:attribute name="defaultTableStyle" type="xsd:string" use="optional"/>
3730     <xsd:attribute name="defaultPivotStyle" type="xsd:string" use="optional"/>
3731 </xsd:complexType>
3732 <xsd:complexType name="CT_TableStyle">
3733     <xsd:sequence>
3734         <xsd:element name="tableStyleElement" type="CT_TableStyleElement" minOccurs="0"
3735             maxOccurs="unbounded"/>
3736     </xsd:sequence>
3737     <xsd:attribute name="name" type="xsd:string" use="required"/>

```



```

3738     <xsd:attribute name="pivot" type="xsd:boolean" use="optional" default="true"/>
3739     <xsd:attribute name="table" type="xsd:boolean" use="optional" default="true"/>
3740     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3741 </xsd:complexType>
3742 <xsd:complexType name="CT_TableStyleElement">
3743     <xsd:attribute name="type" type="ST_TableStyleType" use="required"/>
3744     <xsd:attribute name="size" type="xsd:unsignedInt" use="optional" default="1"/>
3745     <xsd:attribute name="dxId" type="ST_DxId" use="optional"/>
3746 </xsd:complexType>
3747 <xsd:simpleType name="ST_TableStyleType">
3748     <xsd:restriction base="xsd:string">
3749         <xsd:enumeration value="wholeTable"/>
3750         <xsd:enumeration value="headerRow"/>
3751         <xsd:enumeration value="totalRow"/>
3752         <xsd:enumeration value="firstColumn"/>
3753         <xsd:enumeration value="lastColumn"/>
3754         <xsd:enumeration value="firstRowStripe"/>
3755         <xsd:enumeration value="secondRowStripe"/>
3756         <xsd:enumeration value="firstColumnStripe"/>
3757         <xsd:enumeration value="secondColumnStripe"/>
3758         <xsd:enumeration value="firstHeaderCell"/>
3759         <xsd:enumeration value="lastHeaderCell"/>
3760         <xsd:enumeration value="firstTotalCell"/>
3761         <xsd:enumeration value="lastTotalCell"/>
3762         <xsd:enumeration value="firstSubtotalColumn"/>
3763         <xsd:enumeration value="secondSubtotalColumn"/>
3764         <xsd:enumeration value="thirdSubtotalColumn"/>
3765         <xsd:enumeration value="firstSubtotalRow"/>
3766         <xsd:enumeration value="secondSubtotalRow"/>
3767         <xsd:enumeration value="thirdSubtotalRow"/>
3768         <xsd:enumeration value="blankRow"/>
3769         <xsd:enumeration value="firstColumnSubheading"/>
3770         <xsd:enumeration value="secondColumnSubheading"/>
3771         <xsd:enumeration value="thirdColumnSubheading"/>
3772         <xsd:enumeration value="firstRowSubheading"/>
3773         <xsd:enumeration value="secondRowSubheading"/>
3774         <xsd:enumeration value="thirdRowSubheading"/>
3775         <xsd:enumeration value="pageFieldLabels"/>
3776         <xsd:enumeration value="pageFieldValues"/>
3777     </xsd:restriction>
3778 </xsd:simpleType>
3779 <xsd:complexType name="CT_BooleanProperty">
3780     <xsd:attribute name="val" type="xsd:boolean" use="optional" default="true"/>
3781 </xsd:complexType>
3782 <xsd:complexType name="CT_FontSize">
3783     <xsd:attribute name="val" type="xsd:double" use="required"/>
3784 </xsd:complexType>
3785 <xsd:complexType name="CT_IntProperty">
3786     <xsd:attribute name="val" type="xsd:int" use="required"/>
3787 </xsd:complexType>
3788 <xsd:complexType name="CT_FontName">
3789     <xsd:attribute name="val" type="s:ST_Xstring" use="required"/>
3790 </xsd:complexType>

```

```

3791 <xsd:complexType name="CT_VerticalAlignFontProperty">
3792   <xsd:attribute name="val" type="s:ST_VerticalAlignRun" use="required"/>
3793 </xsd:complexType>
3794 <xsd:complexType name="CT_FontScheme">
3795   <xsd:attribute name="val" type="ST_FontScheme" use="required"/>
3796 </xsd:complexType>
3797 <xsd:simpleType name="ST_FontScheme">
3798   <xsd:restriction base="xsd:string">
3799     <xsd:enumeration value="none"/>
3800     <xsd:enumeration value="major"/>
3801     <xsd:enumeration value="minor"/>
3802   </xsd:restriction>
3803 </xsd:simpleType>
3804 <xsd:complexType name="CT_UnderlineProperty">
3805   <xsd:attribute name="val" type="ST_UnderlineValues" use="optional" default="single"/>
3806 </xsd:complexType>
3807 <xsd:simpleType name="ST_UnderlineValues">
3808   <xsd:restriction base="xsd:string">
3809     <xsd:enumeration value="single"/>
3810     <xsd:enumeration value="double"/>
3811     <xsd:enumeration value="singleAccounting"/>
3812     <xsd:enumeration value="doubleAccounting"/>
3813     <xsd:enumeration value="none"/>
3814   </xsd:restriction>
3815 </xsd:simpleType>
3816 <xsd:complexType name="CT_Font">
3817   <xsd:choice maxOccurs="unbounded">
3818     <xsd:element name="name" type="CT_FontName" minOccurs="0" maxOccurs="1"/>
3819     <xsd:element name="charset" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
3820     <xsd:element name="family" type="CT_FontFamily" minOccurs="0" maxOccurs="1"/>
3821     <xsd:element name="b" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3822     <xsd:element name="i" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3823     <xsd:element name="strike" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3824     <xsd:element name="outline" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3825     <xsd:element name="shadow" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3826     <xsd:element name="condense" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3827     <xsd:element name="extend" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3828     <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3829     <xsd:element name="sz" type="CT_FontSize" minOccurs="0" maxOccurs="1"/>
3830     <xsd:element name="u" type="CT_UnderlineProperty" minOccurs="0" maxOccurs="1"/>
3831     <xsd:element name="vertAlign" type="CT_VerticalAlignFontProperty" minOccurs="0"
3832       maxOccurs="1"/>
3833     <xsd:element name="scheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
3834   </xsd:choice>
3835 </xsd:complexType>
3836 <xsd:complexType name="CT_FontFamily">
3837   <xsd:attribute name="val" type="ST_FontFamily" use="required"/>
3838 </xsd:complexType>
3839 <xsd:simpleType name="ST_FontFamily">
3840   <xsd:restriction base="xsd:integer">
3841     <xsd:minInclusive value="0"/>
3842     <xsd:maxInclusive value="14"/>
3843   </xsd:restriction>

```

```

3844 </xsd:simpleType>
3845 <xsd:attributeGroup name="AG_AutoFormat">
3846   <xsd:attribute name="autoFormatId" type="xsd:unsignedInt"/>
3847   <xsd:attribute name="applyNumberFormats" type="xsd:boolean"/>
3848   <xsd:attribute name="applyBorderFormats" type="xsd:boolean"/>
3849   <xsd:attribute name="applyFontFormats" type="xsd:boolean"/>
3850   <xsd:attribute name="applyPatternFormats" type="xsd:boolean"/>
3851   <xsd:attribute name="applyAlignmentFormats" type="xsd:boolean"/>
3852   <xsd:attribute name="applyWidthHeightFormats" type="xsd:boolean"/>
3853 </xsd:attributeGroup>
3854 <xsd:element name="externalLink" type="CT_ExternalLink"/>
3855 <xsd:complexType name="CT_ExternalLink">
3856   <xsd:sequence>
3857     <xsd:choice>
3858       <xsd:element name="externalBook" type="CT_ExternalBook" minOccurs="0" maxOccurs="1"/>
3859       <xsd:element name="ddeLink" type="CT_DdeLink" minOccurs="0" maxOccurs="1"/>
3860       <xsd:element name="oleLink" type="CT_OleLink" minOccurs="0" maxOccurs="1"/>
3861     </xsd:choice>
3862     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
3863   </xsd:sequence>
3864 </xsd:complexType>
3865 <xsd:complexType name="CT_ExternalBook">
3866   <xsd:sequence>
3867     <xsd:element name="sheetNames" type="CT_ExternalSheetNames" minOccurs="0" maxOccurs="1"/>
3868     <xsd:element name="definedNames" type="CT_ExternalDefinedNames" minOccurs="0"
3869       maxOccurs="1"/>
3870     <xsd:element name="sheetDataSet" type="CT_ExternalSheetDataSet" minOccurs="0"
3871       maxOccurs="1"/>
3872   </xsd:sequence>
3873   <xsd:attribute ref="r:id" use="required"/>
3874 </xsd:complexType>
3875 <xsd:complexType name="CT_ExternalSheetNames">
3876   <xsd:sequence>
3877     <xsd:element name="sheetName" minOccurs="1" maxOccurs="unbounded"
3878       type="CT_ExternalSheetName"/>
3879   </xsd:sequence>
3880 </xsd:complexType>
3881 <xsd:complexType name="CT_ExternalSheetName">
3882   <xsd:attribute name="val" type="s:ST_Xstring"/>
3883 </xsd:complexType>
3884 <xsd:complexType name="CT_ExternalDefinedNames">
3885   <xsd:sequence>
3886     <xsd:element name="definedName" type="CT_ExternalDefinedName" minOccurs="0"
3887       maxOccurs="unbounded"/>
3888   </xsd:sequence>
3889 </xsd:complexType>
3890 <xsd:complexType name="CT_ExternalDefinedName">
3891   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3892   <xsd:attribute name="refersTo" type="s:ST_Xstring" use="optional"/>
3893   <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="optional"/>
3894 </xsd:complexType>
3895 <xsd:complexType name="CT_ExternalSheetDataSet">
3896   <xsd:sequence>

```

```

3897     <xsd:element name="sheetData" type="CT_ExternalSheetData" minOccurs="1"
3898         maxOccurs="unbounded"/>
3899     </xsd:sequence>
3900 </xsd:complexType>
3901 <xsd:complexType name="CT_ExternalSheetData">
3902     <xsd:sequence>
3903         <xsd:element name="row" type="CT_ExternalRow" minOccurs="0" maxOccurs="unbounded"/>
3904     </xsd:sequence>
3905     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
3906     <xsd:attribute name="refreshError" type="xsd:boolean" use="optional" default="false"/>
3907 </xsd:complexType>
3908 <xsd:complexType name="CT_ExternalRow">
3909     <xsd:sequence>
3910         <xsd:element name="cell" type="CT_ExternalCell" minOccurs="0" maxOccurs="unbounded"/>
3911     </xsd:sequence>
3912     <xsd:attribute name="r" type="xsd:unsignedInt" use="required"/>
3913 </xsd:complexType>
3914 <xsd:complexType name="CT_ExternalCell">
3915     <xsd:sequence>
3916         <xsd:element name="v" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
3917     </xsd:sequence>
3918     <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
3919     <xsd:attribute name="t" type="ST_CellType" use="optional" default="n"/>
3920     <xsd:attribute name="vm" type="xsd:unsignedInt" use="optional" default="0"/>
3921 </xsd:complexType>
3922 <xsd:complexType name="CT_DdeLink">
3923     <xsd:sequence>
3924         <xsd:element name="ddeItems" type="CT_DdeItems" minOccurs="0" maxOccurs="1"/>
3925     </xsd:sequence>
3926     <xsd:attribute name="ddeService" type="s:ST Xstring" use="required"/>
3927     <xsd:attribute name="ddeTopic" type="s:ST Xstring" use="required"/>
3928 </xsd:complexType>
3929 <xsd:complexType name="CT_DdeItems">
3930     <xsd:sequence>
3931         <xsd:element name="ddeItem" type="CT_DdeItem" minOccurs="0" maxOccurs="unbounded"/>
3932     </xsd:sequence>
3933 </xsd:complexType>
3934 <xsd:complexType name="CT_DdeItem">
3935     <xsd:sequence>
3936         <xsd:element name="values" type="CT_DdeValues" minOccurs="0" maxOccurs="1"/>
3937     </xsd:sequence>
3938     <xsd:attribute name="name" type="s:ST Xstring" default="0"/>
3939     <xsd:attribute name="ole" type="xsd:boolean" use="optional" default="false"/>
3940     <xsd:attribute name="advise" type="xsd:boolean" use="optional" default="false"/>
3941     <xsd:attribute name="preferPic" type="xsd:boolean" use="optional" default="false"/>
3942 </xsd:complexType>
3943 <xsd:complexType name="CT_DdeValues">
3944     <xsd:sequence>
3945         <xsd:element name="value" minOccurs="1" maxOccurs="unbounded" type="CT_DdeValue"/>
3946     </xsd:sequence>
3947     <xsd:attribute name="rows" type="xsd:unsignedInt" use="optional" default="1"/>
3948     <xsd:attribute name="cols" type="xsd:unsignedInt" use="optional" default="1"/>
3949 </xsd:complexType>

```

```

3950 <xsd:complexType name="CT_DdeValue">
3951   <xsd:sequence>
3952     <xsd:element name="val" type="s:ST Xstring" minOccurs="1" maxOccurs="1"/>
3953   </xsd:sequence>
3954   <xsd:attribute name="t" type="ST DdeValueType" use="optional" default="n"/>
3955 </xsd:complexType>
3956 <xsd:simpleType name="ST_DdeValueType">
3957   <xsd:restriction base="xsd:string">
3958     <xsd:enumeration value="nil"/>
3959     <xsd:enumeration value="b"/>
3960     <xsd:enumeration value="n"/>
3961     <xsd:enumeration value="e"/>
3962     <xsd:enumeration value="str"/>
3963   </xsd:restriction>
3964 </xsd:simpleType>
3965 <xsd:complexType name="CT_OleLink">
3966   <xsd:sequence>
3967     <xsd:element name="oleItems" type="CT OleItems" minOccurs="0" maxOccurs="1"/>
3968   </xsd:sequence>
3969   <xsd:attribute ref="r:id" use="required"/>
3970   <xsd:attribute name="progId" type="s:ST Xstring" use="required"/>
3971 </xsd:complexType>
3972 <xsd:complexType name="CT_OleItems">
3973   <xsd:sequence>
3974     <xsd:element name="oleItem" type="CT OleItem" minOccurs="0" maxOccurs="unbounded"/>
3975   </xsd:sequence>
3976 </xsd:complexType>
3977 <xsd:complexType name="CT_OleItem">
3978   <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
3979   <xsd:attribute name="icon" type="xsd:boolean" use="optional" default="false"/>
3980   <xsd:attribute name="advise" type="xsd:boolean" use="optional" default="false"/>
3981   <xsd:attribute name="preferPic" type="xsd:boolean" use="optional" default="false"/>
3982 </xsd:complexType>
3983 <xsd:element name="table" type="CT Table"/>
3984 <xsd:complexType name="CT_Table">
3985   <xsd:sequence>
3986     <xsd:element name="autoFilter" type="CT AutoFilter" minOccurs="0" maxOccurs="1"/>
3987     <xsd:element name="sortState" type="CT SortState" minOccurs="0" maxOccurs="1"/>
3988     <xsd:element name="tableColumns" type="CT TableColumns" minOccurs="1" maxOccurs="1"/>
3989     <xsd:element name="tableStyleInfo" type="CT TableStyleInfo" minOccurs="0" maxOccurs="1"/>
3990     <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
3991   </xsd:sequence>
3992   <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3993   <xsd:attribute name="name" type="s:ST Xstring" use="optional"/>
3994   <xsd:attribute name="displayName" type="s:ST Xstring" use="required"/>
3995   <xsd:attribute name="comment" type="s:ST Xstring" use="optional"/>
3996   <xsd:attribute name="ref" type="ST Ref" use="required"/>
3997   <xsd:attribute name="tableType" type="ST TableType" use="optional" default="worksheet"/>
3998   <xsd:attribute name="headerRowCount" type="xsd:unsignedInt" use="optional" default="1"/>
3999   <xsd:attribute name="insertRow" type="xsd:boolean" use="optional" default="false"/>
4000   <xsd:attribute name="insertRowShift" type="xsd:boolean" use="optional" default="false"/>
4001   <xsd:attribute name="totalsRowCount" type="xsd:unsignedInt" use="optional" default="0"/>
4002   <xsd:attribute name="totalsRowShown" type="xsd:boolean" use="optional" default="true"/>

```

```

4003     <xsd:attribute name="published" type="xsd:boolean" use="optional" default="false"/>
4004     <xsd:attribute name="headerRowDxfId" type="ST_DxfId" use="optional"/>
4005     <xsd:attribute name="dataDxfId" type="ST_DxfId" use="optional"/>
4006     <xsd:attribute name="totalsRowDxfId" type="ST_DxfId" use="optional"/>
4007     <xsd:attribute name="headerRowBorderDxfId" type="ST_DxfId" use="optional"/>
4008     <xsd:attribute name="tableBorderDxfId" type="ST_DxfId" use="optional"/>
4009     <xsd:attribute name="totalsRowBorderDxfId" type="ST_DxfId" use="optional"/>
4010     <xsd:attribute name="headerRowCellStyle" type="s:ST_Xstring" use="optional"/>
4011     <xsd:attribute name="dataCellStyle" type="s:ST_Xstring" use="optional"/>
4012     <xsd:attribute name="totalsRowCellStyle" type="s:ST_Xstring" use="optional"/>
4013     <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="optional"/>
4014 </xsd:complexType>
4015 <xsd:simpleType name="ST_TableType">
4016     <xsd:restriction base="xsd:string">
4017         <xsd:enumeration value="worksheet"/>
4018         <xsd:enumeration value="xml"/>
4019         <xsd:enumeration value="queryTable"/>
4020     </xsd:restriction>
4021 </xsd:simpleType>
4022 <xsd:complexType name="CT_TableStyleInfo">
4023     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
4024     <xsd:attribute name="showFirstColumn" type="xsd:boolean" use="optional"/>
4025     <xsd:attribute name="showLastColumn" type="xsd:boolean" use="optional"/>
4026     <xsd:attribute name="showRowStripes" type="xsd:boolean" use="optional"/>
4027     <xsd:attribute name="showColumnStripes" type="xsd:boolean" use="optional"/>
4028 </xsd:complexType>
4029 <xsd:complexType name="CT_TableColumns">
4030     <xsd:sequence>
4031         <xsd:element name="tableColumn" type="CT_TableColumn" minOccurs="1"
4032             maxOccurs="unbounded"/>
4033     </xsd:sequence>
4034     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
4035 </xsd:complexType>
4036 <xsd:complexType name="CT_TableColumn">
4037     <xsd:sequence>
4038         <xsd:element name="calculatedColumnFormula" type="CT_TableFormula" minOccurs="0"
4039             maxOccurs="1"/>
4040         <xsd:element name="totalsRowFormula" type="CT_TableFormula" minOccurs="0" maxOccurs="1"/>
4041         <xsd:element name="xmlColumnPr" type="CT_XmlColumnPr" minOccurs="0" maxOccurs="1"/>
4042         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4043     </xsd:sequence>
4044     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
4045     <xsd:attribute name="uniqueName" type="s:ST_Xstring" use="optional"/>
4046     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
4047     <xsd:attribute name="totalsRowFunction" type="ST_TotalsRowFunction" use="optional"
4048         default="none"/>
4049     <xsd:attribute name="totalsRowLabel" type="s:ST_Xstring" use="optional"/>
4050     <xsd:attribute name="queryTableFieldId" type="xsd:unsignedInt" use="optional"/>
4051     <xsd:attribute name="headerRowDxfId" type="ST_DxfId" use="optional"/>
4052     <xsd:attribute name="dataDxfId" type="ST_DxfId" use="optional"/>
4053     <xsd:attribute name="totalsRowDxfId" type="ST_DxfId" use="optional"/>
4054     <xsd:attribute name="headerRowCellStyle" type="s:ST_Xstring" use="optional"/>
4055     <xsd:attribute name="dataCellStyle" type="s:ST_Xstring" use="optional"/>

```

```

4056     <xsd:attribute name="totalsRowCellStyle" type="s:ST_Xstring" use="optional"/>
4057 </xsd:complexType>
4058 <xsd:complexType name="CT_TableFormula">
4059     <xsd:simpleContent>
4060         <xsd:extension base="ST_Formula">
4061             <xsd:attribute name="array" type="xsd:boolean" default="false"/>
4062         </xsd:extension>
4063     </xsd:simpleContent>
4064 </xsd:complexType>
4065 <xsd:simpleType name="ST_TotalsRowFunction">
4066     <xsd:restriction base="xsd:string">
4067         <xsd:enumeration value="none"/>
4068         <xsd:enumeration value="sum"/>
4069         <xsd:enumeration value="min"/>
4070         <xsd:enumeration value="max"/>
4071         <xsd:enumeration value="average"/>
4072         <xsd:enumeration value="count"/>
4073         <xsd:enumeration value="countNums"/>
4074         <xsd:enumeration value="stdDev"/>
4075         <xsd:enumeration value="var"/>
4076         <xsd:enumeration value="custom"/>
4077     </xsd:restriction>
4078 </xsd:simpleType>
4079 <xsd:complexType name="CT_XmlColumnPr">
4080     <xsd:sequence>
4081         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4082     </xsd:sequence>
4083     <xsd:attribute name="mapId" type="xsd:unsignedInt" use="required"/>
4084     <xsd:attribute name="xpath" type="s:ST_Xstring" use="required"/>
4085     <xsd:attribute name="denormalized" type="xsd:boolean" use="optional" default="false"/>
4086     <xsd:attribute name="xmlDataType" type="ST_XmlDataType" use="required"/>
4087 </xsd:complexType>
4088 <xsd:simpleType name="ST_XmlDataType">
4089     <xsd:restriction base="xsd:string"/>
4090 </xsd:simpleType>
4091 <xsd:element name="volTypes" type="CT_VolTypes"/>
4092 <xsd:complexType name="CT_VolTypes">
4093     <xsd:sequence>
4094         <xsd:element name="volType" type="CT_VolType" minOccurs="1" maxOccurs="unbounded"/>
4095         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
4096     </xsd:sequence>
4097 </xsd:complexType>
4098 <xsd:complexType name="CT_VolType">
4099     <xsd:sequence>
4100         <xsd:element name="main" type="CT_VolMain" minOccurs="1" maxOccurs="unbounded"/>
4101     </xsd:sequence>
4102     <xsd:attribute name="type" type="ST_VolDepType" use="required"/>
4103 </xsd:complexType>
4104 <xsd:complexType name="CT_VolMain">
4105     <xsd:sequence>
4106         <xsd:element name="tp" type="CT_VolTopic" minOccurs="1" maxOccurs="unbounded"/>
4107     </xsd:sequence>
4108     <xsd:attribute name="first" type="s:ST_Xstring" use="required"/>

```

```

4109 </xsd:complexType>
4110 <xsd:complexType name="CT_VolTopic">
4111   <xsd:sequence>
4112     <xsd:element name="v" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
4113     <xsd:element name="stp" type="s:ST_Xstring" minOccurs="0" maxOccurs="unbounded"/>
4114     <xsd:element name="tr" type="CT_VolTopicRef" minOccurs="1" maxOccurs="unbounded"/>
4115   </xsd:sequence>
4116   <xsd:attribute name="t" type="ST_VolValueType" use="optional" default="n"/>
4117 </xsd:complexType>
4118 <xsd:complexType name="CT_VolTopicRef">
4119   <xsd:attribute name="r" type="ST_CellRef" use="required"/>
4120   <xsd:attribute name="s" type="xsd:unsignedInt" use="required"/>
4121 </xsd:complexType>
4122 <xsd:simpleType name="ST_VolDepType">
4123   <xsd:restriction base="xsd:string">
4124     <xsd:enumeration value="realTimeData"/>
4125     <xsd:enumeration value="olapFunctions"/>
4126   </xsd:restriction>
4127 </xsd:simpleType>
4128 <xsd:simpleType name="ST_VolValueType">
4129   <xsd:restriction base="xsd:string">
4130     <xsd:enumeration value="b"/>
4131     <xsd:enumeration value="n"/>
4132     <xsd:enumeration value="e"/>
4133     <xsd:enumeration value="s"/>
4134   </xsd:restriction>
4135 </xsd:simpleType>
4136 <xsd:element name="workbook" type="CT_Workbook"/>
4137 <xsd:complexType name="CT_Workbook">
4138   <xsd:sequence>
4139     <xsd:element name="fileVersion" type="CT_FileVersion" minOccurs="0" maxOccurs="1"/>
4140     <xsd:element name="fileSharing" type="CT_FileSharing" minOccurs="0" maxOccurs="1"/>
4141     <xsd:element name="workbookPr" type="CT_WorkbookPr" minOccurs="0" maxOccurs="1"/>
4142     <xsd:element name="workbookProtection" type="CT_WorkbookProtection" minOccurs="0"
4143       maxOccurs="1"/>
4144     <xsd:element name="bookViews" type="CT_BookViews" minOccurs="0" maxOccurs="1"/>
4145     <xsd:element name="sheets" type="CT_Sheets" minOccurs="1" maxOccurs="1"/>
4146     <xsd:element name="functionGroups" type="CT_FunctionGroups" minOccurs="0" maxOccurs="1"/>
4147     <xsd:element name="externalReferences" type="CT_ExternalReferences" minOccurs="0"
4148       maxOccurs="1"/>
4149     <xsd:element name="definedNames" type="CT_DefinedNames" minOccurs="0" maxOccurs="1"/>
4150     <xsd:element name="calcPr" type="CT_CalcPr" minOccurs="0" maxOccurs="1"/>
4151     <xsd:element name="oleSize" type="CT_OleSize" minOccurs="0" maxOccurs="1"/>
4152     <xsd:element name="customWorkbookViews" type="CT_CustomWorkbookViews" minOccurs="0"
4153       maxOccurs="1"/>
4154     <xsd:element name="pivotCaches" type="CT_PivotCaches" minOccurs="0" maxOccurs="1"/>
4155     <xsd:element name="smartTagPr" type="CT_SmartTagPr" minOccurs="0" maxOccurs="1"/>
4156     <xsd:element name="smartTagTypes" type="CT_SmartTagTypes" minOccurs="0" maxOccurs="1"/>
4157     <xsd:element name="webPublishing" type="CT_WebPublishing" minOccurs="0" maxOccurs="1"/>
4158     <xsd:element name="fileRecoveryPr" type="CT_FileRecoveryPr" minOccurs="0"
4159       maxOccurs="unbounded"/>
4160     <xsd:element name="webPublishObjects" type="CT_WebPublishObjects" minOccurs="0"
4161       maxOccurs="1"/>

```



```

4162     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4163   </xsd:sequence>
4164   <xsd:attribute name="conformance" type="s:ST_ConformanceClass"/>
4165 </xsd:complexType>
4166 <xsd:complexType name="CT_FileVersion">
4167   <xsd:attribute name="appName" type="xsd:string" use="optional"/>
4168   <xsd:attribute name="lastEdited" type="xsd:string" use="optional"/>
4169   <xsd:attribute name="lowestEdited" type="xsd:string" use="optional"/>
4170   <xsd:attribute name="rupBuild" type="xsd:string" use="optional"/>
4171   <xsd:attribute name="codeName" type="s:ST_Guid" use="optional"/>
4172 </xsd:complexType>
4173 <xsd:complexType name="CT_BookViews">
4174   <xsd:sequence>
4175     <xsd:element name="workbookView" type="CT_BookView" minOccurs="1" maxOccurs="unbounded"/>
4176   </xsd:sequence>
4177 </xsd:complexType>
4178 <xsd:complexType name="CT_BookView">
4179   <xsd:sequence>
4180     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4181   </xsd:sequence>
4182   <xsd:attribute name="visibility" type="ST_Visibility" use="optional" default="visible"/>
4183   <xsd:attribute name="minimized" type="xsd:boolean" use="optional" default="false"/>
4184   <xsd:attribute name="showHorizontalScroll" type="xsd:boolean" use="optional" default="true"/>
4185   <xsd:attribute name="showVerticalScroll" type="xsd:boolean" use="optional" default="true"/>
4186   <xsd:attribute name="showSheetTabs" type="xsd:boolean" use="optional" default="true"/>
4187   <xsd:attribute name="xWindow" type="xsd:int" use="optional"/>
4188   <xsd:attribute name="yWindow" type="xsd:int" use="optional"/>
4189   <xsd:attribute name="windowWidth" type="xsd:unsignedInt" use="optional"/>
4190   <xsd:attribute name="windowHeight" type="xsd:unsignedInt" use="optional"/>
4191   <xsd:attribute name="tabRatio" type="xsd:unsignedInt" use="optional" default="600"/>
4192   <xsd:attribute name="firstSheet" type="xsd:unsignedInt" use="optional" default="0"/>
4193   <xsd:attribute name="activeTab" type="xsd:unsignedInt" use="optional" default="0"/>
4194   <xsd:attribute name="autoFilterDateGrouping" type="xsd:boolean" use="optional"
4195     default="true"/>
4196 </xsd:complexType>
4197 <xsd:simpleType name="ST_Visibility">
4198   <xsd:restriction base="xsd:string">
4199     <xsd:enumeration value="visible"/>
4200     <xsd:enumeration value="hidden"/>
4201     <xsd:enumeration value="veryHidden"/>
4202   </xsd:restriction>
4203 </xsd:simpleType>
4204 <xsd:complexType name="CT_CustomWorkbookViews">
4205   <xsd:sequence>
4206     <xsd:element name="customWorkbookView" minOccurs="1" maxOccurs="unbounded"
4207       type="CT_CustomWorkbookView"/>
4208   </xsd:sequence>
4209 </xsd:complexType>
4210 <xsd:complexType name="CT_CustomWorkbookView">
4211   <xsd:sequence>
4212     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
4213   </xsd:sequence>
4214   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>

```

```

4215 <xsd:attribute name="guid" type="s:ST Guid" use="required"/>
4216 <xsd:attribute name="autoUpdate" type="xsd:boolean" use="optional" default="false"/>
4217 <xsd:attribute name="mergeInterval" type="xsd:unsignedInt" use="optional"/>
4218 <xsd:attribute name="changesSavedWin" type="xsd:boolean" use="optional" default="false"/>
4219 <xsd:attribute name="onlySync" type="xsd:boolean" use="optional" default="false"/>
4220 <xsd:attribute name="personalView" type="xsd:boolean" use="optional" default="false"/>
4221 <xsd:attribute name="includePrintSettings" type="xsd:boolean" use="optional" default="true"/>
4222 <xsd:attribute name="includeHiddenRowCol" type="xsd:boolean" use="optional" default="true"/>
4223 <xsd:attribute name="maximized" type="xsd:boolean" use="optional" default="false"/>
4224 <xsd:attribute name="minimized" type="xsd:boolean" use="optional" default="false"/>
4225 <xsd:attribute name="showHorizontalScroll" type="xsd:boolean" use="optional" default="true"/>
4226 <xsd:attribute name="showVerticalScroll" type="xsd:boolean" use="optional" default="true"/>
4227 <xsd:attribute name="showSheetTabs" type="xsd:boolean" use="optional" default="true"/>
4228 <xsd:attribute name="xWindow" type="xsd:int" use="optional" default="0"/>
4229 <xsd:attribute name="yWindow" type="xsd:int" use="optional" default="0"/>
4230 <xsd:attribute name="windowWidth" type="xsd:unsignedInt" use="required"/>
4231 <xsd:attribute name="windowHeight" type="xsd:unsignedInt" use="required"/>
4232 <xsd:attribute name="tabRatio" type="xsd:unsignedInt" use="optional" default="600"/>
4233 <xsd:attribute name="activeSheetId" type="xsd:unsignedInt" use="required"/>
4234 <xsd:attribute name="showFormulaBar" type="xsd:boolean" use="optional" default="true"/>
4235 <xsd:attribute name="showStatusbar" type="xsd:boolean" use="optional" default="true"/>
4236 <xsd:attribute name="showComments" type="ST Comments" use="optional" default="commIndicator"/>
4237 <xsd:attribute name="showObjects" type="ST Objects" use="optional" default="all"/>
4238 </xsd:complexType>
4239 <xsd:simpleType name="ST_Comments">
4240 <xsd:restriction base="xsd:string">
4241 <xsd:enumeration value="commNone"/>
4242 <xsd:enumeration value="commIndicator"/>
4243 <xsd:enumeration value="commIndAndComment"/>
4244 </xsd:restriction>
4245 </xsd:simpleType>
4246 <xsd:simpleType name="ST_Objects">
4247 <xsd:restriction base="xsd:string">
4248 <xsd:enumeration value="all"/>
4249 <xsd:enumeration value="placeholders"/>
4250 <xsd:enumeration value="none"/>
4251 </xsd:restriction>
4252 </xsd:simpleType>
4253 <xsd:complexType name="CT_Sheets">
4254 <xsd:sequence>
4255 <xsd:element name="sheet" type="CT Sheet" minOccurs="1" maxOccurs="unbounded"/>
4256 </xsd:sequence>
4257 </xsd:complexType>
4258 <xsd:complexType name="CT_Sheet">
4259 <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
4260 <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
4261 <xsd:attribute name="state" type="ST SheetState" use="optional" default="visible"/>
4262 <xsd:attribute ref="r:id" use="required"/>
4263 </xsd:complexType>
4264 <xsd:simpleType name="ST_SheetState">
4265 <xsd:restriction base="xsd:string">
4266 <xsd:enumeration value="visible"/>
4267 <xsd:enumeration value="hidden"/>

```

```

4268         <xsd:enumeration value="veryHidden"/>
4269     </xsd:restriction>
4270 </xsd:simpleType>
4271 <xsd:complexType name="CT_WorkbookPr">
4272     <xsd:attribute name="date1904" type="xsd:boolean" use="optional" default="false"/>
4273     <xsd:attribute name="showObjects" type="ST_Objects" use="optional" default="all"/>
4274     <xsd:attribute name="showBorderUnselectedTables" type="xsd:boolean" use="optional"
4275         default="true"/>
4276     <xsd:attribute name="filterPrivacy" type="xsd:boolean" use="optional" default="false"/>
4277     <xsd:attribute name="promptedSolutions" type="xsd:boolean" use="optional" default="false"/>
4278     <xsd:attribute name="showInkAnnotation" type="xsd:boolean" use="optional" default="true"/>
4279     <xsd:attribute name="backupFile" type="xsd:boolean" use="optional" default="false"/>
4280     <xsd:attribute name="saveExternalLinkValues" type="xsd:boolean" use="optional"
4281         default="true"/>
4282     <xsd:attribute name="updateLinks" type="ST_UpdateLinks" use="optional" default="userSet"/>
4283     <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
4284     <xsd:attribute name="hidePivotFieldList" type="xsd:boolean" use="optional" default="false"/>
4285     <xsd:attribute name="showPivotChartFilter" type="xsd:boolean" default="false"/>
4286     <xsd:attribute name="allowRefreshQuery" type="xsd:boolean" use="optional" default="false"/>
4287     <xsd:attribute name="publishItems" type="xsd:boolean" use="optional" default="false"/>
4288     <xsd:attribute name="checkCompatibility" type="xsd:boolean" use="optional" default="false"/>
4289     <xsd:attribute name="autoCompressPictures" type="xsd:boolean" use="optional" default="true"/>
4290     <xsd:attribute name="refreshAllConnections" type="xsd:boolean" use="optional"
4291         default="false"/>
4292     <xsd:attribute name="defaultThemeVersion" type="xsd:unsignedInt" use="optional"/>
4293 </xsd:complexType>
4294 <xsd:simpleType name="ST_UpdateLinks">
4295     <xsd:restriction base="xsd:string">
4296         <xsd:enumeration value="userSet"/>
4297         <xsd:enumeration value="never"/>
4298         <xsd:enumeration value="always"/>
4299     </xsd:restriction>
4300 </xsd:simpleType>
4301 <xsd:complexType name="CT_SmartTagPr">
4302     <xsd:attribute name="embed" type="xsd:boolean" use="optional" default="false"/>
4303     <xsd:attribute name="show" type="ST_SmartTagShow" use="optional" default="all"/>
4304 </xsd:complexType>
4305 <xsd:simpleType name="ST_SmartTagShow">
4306     <xsd:restriction base="xsd:string">
4307         <xsd:enumeration value="all"/>
4308         <xsd:enumeration value="none"/>
4309         <xsd:enumeration value="noIndicator"/>
4310     </xsd:restriction>
4311 </xsd:simpleType>
4312 <xsd:complexType name="CT_SmartTagTypes">
4313     <xsd:sequence>
4314         <xsd:element name="smartTagType" type="CT_SmartTagType" minOccurs="0"
4315             maxOccurs="unbounded"/>
4316     </xsd:sequence>
4317 </xsd:complexType>
4318 <xsd:complexType name="CT_SmartTagType">
4319     <xsd:attribute name="namespaceUri" type="s:ST_Xstring" use="optional"/>
4320     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>

```

```

4321     <xsd:attribute name="url" type="s:ST Xstring" use="optional"/>
4322 </xsd:complexType>
4323 <xsd:complexType name="CT_FileRecoveryPr">
4324     <xsd:attribute name="autoRecover" type="xsd:boolean" use="optional" default="true"/>
4325     <xsd:attribute name="crashSave" type="xsd:boolean" use="optional" default="false"/>
4326     <xsd:attribute name="dataExtractLoad" type="xsd:boolean" use="optional" default="false"/>
4327     <xsd:attribute name="repairLoad" type="xsd:boolean" use="optional" default="false"/>
4328 </xsd:complexType>
4329 <xsd:complexType name="CT_CalcPr">
4330     <xsd:attribute name="calcId" type="xsd:unsignedInt"/>
4331     <xsd:attribute name="calcMode" type="ST_CalcMode" use="optional" default="auto"/>
4332     <xsd:attribute name="fullCalcOnLoad" type="xsd:boolean" use="optional" default="false"/>
4333     <xsd:attribute name="refMode" type="ST_RefMode" use="optional" default="A1"/>
4334     <xsd:attribute name="iterate" type="xsd:boolean" use="optional" default="false"/>
4335     <xsd:attribute name="iterateCount" type="xsd:unsignedInt" use="optional" default="100"/>
4336     <xsd:attribute name="iterateDelta" type="xsd:double" use="optional" default="0.001"/>
4337     <xsd:attribute name="fullPrecision" type="xsd:boolean" use="optional" default="true"/>
4338     <xsd:attribute name="calcCompleted" type="xsd:boolean" use="optional" default="true"/>
4339     <xsd:attribute name="calcOnSave" type="xsd:boolean" use="optional" default="true"/>
4340     <xsd:attribute name="concurrentCalc" type="xsd:boolean" use="optional" default="true"/>
4341     <xsd:attribute name="concurrentManualCount" type="xsd:unsignedInt" use="optional"/>
4342     <xsd:attribute name="forceFullCalc" type="xsd:boolean" use="optional"/>
4343 </xsd:complexType>
4344 <xsd:simpleType name="ST_CalcMode">
4345     <xsd:restriction base="xsd:string">
4346         <xsd:enumeration value="manual"/>
4347         <xsd:enumeration value="auto"/>
4348         <xsd:enumeration value="autoNoTable"/>
4349     </xsd:restriction>
4350 </xsd:simpleType>
4351 <xsd:simpleType name="ST_RefMode">
4352     <xsd:restriction base="xsd:string">
4353         <xsd:enumeration value="A1"/>
4354         <xsd:enumeration value="R1C1"/>
4355     </xsd:restriction>
4356 </xsd:simpleType>
4357 <xsd:complexType name="CT_DefinedNames">
4358     <xsd:sequence>
4359         <xsd:element name="definedName" type="CT_DefinedName" minOccurs="0"
4360             maxOccurs="unbounded"/>
4361     </xsd:sequence>
4362 </xsd:complexType>
4363 <xsd:complexType name="CT_DefinedName">
4364     <xsd:simpleContent>
4365         <xsd:extension base="ST_Formula">
4366             <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
4367             <xsd:attribute name="comment" type="s:ST Xstring" use="optional"/>
4368             <xsd:attribute name="customMenu" type="s:ST Xstring" use="optional"/>
4369             <xsd:attribute name="description" type="s:ST Xstring" use="optional"/>
4370             <xsd:attribute name="help" type="s:ST Xstring" use="optional"/>
4371             <xsd:attribute name="statusBar" type="s:ST Xstring" use="optional"/>
4372             <xsd:attribute name="localSheetId" type="xsd:unsignedInt" use="optional"/>
4373             <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>

```

```

4374         <xsd:attribute name="function" type="xsd:boolean" use="optional" default="false"/>
4375         <xsd:attribute name="vbProcedure" type="xsd:boolean" use="optional" default="false"/>
4376         <xsd:attribute name="xlm" type="xsd:boolean" use="optional" default="false"/>
4377         <xsd:attribute name="functionGroupId" type="xsd:unsignedInt" use="optional"/>
4378         <xsd:attribute name="shortcutKey" type="s:ST_Xstring" use="optional"/>
4379         <xsd:attribute name="publishToServer" type="xsd:boolean" use="optional"
4380             default="false"/>
4381         <xsd:attribute name="workbookParameter" type="xsd:boolean" use="optional"
4382             default="false"/>
4383     </xsd:extension>
4384 </xsd:simpleContent>
4385 </xsd:complexType>
4386 <xsd:complexType name="CT_ExternalReferences">
4387     <xsd:sequence>
4388         <xsd:element name="externalReference" type="CT_ExternalReference" minOccurs="1"
4389             maxOccurs="unbounded"/>
4390     </xsd:sequence>
4391 </xsd:complexType>
4392 <xsd:complexType name="CT_ExternalReference">
4393     <xsd:attribute ref="r:id" use="required"/>
4394 </xsd:complexType>
4395 <xsd:complexType name="CT_SheetBackgroundPicture">
4396     <xsd:attribute ref="r:id" use="required"/>
4397 </xsd:complexType>
4398 <xsd:complexType name="CT_PivotCaches">
4399     <xsd:sequence>
4400         <xsd:element name="pivotCache" type="CT_PivotCache" minOccurs="1" maxOccurs="unbounded"/>
4401     </xsd:sequence>
4402 </xsd:complexType>
4403 <xsd:complexType name="CT_PivotCache">
4404     <xsd:attribute name="cacheId" type="xsd:unsignedInt" use="required"/>
4405     <xsd:attribute ref="r:id" use="required"/>
4406 </xsd:complexType>
4407 <xsd:complexType name="CT_FileSharing">
4408     <xsd:attribute name="readOnlyRecommended" type="xsd:boolean" use="optional" default="false"/>
4409     <xsd:attribute name="userName" type="s:ST_Xstring"/>
4410     <xsd:attribute name="reservationPassword" type="ST_UnsignedShortHex"/>
4411     <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
4412     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
4413     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
4414     <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
4415 </xsd:complexType>
4416 <xsd:complexType name="CT_OleSize">
4417     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
4418 </xsd:complexType>
4419 <xsd:complexType name="CT_WorkbookProtection">
4420     <xsd:attribute name="workbookPassword" type="ST_UnsignedShortHex" use="optional"/>
4421     <xsd:attribute name="workbookPasswordCharacterSet" type="xsd:string" use="optional"/>
4422     <xsd:attribute name="revisionsPassword" type="ST_UnsignedShortHex" use="optional"/>
4423     <xsd:attribute name="revisionsPasswordCharacterSet" type="xsd:string" use="optional"/>
4424     <xsd:attribute name="lockStructure" type="xsd:boolean" use="optional" default="false"/>
4425     <xsd:attribute name="lockWindows" type="xsd:boolean" use="optional" default="false"/>
4426     <xsd:attribute name="lockRevision" type="xsd:boolean" use="optional" default="false"/>

```

```

4427     <xsd:attribute name="revisionsAlgorithmName" type="s:ST_Xstring" use="optional"/>
4428     <xsd:attribute name="revisionsHashValue" type="xsd:base64Binary" use="optional"/>
4429     <xsd:attribute name="revisionsSaltValue" type="xsd:base64Binary" use="optional"/>
4430     <xsd:attribute name="revisionsSpinCount" type="xsd:unsignedInt" use="optional"/>
4431     <xsd:attribute name="workbookAlgorithmName" type="s:ST_Xstring" use="optional"/>
4432     <xsd:attribute name="workbookHashValue" type="xsd:base64Binary" use="optional"/>
4433     <xsd:attribute name="workbookSaltValue" type="xsd:base64Binary" use="optional"/>
4434     <xsd:attribute name="workbookSpinCount" type="xsd:unsignedInt" use="optional"/>
4435 </xsd:complexType>
4436 <xsd:complexType name="CT_WebPublishing">
4437     <xsd:attribute name="css" type="xsd:boolean" use="optional" default="true"/>
4438     <xsd:attribute name="thicket" type="xsd:boolean" use="optional" default="true"/>
4439     <xsd:attribute name="longFileNames" type="xsd:boolean" use="optional" default="true"/>
4440     <xsd:attribute name="vml" type="xsd:boolean" use="optional" default="false"/>
4441     <xsd:attribute name="allowPng" type="xsd:boolean" use="optional" default="false"/>
4442     <xsd:attribute name="targetScreenSize" type="ST_TargetScreenSize" use="optional"
4443         default="800x600"/>
4444     <xsd:attribute name="dpi" type="xsd:unsignedInt" use="optional" default="96"/>
4445     <xsd:attribute name="codePage" type="xsd:unsignedInt" use="optional"/>
4446     <xsd:attribute name="characterSet" type="xsd:string" use="optional"/>
4447 </xsd:complexType>
4448 <xsd:simpleType name="ST_TargetScreenSize">
4449     <xsd:restriction base="xsd:string">
4450         <xsd:enumeration value="544x376"/>
4451         <xsd:enumeration value="640x480"/>
4452         <xsd:enumeration value="720x512"/>
4453         <xsd:enumeration value="800x600"/>
4454         <xsd:enumeration value="1024x768"/>
4455         <xsd:enumeration value="1152x882"/>
4456         <xsd:enumeration value="1152x900"/>
4457         <xsd:enumeration value="1280x1024"/>
4458         <xsd:enumeration value="1600x1200"/>
4459         <xsd:enumeration value="1800x1440"/>
4460         <xsd:enumeration value="1920x1200"/>
4461     </xsd:restriction>
4462 </xsd:simpleType>
4463 <xsd:complexType name="CT_FunctionGroups">
4464     <xsd:sequence maxOccurs="unbounded">
4465         <xsd:element name="functionGroup" type="CT_FunctionGroup" minOccurs="0"/>
4466     </xsd:sequence>
4467     <xsd:attribute name="builtInGroupCount" type="xsd:unsignedInt" default="16" use="optional"/>
4468 </xsd:complexType>
4469 <xsd:complexType name="CT_FunctionGroup">
4470     <xsd:attribute name="name" type="s:ST_Xstring"/>
4471 </xsd:complexType>
4472 <xsd:complexType name="CT_WebPublishObjects">
4473     <xsd:sequence>
4474         <xsd:element name="webPublishObject" type="CT_WebPublishObject" minOccurs="1"
4475             maxOccurs="unbounded"/>
4476     </xsd:sequence>
4477     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
4478 </xsd:complexType>
4479 <xsd:complexType name="CT_WebPublishObject">

```

```

4480     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
4481     <xsd:attribute name="divId" type="s:ST_Xstring" use="required"/>
4482     <xsd:attribute name="sourceObject" type="s:ST_Xstring" use="optional"/>
4483     <xsd:attribute name="destinationFile" type="s:ST_Xstring" use="required"/>
4484     <xsd:attribute name="title" type="s:ST_Xstring" use="optional"/>
4485     <xsd:attribute name="autoRepublish" type="xsd:boolean" use="optional" default="false"/>
4486   </xsd:complexType>
4487 </xsd:schema>

```

A.3 PresentationML

This schema is available in the file pml.xsd.

```

1  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2    xmlns="http://schemas.openxmlformats.org/presentationml/2006/main"
3    xmlns:p="http://schemas.openxmlformats.org/presentationml/2006/main"
4    xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
5    xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
6    xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7    elementFormDefault="qualified"
8    targetNamespace="http://schemas.openxmlformats.org/presentationml/2006/main">
9    <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
10      schemaLocation="shared-relationshipReference.xsd"/>
11    <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
12      main.xsd"/>
13    <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
14      schemaLocation="shared-commonSimpleTypes.xsd"/>
15    <xsd:simpleType name="ST_TransitionSideDirectionType">
16      <xsd:restriction base="xsd:token">
17        <xsd:enumeration value="l"/>
18        <xsd:enumeration value="u"/>
19        <xsd:enumeration value="r"/>
20        <xsd:enumeration value="d"/>
21      </xsd:restriction>
22    </xsd:simpleType>
23    <xsd:simpleType name="ST_TransitionCornerDirectionType">
24      <xsd:restriction base="xsd:token">
25        <xsd:enumeration value="lu"/>
26        <xsd:enumeration value="ru"/>
27        <xsd:enumeration value="ld"/>
28        <xsd:enumeration value="rd"/>
29      </xsd:restriction>
30    </xsd:simpleType>
31    <xsd:simpleType name="ST_TransitionInOutDirectionType">
32      <xsd:restriction base="xsd:token">
33        <xsd:enumeration value="out"/>
34        <xsd:enumeration value="in"/>
35      </xsd:restriction>
36    </xsd:simpleType>
37    <xsd:complexType name="CT_SideDirectionTransition">
38      <xsd:attribute name="dir" type="ST_TransitionSideDirectionType" use="optional" default="l"/>
39    </xsd:complexType>
40    <xsd:complexType name="CT_CornerDirectionTransition">

```

```

41     <xsd:attribute name="dir" type="ST_TransitionCornerDirectionType" use="optional"
42         default="lu"/>
43 </xsd:complexType>
44 <xsd:simpleType name="ST_TransitionEightDirectionType">
45     <xsd:union memberTypes="ST_TransitionSideDirectionType ST_TransitionCornerDirectionType"/>
46 </xsd:simpleType>
47 <xsd:complexType name="CT_EightDirectionTransition">
48     <xsd:attribute name="dir" type="ST_TransitionEightDirectionType" use="optional" default="l"/>
49 </xsd:complexType>
50 <xsd:complexType name="CT_OrientationTransition">
51     <xsd:attribute name="dir" type="ST_Direction" use="optional" default="horz"/>
52 </xsd:complexType>
53 <xsd:complexType name="CT_InOutTransition">
54     <xsd:attribute name="dir" type="ST_TransitionInOutDirectionType" use="optional"
55         default="out"/>
56 </xsd:complexType>
57 <xsd:complexType name="CT_OptionalBlackTransition">
58     <xsd:attribute name="thruBlk" type="xsd:boolean" use="optional" default="false"/>
59 </xsd:complexType>
60 <xsd:complexType name="CT_SplitTransition">
61     <xsd:attribute name="orient" type="ST_Direction" use="optional" default="horz"/>
62     <xsd:attribute name="dir" type="ST_TransitionInOutDirectionType" use="optional"
63         default="out"/>
64 </xsd:complexType>
65 <xsd:complexType name="CT_WheelTransition">
66     <xsd:attribute name="spokes" type="xsd:unsignedInt" use="optional" default="4"/>
67 </xsd:complexType>
68 <xsd:complexType name="CT_TransitionStartSoundAction">
69     <xsd:sequence>
70         <xsd:element minOccurs="1" maxOccurs="1" name="snd" type="a:CT_EmbeddedWAVAudioFile"/>
71     </xsd:sequence>
72     <xsd:attribute name="loop" type="xsd:boolean" use="optional" default="false"/>
73 </xsd:complexType>
74 <xsd:complexType name="CT_TransitionSoundAction">
75     <xsd:choice minOccurs="1" maxOccurs="1">
76         <xsd:element name="stSnd" type="CT_TransitionStartSoundAction"/>
77         <xsd:element name="endSnd" type="CT_Empty"/>
78     </xsd:choice>
79 </xsd:complexType>
80 <xsd:simpleType name="ST_TransitionSpeed">
81     <xsd:restriction base="xsd:token">
82         <xsd:enumeration value="slow"/>
83         <xsd:enumeration value="med"/>
84         <xsd:enumeration value="fast"/>
85     </xsd:restriction>
86 </xsd:simpleType>
87 <xsd:complexType name="CT_SlideTransition">
88     <xsd:sequence>
89         <xsd:choice minOccurs="0" maxOccurs="1">
90             <xsd:element name="blinds" type="CT_OrientationTransition"/>
91             <xsd:element name="checker" type="CT_OrientationTransition"/>
92             <xsd:element name="circle" type="CT_Empty"/>
93             <xsd:element name="dissolve" type="CT_Empty"/>

```



```

94      <xsd:element name="comb" type="CT_OrientationTransition"/>
95      <xsd:element name="cover" type="CT_EightDirectionTransition"/>
96      <xsd:element name="cut" type="CT_OptionalBlackTransition"/>
97      <xsd:element name="diamond" type="CT_Empty"/>
98      <xsd:element name="fade" type="CT_OptionalBlackTransition"/>
99      <xsd:element name="newsflash" type="CT_Empty"/>
100     <xsd:element name="plus" type="CT_Empty"/>
101     <xsd:element name="pull" type="CT_EightDirectionTransition"/>
102     <xsd:element name="push" type="CT_SideDirectionTransition"/>
103     <xsd:element name="random" type="CT_Empty"/>
104     <xsd:element name="randomBar" type="CT_OrientationTransition"/>
105     <xsd:element name="split" type="CT_SplitTransition"/>
106     <xsd:element name="strips" type="CT_CornerDirectionTransition"/>
107     <xsd:element name="wedge" type="CT_Empty"/>
108     <xsd:element name="wheel" type="CT_WheelTransition"/>
109     <xsd:element name="wipe" type="CT_SideDirectionTransition"/>
110     <xsd:element name="zoom" type="CT_InOutTransition"/>
111   </xsd:choice>
112   <xsd:element name="sndAc" minOccurs="0" maxOccurs="1" type="CT_TransitionSoundAction"/>
113   <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
114 </xsd:sequence>
115 <xsd:attribute name="spd" type="ST_TransitionSpeed" use="optional" default="fast"/>
116 <xsd:attribute name="advClick" type="xsd:boolean" use="optional" default="true"/>
117 <xsd:attribute name="advTm" type="xsd:unsignedInt" use="optional"/>
118 </xsd:complexType>
119 <xsd:simpleType name="ST_TLTimeIndefinite">
120   <xsd:restriction base="xsd:token">
121     <xsd:enumeration value="indefinite"/>
122   </xsd:restriction>
123 </xsd:simpleType>
124 <xsd:simpleType name="ST_TLTime">
125   <xsd:union memberTypes="xsd:unsignedInt ST_TLTimeIndefinite"/>
126 </xsd:simpleType>
127 <xsd:simpleType name="ST_TLTimeNodeID">
128   <xsd:restriction base="xsd:unsignedInt"/>
129 </xsd:simpleType>
130 <xsd:complexType name="CT_TLIterateIntervalTime">
131   <xsd:attribute name="val" type="ST_TLTime" use="required"/>
132 </xsd:complexType>
133 <xsd:complexType name="CT_TLIterateIntervalPercentage">
134   <xsd:attribute name="val" type="a:ST_PositivePercentage" use="required"/>
135 </xsd:complexType>
136 <xsd:simpleType name="ST_IterateType">
137   <xsd:restriction base="xsd:token">
138     <xsd:enumeration value="el"/>
139     <xsd:enumeration value="wd"/>
140     <xsd:enumeration value="lt"/>
141   </xsd:restriction>
142 </xsd:simpleType>
143 <xsd:complexType name="CT_TLIterateData">
144   <xsd:choice minOccurs="1" maxOccurs="1">
145     <xsd:element name="tmAbs" type="CT_TLIterateIntervalTime"/>
146     <xsd:element name="tmPct" type="CT_TLIterateIntervalPercentage"/>

```

```

147     </xsd:choice>
148     <xsd:attribute name="type" type="ST_IterateType" use="optional" default="el"/>
149     <xsd:attribute name="backwards" type="xsd:boolean" use="optional" default="false"/>
150 </xsd:complexType>
151 <xsd:complexType name="CT_TLSubShapeId">
152     <xsd:attribute name="spid" type="a:ST_ShapeID" use="required"/>
153 </xsd:complexType>
154 <xsd:complexType name="CT_TLTextTargetElement">
155     <xsd:choice minOccurs="0" maxOccurs="1">
156         <xsd:element name="charRg" type="CT_IndexRange"/>
157         <xsd:element name="pRg" type="CT_IndexRange"/>
158     </xsd:choice>
159 </xsd:complexType>
160 <xsd:simpleType name="ST_TLChartSubelementType">
161     <xsd:restriction base="xsd:token">
162         <xsd:enumeration value="gridLegend"/>
163         <xsd:enumeration value="series"/>
164         <xsd:enumeration value="category"/>
165         <xsd:enumeration value="ptInSeries"/>
166         <xsd:enumeration value="ptInCategory"/>
167     </xsd:restriction>
168 </xsd:simpleType>
169 <xsd:complexType name="CT_TLOleChartTargetElement">
170     <xsd:attribute name="type" type="ST_TLChartSubelementType" use="required"/>
171     <xsd:attribute name="lvl" type="xsd:unsignedInt" use="optional" default="0"/>
172 </xsd:complexType>
173 <xsd:complexType name="CT_TLShapeTargetElement">
174     <xsd:choice minOccurs="0" maxOccurs="1">
175         <xsd:element name="bg" type="CT_Empty"/>
176         <xsd:element name="subSp" type="CT_TLSubShapeId"/>
177         <xsd:element name="oleChartEl" type="CT_TLOleChartTargetElement"/>
178         <xsd:element name="txEl" type="CT_TLTextTargetElement"/>
179         <xsd:element name="graphicEl" type="a:CT_AnimationElementChoice"/>
180     </xsd:choice>
181     <xsd:attribute name="spid" type="a:ST_DrawingElementId" use="required"/>
182 </xsd:complexType>
183 <xsd:complexType name="CT_TLTimeTargetElement">
184     <xsd:choice minOccurs="1" maxOccurs="1">
185         <xsd:element name="sldTgt" type="CT_Empty"/>
186         <xsd:element name="sndTgt" type="a:CT_EmbeddedWAVAudioFile"/>
187         <xsd:element name="spTgt" type="CT_TLShapeTargetElement"/>
188         <xsd:element name="inkTgt" type="CT_TLSubShapeId"/>
189     </xsd:choice>
190 </xsd:complexType>
191 <xsd:complexType name="CT_TLTriggerTimeNodeID">
192     <xsd:attribute name="val" type="ST_TLTimeNodeID" use="required"/>
193 </xsd:complexType>
194 <xsd:simpleType name="ST_TLTriggerRuntimeNode">
195     <xsd:restriction base="xsd:token">
196         <xsd:enumeration value="first"/>
197         <xsd:enumeration value="last"/>
198         <xsd:enumeration value="all"/>
199     </xsd:restriction>

```

```

200 </xsd:simpleType>
201 <xsd:complexType name="CT_TLTriggerRuntimeNode">
202   <xsd:attribute name="val" type="ST_TLTriggerRuntimeNode" use="required"/>
203 </xsd:complexType>
204 <xsd:simpleType name="ST_TLTriggerEvent">
205   <xsd:restriction base="xsd:token">
206     <xsd:enumeration value="onBegin"/>
207     <xsd:enumeration value="onEnd"/>
208     <xsd:enumeration value="begin"/>
209     <xsd:enumeration value="end"/>
210     <xsd:enumeration value="onClick"/>
211     <xsd:enumeration value="onDbClick"/>
212     <xsd:enumeration value="onMouseOver"/>
213     <xsd:enumeration value="onMouseOut"/>
214     <xsd:enumeration value="onNext"/>
215     <xsd:enumeration value="onPrev"/>
216     <xsd:enumeration value="onStopAudio"/>
217   </xsd:restriction>
218 </xsd:simpleType>
219 <xsd:complexType name="CT_TLTimeCondition">
220   <xsd:choice minOccurs="0" maxOccurs="1">
221     <xsd:element name="tgtEl" type="CT_TLTimeTargetElement"/>
222     <xsd:element name="tn" type="CT_TLTriggerTimeNodeID"/>
223     <xsd:element name="rtn" type="CT_TLTriggerRuntimeNode"/>
224   </xsd:choice>
225   <xsd:attribute name="evt" use="optional" type="ST_TLTriggerEvent"/>
226   <xsd:attribute name="delay" type="ST_TLTime" use="optional"/>
227 </xsd:complexType>
228 <xsd:complexType name="CT_TLTimeConditionList">
229   <xsd:sequence>
230     <xsd:element name="cond" type="CT_TLTimeCondition" minOccurs="1" maxOccurs="unbounded"/>
231   </xsd:sequence>
232 </xsd:complexType>
233 <xsd:complexType name="CT_TimeNodeList">
234   <xsd:choice minOccurs="1" maxOccurs="unbounded">
235     <xsd:element name="par" type="CT_TLTimeNodeParallel"/>
236     <xsd:element name="seq" type="CT_TLTimeNodeSequence"/>
237     <xsd:element name="excl" type="CT_TLTimeNodeExclusive"/>
238     <xsd:element name="anim" type="CT_TLAnimateBehavior"/>
239     <xsd:element name="animClr" type="CT_TLAnimateColorBehavior"/>
240     <xsd:element name="animEffect" type="CT_TLAnimateEffectBehavior"/>
241     <xsd:element name="animMotion" type="CT_TLAnimateMotionBehavior"/>
242     <xsd:element name="animRot" type="CT_TLAnimateRotationBehavior"/>
243     <xsd:element name="animScale" type="CT_TLAnimateScaleBehavior"/>
244     <xsd:element name="cmd" type="CT_TLCommandBehavior"/>
245     <xsd:element name="set" type="CT_TLSetBehavior"/>
246     <xsd:element name="audio" type="CT_TLMediaNodeAudio"/>
247     <xsd:element name="video" type="CT_TLMediaNodeVideo"/>
248   </xsd:choice>
249 </xsd:complexType>
250 <xsd:simpleType name="ST_TLTimeNodePresetClassType">
251   <xsd:restriction base="xsd:token">
252     <xsd:enumeration value="entr"/>

```

```

253         <xsd:enumeration value="exit"/>
254         <xsd:enumeration value="emph"/>
255         <xsd:enumeration value="path"/>
256         <xsd:enumeration value="verb"/>
257         <xsd:enumeration value="mediacall"/>
258     </xsd:restriction>
259 </xsd:simpleType>
260 <xsd:simpleType name="ST_TLTimeNodeRestartType">
261     <xsd:restriction base="xsd:token">
262         <xsd:enumeration value="always"/>
263         <xsd:enumeration value="whenNotActive"/>
264         <xsd:enumeration value="never"/>
265     </xsd:restriction>
266 </xsd:simpleType>
267 <xsd:simpleType name="ST_TLTimeNodeFillType">
268     <xsd:restriction base="xsd:token">
269         <xsd:enumeration value="remove"/>
270         <xsd:enumeration value="freeze"/>
271         <xsd:enumeration value="hold"/>
272         <xsd:enumeration value="transition"/>
273     </xsd:restriction>
274 </xsd:simpleType>
275 <xsd:simpleType name="ST_TLTimeNodeSyncType">
276     <xsd:restriction base="xsd:token">
277         <xsd:enumeration value="canSlip"/>
278         <xsd:enumeration value="locked"/>
279     </xsd:restriction>
280 </xsd:simpleType>
281 <xsd:simpleType name="ST_TLTimeNodeMasterRelation">
282     <xsd:restriction base="xsd:token">
283         <xsd:enumeration value="sameClick"/>
284         <xsd:enumeration value="lastClick"/>
285         <xsd:enumeration value="nextClick"/>
286     </xsd:restriction>
287 </xsd:simpleType>
288 <xsd:simpleType name="ST_TLTimeNodeType">
289     <xsd:restriction base="xsd:token">
290         <xsd:enumeration value="clickEffect"/>
291         <xsd:enumeration value="withEffect"/>
292         <xsd:enumeration value="afterEffect"/>
293         <xsd:enumeration value="mainSeq"/>
294         <xsd:enumeration value="interactiveSeq"/>
295         <xsd:enumeration value="clickPar"/>
296         <xsd:enumeration value="withGroup"/>
297         <xsd:enumeration value="afterGroup"/>
298         <xsd:enumeration value="tmRoot"/>
299     </xsd:restriction>
300 </xsd:simpleType>
301 <xsd:complexType name="CT_TLCommonTimeNodeData">
302     <xsd:sequence>
303         <xsd:element name="stCondLst" type="CT_TLTimeConditionList" minOccurs="0" maxOccurs="1"/>
304         <xsd:element name="endCondLst" type="CT_TLTimeConditionList" minOccurs="0" maxOccurs="1"/>
305         <xsd:element name="endSync" type="CT_TLTimeCondition" minOccurs="0" maxOccurs="1"/>

```

```

306     <xsd:element name="iterate" type="CT_TLIterateData" minOccurs="0" maxOccurs="1"/>
307     <xsd:element name="childTnLst" type="CT_TimeNodeList" minOccurs="0" maxOccurs="1"/>
308     <xsd:element name="subTnLst" type="CT_TimeNodeList" minOccurs="0" maxOccurs="1"/>
309 </xsd:sequence>
310 <xsd:attribute name="id" type="ST_TLTimeNodeID" use="optional"/>
311 <xsd:attribute name="presetID" type="xsd:int" use="optional"/>
312 <xsd:attribute name="presetClass" type="ST_TLTimeNodePresetClassType" use="optional"/>
313 <xsd:attribute name="presetSubtype" type="xsd:int" use="optional"/>
314 <xsd:attribute name="dur" type="ST_TLTime" use="optional"/>
315 <xsd:attribute name="repeatCount" type="ST_TLTime" use="optional" default="1000"/>
316 <xsd:attribute name="repeatDur" type="ST_TLTime" use="optional"/>
317 <xsd:attribute name="spd" type="a:ST_Percentage" use="optional" default="100%"/>
318 <xsd:attribute name="accel" type="a:ST_PositiveFixedPercentage" use="optional" default="0%"/>
319 <xsd:attribute name="decel" type="a:ST_PositiveFixedPercentage" use="optional" default="0%"/>
320 <xsd:attribute name="autoRev" type="xsd:boolean" use="optional" default="false"/>
321 <xsd:attribute name="restart" type="ST_TLTimeNodeRestartType" use="optional"/>
322 <xsd:attribute name="fill" type="ST_TLTimeNodeFillType" use="optional"/>
323 <xsd:attribute name="syncBehavior" type="ST_TLTimeNodeSyncType" use="optional"/>
324 <xsd:attribute name="tmFilter" type="xsd:string" use="optional"/>
325 <xsd:attribute name="evtFilter" type="xsd:string" use="optional"/>
326 <xsd:attribute name="display" type="xsd:boolean" use="optional"/>
327 <xsd:attribute name="masterRel" type="ST_TLTimeNodeMasterRelation" use="optional"/>
328 <xsd:attribute name="bldLvl" type="xsd:int" use="optional"/>
329 <xsd:attribute name="grpId" type="xsd:unsignedInt" use="optional"/>
330 <xsd:attribute name="afterEffect" type="xsd:boolean" use="optional"/>
331 <xsd:attribute name="nodeType" type="ST_TLTimeNodeType" use="optional"/>
332 <xsd:attribute name="nodePh" type="xsd:boolean" use="optional"/>
333 </xsd:complexType>
334 <xsd:complexType name="CT_TLTimeNodeParallel">
335   <xsd:sequence>
336     <xsd:element name="cTn" type="CT_TLCommonTimeNodeData" minOccurs="1" maxOccurs="1"/>
337   </xsd:sequence>
338 </xsd:complexType>
339 <xsd:simpleType name="ST_TLNextActionType">
340   <xsd:restriction base="xsd:token">
341     <xsd:enumeration value="none"/>
342     <xsd:enumeration value="seek"/>
343   </xsd:restriction>
344 </xsd:simpleType>
345 <xsd:simpleType name="ST_TLPreviousActionType">
346   <xsd:restriction base="xsd:token">
347     <xsd:enumeration value="none"/>
348     <xsd:enumeration value="skipTimed"/>
349   </xsd:restriction>
350 </xsd:simpleType>
351 <xsd:complexType name="CT_TLTimeNodeSequence">
352   <xsd:sequence>
353     <xsd:element name="cTn" type="CT_TLCommonTimeNodeData" minOccurs="1" maxOccurs="1"/>
354     <xsd:element name="prevCondLst" type="CT_TLTimeConditionList" minOccurs="0"
355       maxOccurs="1"/>
356     <xsd:element name="nextCondLst" type="CT_TLTimeConditionList" minOccurs="0"
357       maxOccurs="1"/>
358   </xsd:sequence>

```

```

359     <xsd:attribute name="concurrent" type="xsd:boolean" use="optional"/>
360     <xsd:attribute name="prevAc" type="ST_TLPreviousActionType" use="optional"/>
361     <xsd:attribute name="nextAc" type="ST_TLNextActionType" use="optional"/>
362 </xsd:complexType>
363 <xsd:complexType name="CT_TLTimeNodeExclusive">
364     <xsd:sequence>
365         <xsd:element name="cTn" type="CT_TLCommonTimeNodeData" minOccurs="1" maxOccurs="1"/>
366     </xsd:sequence>
367 </xsd:complexType>
368 <xsd:complexType name="CT_TLBehaviorAttributeNameList">
369     <xsd:sequence>
370         <xsd:element name="attrName" type="xsd:string" minOccurs="1" maxOccurs="unbounded"/>
371     </xsd:sequence>
372 </xsd:complexType>
373 <xsd:simpleType name="ST_TLBehaviorAdditiveType">
374     <xsd:restriction base="xsd:token">
375         <xsd:enumeration value="base"/>
376         <xsd:enumeration value="sum"/>
377         <xsd:enumeration value="repl"/>
378         <xsd:enumeration value="mult"/>
379         <xsd:enumeration value="none"/>
380     </xsd:restriction>
381 </xsd:simpleType>
382 <xsd:simpleType name="ST_TLBehaviorAccumulateType">
383     <xsd:restriction base="xsd:token">
384         <xsd:enumeration value="none"/>
385         <xsd:enumeration value="always"/>
386     </xsd:restriction>
387 </xsd:simpleType>
388 <xsd:simpleType name="ST_TLBehaviorTransformType">
389     <xsd:restriction base="xsd:token">
390         <xsd:enumeration value="pt"/>
391         <xsd:enumeration value="img"/>
392     </xsd:restriction>
393 </xsd:simpleType>
394 <xsd:simpleType name="ST_TLBehaviorOverrideType">
395     <xsd:restriction base="xsd:token">
396         <xsd:enumeration value="normal"/>
397         <xsd:enumeration value="childStyle"/>
398     </xsd:restriction>
399 </xsd:simpleType>
400 <xsd:complexType name="CT_TLCommonBehaviorData">
401     <xsd:sequence>
402         <xsd:element name="cTn" type="CT_TLCommonTimeNodeData" minOccurs="1" maxOccurs="1"/>
403         <xsd:element name="tgtEl" type="CT_TLTimeTargetElement" minOccurs="1" maxOccurs="1"/>
404         <xsd:element name="attrNameList" type="CT_TLBehaviorAttributeNameList" minOccurs="0"
405             maxOccurs="1"/>
406     </xsd:sequence>
407     <xsd:attribute name="additive" type="ST_TLBehaviorAdditiveType" use="optional"/>
408     <xsd:attribute name="accumulate" type="ST_TLBehaviorAccumulateType" use="optional"/>
409     <xsd:attribute name="xfrmType" type="ST_TLBehaviorTransformType" use="optional"/>
410     <xsd:attribute name="from" type="xsd:string" use="optional"/>
411     <xsd:attribute name="to" type="xsd:string" use="optional"/>

```

```

412     <xsd:attribute name="by" type="xsd:string" use="optional"/>
413     <xsd:attribute name="rctx" type="xsd:string" use="optional"/>
414     <xsd:attribute name="override" type="ST_TLBehaviorOverrideType" use="optional"/>
415 </xsd:complexType>
416 <xsd:complexType name="CT_TLAnimVariantBooleanVal">
417     <xsd:attribute name="val" type="xsd:boolean" use="required"/>
418 </xsd:complexType>
419 <xsd:complexType name="CT_TLAnimVariantIntegerVal">
420     <xsd:attribute name="val" type="xsd:int" use="required"/>
421 </xsd:complexType>
422 <xsd:complexType name="CT_TLAnimVariantFloatVal">
423     <xsd:attribute name="val" type="xsd:float" use="required"/>
424 </xsd:complexType>
425 <xsd:complexType name="CT_TLAnimVariantStringVal">
426     <xsd:attribute name="val" type="xsd:string" use="required"/>
427 </xsd:complexType>
428 <xsd:complexType name="CT_TLAnimVariant">
429     <xsd:choice minOccurs="1" maxOccurs="1">
430         <xsd:element name="boolVal" type="CT_TLAnimVariantBooleanVal"/>
431         <xsd:element name="intVal" type="CT_TLAnimVariantIntegerVal"/>
432         <xsd:element name="fltVal" type="CT_TLAnimVariantFloatVal"/>
433         <xsd:element name="strVal" type="CT_TLAnimVariantStringVal"/>
434         <xsd:element name="clrVal" type="a:CT_Color"/>
435     </xsd:choice>
436 </xsd:complexType>
437 <xsd:simpleType name="ST_TLTimeAnimateValueTime">
438     <xsd:union memberTypes="a:ST_PositiveFixedPercentage ST_TLTimeIndefinite"/>
439 </xsd:simpleType>
440 <xsd:complexType name="CT_TLTimeAnimateValue">
441     <xsd:sequence>
442         <xsd:element name="val" type="CT_TLAnimVariant" minOccurs="0" maxOccurs="1"/>
443     </xsd:sequence>
444     <xsd:attribute name="tm" type="ST_TLTimeAnimateValueTime" use="optional"
445         default="indefinite"/>
446     <xsd:attribute name="fmla" type="xsd:string" use="optional" default=""/>
447 </xsd:complexType>
448 <xsd:complexType name="CT_TLTimeAnimateValueList">
449     <xsd:sequence>
450         <xsd:element name="tav" type="CT_TLTimeAnimateValue" minOccurs="0" maxOccurs="unbounded"/>
451     </xsd:sequence>
452 </xsd:complexType>
453 <xsd:simpleType name="ST_TLAnimateBehaviorCalcMode">
454     <xsd:restriction base="xsd:token">
455         <xsd:enumeration value="discrete"/>
456         <xsd:enumeration value="lin"/>
457         <xsd:enumeration value="fmla"/>
458     </xsd:restriction>
459 </xsd:simpleType>
460 <xsd:simpleType name="ST_TLAnimateBehaviorValueType">
461     <xsd:restriction base="xsd:token">
462         <xsd:enumeration value="str"/>
463         <xsd:enumeration value="num"/>
464         <xsd:enumeration value="clr"/>

```

```

465     </xsd:restriction>
466 </xsd:simpleType>
467 <xsd:complexType name="CT_TLAnimateBehavior">
468     <xsd:sequence>
469         <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
470         <xsd:element name="tavLst" type="CT_TLTimeAnimateValueList" minOccurs="0" maxOccurs="1"/>
471     </xsd:sequence>
472     <xsd:attribute name="by" type="xsd:string" use="optional"/>
473     <xsd:attribute name="from" type="xsd:string" use="optional"/>
474     <xsd:attribute name="to" type="xsd:string" use="optional"/>
475     <xsd:attribute name="calcmode" type="ST_TLAnimateBehaviorCalcMode" use="optional"/>
476     <xsd:attribute name="valueType" type="ST_TLAnimateBehaviorValueType" use="optional"/>
477 </xsd:complexType>
478 <xsd:complexType name="CT_TLByRgbColorTransform">
479     <xsd:attribute name="r" type="a:ST_FixedPercentage" use="required"/>
480     <xsd:attribute name="g" type="a:ST_FixedPercentage" use="required"/>
481     <xsd:attribute name="b" type="a:ST_FixedPercentage" use="required"/>
482 </xsd:complexType>
483 <xsd:complexType name="CT_TLByHslColorTransform">
484     <xsd:attribute name="h" type="a:ST_Angle" use="required"/>
485     <xsd:attribute name="s" type="a:ST_FixedPercentage" use="required"/>
486     <xsd:attribute name="l" type="a:ST_FixedPercentage" use="required"/>
487 </xsd:complexType>
488 <xsd:complexType name="CT_TLByAnimateColorTransform">
489     <xsd:choice minOccurs="1" maxOccurs="1">
490         <xsd:element name="rgb" type="CT_TLByRgbColorTransform"/>
491         <xsd:element name="hsl" type="CT_TLByHslColorTransform"/>
492     </xsd:choice>
493 </xsd:complexType>
494 <xsd:simpleType name="ST_TLAnimateColorSpace">
495     <xsd:restriction base="xsd:token">
496         <xsd:enumeration value="rgb"/>
497         <xsd:enumeration value="hsl"/>
498     </xsd:restriction>
499 </xsd:simpleType>
500 <xsd:simpleType name="ST_TLAnimateColorDirection">
501     <xsd:restriction base="xsd:token">
502         <xsd:enumeration value="cw"/>
503         <xsd:enumeration value="ccw"/>
504     </xsd:restriction>
505 </xsd:simpleType>
506 <xsd:complexType name="CT_TLAnimateColorBehavior">
507     <xsd:sequence>
508         <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
509         <xsd:element name="by" type="CT_TLByAnimateColorTransform" minOccurs="0" maxOccurs="1"/>
510         <xsd:element name="from" type="a:CT_Color" minOccurs="0" maxOccurs="1"/>
511         <xsd:element name="to" type="a:CT_Color" minOccurs="0" maxOccurs="1"/>
512     </xsd:sequence>
513     <xsd:attribute name="clrSpc" type="ST_TLAnimateColorSpace" use="optional"/>
514     <xsd:attribute name="dir" type="ST_TLAnimateColorDirection" use="optional"/>
515 </xsd:complexType>
516 <xsd:simpleType name="ST_TLAnimateEffectTransition">
517     <xsd:restriction base="xsd:token">

```



```

518         <xsd:enumeration value="in"/>
519         <xsd:enumeration value="out"/>
520         <xsd:enumeration value="none"/>
521     </xsd:restriction>
522 </xsd:simpleType>
523 <xsd:complexType name="CT_TLAnimateEffectBehavior">
524     <xsd:sequence>
525         <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
526         <xsd:element name="progress" type="CT_TLAnimVariant" minOccurs="0" maxOccurs="1"/>
527     </xsd:sequence>
528     <xsd:attribute name="transition" type="ST_TLAnimateEffectTransition" use="optional"/>
529     <xsd:attribute name="filter" type="xsd:string" use="optional"/>
530     <xsd:attribute name="prLst" type="xsd:string" use="optional"/>
531 </xsd:complexType>
532 <xsd:simpleType name="ST_TLAnimateMotionBehaviorOrigin">
533     <xsd:restriction base="xsd:token">
534         <xsd:enumeration value="parent"/>
535         <xsd:enumeration value="layout"/>
536     </xsd:restriction>
537 </xsd:simpleType>
538 <xsd:simpleType name="ST_TLAnimateMotionPathEditMode">
539     <xsd:restriction base="xsd:token">
540         <xsd:enumeration value="relative"/>
541         <xsd:enumeration value="fixed"/>
542     </xsd:restriction>
543 </xsd:simpleType>
544 <xsd:complexType name="CT_TLPoint">
545     <xsd:attribute name="x" type="a:ST_Percentage" use="required"/>
546     <xsd:attribute name="y" type="a:ST_Percentage" use="required"/>
547 </xsd:complexType>
548 <xsd:complexType name="CT_TLAnimateMotionBehavior">
549     <xsd:sequence>
550         <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
551         <xsd:element name="by" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
552         <xsd:element name="from" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
553         <xsd:element name="to" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
554         <xsd:element name="rCtr" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
555     </xsd:sequence>
556     <xsd:attribute name="origin" type="ST_TLAnimateMotionBehaviorOrigin" use="optional"/>
557     <xsd:attribute name="path" type="xsd:string" use="optional"/>
558     <xsd:attribute name="pathEditMode" type="ST_TLAnimateMotionPathEditMode" use="optional"/>
559     <xsd:attribute name="rAng" type="a:ST_Angle" use="optional"/>
560     <xsd:attribute name="ptsTypes" type="xsd:string" use="optional"/>
561 </xsd:complexType>
562 <xsd:complexType name="CT_TLAnimateRotationBehavior">
563     <xsd:sequence>
564         <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
565     </xsd:sequence>
566     <xsd:attribute name="by" type="a:ST_Angle" use="optional"/>
567     <xsd:attribute name="from" type="a:ST_Angle" use="optional"/>
568     <xsd:attribute name="to" type="a:ST_Angle" use="optional"/>
569 </xsd:complexType>
570 <xsd:complexType name="CT_TLAnimateScaleBehavior">

```

```

571     <xsd:sequence>
572       <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
573       <xsd:element name="by" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
574       <xsd:element name="from" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
575       <xsd:element name="to" type="CT_TLPoint" minOccurs="0" maxOccurs="1"/>
576     </xsd:sequence>
577     <xsd:attribute name="zoomContents" type="xsd:boolean" use="optional"/>
578   </xsd:complexType>
579   <xsd:simpleType name="ST_TLCommandType">
580     <xsd:restriction base="xsd:token">
581       <xsd:enumeration value="evt"/>
582       <xsd:enumeration value="call"/>
583       <xsd:enumeration value="verb"/>
584     </xsd:restriction>
585   </xsd:simpleType>
586   <xsd:complexType name="CT_TLCommandBehavior">
587     <xsd:sequence>
588       <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
589     </xsd:sequence>
590     <xsd:attribute type="ST_TLCommandType" name="type" use="optional"/>
591     <xsd:attribute name="cmd" type="xsd:string" use="optional"/>
592   </xsd:complexType>
593   <xsd:complexType name="CT_TLSetBehavior">
594     <xsd:sequence>
595       <xsd:element name="cBhvr" type="CT_TLCommonBehaviorData" minOccurs="1" maxOccurs="1"/>
596       <xsd:element name="to" type="CT_TLAnimVariant" minOccurs="0" maxOccurs="1"/>
597     </xsd:sequence>
598   </xsd:complexType>
599   <xsd:complexType name="CT_TLCommonMediaNodeData">
600     <xsd:sequence>
601       <xsd:element name="cTn" type="CT_TLCommonTimeNodeData" minOccurs="1" maxOccurs="1"/>
602       <xsd:element name="tgtEl" type="CT_TLTimeTargetElement" minOccurs="1" maxOccurs="1"/>
603     </xsd:sequence>
604     <xsd:attribute name="vol" type="a:ST_PositiveFixedPercentage" default="50%" use="optional"/>
605     <xsd:attribute name="mute" type="xsd:boolean" use="optional" default="false"/>
606     <xsd:attribute name="numSld" type="xsd:unsignedInt" use="optional" default="1"/>
607     <xsd:attribute name="showWhenStopped" type="xsd:boolean" use="optional" default="true"/>
608   </xsd:complexType>
609   <xsd:complexType name="CT_TLMediaNodeAudio">
610     <xsd:sequence>
611       <xsd:element name="cMediaNode" type="CT_TLCommonMediaNodeData" minOccurs="1"
612         maxOccurs="1"/>
613     </xsd:sequence>
614     <xsd:attribute name="isNarration" type="xsd:boolean" use="optional" default="false"/>
615   </xsd:complexType>
616   <xsd:complexType name="CT_TLMediaNodeVideo">
617     <xsd:sequence>
618       <xsd:element name="cMediaNode" type="CT_TLCommonMediaNodeData" minOccurs="1"
619         maxOccurs="1"/>
620     </xsd:sequence>
621     <xsd:attribute name="fullScr" type="xsd:boolean" use="optional" default="false"/>
622   </xsd:complexType>
623   <xsd:attributeGroup name="AG_TLBuild">

```

```

624     <xsd:attribute name="spid" type="a:ST_DrawingElementId" use="required"/>
625     <xsd:attribute name="grpId" type="xsd:unsignedInt" use="required"/>
626     <xsd:attribute name="uiExpand" type="xsd:boolean" use="optional" default="false"/>
627 </xsd:attributeGroup>
628 <xsd:complexType name="CT_TLTemplate">
629     <xsd:sequence>
630         <xsd:element name="tnLst" type="CT_TimeNodeList" minOccurs="1" maxOccurs="1"/>
631     </xsd:sequence>
632     <xsd:attribute name="lvl" type="xsd:unsignedInt" use="optional" default="0"/>
633 </xsd:complexType>
634 <xsd:complexType name="CT_TLTemplateList">
635     <xsd:sequence>
636         <xsd:element name="tmpl" type="CT_TLTemplate" minOccurs="0" maxOccurs="9"/>
637     </xsd:sequence>
638 </xsd:complexType>
639 <xsd:simpleType name="ST_TLParaBuildType">
640     <xsd:restriction base="xsd:token">
641         <xsd:enumeration value="allAtOnce"/>
642         <xsd:enumeration value="p"/>
643         <xsd:enumeration value="cust"/>
644         <xsd:enumeration value="whole"/>
645     </xsd:restriction>
646 </xsd:simpleType>
647 <xsd:complexType name="CT_TLBuildParagraph">
648     <xsd:sequence>
649         <xsd:element name="tmplLst" type="CT_TLTemplateList" minOccurs="0" maxOccurs="1"/>
650     </xsd:sequence>
651     <xsd:attributeGroup ref="AG_TLBuild"/>
652     <xsd:attribute name="build" type="ST_TLParaBuildType" use="optional" default="whole"/>
653     <xsd:attribute name="bldLvl" type="xsd:unsignedInt" use="optional" default="1"/>
654     <xsd:attribute name="animBg" type="xsd:boolean" use="optional" default="false"/>
655     <xsd:attribute name="autoUpdateAnimBg" type="xsd:boolean" default="true" use="optional"/>
656     <xsd:attribute name="rev" type="xsd:boolean" use="optional" default="false"/>
657     <xsd:attribute name="advAuto" type="ST_TLTime" use="optional" default="indefinite"/>
658 </xsd:complexType>
659 <xsd:simpleType name="ST_TLDiagramBuildType">
660     <xsd:restriction base="xsd:token">
661         <xsd:enumeration value="whole"/>
662         <xsd:enumeration value="depthByNode"/>
663         <xsd:enumeration value="depthByBranch"/>
664         <xsd:enumeration value="breadthByNode"/>
665         <xsd:enumeration value="breadthByLvl"/>
666         <xsd:enumeration value="cw"/>
667         <xsd:enumeration value="cwIn"/>
668         <xsd:enumeration value="cwOut"/>
669         <xsd:enumeration value="ccw"/>
670         <xsd:enumeration value="ccwIn"/>
671         <xsd:enumeration value="ccwOut"/>
672         <xsd:enumeration value="inByRing"/>
673         <xsd:enumeration value="outByRing"/>
674         <xsd:enumeration value="up"/>
675         <xsd:enumeration value="down"/>
676         <xsd:enumeration value="allAtOnce"/>

```

```

677     <xsd:enumeration value="cust"/>
678   </xsd:restriction>
679 </xsd:simpleType>
680 <xsd:complexType name="CT_TLBuildDiagram">
681   <xsd:attributeGroup ref="AG_TLBuild"/>
682   <xsd:attribute name="bld" type="ST_TLDiagramBuildType" use="optional" default="whole"/>
683 </xsd:complexType>
684 <xsd:simpleType name="ST_TLOleChartBuildType">
685   <xsd:restriction base="xsd:token">
686     <xsd:enumeration value="allAtOnce"/>
687     <xsd:enumeration value="series"/>
688     <xsd:enumeration value="category"/>
689     <xsd:enumeration value="seriesEl"/>
690     <xsd:enumeration value="categoryEl"/>
691   </xsd:restriction>
692 </xsd:simpleType>
693 <xsd:complexType name="CT_TLOleBuildChart">
694   <xsd:attributeGroup ref="AG_TLBuild"/>
695   <xsd:attribute name="bld" type="ST_TLOleChartBuildType" use="optional" default="allAtOnce"/>
696   <xsd:attribute name="animBg" type="xsd:boolean" use="optional" default="true"/>
697 </xsd:complexType>
698 <xsd:complexType name="CT_TLGraphicalObjectBuild">
699   <xsd:choice minOccurs="1" maxOccurs="1">
700     <xsd:element name="bldAsOne" type="CT_Empty"/>
701     <xsd:element name="bldSub" type="a:CT_AnimationGraphicalObjectBuildProperties"/>
702   </xsd:choice>
703   <xsd:attributeGroup ref="AG_TLBuild"/>
704 </xsd:complexType>
705 <xsd:complexType name="CT_BuildList">
706   <xsd:choice minOccurs="1" maxOccurs="unbounded">
707     <xsd:element name="bldP" type="CT_TLBuildParagraph"/>
708     <xsd:element name="bldDgm" type="CT_TLBuildDiagram"/>
709     <xsd:element name="bldOleChart" type="CT_TLOleBuildChart"/>
710     <xsd:element name="bldGraphic" type="CT_TLGraphicalObjectBuild"/>
711   </xsd:choice>
712 </xsd:complexType>
713 <xsd:complexType name="CT_SlideTiming">
714   <xsd:sequence>
715     <xsd:element name="tnLst" type="CT_TimeNodeList" minOccurs="0" maxOccurs="1"/>
716     <xsd:element name="bldLst" type="CT_BuildList" minOccurs="0" maxOccurs="1"/>
717     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
718   </xsd:sequence>
719 </xsd:complexType>
720 <xsd:complexType name="CT_Empty"/>
721 <xsd:simpleType name="ST_Name">
722   <xsd:restriction base="xsd:string"/>
723 </xsd:simpleType>
724 <xsd:simpleType name="ST_Direction">
725   <xsd:restriction base="xsd:token">
726     <xsd:enumeration value="horz"/>
727     <xsd:enumeration value="vert"/>
728   </xsd:restriction>
729 </xsd:simpleType>

```

```

730 <xsd:simpleType name="ST_Index">
731   <xsd:restriction base="xsd:unsignedInt"/>
732 </xsd:simpleType>
733 <xsd:complexType name="CT_IndexRange">
734   <xsd:attribute name="st" type="ST_Index" use="required"/>
735   <xsd:attribute name="end" type="ST_Index" use="required"/>
736 </xsd:complexType>
737 <xsd:complexType name="CT_SlideRelationshipListEntry">
738   <xsd:attribute ref="r:id" use="required"/>
739 </xsd:complexType>
740 <xsd:complexType name="CT_SlideRelationshipList">
741   <xsd:sequence>
742     <xsd:element name="sld" type="CT_SlideRelationshipListEntry" minOccurs="0"
743       maxOccurs="unbounded"/>
744   </xsd:sequence>
745 </xsd:complexType>
746 <xsd:complexType name="CT_CustomShowId">
747   <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
748 </xsd:complexType>
749 <xsd:group name="EG_SlideListChoice">
750   <xsd:choice>
751     <xsd:element name="sldAll" type="CT_Empty"/>
752     <xsd:element name="sldRg" type="CT_IndexRange"/>
753     <xsd:element name="custShow" type="CT_CustomShowId"/>
754   </xsd:choice>
755 </xsd:group>
756 <xsd:complexType name="CT_CustomerData">
757   <xsd:attribute ref="r:id" use="required"/>
758 </xsd:complexType>
759 <xsd:complexType name="CT_TagsData">
760   <xsd:attribute ref="r:id" use="required"/>
761 </xsd:complexType>
762 <xsd:complexType name="CT_CustomerDataList">
763   <xsd:sequence minOccurs="0" maxOccurs="1">
764     <xsd:element name="custData" type="CT_CustomerData" minOccurs="0" maxOccurs="unbounded"/>
765     <xsd:element name="tags" type="CT_TagsData" minOccurs="0" maxOccurs="1"/>
766   </xsd:sequence>
767 </xsd:complexType>
768 <xsd:complexType name="CT_Extension">
769   <xsd:sequence>
770     <xsd:any processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
771   </xsd:sequence>
772   <xsd:attribute name="uri" type="xsd:token" use="required"/>
773 </xsd:complexType>
774 <xsd:group name="EG_ExtensionList">
775   <xsd:sequence>
776     <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
777   </xsd:sequence>
778 </xsd:group>
779 <xsd:complexType name="CT_ExtensionList">
780   <xsd:sequence>
781     <xsd:group ref="EG_ExtensionList" minOccurs="0" maxOccurs="1"/>
782   </xsd:sequence>

```

```

783 </xsd:complexType>
784 <xsd:complexType name="CT_ExtensionListModify">
785   <xsd:sequence>
786     <xsd:group ref="EG_ExtensionList" minOccurs="0" maxOccurs="1"/>
787   </xsd:sequence>
788   <xsd:attribute name="mod" type="xsd:boolean" use="optional" default="false"/>
789 </xsd:complexType>
790 <xsd:complexType name="CT_CommentAuthor">
791   <xsd:sequence>
792     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
793   </xsd:sequence>
794   <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
795   <xsd:attribute name="name" type="ST_Name" use="required"/>
796   <xsd:attribute name="initials" type="ST_Name" use="required"/>
797   <xsd:attribute name="lastIdx" type="xsd:unsignedInt" use="required"/>
798   <xsd:attribute name="clrIdx" type="xsd:unsignedInt" use="required"/>
799 </xsd:complexType>
800 <xsd:complexType name="CT_CommentAuthorList">
801   <xsd:sequence>
802     <xsd:element name="cmAuthor" type="CT_CommentAuthor" minOccurs="0" maxOccurs="unbounded"/>
803   </xsd:sequence>
804 </xsd:complexType>
805 <xsd:element name="cmAuthorLst" type="CT_CommentAuthorList"/>
806 <xsd:complexType name="CT_Comment">
807   <xsd:sequence>
808     <xsd:element name="pos" type="a:CT_Point2D" minOccurs="1" maxOccurs="1"/>
809     <xsd:element name="text" type="xsd:string" minOccurs="1" maxOccurs="1"/>
810     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
811   </xsd:sequence>
812   <xsd:attribute name="authorId" type="xsd:unsignedInt" use="required"/>
813   <xsd:attribute name="dt" type="xsd:dateTime" use="optional"/>
814   <xsd:attribute name="idx" type="ST_Index" use="required"/>
815 </xsd:complexType>
816 <xsd:complexType name="CT_CommentList">
817   <xsd:sequence>
818     <xsd:element name="cm" type="CT_Comment" minOccurs="0" maxOccurs="unbounded"/>
819   </xsd:sequence>
820 </xsd:complexType>
821 <xsd:element name="cmLst" type="CT_CommentList"/>
822 <xsd:attributeGroup name="AG_Ole">
823   <xsd:attribute name="spid" type="a:ST_ShapeID" use="optional"/>
824   <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
825   <xsd:attribute name="showAsIcon" type="xsd:boolean" use="optional" default="false"/>
826   <xsd:attribute ref="r:id" use="optional"/>
827   <xsd:attribute name="imgW" type="a:ST_PositiveCoordinate32" use="optional"/>
828   <xsd:attribute name="imgH" type="a:ST_PositiveCoordinate32" use="optional"/>
829 </xsd:attributeGroup>
830 <xsd:simpleType name="ST_OleObjectFollowColorScheme">
831   <xsd:restriction base="xsd:token">
832     <xsd:enumeration value="none"/>
833     <xsd:enumeration value="full"/>
834     <xsd:enumeration value="textAndBackground"/>
835   </xsd:restriction>

```

```

836 </xsd:simpleType>
837 <xsd:complexType name="CT_OleObjectEmbed">
838   <xsd:sequence>
839     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
840   </xsd:sequence>
841   <xsd:attribute name="followColorScheme" type="ST_OleObjectFollowColorScheme" use="optional"
842     default="none"/>
843 </xsd:complexType>
844 <xsd:complexType name="CT_OleObjectLink">
845   <xsd:sequence>
846     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
847   </xsd:sequence>
848   <xsd:attribute name="updateAutomatic" type="xsd:boolean" use="optional" default="false"/>
849 </xsd:complexType>
850 <xsd:complexType name="CT_OleObject">
851   <xsd:sequence>
852     <xsd:choice minOccurs="1" maxOccurs="1">
853       <xsd:element name="embed" type="CT_OleObjectEmbed"/>
854       <xsd:element name="link" type="CT_OleObjectLink"/>
855     </xsd:choice>
856     <xsd:element name="pic" type="CT_Picture" minOccurs="0" maxOccurs="1"/>
857   </xsd:sequence>
858   <xsd:attributeGroup ref="AG_Ole"/>
859   <xsd:attribute name="progId" type="xsd:string" use="optional"/>
860 </xsd:complexType>
861 <xsd:element name="oleObj" type="CT_OleObject"/>
862 <xsd:complexType name="CT_Control">
863   <xsd:sequence>
864     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
865     <xsd:element name="pic" type="CT_Picture" minOccurs="0" maxOccurs="1"/>
866   </xsd:sequence>
867   <xsd:attributeGroup ref="AG_Ole"/>
868 </xsd:complexType>
869 <xsd:complexType name="CT_ControlList">
870   <xsd:sequence>
871     <xsd:element name="control" type="CT_Control" minOccurs="0" maxOccurs="unbounded"/>
872   </xsd:sequence>
873 </xsd:complexType>
874 <xsd:simpleType name="ST_SlideId">
875   <xsd:restriction base="xsd:unsignedInt">
876     <xsd:minInclusive value="256"/>
877     <xsd:maxExclusive value="2147483648"/>
878   </xsd:restriction>
879 </xsd:simpleType>
880 <xsd:complexType name="CT_SlideIdListEntry">
881   <xsd:sequence>
882     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
883   </xsd:sequence>
884   <xsd:attribute name="id" type="ST_SlideId" use="required"/>
885   <xsd:attribute ref="r:id" use="required"/>
886 </xsd:complexType>
887 <xsd:complexType name="CT_SlideIdList">
888   <xsd:sequence>

```

```

889     <xsd:element name="sldId" type="CT_SlideIdListEntry" minOccurs="0" maxOccurs="unbounded"/>
890   </xsd:sequence>
891 </xsd:complexType>
892 <xsd:simpleType name="ST_SlideMasterId">
893   <xsd:restriction base="xsd:unsignedInt">
894     <xsd:minInclusive value="2147483648"/>
895   </xsd:restriction>
896 </xsd:simpleType>
897 <xsd:complexType name="CT_SlideMasterIdListEntry">
898   <xsd:sequence>
899     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
900   </xsd:sequence>
901   <xsd:attribute name="id" type="ST_SlideMasterId" use="optional"/>
902   <xsd:attribute ref="r:id" use="required"/>
903 </xsd:complexType>
904 <xsd:complexType name="CT_SlideMasterIdList">
905   <xsd:sequence>
906     <xsd:element name="sldMasterId" type="CT_SlideMasterIdListEntry" minOccurs="0"
907       maxOccurs="unbounded"/>
908   </xsd:sequence>
909 </xsd:complexType>
910 <xsd:complexType name="CT_NotesMasterIdListEntry">
911   <xsd:sequence>
912     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
913   </xsd:sequence>
914   <xsd:attribute ref="r:id" use="required"/>
915 </xsd:complexType>
916 <xsd:complexType name="CT_NotesMasterIdList">
917   <xsd:sequence>
918     <xsd:element name="notesMasterId" type="CT_NotesMasterIdListEntry" minOccurs="0"
919       maxOccurs="1"/>
920   </xsd:sequence>
921 </xsd:complexType>
922 <xsd:complexType name="CT_HandoutMasterIdListEntry">
923   <xsd:sequence>
924     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
925   </xsd:sequence>
926   <xsd:attribute ref="r:id" use="required"/>
927 </xsd:complexType>
928 <xsd:complexType name="CT_HandoutMasterIdList">
929   <xsd:sequence>
930     <xsd:element name="handoutMasterId" type="CT_HandoutMasterIdListEntry" minOccurs="0"
931       maxOccurs="1"/>
932   </xsd:sequence>
933 </xsd:complexType>
934 <xsd:complexType name="CT_EmbeddedFontDataId">
935   <xsd:attribute ref="r:id" use="required"/>
936 </xsd:complexType>
937 <xsd:complexType name="CT_EmbeddedFontListEntry">
938   <xsd:sequence>
939     <xsd:element name="font" type="a:CT_TextFont" minOccurs="1" maxOccurs="1"/>
940     <xsd:element name="regular" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
941     <xsd:element name="bold" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>

```



```

942     <xsd:element name="italic" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
943     <xsd:element name="boldItalic" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
944   </xsd:sequence>
945 </xsd:complexType>
946 <xsd:complexType name="CT_EmbeddedFontList">
947   <xsd:sequence>
948     <xsd:element name="embeddedFont" type="CT_EmbeddedFontListEntry" minOccurs="0"
949       maxOccurs="unbounded"/>
950   </xsd:sequence>
951 </xsd:complexType>
952 <xsd:complexType name="CT_SmartTags">
953   <xsd:attribute ref="r:id" use="required"/>
954 </xsd:complexType>
955 <xsd:complexType name="CT_CustomShow">
956   <xsd:sequence>
957     <xsd:element name="sldLst" type="CT_SlideRelationshipList" minOccurs="1" maxOccurs="1"/>
958     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
959   </xsd:sequence>
960   <xsd:attribute name="name" type="ST_Name" use="required"/>
961   <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
962 </xsd:complexType>
963 <xsd:complexType name="CT_CustomShowList">
964   <xsd:sequence>
965     <xsd:element name="custShow" type="CT_CustomShow" minOccurs="0" maxOccurs="unbounded"/>
966   </xsd:sequence>
967 </xsd:complexType>
968 <xsd:simpleType name="ST_PhotoAlbumLayout">
969   <xsd:restriction base="xsd:token">
970     <xsd:enumeration value="fitToSlide"/>
971     <xsd:enumeration value="1pic"/>
972     <xsd:enumeration value="2pic"/>
973     <xsd:enumeration value="4pic"/>
974     <xsd:enumeration value="1picTitle"/>
975     <xsd:enumeration value="2picTitle"/>
976     <xsd:enumeration value="4picTitle"/>
977   </xsd:restriction>
978 </xsd:simpleType>
979 <xsd:simpleType name="ST_PhotoAlbumFrameShape">
980   <xsd:restriction base="xsd:token">
981     <xsd:enumeration value="frameStyle1"/>
982     <xsd:enumeration value="frameStyle2"/>
983     <xsd:enumeration value="frameStyle3"/>
984     <xsd:enumeration value="frameStyle4"/>
985     <xsd:enumeration value="frameStyle5"/>
986     <xsd:enumeration value="frameStyle6"/>
987     <xsd:enumeration value="frameStyle7"/>
988   </xsd:restriction>
989 </xsd:simpleType>
990 <xsd:complexType name="CT_PhotoAlbum">
991   <xsd:sequence>
992     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
993   </xsd:sequence>
994   <xsd:attribute name="bw" type="xsd:boolean" use="optional" default="false"/>

```

```

995     <xsd:attribute name="showCaptions" type="xsd:boolean" use="optional" default="false"/>
996     <xsd:attribute name="layout" type="ST_PhotoAlbumLayout" use="optional" default="fitToSlide"/>
997     <xsd:attribute name="frame" type="ST_PhotoAlbumFrameShape" use="optional"
998         default="frameStyle1"/>
999 </xsd:complexType>
1000 <xsd:simpleType name="ST_SlideSizeCoordinate">
1001     <xsd:restriction base="a:ST_PositiveCoordinate32">
1002         <xsd:minInclusive value="914400"/>
1003         <xsd:maxInclusive value="51206400"/>
1004     </xsd:restriction>
1005 </xsd:simpleType>
1006 <xsd:simpleType name="ST_SlideSizeType">
1007     <xsd:restriction base="xsd:token">
1008         <xsd:enumeration value="screen4x3"/>
1009         <xsd:enumeration value="letter"/>
1010         <xsd:enumeration value="A4"/>
1011         <xsd:enumeration value="35mm"/>
1012         <xsd:enumeration value="overhead"/>
1013         <xsd:enumeration value="banner"/>
1014         <xsd:enumeration value="custom"/>
1015         <xsd:enumeration value="ledger"/>
1016         <xsd:enumeration value="A3"/>
1017         <xsd:enumeration value="B4ISO"/>
1018         <xsd:enumeration value="B5ISO"/>
1019         <xsd:enumeration value="B4JIS"/>
1020         <xsd:enumeration value="B5JIS"/>
1021         <xsd:enumeration value="hagakiCard"/>
1022         <xsd:enumeration value="screen16x9"/>
1023         <xsd:enumeration value="screen16x10"/>
1024     </xsd:restriction>
1025 </xsd:simpleType>
1026 <xsd:complexType name="CT_SlideSize">
1027     <xsd:attribute name="cx" type="ST_SlideSizeCoordinate" use="required"/>
1028     <xsd:attribute name="cy" type="ST_SlideSizeCoordinate" use="required"/>
1029     <xsd:attribute name="type" type="ST_SlideSizeType" use="optional" default="custom"/>
1030 </xsd:complexType>
1031 <xsd:complexType name="CT_Kinsoku">
1032     <xsd:attribute name="lang" type="xsd:string" use="optional"/>
1033     <xsd:attribute name="invalStChars" type="xsd:string" use="required"/>
1034     <xsd:attribute name="invalEndChars" type="xsd:string" use="required"/>
1035 </xsd:complexType>
1036 <xsd:simpleType name="ST_BookmarkIdSeed">
1037     <xsd:restriction base="xsd:unsignedInt">
1038         <xsd:minInclusive value="1"/>
1039         <xsd:maxExclusive value="2147483648"/>
1040     </xsd:restriction>
1041 </xsd:simpleType>
1042 <xsd:complexType name="CT_ModifyVerifier">
1043     <xsd:attribute name="algorithmName" type="xsd:string" use="optional"/>
1044     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
1045     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
1046     <xsd:attribute name="spinValue" type="xsd:unsignedInt" use="optional"/>
1047     <xsd:attribute name="cryptProviderType" type="s:ST_CryptProv" use="optional"/>

```

```

1048 <xsd:attribute name="cryptAlgorithmClass" type="s:ST AlgClass" use="optional"/>
1049 <xsd:attribute name="cryptAlgorithmType" type="s:ST AlgType" use="optional"/>
1050 <xsd:attribute name="cryptAlgorithmSid" type="xsd:unsignedInt" use="optional"/>
1051 <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
1052 <xsd:attribute name="saltData" type="xsd:base64Binary" use="optional"/>
1053 <xsd:attribute name="hashData" type="xsd:base64Binary" use="optional"/>
1054 <xsd:attribute name="cryptProvider" type="xsd:string" use="optional"/>
1055 <xsd:attribute name="algIdExt" type="xsd:unsignedInt" use="optional"/>
1056 <xsd:attribute name="algIdExtSource" type="xsd:string" use="optional"/>
1057 <xsd:attribute name="cryptProviderTypeExt" type="xsd:unsignedInt" use="optional"/>
1058 <xsd:attribute name="cryptProviderTypeExtSource" type="xsd:string" use="optional"/>
1059 </xsd:complexType>
1060 <xsd:complexType name="CT_Presentation">
1061   <xsd:sequence>
1062     <xsd:element name="sldMasterIdLst" type="CT SlideMasterIdList" minOccurs="0"
1063       maxOccurs="1"/>
1064     <xsd:element name="notesMasterIdLst" type="CT NotesMasterIdList" minOccurs="0"
1065       maxOccurs="1"/>
1066     <xsd:element name="handoutMasterIdLst" type="CT HandoutMasterIdList" minOccurs="0"
1067       maxOccurs="1"/>
1068     <xsd:element name="sldIdLst" type="CT SlideIdList" minOccurs="0" maxOccurs="1"/>
1069     <xsd:element name="sldSz" type="CT SlideSize" minOccurs="0" maxOccurs="1"/>
1070     <xsd:element name="notesSz" type="a:CT PositiveSize2D" minOccurs="1" maxOccurs="1"/>
1071     <xsd:element name="smartTags" type="CT SmartTags" minOccurs="0" maxOccurs="1"/>
1072     <xsd:element name="embeddedFontLst" type="CT EmbeddedFontList" minOccurs="0"
1073       maxOccurs="1"/>
1074     <xsd:element name="custShowLst" type="CT CustomShowList" minOccurs="0" maxOccurs="1"/>
1075     <xsd:element name="photoAlbum" type="CT PhotoAlbum" minOccurs="0" maxOccurs="1"/>
1076     <xsd:element name="custDataLst" type="CT CustomerDataList" minOccurs="0" maxOccurs="1"/>
1077     <xsd:element name="kinsoku" type="CT Kinsoku" minOccurs="0"/>
1078     <xsd:element name="defaultTextStyle" type="a:CT TextListStyle" minOccurs="0"
1079       maxOccurs="1"/>
1080     <xsd:element name="modifyVerifier" type="CT ModifyVerifier" minOccurs="0" maxOccurs="1"/>
1081     <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1082   </xsd:sequence>
1083   <xsd:attribute name="serverZoom" type="a:ST Percentage" use="optional" default="50%"/>
1084   <xsd:attribute name="firstSlideNum" type="xsd:int" use="optional" default="1"/>
1085   <xsd:attribute name="showSpecialPlsOnTitleSld" type="xsd:boolean" use="optional"
1086     default="true"/>
1087   <xsd:attribute name="rtl" type="xsd:boolean" use="optional" default="false"/>
1088   <xsd:attribute name="removePersonalInfoOnSave" type="xsd:boolean" use="optional"
1089     default="false"/>
1090   <xsd:attribute name="compatMode" type="xsd:boolean" use="optional" default="false"/>
1091   <xsd:attribute name="strictFirstAndLastChars" type="xsd:boolean" use="optional"
1092     default="true"/>
1093   <xsd:attribute name="embedTrueTypeFonts" type="xsd:boolean" use="optional" default="false"/>
1094   <xsd:attribute name="saveSubsetFonts" type="xsd:boolean" use="optional" default="false"/>
1095   <xsd:attribute name="autoCompressPictures" type="xsd:boolean" use="optional" default="true"/>
1096   <xsd:attribute name="bookmarkIdSeed" type="ST BookmarkIdSeed" use="optional" default="1"/>
1097   <xsd:attribute name="conformance" type="s:ST ConformanceClass"/>
1098 </xsd:complexType>
1099 <xsd:element name="presentation" type="CT Presentation"/>
1100 <xsd:complexType name="CT_HtmlPublishProperties">

```

```

1101     <xsd:sequence>
1102         <xsd:group ref="EG_SlideListChoice" minOccurs="1" maxOccurs="1"/>
1103         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1104     </xsd:sequence>
1105     <xsd:attribute name="showSpeakerNotes" type="xsd:boolean" use="optional" default="true"/>
1106     <xsd:attribute name="target" type="xsd:string" use="optional"/>
1107     <xsd:attribute name="title" type="xsd:string" use="optional" default=""/>
1108     <xsd:attribute ref="r:id" use="required"/>
1109 </xsd:complexType>
1110 <xsd:simpleType name="ST_WebColorType">
1111     <xsd:restriction base="xsd:token">
1112         <xsd:enumeration value="none"/>
1113         <xsd:enumeration value="browser"/>
1114         <xsd:enumeration value="presentationText"/>
1115         <xsd:enumeration value="presentationAccent"/>
1116         <xsd:enumeration value="whiteTextOnBlack"/>
1117         <xsd:enumeration value="blackTextOnWhite"/>
1118     </xsd:restriction>
1119 </xsd:simpleType>
1120 <xsd:simpleType name="ST_WebScreenSize">
1121     <xsd:restriction base="xsd:token">
1122         <xsd:enumeration value="544x376"/>
1123         <xsd:enumeration value="640x480"/>
1124         <xsd:enumeration value="720x512"/>
1125         <xsd:enumeration value="800x600"/>
1126         <xsd:enumeration value="1024x768"/>
1127         <xsd:enumeration value="1152x882"/>
1128         <xsd:enumeration value="1152x900"/>
1129         <xsd:enumeration value="1280x1024"/>
1130         <xsd:enumeration value="1600x1200"/>
1131         <xsd:enumeration value="1800x1400"/>
1132         <xsd:enumeration value="1920x1200"/>
1133     </xsd:restriction>
1134 </xsd:simpleType>
1135 <xsd:simpleType name="ST_WebEncoding">
1136     <xsd:restriction base="xsd:string"/>
1137 </xsd:simpleType>
1138 <xsd:complexType name="CT_WebProperties">
1139     <xsd:sequence>
1140         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1141     </xsd:sequence>
1142     <xsd:attribute name="showAnimation" type="xsd:boolean" use="optional" default="false"/>
1143     <xsd:attribute name="resizeGraphics" type="xsd:boolean" use="optional" default="true"/>
1144     <xsd:attribute name="allowPng" type="xsd:boolean" use="optional" default="false"/>
1145     <xsd:attribute name="relyOnVml" type="xsd:boolean" use="optional" default="false"/>
1146     <xsd:attribute name="organizeInFolders" type="xsd:boolean" use="optional" default="true"/>
1147     <xsd:attribute name="useLongFileNames" type="xsd:boolean" use="optional" default="true"/>
1148     <xsd:attribute name="imgSz" type="ST_WebScreenSize" use="optional" default="800x600"/>
1149     <xsd:attribute name="encoding" type="ST_WebEncoding" use="optional" default=""/>
1150     <xsd:attribute name="clr" type="ST_WebColorType" use="optional" default="whiteTextOnBlack"/>
1151 </xsd:complexType>
1152 <xsd:simpleType name="ST_PrintWhat">
1153     <xsd:restriction base="xsd:token">

```

```

1154         <xsd:enumeration value="slides"/>
1155         <xsd:enumeration value="handouts1"/>
1156         <xsd:enumeration value="handouts2"/>
1157         <xsd:enumeration value="handouts3"/>
1158         <xsd:enumeration value="handouts4"/>
1159         <xsd:enumeration value="handouts6"/>
1160         <xsd:enumeration value="handouts9"/>
1161         <xsd:enumeration value="notes"/>
1162         <xsd:enumeration value="outline"/>
1163     </xsd:restriction>
1164 </xsd:simpleType>
1165 <xsd:simpleType name="ST_PrintColorMode">
1166     <xsd:restriction base="xsd:token">
1167         <xsd:enumeration value="bw"/>
1168         <xsd:enumeration value="gray"/>
1169         <xsd:enumeration value="clr"/>
1170     </xsd:restriction>
1171 </xsd:simpleType>
1172 <xsd:complexType name="CT_PrintProperties">
1173     <xsd:sequence>
1174         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1175     </xsd:sequence>
1176     <xsd:attribute name="prnWhat" type="ST_PrintWhat" use="optional" default="slides"/>
1177     <xsd:attribute name="clrMode" type="ST_PrintColorMode" use="optional" default="clr"/>
1178     <xsd:attribute name="hiddenSlides" type="xsd:boolean" use="optional" default="false"/>
1179     <xsd:attribute name="scaleToFitPaper" type="xsd:boolean" use="optional" default="false"/>
1180     <xsd:attribute name="frameSlides" type="xsd:boolean" use="optional" default="false"/>
1181 </xsd:complexType>
1182 <xsd:complexType name="CT_ShowInfoBrowse">
1183     <xsd:attribute name="showScrollbar" type="xsd:boolean" use="optional" default="true"/>
1184 </xsd:complexType>
1185 <xsd:complexType name="CT_ShowInfoKiosk">
1186     <xsd:attribute name="restart" type="xsd:unsignedInt" use="optional" default="300000"/>
1187 </xsd:complexType>
1188 <xsd:group name="EG_ShowType">
1189     <xsd:choice>
1190         <xsd:element name="present" type="CT_Empty"/>
1191         <xsd:element name="browse" type="CT_ShowInfoBrowse"/>
1192         <xsd:element name="kiosk" type="CT_ShowInfoKiosk"/>
1193     </xsd:choice>
1194 </xsd:group>
1195 <xsd:complexType name="CT_ShowProperties">
1196     <xsd:sequence minOccurs="0" maxOccurs="1">
1197         <xsd:group ref="EG_ShowType" minOccurs="0" maxOccurs="1"/>
1198         <xsd:group ref="EG_SlideListChoice" minOccurs="0" maxOccurs="1"/>
1199         <xsd:element name="penClr" type="a:CT_Color" minOccurs="0" maxOccurs="1"/>
1200         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1201     </xsd:sequence>
1202     <xsd:attribute name="loop" type="xsd:boolean" use="optional" default="false"/>
1203     <xsd:attribute name="showNarration" type="xsd:boolean" use="optional" default="false"/>
1204     <xsd:attribute name="showAnimation" type="xsd:boolean" use="optional" default="true"/>
1205     <xsd:attribute name="useTimings" type="xsd:boolean" use="optional" default="true"/>
1206 </xsd:complexType>

```

```

1207 <xsd:complexType name="CT_PresentationProperties">
1208   <xsd:sequence>
1209     <xsd:element name="htmlPubPr" type="CT_HtmlPublishProperties" minOccurs="0"
1210       maxOccurs="1"/>
1211     <xsd:element name="webPr" type="CT_WebProperties" minOccurs="0" maxOccurs="1"/>
1212     <xsd:element name="prnPr" type="CT_PrintProperties" minOccurs="0" maxOccurs="1"/>
1213     <xsd:element name="showPr" type="CT_ShowProperties" minOccurs="0" maxOccurs="1"/>
1214     <xsd:element name="clrMru" type="a:CT_ColorMRU" minOccurs="0" maxOccurs="1"/>
1215     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1216   </xsd:sequence>
1217 </xsd:complexType>
1218 <xsd:element name="presentationPr" type="CT_PresentationProperties"/>
1219 <xsd:complexType name="CT_HeaderFooter">
1220   <xsd:sequence>
1221     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1222   </xsd:sequence>
1223   <xsd:attribute name="sldNum" type="xsd:boolean" use="optional" default="true"/>
1224   <xsd:attribute name="hdr" type="xsd:boolean" use="optional" default="true"/>
1225   <xsd:attribute name="ftr" type="xsd:boolean" use="optional" default="true"/>
1226   <xsd:attribute name="dt" type="xsd:boolean" use="optional" default="true"/>
1227 </xsd:complexType>
1228 <xsd:simpleType name="ST_PlaceholderType">
1229   <xsd:restriction base="xsd:token">
1230     <xsd:enumeration value="title"/>
1231     <xsd:enumeration value="body"/>
1232     <xsd:enumeration value="ctrTitle"/>
1233     <xsd:enumeration value="subTitle"/>
1234     <xsd:enumeration value="dt"/>
1235     <xsd:enumeration value="sldNum"/>
1236     <xsd:enumeration value="ftr"/>
1237     <xsd:enumeration value="hdr"/>
1238     <xsd:enumeration value="obj"/>
1239     <xsd:enumeration value="chart"/>
1240     <xsd:enumeration value="tbl"/>
1241     <xsd:enumeration value="clipArt"/>
1242     <xsd:enumeration value="dgm"/>
1243     <xsd:enumeration value="media"/>
1244     <xsd:enumeration value="sldImg"/>
1245     <xsd:enumeration value="pic"/>
1246   </xsd:restriction>
1247 </xsd:simpleType>
1248 <xsd:simpleType name="ST_PlaceholderSize">
1249   <xsd:restriction base="xsd:token">
1250     <xsd:enumeration value="full"/>
1251     <xsd:enumeration value="half"/>
1252     <xsd:enumeration value="quarter"/>
1253   </xsd:restriction>
1254 </xsd:simpleType>
1255 <xsd:complexType name="CT_Placeholder">
1256   <xsd:sequence>
1257     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1258   </xsd:sequence>
1259   <xsd:attribute name="type" type="ST_PlaceholderType" use="optional" default="obj"/>

```

```

1260     <xsd:attribute name="orient" type="ST_Direction" use="optional" default="horz"/>
1261     <xsd:attribute name="sz" type="ST_PlaceholderSize" use="optional" default="full"/>
1262     <xsd:attribute name="idx" type="xsd:unsignedInt" use="optional" default="0"/>
1263     <xsd:attribute name="hasCustomPrompt" type="xsd:boolean" use="optional" default="false"/>
1264 </xsd:complexType>
1265 <xsd:complexType name="CT_ApplicationNonVisualDrawingProps">
1266     <xsd:sequence>
1267         <xsd:element name="ph" type="CT_Placeholder" minOccurs="0" maxOccurs="1"/>
1268         <xsd:group ref="a:EG_Media" minOccurs="0" maxOccurs="1"/>
1269         <xsd:element name="custDataLst" type="CT_CustomerDataList" minOccurs="0" maxOccurs="1"/>
1270         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1271     </xsd:sequence>
1272     <xsd:attribute name="isPhoto" type="xsd:boolean" use="optional" default="false"/>
1273     <xsd:attribute name="userDrawn" type="xsd:boolean" use="optional" default="false"/>
1274 </xsd:complexType>
1275 <xsd:complexType name="CT_ShapeNonVisual">
1276     <xsd:sequence>
1277         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1278         <xsd:element name="cNvSpPr" type="a:CT_NonVisualDrawingShapeProps" minOccurs="1"
1279             maxOccurs="1"/>
1280         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1281             maxOccurs="1"/>
1282     </xsd:sequence>
1283 </xsd:complexType>
1284 <xsd:complexType name="CT_Shape">
1285     <xsd:sequence>
1286         <xsd:element name="nvSpPr" type="CT_ShapeNonVisual" minOccurs="1" maxOccurs="1"/>
1287         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1288         <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1289         <xsd:element name="txBody" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1290         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1291     </xsd:sequence>
1292     <xsd:attribute name="useBgFill" type="xsd:boolean" use="optional" default="false"/>
1293 </xsd:complexType>
1294 <xsd:complexType name="CT_ConnectorNonVisual">
1295     <xsd:sequence>
1296         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1297         <xsd:element name="cNvCxnSpPr" type="a:CT_NonVisualConnectorProperties" minOccurs="1"
1298             maxOccurs="1"/>
1299         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1300             maxOccurs="1"/>
1301     </xsd:sequence>
1302 </xsd:complexType>
1303 <xsd:complexType name="CT_Connector">
1304     <xsd:sequence>
1305         <xsd:element name="nvCxnSpPr" type="CT_ConnectorNonVisual" minOccurs="1" maxOccurs="1"/>
1306         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1307         <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1308         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1309     </xsd:sequence>
1310 </xsd:complexType>
1311 <xsd:complexType name="CT_PictureNonVisual">
1312     <xsd:sequence>

```

```

1313     <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1314     <xsd:element name="cNvPicPr" type="a:CT_NonVisualPictureProperties" minOccurs="1"
1315         maxOccurs="1"/>
1316     <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1317         maxOccurs="1"/>
1318 </xsd:sequence>
1319 </xsd:complexType>
1320 <xsd:complexType name="CT_Picture">
1321     <xsd:sequence>
1322         <xsd:element name="nvPicPr" type="CT_PictureNonVisual" minOccurs="1" maxOccurs="1"/>
1323         <xsd:element name="blipFill" type="a:CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
1324         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1325         <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1326         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1327     </xsd:sequence>
1328 </xsd:complexType>
1329 <xsd:complexType name="CT_GraphicalObjectFrameNonVisual">
1330     <xsd:sequence>
1331         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1332         <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
1333             minOccurs="1" maxOccurs="1"/>
1334         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1335             maxOccurs="1"/>
1336     </xsd:sequence>
1337 </xsd:complexType>
1338 <xsd:complexType name="CT_GraphicalObjectFrame">
1339     <xsd:sequence>
1340         <xsd:element name="nvGraphicFramePr" type="CT_GraphicalObjectFrameNonVisual" minOccurs="1"
1341             maxOccurs="1"/>
1342         <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="1" maxOccurs="1"/>
1343         <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
1344         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1345     </xsd:sequence>
1346     <xsd:attribute name="bwMode" type="a:ST_BlackWhiteMode" use="optional"/>
1347 </xsd:complexType>
1348 <xsd:complexType name="CT_GroupShapeNonVisual">
1349     <xsd:sequence>
1350         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1351         <xsd:element name="cNvGrpSpPr" type="a:CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
1352             maxOccurs="1"/>
1353         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1354             maxOccurs="1"/>
1355     </xsd:sequence>
1356 </xsd:complexType>
1357 <xsd:complexType name="CT_GroupShape">
1358     <xsd:sequence>
1359         <xsd:element name="nvGrpSpPr" type="CT_GroupShapeNonVisual" minOccurs="1" maxOccurs="1"/>
1360         <xsd:element name="grpSpPr" type="a:CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
1361         <xsd:choice minOccurs="0" maxOccurs="unbounded">
1362             <xsd:element name="sp" type="CT_Shape"/>
1363             <xsd:element name="grpSp" type="CT_GroupShape"/>
1364             <xsd:element name="graphicFrame" type="CT_GraphicalObjectFrame"/>
1365             <xsd:element name="cxnSp" type="CT_Connector"/>

```



```

1366         <xsd:element name="pic" type="CT Picture"/>
1367         <xsd:element name="contentPart" type="CT Rel"/>
1368     </xsd:choice>
1369     <xsd:element name="extLst" type="CT ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1370 </xsd:sequence>
1371 </xsd:complexType>
1372 <xsd:complexType name="CT_Rel">
1373     <xsd:attribute ref="r:id" use="required"/>
1374 </xsd:complexType>
1375 <xsd:group name="EG_TopLevelSlide">
1376     <xsd:sequence>
1377         <xsd:element name="clrMap" type="a:CT ColorMapping" minOccurs="1" maxOccurs="1"/>
1378     </xsd:sequence>
1379 </xsd:group>
1380 <xsd:group name="EG_ChildSlide">
1381     <xsd:sequence>
1382         <xsd:element name="clrMapOvr" type="a:CT ColorMappingOverride" minOccurs="0"
1383             maxOccurs="1"/>
1384     </xsd:sequence>
1385 </xsd:group>
1386 <xsd:attributeGroup name="AG_ChildSlide">
1387     <xsd:attribute name="showMasterSp" type="xsd:boolean" use="optional" default="true"/>
1388     <xsd:attribute name="showMasterPhAnim" type="xsd:boolean" use="optional" default="true"/>
1389 </xsd:attributeGroup>
1390 <xsd:complexType name="CT_BackgroundProperties">
1391     <xsd:sequence>
1392         <xsd:group ref="a:EG FillProperties" minOccurs="1" maxOccurs="1"/>
1393         <xsd:group ref="a:EG EffectProperties" minOccurs="0" maxOccurs="1"/>
1394         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1395     </xsd:sequence>
1396     <xsd:attribute name="shadeToTitle" type="xsd:boolean" use="optional" default="false"/>
1397 </xsd:complexType>
1398 <xsd:group name="EG_Background">
1399     <xsd:choice>
1400         <xsd:element name="bgPr" type="CT BackgroundProperties"/>
1401         <xsd:element name="bgRef" type="a:CT StyleMatrixReference"/>
1402     </xsd:choice>
1403 </xsd:group>
1404 <xsd:complexType name="CT_Background">
1405     <xsd:sequence>
1406         <xsd:group ref="EG_Background"/>
1407     </xsd:sequence>
1408     <xsd:attribute name="bwMode" type="a:ST BlackWhiteMode" use="optional" default="white"/>
1409 </xsd:complexType>
1410 <xsd:complexType name="CT_CommonSlideData">
1411     <xsd:sequence>
1412         <xsd:element name="bg" type="CT Background" minOccurs="0" maxOccurs="1"/>
1413         <xsd:element name="spTree" type="CT GroupShape" minOccurs="1" maxOccurs="1"/>
1414         <xsd:element name="custDataLst" type="CT CustomerDataList" minOccurs="0" maxOccurs="1"/>
1415         <xsd:element name="controls" type="CT ControlList" minOccurs="0" maxOccurs="1"/>
1416         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1417     </xsd:sequence>
1418     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>

```

```

1419 </xsd:complexType>
1420 <xsd:complexType name="CT_Slide">
1421   <xsd:sequence minOccurs="1" maxOccurs="1">
1422     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1423     <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1424     <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1425     <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1426     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1427   </xsd:sequence>
1428   <xsd:attributeGroup ref="AG_ChildSlide"/>
1429   <xsd:attribute name="show" type="xsd:boolean" use="optional" default="true"/>
1430 </xsd:complexType>
1431 <xsd:element name="sld" type="CT_Slide"/>
1432 <xsd:simpleType name="ST_SlideLayoutType">
1433   <xsd:restriction base="xsd:token">
1434     <xsd:enumeration value="title"/>
1435     <xsd:enumeration value="tx"/>
1436     <xsd:enumeration value="twoColTx"/>
1437     <xsd:enumeration value="tbl"/>
1438     <xsd:enumeration value="txAndChart"/>
1439     <xsd:enumeration value="chartAndTx"/>
1440     <xsd:enumeration value="dgm"/>
1441     <xsd:enumeration value="chart"/>
1442     <xsd:enumeration value="txAndClipArt"/>
1443     <xsd:enumeration value="clipArtAndTx"/>
1444     <xsd:enumeration value="titleOnly"/>
1445     <xsd:enumeration value="blank"/>
1446     <xsd:enumeration value="txAndObj"/>
1447     <xsd:enumeration value="objAndTx"/>
1448     <xsd:enumeration value="objOnly"/>
1449     <xsd:enumeration value="obj"/>
1450     <xsd:enumeration value="txAndMedia"/>
1451     <xsd:enumeration value="mediaAndTx"/>
1452     <xsd:enumeration value="objOverTx"/>
1453     <xsd:enumeration value="txOverObj"/>
1454     <xsd:enumeration value="txAndTwoObj"/>
1455     <xsd:enumeration value="twoObjAndTx"/>
1456     <xsd:enumeration value="twoObjOverTx"/>
1457     <xsd:enumeration value="fourObj"/>
1458     <xsd:enumeration value="vertTx"/>
1459     <xsd:enumeration value="clipArtAndVertTx"/>
1460     <xsd:enumeration value="vertTitleAndTx"/>
1461     <xsd:enumeration value="vertTitleAndTxOverChart"/>
1462     <xsd:enumeration value="twoObj"/>
1463     <xsd:enumeration value="objAndTwoObj"/>
1464     <xsd:enumeration value="twoObjAndObj"/>
1465     <xsd:enumeration value="cust"/>
1466     <xsd:enumeration value="secHead"/>
1467     <xsd:enumeration value="twoTxTwoObj"/>
1468     <xsd:enumeration value="objTx"/>
1469     <xsd:enumeration value="picTx"/>
1470   </xsd:restriction>
1471 </xsd:simpleType>

```

```

1472 <xsd:complexType name="CT_SlideLayout">
1473   <xsd:sequence minOccurs="1" maxOccurs="1">
1474     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1475     <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1476     <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1477     <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1478     <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1479     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1480   </xsd:sequence>
1481   <xsd:attributeGroup ref="AG_ChildSlide"/>
1482   <xsd:attribute name="matchingName" type="xsd:string" use="optional" default=""/>
1483   <xsd:attribute name="type" type="ST_SlideLayoutType" use="optional" default="cust"/>
1484   <xsd:attribute name="preserve" type="xsd:boolean" use="optional" default="false"/>
1485   <xsd:attribute name="userDrawn" type="xsd:boolean" use="optional" default="false"/>
1486 </xsd:complexType>
1487 <xsd:element name="sldLayout" type="CT_SlideLayout"/>
1488 <xsd:complexType name="CT_SlideMasterTextStyles">
1489   <xsd:sequence>
1490     <xsd:element name="titleStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1491     <xsd:element name="bodyStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1492     <xsd:element name="otherStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1493     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1494   </xsd:sequence>
1495 </xsd:complexType>
1496 <xsd:simpleType name="ST_SlideLayoutId">
1497   <xsd:restriction base="xsd:unsignedInt">
1498     <xsd:minInclusive value="2147483648"/>
1499   </xsd:restriction>
1500 </xsd:simpleType>
1501 <xsd:complexType name="CT_SlideLayoutIdListEntry">
1502   <xsd:sequence>
1503     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1504   </xsd:sequence>
1505   <xsd:attribute name="id" type="ST_SlideLayoutId" use="optional"/>
1506   <xsd:attribute ref="r:id" use="required"/>
1507 </xsd:complexType>
1508 <xsd:complexType name="CT_SlideLayoutIdList">
1509   <xsd:sequence>
1510     <xsd:element name="sldLayoutId" type="CT_SlideLayoutIdListEntry" minOccurs="0"
1511       maxOccurs="unbounded"/>
1512   </xsd:sequence>
1513 </xsd:complexType>
1514 <xsd:complexType name="CT_SlideMaster">
1515   <xsd:sequence minOccurs="1" maxOccurs="1">
1516     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1517     <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1518     <xsd:element name="sldLayoutIdList" type="CT_SlideLayoutIdList" minOccurs="0"
1519       maxOccurs="1"/>
1520     <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1521     <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1522     <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1523     <xsd:element name="txStyles" type="CT_SlideMasterTextStyles" minOccurs="0" maxOccurs="1"/>
1524     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>

```

```

1525     </xsd:sequence>
1526     <xsd:attribute name="preserve" type="xsd:boolean" use="optional" default="false"/>
1527 </xsd:complexType>
1528 <xsd:element name="sldMaster" type="CT_SlideMaster"/>
1529 <xsd:complexType name="CT_HandoutMaster">
1530     <xsd:sequence>
1531         <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1532         <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1533         <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1534         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1535     </xsd:sequence>
1536 </xsd:complexType>
1537 <xsd:element name="handoutMaster" type="CT_HandoutMaster"/>
1538 <xsd:complexType name="CT_NotesMaster">
1539     <xsd:sequence>
1540         <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1541         <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1542         <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1543         <xsd:element name="notesStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1544         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1545     </xsd:sequence>
1546 </xsd:complexType>
1547 <xsd:element name="notesMaster" type="CT_NotesMaster"/>
1548 <xsd:complexType name="CT_NotesSlide">
1549     <xsd:sequence minOccurs="1" maxOccurs="1">
1550         <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1551         <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1552         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1553     </xsd:sequence>
1554     <xsd:attributeGroup ref="AG_ChildSlide"/>
1555 </xsd:complexType>
1556 <xsd:element name="notes" type="CT_NotesSlide"/>
1557 <xsd:complexType name="CT_SlideSyncProperties">
1558     <xsd:sequence>
1559         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1560     </xsd:sequence>
1561     <xsd:attribute name="serverSldId" type="xsd:string" use="required"/>
1562     <xsd:attribute name="serverSldModifiedTime" type="xsd:dateTime" use="required"/>
1563     <xsd:attribute name="clientInsertedTime" type="xsd:dateTime" use="required"/>
1564 </xsd:complexType>
1565 <xsd:element name="sldSyncPr" type="CT_SlideSyncProperties"/>
1566 <xsd:complexType name="CT_StringTag">
1567     <xsd:attribute name="name" type="xsd:string" use="required"/>
1568     <xsd:attribute name="val" type="xsd:string" use="required"/>
1569 </xsd:complexType>
1570 <xsd:complexType name="CT_TagList">
1571     <xsd:sequence>
1572         <xsd:element name="tag" type="CT_StringTag" minOccurs="0" maxOccurs="unbounded"/>
1573     </xsd:sequence>
1574 </xsd:complexType>
1575 <xsd:element name="tagLst" type="CT_TagList"/>
1576 <xsd:simpleType name="ST_SplitterBarState">
1577     <xsd:restriction base="xsd:token">

```

```

1578         <xsd:enumeration value="minimized"/>
1579         <xsd:enumeration value="restored"/>
1580         <xsd:enumeration value="maximized"/>
1581     </xsd:restriction>
1582 </xsd:simpleType>
1583 <xsd:simpleType name="ST_ViewType">
1584     <xsd:restriction base="xsd:token">
1585         <xsd:enumeration value="sldView"/>
1586         <xsd:enumeration value="sldMasterView"/>
1587         <xsd:enumeration value="notesView"/>
1588         <xsd:enumeration value="handoutView"/>
1589         <xsd:enumeration value="notesMasterView"/>
1590         <xsd:enumeration value="outlineView"/>
1591         <xsd:enumeration value="sldSorterView"/>
1592         <xsd:enumeration value="sldThumbnailView"/>
1593     </xsd:restriction>
1594 </xsd:simpleType>
1595 <xsd:complexType name="CT_NormalViewPortion">
1596     <xsd:attribute name="sz" type="a:ST_PositiveFixedPercentage" use="required"/>
1597     <xsd:attribute name="autoAdjust" type="xsd:boolean" use="optional" default="true"/>
1598 </xsd:complexType>
1599 <xsd:complexType name="CT_NormalViewProperties">
1600     <xsd:sequence>
1601         <xsd:element name="restoredLeft" type="CT_NormalViewPortion" minOccurs="1" maxOccurs="1"/>
1602         <xsd:element name="restoredTop" type="CT_NormalViewPortion" minOccurs="1" maxOccurs="1"/>
1603         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1604     </xsd:sequence>
1605     <xsd:attribute name="showOutlineIcons" type="xsd:boolean" use="optional" default="true"/>
1606     <xsd:attribute name="snapVertSplitter" type="xsd:boolean" use="optional" default="false"/>
1607     <xsd:attribute name="vertBarState" type="ST_SplitterBarState" use="optional"
1608         default="restored"/>
1609     <xsd:attribute name="horzBarState" type="ST_SplitterBarState" use="optional"
1610         default="restored"/>
1611     <xsd:attribute name="preferSingleView" type="xsd:boolean" use="optional" default="false"/>
1612 </xsd:complexType>
1613 <xsd:complexType name="CT_CommonViewProperties">
1614     <xsd:sequence>
1615         <xsd:element name="scale" type="a:CT_Scale2D" minOccurs="1" maxOccurs="1"/>
1616         <xsd:element name="origin" type="a:CT_Point2D" minOccurs="1" maxOccurs="1"/>
1617     </xsd:sequence>
1618     <xsd:attribute name="varScale" type="xsd:boolean" use="optional" default="false"/>
1619 </xsd:complexType>
1620 <xsd:complexType name="CT_NotesTextViewProperties">
1621     <xsd:sequence minOccurs="1" maxOccurs="1">
1622         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1623         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1624     </xsd:sequence>
1625 </xsd:complexType>
1626 <xsd:complexType name="CT_OutlineViewSlideEntry">
1627     <xsd:attribute ref="r:id" use="required"/>
1628     <xsd:attribute name="collapse" type="xsd:boolean" use="optional" default="false"/>
1629 </xsd:complexType>
1630 <xsd:complexType name="CT_OutlineViewSlidelist">

```

```

1631     <xsd:sequence>
1632         <xsd:element name="sld" type="CT_OutlineViewSlideEntry" minOccurs="0"
1633             maxOccurs="unbounded"/>
1634     </xsd:sequence>
1635 </xsd:complexType>
1636 <xsd:complexType name="CT_OutlineViewProperties">
1637     <xsd:sequence minOccurs="1" maxOccurs="1">
1638         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1639         <xsd:element name="sldLst" type="CT_OutlineViewSlideList" minOccurs="0" maxOccurs="1"/>
1640         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1641     </xsd:sequence>
1642 </xsd:complexType>
1643 <xsd:complexType name="CT_SlideSorterViewProperties">
1644     <xsd:sequence minOccurs="1" maxOccurs="1">
1645         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1646         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1647     </xsd:sequence>
1648     <xsd:attribute name="showFormatting" type="xsd:boolean" use="optional" default="true"/>
1649 </xsd:complexType>
1650 <xsd:complexType name="CT_Guide">
1651     <xsd:attribute name="orient" type="ST_Direction" use="optional" default="vert"/>
1652     <xsd:attribute name="pos" type="a:ST_Coordinate32" use="optional" default="0"/>
1653 </xsd:complexType>
1654 <xsd:complexType name="CT_GuideList">
1655     <xsd:sequence minOccurs="0" maxOccurs="1">
1656         <xsd:element name="guide" type="CT_Guide" minOccurs="0" maxOccurs="unbounded"/>
1657     </xsd:sequence>
1658 </xsd:complexType>
1659 <xsd:complexType name="CT_CommonSlideViewProperties">
1660     <xsd:sequence>
1661         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1662         <xsd:element name="guideLst" type="CT_GuideList" minOccurs="0" maxOccurs="1"/>
1663     </xsd:sequence>
1664     <xsd:attribute name="snapToGrid" type="xsd:boolean" use="optional" default="true"/>
1665     <xsd:attribute name="snapToObjects" type="xsd:boolean" use="optional" default="false"/>
1666     <xsd:attribute name="showGuides" type="xsd:boolean" use="optional" default="false"/>
1667 </xsd:complexType>
1668 <xsd:complexType name="CT_SlideViewProperties">
1669     <xsd:sequence>
1670         <xsd:element name="cSldViewPr" type="CT_CommonSlideViewProperties" minOccurs="1"
1671             maxOccurs="1"/>
1672         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1673     </xsd:sequence>
1674 </xsd:complexType>
1675 <xsd:complexType name="CT_NotesViewProperties">
1676     <xsd:sequence>
1677         <xsd:element name="cSldViewPr" type="CT_CommonSlideViewProperties" minOccurs="1"
1678             maxOccurs="1"/>
1679         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1680     </xsd:sequence>
1681 </xsd:complexType>
1682 <xsd:complexType name="CT_ViewProperties">
1683     <xsd:sequence minOccurs="0" maxOccurs="1">

```

```

1684     <xsd:element name="normalViewPr" type="CT_NormalViewProperties" minOccurs="0"
1685         maxOccurs="1"/>
1686     <xsd:element name="slideViewPr" type="CT_SlideViewProperties" minOccurs="0"
1687         maxOccurs="1"/>
1688     <xsd:element name="outlineViewPr" type="CT_OutlineViewProperties" minOccurs="0"
1689         maxOccurs="1"/>
1690     <xsd:element name="notesTextViewPr" type="CT_NotesTextViewProperties" minOccurs="0"
1691         maxOccurs="1"/>
1692     <xsd:element name="sorterViewPr" type="CT_SlideSorterViewProperties" minOccurs="0"
1693         maxOccurs="1"/>
1694     <xsd:element name="notesViewPr" type="CT_NotesViewProperties" minOccurs="0"
1695         maxOccurs="1"/>
1696     <xsd:element name="gridSpacing" type="a:CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
1697     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1698 </xsd:sequence>
1699     <xsd:attribute name="lastView" type="ST_ViewType" use="optional" default="sldView"/>
1700     <xsd:attribute name="showComments" type="xsd:boolean" use="optional" default="true"/>
1701 </xsd:complexType>
1702     <xsd:element name="viewPr" type="CT_ViewProperties"/>
1703 </xsd:schema>

```

A.4 DrawingML - Framework

A.4.1 DrawingML - Main

This schema is available in the file dml-main.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
3   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4   xmlns="http://schemas.openxmlformats.org/drawingml/2006/main"
5   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/main"
6   elementFormDefault="qualified">
7   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
8       schemaLocation="shared-relationshipReference.xsd"/>
9   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
10       schemaLocation="shared-commonSimpleTypes.xsd"/>
11   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/diagram"
12       schemaLocation="dml-diagram.xsd"/>
13   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/chart"
14       schemaLocation="dml-chart.xsd"/>
15   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/picture"
16       schemaLocation="dml-picture.xsd"/>
17   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas"
18       schemaLocation="dml-lockedCanvas.xsd"/>
19   <xsd:complexType name="CT_AudioFile">
20       <xsd:sequence>
21           <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
22       </xsd:sequence>
23       <xsd:attribute ref="r:link" use="required"/>
24       <xsd:attribute name="contentType" type="xsd:string" use="optional"/>
25   </xsd:complexType>
26   <xsd:complexType name="CT_VideoFile">

```

```

27     <xsd:sequence>
28         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
29     </xsd:sequence>
30     <xsd:attribute ref="r:link" use="required"/>
31     <xsd:attribute name="contentType" type="xsd:string" use="optional"/>
32 </xsd:complexType>
33 <xsd:complexType name="CT_QuickTimeFile">
34     <xsd:sequence>
35         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
36     </xsd:sequence>
37     <xsd:attribute ref="r:link" use="required"/>
38 </xsd:complexType>
39 <xsd:complexType name="CT_AudioCDTime">
40     <xsd:attribute name="track" type="xsd:unsignedByte" use="required"/>
41     <xsd:attribute name="time" type="xsd:unsignedInt" use="optional" default="0"/>
42 </xsd:complexType>
43 <xsd:complexType name="CT_AudioCD">
44     <xsd:sequence>
45         <xsd:element name="st" type="CT_AudioCDTime" minOccurs="1" maxOccurs="1"/>
46         <xsd:element name="end" type="CT_AudioCDTime" minOccurs="1" maxOccurs="1"/>
47         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
48     </xsd:sequence>
49 </xsd:complexType>
50 <xsd:group name="EG_Media">
51     <xsd:choice>
52         <xsd:element name="audioCd" type="CT_AudioCD"/>
53         <xsd:element name="wavAudioFile" type="CT_EmbeddedWAVAudioFile"/>
54         <xsd:element name="audioFile" type="CT_AudioFile"/>
55         <xsd:element name="videoFile" type="CT_VideoFile"/>
56         <xsd:element name="quickTimeFile" type="CT_QuickTimeFile"/>
57     </xsd:choice>
58 </xsd:group>
59 <xsd:element name="videoFile" type="CT_VideoFile"/>
60 <xsd:simpleType name="ST_StyleMatrixColumnIndex">
61     <xsd:restriction base="xsd:unsignedInt"/>
62 </xsd:simpleType>
63 <xsd:simpleType name="ST_FontCollectionIndex">
64     <xsd:restriction base="xsd:token">
65         <xsd:enumeration value="major"/>
66         <xsd:enumeration value="minor"/>
67         <xsd:enumeration value="none"/>
68     </xsd:restriction>
69 </xsd:simpleType>
70 <xsd:simpleType name="ST_ColorSchemeIndex">
71     <xsd:restriction base="xsd:token">
72         <xsd:enumeration value="dk1"/>
73         <xsd:enumeration value="lt1"/>
74         <xsd:enumeration value="dk2"/>
75         <xsd:enumeration value="lt2"/>
76         <xsd:enumeration value="accent1"/>
77         <xsd:enumeration value="accent2"/>
78         <xsd:enumeration value="accent3"/>
79         <xsd:enumeration value="accent4"/>

```



```

80      <xsd:enumeration value="accent5"/>
81      <xsd:enumeration value="accent6"/>
82      <xsd:enumeration value="hlink"/>
83      <xsd:enumeration value="folHlink"/>
84    </xsd:restriction>
85  </xsd:simpleType>
86  <xsd:complexType name="CT_ColorScheme">
87    <xsd:sequence>
88      <xsd:element name="dk1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
89      <xsd:element name="lt1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
90      <xsd:element name="dk2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
91      <xsd:element name="lt2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
92      <xsd:element name="accent1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
93      <xsd:element name="accent2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
94      <xsd:element name="accent3" type="CT_Color" minOccurs="1" maxOccurs="1"/>
95      <xsd:element name="accent4" type="CT_Color" minOccurs="1" maxOccurs="1"/>
96      <xsd:element name="accent5" type="CT_Color" minOccurs="1" maxOccurs="1"/>
97      <xsd:element name="accent6" type="CT_Color" minOccurs="1" maxOccurs="1"/>
98      <xsd:element name="hlink" type="CT_Color" minOccurs="1" maxOccurs="1"/>
99      <xsd:element name="folHlink" type="CT_Color" minOccurs="1" maxOccurs="1"/>
100     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
101   </xsd:sequence>
102   <xsd:attribute name="name" type="xsd:string" use="required"/>
103 </xsd:complexType>
104 <xsd:complexType name="CT_CustomColor">
105   <xsd:sequence>
106     <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
107   </xsd:sequence>
108   <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
109 </xsd:complexType>
110 <xsd:complexType name="CT_SupplementalFont">
111   <xsd:attribute name="script" type="xsd:string" use="required"/>
112   <xsd:attribute name="typeface" type="ST_TextTypeface" use="required"/>
113 </xsd:complexType>
114 <xsd:complexType name="CT_CustomColorList">
115   <xsd:sequence>
116     <xsd:element name="custClr" type="CT_CustomColor" minOccurs="0" maxOccurs="unbounded"/>
117   </xsd:sequence>
118 </xsd:complexType>
119 <xsd:complexType name="CT_FontCollection">
120   <xsd:sequence>
121     <xsd:element name="latin" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
122     <xsd:element name="ea" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
123     <xsd:element name="cs" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
124     <xsd:element name="font" type="CT_SupplementalFont" minOccurs="0" maxOccurs="unbounded"/>
125     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
126   </xsd:sequence>
127 </xsd:complexType>
128 <xsd:complexType name="CT_EffectStyleItem">
129   <xsd:sequence>
130     <xsd:group ref="EG_EffectProperties" minOccurs="1" maxOccurs="1"/>
131     <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
132     <xsd:element name="sp3d" type="CT_Shape3D" minOccurs="0" maxOccurs="1"/>

```

```

133     </xsd:sequence>
134 </xsd:complexType>
135 <xsd:complexType name="CT_FontScheme">
136     <xsd:sequence>
137         <xsd:element name="majorFont" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
138         <xsd:element name="minorFont" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
139         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
140     </xsd:sequence>
141     <xsd:attribute name="name" type="xsd:string" use="required"/>
142 </xsd:complexType>
143 <xsd:complexType name="CT_FillStyleList">
144     <xsd:sequence>
145         <xsd:group ref="EG_FillProperties" minOccurs="3" maxOccurs="unbounded"/>
146     </xsd:sequence>
147 </xsd:complexType>
148 <xsd:complexType name="CT_LineStyleList">
149     <xsd:sequence>
150         <xsd:element name="ln" type="CT_LineProperties" minOccurs="3" maxOccurs="unbounded"/>
151     </xsd:sequence>
152 </xsd:complexType>
153 <xsd:complexType name="CT_EffectStyleList">
154     <xsd:sequence>
155         <xsd:element name="effectStyle" type="CT_EffectStyleItem" minOccurs="3"
156             maxOccurs="unbounded"/>
157     </xsd:sequence>
158 </xsd:complexType>
159 <xsd:complexType name="CT_BackgroundFillStyleList">
160     <xsd:sequence>
161         <xsd:group ref="EG_FillProperties" minOccurs="3" maxOccurs="unbounded"/>
162     </xsd:sequence>
163 </xsd:complexType>
164 <xsd:complexType name="CT_StyleMatrix">
165     <xsd:sequence>
166         <xsd:element name="fillStyleLst" type="CT_FillStyleList" minOccurs="1" maxOccurs="1"/>
167         <xsd:element name="lnStyleLst" type="CT_LineStyleList" minOccurs="1" maxOccurs="1"/>
168         <xsd:element name="effectStyleLst" type="CT_EffectStyleList" minOccurs="1" maxOccurs="1"/>
169         <xsd:element name="bgFillStyleLst" type="CT_BackgroundFillStyleList" minOccurs="1"
170             maxOccurs="1"/>
171     </xsd:sequence>
172     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
173 </xsd:complexType>
174 <xsd:complexType name="CT_BaseStyles">
175     <xsd:sequence>
176         <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="1" maxOccurs="1"/>
177         <xsd:element name="fontScheme" type="CT_FontScheme" minOccurs="1" maxOccurs="1"/>
178         <xsd:element name="fmtScheme" type="CT_StyleMatrix" minOccurs="1" maxOccurs="1"/>
179         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
180     </xsd:sequence>
181 </xsd:complexType>
182 <xsd:complexType name="CT_OfficeArtExtension">
183     <xsd:sequence>
184         <xsd:any processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
185     </xsd:sequence>

```

```

186     <xsd:attribute name="uri" type="xsd:token" use="required"/>
187 </xsd:complexType>
188 <xsd:simpleType name="ST_Coordinate">
189     <xsd:union memberTypes="ST_CoordinateUnqualified s:ST_UniversalMeasure"/>
190 </xsd:simpleType>
191 <xsd:simpleType name="ST_CoordinateUnqualified">
192     <xsd:restriction base="xsd:long">
193         <xsd:minInclusive value="-27273042329600"/>
194         <xsd:maxInclusive value="27273042316900"/>
195     </xsd:restriction>
196 </xsd:simpleType>
197 <xsd:simpleType name="ST_Coordinate32">
198     <xsd:union memberTypes="ST_Coordinate32Unqualified s:ST_UniversalMeasure"/>
199 </xsd:simpleType>
200 <xsd:simpleType name="ST_Coordinate32Unqualified">
201     <xsd:restriction base="xsd:int"/>
202 </xsd:simpleType>
203 <xsd:simpleType name="ST_PositiveCoordinate">
204     <xsd:restriction base="xsd:long">
205         <xsd:minInclusive value="0"/>
206         <xsd:maxInclusive value="27273042316900"/>
207     </xsd:restriction>
208 </xsd:simpleType>
209 <xsd:simpleType name="ST_PositiveCoordinate32">
210     <xsd:restriction base="ST_Coordinate32Unqualified">
211         <xsd:minInclusive value="0"/>
212     </xsd:restriction>
213 </xsd:simpleType>
214 <xsd:simpleType name="ST_Angle">
215     <xsd:restriction base="xsd:int"/>
216 </xsd:simpleType>
217 <xsd:complexType name="CT_Angle">
218     <xsd:attribute name="val" type="ST_Angle" use="required"/>
219 </xsd:complexType>
220 <xsd:simpleType name="ST_FixedAngle">
221     <xsd:restriction base="ST_Angle">
222         <xsd:minExclusive value="-5400000"/>
223         <xsd:maxExclusive value="5400000"/>
224     </xsd:restriction>
225 </xsd:simpleType>
226 <xsd:simpleType name="ST_PositiveFixedAngle">
227     <xsd:restriction base="ST_Angle">
228         <xsd:minInclusive value="0"/>
229         <xsd:maxExclusive value="21600000"/>
230     </xsd:restriction>
231 </xsd:simpleType>
232 <xsd:complexType name="CT_PositiveFixedAngle">
233     <xsd:attribute name="val" type="ST_PositiveFixedAngle" use="required"/>
234 </xsd:complexType>
235 <xsd:simpleType name="ST_Percentage">
236     <xsd:union memberTypes="ST_PercentageDecimal s:ST_Percentage"/>
237 </xsd:simpleType>
238 <xsd:simpleType name="ST_PercentageDecimal">

```

```

239     <xsd:restriction base="xsd:int"/>
240 </xsd:simpleType>
241 <xsd:complexType name="CT_Percentage">
242     <xsd:attribute name="val" type="ST_Percentage" use="required"/>
243 </xsd:complexType>
244 <xsd:simpleType name="ST_PositivePercentage">
245     <xsd:union memberTypes="ST_PositivePercentageDecimal s:ST_PositivePercentage"/>
246 </xsd:simpleType>
247 <xsd:simpleType name="ST_PositivePercentageDecimal">
248     <xsd:restriction base="ST_PercentageDecimal">
249         <xsd:minInclusive value="0"/>
250     </xsd:restriction>
251 </xsd:simpleType>
252 <xsd:complexType name="CT_PositivePercentage">
253     <xsd:attribute name="val" type="ST_PositivePercentage" use="required"/>
254 </xsd:complexType>
255 <xsd:simpleType name="ST_FixedPercentage">
256     <xsd:union memberTypes="ST_FixedPercentageDecimal s:ST_FixedPercentage"/>
257 </xsd:simpleType>
258 <xsd:simpleType name="ST_FixedPercentageDecimal">
259     <xsd:restriction base="ST_PercentageDecimal">
260         <xsd:minInclusive value="-100000"/>
261         <xsd:maxInclusive value="100000"/>
262     </xsd:restriction>
263 </xsd:simpleType>
264 <xsd:complexType name="CT_FixedPercentage">
265     <xsd:attribute name="val" type="ST_FixedPercentage" use="required"/>
266 </xsd:complexType>
267 <xsd:simpleType name="ST_PositiveFixedPercentage">
268     <xsd:union memberTypes="ST_PositiveFixedPercentageDecimal s:ST_PositiveFixedPercentage"/>
269 </xsd:simpleType>
270 <xsd:simpleType name="ST_PositiveFixedPercentageDecimal">
271     <xsd:restriction base="ST_PercentageDecimal">
272         <xsd:minInclusive value="0"/>
273         <xsd:maxInclusive value="100000"/>
274     </xsd:restriction>
275 </xsd:simpleType>
276 <xsd:complexType name="CT_PositiveFixedPercentage">
277     <xsd:attribute name="val" type="ST_PositiveFixedPercentage" use="required"/>
278 </xsd:complexType>
279 <xsd:complexType name="CT_Ratio">
280     <xsd:attribute name="n" type="xsd:long" use="required"/>
281     <xsd:attribute name="d" type="xsd:long" use="required"/>
282 </xsd:complexType>
283 <xsd:complexType name="CT_Point2D">
284     <xsd:attribute name="x" type="ST_Coordinate" use="required"/>
285     <xsd:attribute name="y" type="ST_Coordinate" use="required"/>
286 </xsd:complexType>
287 <xsd:complexType name="CT_PositiveSize2D">
288     <xsd:attribute name="cx" type="ST_PositiveCoordinate" use="required"/>
289     <xsd:attribute name="cy" type="ST_PositiveCoordinate" use="required"/>
290 </xsd:complexType>
291 <xsd:complexType name="CT_ComplementTransform"/>

```

```

292 <xsd:complexType name="CT_InverseTransform"/>
293 <xsd:complexType name="CT_GrayscaleTransform"/>
294 <xsd:complexType name="CT_GammaTransform"/>
295 <xsd:complexType name="CT_InverseGammaTransform"/>
296 <xsd:group name="EG_ColorTransform">
297   <xsd:choice>
298     <xsd:element name="tint" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
299     <xsd:element name="shade" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
300     <xsd:element name="comp" type="CT_ComplementTransform" minOccurs="1" maxOccurs="1"/>
301     <xsd:element name="inv" type="CT_InverseTransform" minOccurs="1" maxOccurs="1"/>
302     <xsd:element name="gray" type="CT_GrayscaleTransform" minOccurs="1" maxOccurs="1"/>
303     <xsd:element name="alpha" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
304     <xsd:element name="alphaOff" type="CT_FixedPercentage" minOccurs="1" maxOccurs="1"/>
305     <xsd:element name="alphaMod" type="CT_PositivePercentage" minOccurs="1" maxOccurs="1"/>
306     <xsd:element name="hue" type="CT_PositiveFixedAngle" minOccurs="1" maxOccurs="1"/>
307     <xsd:element name="hueOff" type="CT_Angle" minOccurs="1" maxOccurs="1"/>
308     <xsd:element name="hueMod" type="CT_PositivePercentage" minOccurs="1" maxOccurs="1"/>
309     <xsd:element name="sat" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
310     <xsd:element name="satOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
311     <xsd:element name="satMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
312     <xsd:element name="lum" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
313     <xsd:element name="lumOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
314     <xsd:element name="lumMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
315     <xsd:element name="red" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
316     <xsd:element name="redOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
317     <xsd:element name="redMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
318     <xsd:element name="green" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
319     <xsd:element name="greenOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
320     <xsd:element name="greenMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
321     <xsd:element name="blue" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
322     <xsd:element name="blueOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
323     <xsd:element name="blueMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
324     <xsd:element name="gamma" type="CT_GammaTransform" minOccurs="1" maxOccurs="1"/>
325     <xsd:element name="invGamma" type="CT_InverseGammaTransform" minOccurs="1" maxOccurs="1"/>
326   </xsd:choice>
327 </xsd:group>
328 <xsd:complexType name="CT_ScRgbColor">
329   <xsd:sequence>
330     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
331   </xsd:sequence>
332   <xsd:attribute name="r" type="ST_Percentage" use="required"/>
333   <xsd:attribute name="g" type="ST_Percentage" use="required"/>
334   <xsd:attribute name="b" type="ST_Percentage" use="required"/>
335 </xsd:complexType>
336 <xsd:complexType name="CT_SRgbColor">
337   <xsd:sequence>
338     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
339   </xsd:sequence>
340   <xsd:attribute name="val" type="s:ST_HexColorRGB" use="required"/>
341 </xsd:complexType>
342 <xsd:complexType name="CT_HslColor">
343   <xsd:sequence>
344     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

```

```

345     </xsd:sequence>
346     <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="required"/>
347     <xsd:attribute name="sat" type="ST_Percentage" use="required"/>
348     <xsd:attribute name="lum" type="ST_Percentage" use="required"/>
349 </xsd:complexType>
350 <xsd:simpleType name="ST_SystemColorVal">
351     <xsd:restriction base="xsd:token">
352         <xsd:enumeration value="scrollBar"/>
353         <xsd:enumeration value="background"/>
354         <xsd:enumeration value="activeCaption"/>
355         <xsd:enumeration value="inactiveCaption"/>
356         <xsd:enumeration value="menu"/>
357         <xsd:enumeration value="window"/>
358         <xsd:enumeration value="windowFrame"/>
359         <xsd:enumeration value="menuText"/>
360         <xsd:enumeration value="windowText"/>
361         <xsd:enumeration value="captionText"/>
362         <xsd:enumeration value="activeBorder"/>
363         <xsd:enumeration value="inactiveBorder"/>
364         <xsd:enumeration value="appWorkspace"/>
365         <xsd:enumeration value="highlight"/>
366         <xsd:enumeration value="highlightText"/>
367         <xsd:enumeration value="btnFace"/>
368         <xsd:enumeration value="btnShadow"/>
369         <xsd:enumeration value="grayText"/>
370         <xsd:enumeration value="btnText"/>
371         <xsd:enumeration value="inactiveCaptionText"/>
372         <xsd:enumeration value="btnHighlight"/>
373         <xsd:enumeration value="3dDkShadow"/>
374         <xsd:enumeration value="3dLight"/>
375         <xsd:enumeration value="infoText"/>
376         <xsd:enumeration value="infoBk"/>
377         <xsd:enumeration value="hotLight"/>
378         <xsd:enumeration value="gradientActiveCaption"/>
379         <xsd:enumeration value="gradientInactiveCaption"/>
380         <xsd:enumeration value="menuHighlight"/>
381         <xsd:enumeration value="menuBar"/>
382     </xsd:restriction>
383 </xsd:simpleType>
384 <xsd:complexType name="CT_SystemColor">
385     <xsd:sequence>
386         <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
387     </xsd:sequence>
388     <xsd:attribute name="val" type="ST_SystemColorVal" use="required"/>
389     <xsd:attribute name="lastClr" type="s:ST_HexColorRGB" use="optional"/>
390 </xsd:complexType>
391 <xsd:simpleType name="ST_SchemeColorVal">
392     <xsd:restriction base="xsd:token">
393         <xsd:enumeration value="bg1"/>
394         <xsd:enumeration value="tx1"/>
395         <xsd:enumeration value="bg2"/>
396         <xsd:enumeration value="tx2"/>
397         <xsd:enumeration value="accent1"/>

```

```

398         <xsd:enumeration value="accent2"/>
399         <xsd:enumeration value="accent3"/>
400         <xsd:enumeration value="accent4"/>
401         <xsd:enumeration value="accent5"/>
402         <xsd:enumeration value="accent6"/>
403         <xsd:enumeration value="hlink"/>
404         <xsd:enumeration value="folHlink"/>
405         <xsd:enumeration value="phClr"/>
406         <xsd:enumeration value="dk1"/>
407         <xsd:enumeration value="lt1"/>
408         <xsd:enumeration value="dk2"/>
409         <xsd:enumeration value="lt2"/>
410     </xsd:restriction>
411 </xsd:simpleType>
412 <xsd:complexType name="CT_SchemeColor">
413     <xsd:sequence>
414         <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
415     </xsd:sequence>
416     <xsd:attribute name="val" type="ST_SchemeColorVal" use="required"/>
417 </xsd:complexType>
418 <xsd:simpleType name="ST_PresetColorVal">
419     <xsd:restriction base="xsd:token">
420         <xsd:enumeration value="aliceBlue"/>
421         <xsd:enumeration value="antiqueWhite"/>
422         <xsd:enumeration value="aqua"/>
423         <xsd:enumeration value="aquamarine"/>
424         <xsd:enumeration value="azure"/>
425         <xsd:enumeration value="beige"/>
426         <xsd:enumeration value="bisque"/>
427         <xsd:enumeration value="black"/>
428         <xsd:enumeration value="blanchedAlmond"/>
429         <xsd:enumeration value="blue"/>
430         <xsd:enumeration value="blueViolet"/>
431         <xsd:enumeration value="brown"/>
432         <xsd:enumeration value="burlyWood"/>
433         <xsd:enumeration value="cadetBlue"/>
434         <xsd:enumeration value="chartreuse"/>
435         <xsd:enumeration value="chocolate"/>
436         <xsd:enumeration value="coral"/>
437         <xsd:enumeration value="cornflowerBlue"/>
438         <xsd:enumeration value="cornsilk"/>
439         <xsd:enumeration value="crimson"/>
440         <xsd:enumeration value="cyan"/>
441         <xsd:enumeration value="darkBlue"/>
442         <xsd:enumeration value="darkCyan"/>
443         <xsd:enumeration value="darkGoldenrod"/>
444         <xsd:enumeration value="darkGray"/>
445         <xsd:enumeration value="darkGrey"/>
446         <xsd:enumeration value="darkGreen"/>
447         <xsd:enumeration value="darkKhaki"/>
448         <xsd:enumeration value="darkMagenta"/>
449         <xsd:enumeration value="darkOliveGreen"/>
450         <xsd:enumeration value="darkOrange"/>

```

```

451     <xsd:enumeration value="darkOrchid"/>
452     <xsd:enumeration value="darkRed"/>
453     <xsd:enumeration value="darkSalmon"/>
454     <xsd:enumeration value="darkSeaGreen"/>
455     <xsd:enumeration value="darkSlateBlue"/>
456     <xsd:enumeration value="darkSlateGray"/>
457     <xsd:enumeration value="darkSlateGrey"/>
458     <xsd:enumeration value="darkTurquoise"/>
459     <xsd:enumeration value="darkViolet"/>
460     <xsd:enumeration value="dkBlue"/>
461     <xsd:enumeration value="dkCyan"/>
462     <xsd:enumeration value="dkGoldenrod"/>
463     <xsd:enumeration value="dkGray"/>
464     <xsd:enumeration value="dkGrey"/>
465     <xsd:enumeration value="dkGreen"/>
466     <xsd:enumeration value="dkKhaki"/>
467     <xsd:enumeration value="dkMagenta"/>
468     <xsd:enumeration value="dkOliveGreen"/>
469     <xsd:enumeration value="dkOrange"/>
470     <xsd:enumeration value="dkOrchid"/>
471     <xsd:enumeration value="dkRed"/>
472     <xsd:enumeration value="dkSalmon"/>
473     <xsd:enumeration value="dkSeaGreen"/>
474     <xsd:enumeration value="dkSlateBlue"/>
475     <xsd:enumeration value="dkSlateGray"/>
476     <xsd:enumeration value="dkSlateGrey"/>
477     <xsd:enumeration value="dkTurquoise"/>
478     <xsd:enumeration value="dkViolet"/>
479     <xsd:enumeration value="deepPink"/>
480     <xsd:enumeration value="deepSkyBlue"/>
481     <xsd:enumeration value="dimGray"/>
482     <xsd:enumeration value="dimGrey"/>
483     <xsd:enumeration value="dodgerBlue"/>
484     <xsd:enumeration value="firebrick"/>
485     <xsd:enumeration value="floralWhite"/>
486     <xsd:enumeration value="forestGreen"/>
487     <xsd:enumeration value="fuchsia"/>
488     <xsd:enumeration value="gainsboro"/>
489     <xsd:enumeration value="ghostWhite"/>
490     <xsd:enumeration value="gold"/>
491     <xsd:enumeration value="goldenrod"/>
492     <xsd:enumeration value="gray"/>
493     <xsd:enumeration value="grey"/>
494     <xsd:enumeration value="green"/>
495     <xsd:enumeration value="greenYellow"/>
496     <xsd:enumeration value="honeydew"/>
497     <xsd:enumeration value="hotPink"/>
498     <xsd:enumeration value="indianRed"/>
499     <xsd:enumeration value="indigo"/>
500     <xsd:enumeration value="ivory"/>
501     <xsd:enumeration value="khaki"/>
502     <xsd:enumeration value="lavender"/>
503     <xsd:enumeration value="lavenderBlush"/>

```



```

504      <xsd:enumeration value="lawnGreen"/>
505      <xsd:enumeration value="lemonChiffon"/>
506      <xsd:enumeration value="lightBlue"/>
507      <xsd:enumeration value="lightCoral"/>
508      <xsd:enumeration value="lightCyan"/>
509      <xsd:enumeration value="lightGoldenrodYellow"/>
510      <xsd:enumeration value="lightGray"/>
511      <xsd:enumeration value="lightGrey"/>
512      <xsd:enumeration value="lightGreen"/>
513      <xsd:enumeration value="lightPink"/>
514      <xsd:enumeration value="lightSalmon"/>
515      <xsd:enumeration value="lightSeaGreen"/>
516      <xsd:enumeration value="lightSkyBlue"/>
517      <xsd:enumeration value="lightSlateGray"/>
518      <xsd:enumeration value="lightSlateGrey"/>
519      <xsd:enumeration value="lightSteelBlue"/>
520      <xsd:enumeration value="lightYellow"/>
521      <xsd:enumeration value="ltBlue"/>
522      <xsd:enumeration value="ltCoral"/>
523      <xsd:enumeration value="ltCyan"/>
524      <xsd:enumeration value="ltGoldenrodYellow"/>
525      <xsd:enumeration value="ltGray"/>
526      <xsd:enumeration value="ltGrey"/>
527      <xsd:enumeration value="ltGreen"/>
528      <xsd:enumeration value="ltPink"/>
529      <xsd:enumeration value="ltSalmon"/>
530      <xsd:enumeration value="ltSeaGreen"/>
531      <xsd:enumeration value="ltSkyBlue"/>
532      <xsd:enumeration value="ltSlateGray"/>
533      <xsd:enumeration value="ltSlateGrey"/>
534      <xsd:enumeration value="ltSteelBlue"/>
535      <xsd:enumeration value="ltYellow"/>
536      <xsd:enumeration value="lime"/>
537      <xsd:enumeration value="limeGreen"/>
538      <xsd:enumeration value="linen"/>
539      <xsd:enumeration value="magenta"/>
540      <xsd:enumeration value="maroon"/>
541      <xsd:enumeration value="medAquamarine"/>
542      <xsd:enumeration value="medBlue"/>
543      <xsd:enumeration value="medOrchid"/>
544      <xsd:enumeration value="medPurple"/>
545      <xsd:enumeration value="medSeaGreen"/>
546      <xsd:enumeration value="medSlateBlue"/>
547      <xsd:enumeration value="medSpringGreen"/>
548      <xsd:enumeration value="medTurquoise"/>
549      <xsd:enumeration value="medVioletRed"/>
550      <xsd:enumeration value="mediumAquamarine"/>
551      <xsd:enumeration value="mediumBlue"/>
552      <xsd:enumeration value="mediumOrchid"/>
553      <xsd:enumeration value="mediumPurple"/>
554      <xsd:enumeration value="mediumSeaGreen"/>
555      <xsd:enumeration value="mediumSlateBlue"/>
556      <xsd:enumeration value="mediumSpringGreen"/>

```

```
557      <xsd:enumeration value="mediumTurquoise"/>
558      <xsd:enumeration value="mediumVioletRed"/>
559      <xsd:enumeration value="midnightBlue"/>
560      <xsd:enumeration value="mintCream"/>
561      <xsd:enumeration value="mistyRose"/>
562      <xsd:enumeration value="moccasin"/>
563      <xsd:enumeration value="navajoWhite"/>
564      <xsd:enumeration value="navy"/>
565      <xsd:enumeration value="oldLace"/>
566      <xsd:enumeration value="olive"/>
567      <xsd:enumeration value="oliveDrab"/>
568      <xsd:enumeration value="orange"/>
569      <xsd:enumeration value="orangeRed"/>
570      <xsd:enumeration value="orchid"/>
571      <xsd:enumeration value="paleGoldenrod"/>
572      <xsd:enumeration value="paleGreen"/>
573      <xsd:enumeration value="paleTurquoise"/>
574      <xsd:enumeration value="paleVioletRed"/>
575      <xsd:enumeration value="papayaWhip"/>
576      <xsd:enumeration value="peachPuff"/>
577      <xsd:enumeration value="peru"/>
578      <xsd:enumeration value="pink"/>
579      <xsd:enumeration value="plum"/>
580      <xsd:enumeration value="powderBlue"/>
581      <xsd:enumeration value="purple"/>
582      <xsd:enumeration value="red"/>
583      <xsd:enumeration value="rosyBrown"/>
584      <xsd:enumeration value="royalBlue"/>
585      <xsd:enumeration value="saddleBrown"/>
586      <xsd:enumeration value="salmon"/>
587      <xsd:enumeration value="sandyBrown"/>
588      <xsd:enumeration value="seaGreen"/>
589      <xsd:enumeration value="seaShell"/>
590      <xsd:enumeration value="sienna"/>
591      <xsd:enumeration value="silver"/>
592      <xsd:enumeration value="skyBlue"/>
593      <xsd:enumeration value="slateBlue"/>
594      <xsd:enumeration value="slateGray"/>
595      <xsd:enumeration value="slateGrey"/>
596      <xsd:enumeration value="snow"/>
597      <xsd:enumeration value="springGreen"/>
598      <xsd:enumeration value="steelBlue"/>
599      <xsd:enumeration value="tan"/>
600      <xsd:enumeration value="teal"/>
601      <xsd:enumeration value="thistle"/>
602      <xsd:enumeration value="tomato"/>
603      <xsd:enumeration value="turquoise"/>
604      <xsd:enumeration value="violet"/>
605      <xsd:enumeration value="wheat"/>
606      <xsd:enumeration value="white"/>
607      <xsd:enumeration value="whiteSmoke"/>
608      <xsd:enumeration value="yellow"/>
609      <xsd:enumeration value="yellowGreen"/>
```

```

610     </xsd:restriction>
611 </xsd:simpleType>
612 <xsd:complexType name="CT_PresetColor">
613     <xsd:sequence>
614         <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
615     </xsd:sequence>
616     <xsd:attribute name="val" type="ST_PresetColorVal" use="required"/>
617 </xsd:complexType>
618 <xsd:group name="EG_OfficeArtExtensionList">
619     <xsd:sequence>
620         <xsd:element name="ext" type="CT_OfficeArtExtension" minOccurs="0" maxOccurs="unbounded"/>
621     </xsd:sequence>
622 </xsd:group>
623 <xsd:complexType name="CT_OfficeArtExtensionList">
624     <xsd:sequence>
625         <xsd:group ref="EG_OfficeArtExtensionList" minOccurs="1" maxOccurs="1"/>
626     </xsd:sequence>
627 </xsd:complexType>
628 <xsd:complexType name="CT_Scale2D">
629     <xsd:sequence>
630         <xsd:element name="sx" type="CT_Ratio" minOccurs="1" maxOccurs="1"/>
631         <xsd:element name="sy" type="CT_Ratio" minOccurs="1" maxOccurs="1"/>
632     </xsd:sequence>
633 </xsd:complexType>
634 <xsd:complexType name="CT_Transform2D">
635     <xsd:sequence>
636         <xsd:element name="off" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
637         <xsd:element name="ext" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
638     </xsd:sequence>
639     <xsd:attribute name="rot" type="ST_Angle" use="optional" default="0"/>
640     <xsd:attribute name="flipH" type="xsd:boolean" use="optional" default="false"/>
641     <xsd:attribute name="flipV" type="xsd:boolean" use="optional" default="false"/>
642 </xsd:complexType>
643 <xsd:complexType name="CT_GroupTransform2D">
644     <xsd:sequence>
645         <xsd:element name="off" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
646         <xsd:element name="ext" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
647         <xsd:element name="chOff" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
648         <xsd:element name="chExt" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
649     </xsd:sequence>
650     <xsd:attribute name="rot" type="ST_Angle" use="optional" default="0"/>
651     <xsd:attribute name="flipH" type="xsd:boolean" use="optional" default="false"/>
652     <xsd:attribute name="flipV" type="xsd:boolean" use="optional" default="false"/>
653 </xsd:complexType>
654 <xsd:complexType name="CT_Point3D">
655     <xsd:attribute name="x" type="ST_Coordinate" use="required"/>
656     <xsd:attribute name="y" type="ST_Coordinate" use="required"/>
657     <xsd:attribute name="z" type="ST_Coordinate" use="required"/>
658 </xsd:complexType>
659 <xsd:complexType name="CT_Vector3D">
660     <xsd:attribute name="dx" type="ST_Coordinate" use="required"/>
661     <xsd:attribute name="dy" type="ST_Coordinate" use="required"/>
662     <xsd:attribute name="dz" type="ST_Coordinate" use="required"/>

```

```

663 </xsd:complexType>
664 <xsd:complexType name="CT_SphereCoords">
665   <xsd:attribute name="lat" type="ST_PositiveFixedAngle" use="required"/>
666   <xsd:attribute name="lon" type="ST_PositiveFixedAngle" use="required"/>
667   <xsd:attribute name="rev" type="ST_PositiveFixedAngle" use="required"/>
668 </xsd:complexType>
669 <xsd:complexType name="CT_RelativeRect">
670   <xsd:attribute name="l" type="ST_Percentage" use="optional" default="0%"/>
671   <xsd:attribute name="t" type="ST_Percentage" use="optional" default="0%"/>
672   <xsd:attribute name="r" type="ST_Percentage" use="optional" default="0%"/>
673   <xsd:attribute name="b" type="ST_Percentage" use="optional" default="0%"/>
674 </xsd:complexType>
675 <xsd:simpleType name="ST_RectAlignment">
676   <xsd:restriction base="xsd:token">
677     <xsd:enumeration value="tl"/>
678     <xsd:enumeration value="t"/>
679     <xsd:enumeration value="tr"/>
680     <xsd:enumeration value="l"/>
681     <xsd:enumeration value="ctr"/>
682     <xsd:enumeration value="r"/>
683     <xsd:enumeration value="bl"/>
684     <xsd:enumeration value="b"/>
685     <xsd:enumeration value="br"/>
686   </xsd:restriction>
687 </xsd:simpleType>
688 <xsd:group name="EG_ColorChoice">
689   <xsd:choice>
690     <xsd:element name="scrgbClr" type="CT_ScRgbColor" minOccurs="1" maxOccurs="1"/>
691     <xsd:element name="srgbClr" type="CT_SRgbColor" minOccurs="1" maxOccurs="1"/>
692     <xsd:element name="hslClr" type="CT_HslColor" minOccurs="1" maxOccurs="1"/>
693     <xsd:element name="sysClr" type="CT_SystemColor" minOccurs="1" maxOccurs="1"/>
694     <xsd:element name="schemeClr" type="CT_SchemeColor" minOccurs="1" maxOccurs="1"/>
695     <xsd:element name="prstClr" type="CT_PresetColor" minOccurs="1" maxOccurs="1"/>
696   </xsd:choice>
697 </xsd:group>
698 <xsd:complexType name="CT_Color">
699   <xsd:sequence>
700     <xsd:group ref="EG_ColorChoice"/>
701   </xsd:sequence>
702 </xsd:complexType>
703 <xsd:complexType name="CT_ColorMRU">
704   <xsd:sequence>
705     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="unbounded"/>
706   </xsd:sequence>
707 </xsd:complexType>
708 <xsd:simpleType name="ST_BlackWhiteMode">
709   <xsd:restriction base="xsd:token">
710     <xsd:enumeration value="clr"/>
711     <xsd:enumeration value="auto"/>
712     <xsd:enumeration value="gray"/>
713     <xsd:enumeration value="ltGray"/>
714     <xsd:enumeration value="invGray"/>
715     <xsd:enumeration value="grayWhite"/>

```

```

716         <xsd:enumeration value="blackGray"/>
717         <xsd:enumeration value="blackWhite"/>
718         <xsd:enumeration value="black"/>
719         <xsd:enumeration value="white"/>
720         <xsd:enumeration value="hidden"/>
721     </xsd:restriction>
722 </xsd:simpleType>
723 <xsd:attributeGroup name="AG_Blob">
724     <xsd:attribute ref="r:embed" use="optional" default=""/>
725     <xsd:attribute ref="r:link" use="optional" default=""/>
726 </xsd:attributeGroup>
727 <xsd:complexType name="CT_EmbeddedWAVAudioFile">
728     <xsd:attribute ref="r:embed" use="required"/>
729     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
730 </xsd:complexType>
731 <xsd:complexType name="CT_Hyperlink">
732     <xsd:sequence>
733         <xsd:element name="snd" type="CT_EmbeddedWAVAudioFile" minOccurs="0" maxOccurs="1"/>
734         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
735     </xsd:sequence>
736     <xsd:attribute ref="r:id" use="optional"/>
737     <xsd:attribute name="invalidUrl" type="xsd:string" use="optional" default=""/>
738     <xsd:attribute name="action" type="xsd:string" use="optional" default=""/>
739     <xsd:attribute name="tgtFrame" type="xsd:string" use="optional" default=""/>
740     <xsd:attribute name="tooltip" type="xsd:string" use="optional" default=""/>
741     <xsd:attribute name="history" type="xsd:boolean" use="optional" default="true"/>
742     <xsd:attribute name="highlightClick" type="xsd:boolean" use="optional" default="false"/>
743     <xsd:attribute name="endSnd" type="xsd:boolean" use="optional" default="false"/>
744 </xsd:complexType>
745 <xsd:simpleType name="ST_DrawingElementId">
746     <xsd:restriction base="xsd:unsignedInt"/>
747 </xsd:simpleType>
748 <xsd:attributeGroup name="AG_Locking">
749     <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
750     <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
751     <xsd:attribute name="noRot" type="xsd:boolean" use="optional" default="false"/>
752     <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
753     <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
754     <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
755     <xsd:attribute name="noEditPoints" type="xsd:boolean" use="optional" default="false"/>
756     <xsd:attribute name="noAdjustHandles" type="xsd:boolean" use="optional" default="false"/>
757     <xsd:attribute name="noChangeArrowheads" type="xsd:boolean" use="optional" default="false"/>
758     <xsd:attribute name="noChangeShapeType" type="xsd:boolean" use="optional" default="false"/>
759 </xsd:attributeGroup>
760 <xsd:complexType name="CT_ConnectorLocking">
761     <xsd:sequence>
762         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
763     </xsd:sequence>
764     <xsd:attributeGroup ref="AG_Locking"/>
765 </xsd:complexType>
766 <xsd:complexType name="CT_ShapeLocking">
767     <xsd:sequence>
768         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

769     </xsd:sequence>
770     <xsd:attributeGroup ref="AG_Locking"/>
771     <xsd:attribute name="noTextEdit" type="xsd:boolean" use="optional" default="false"/>
772 </xsd:complexType>
773 <xsd:complexType name="CT_PictureLocking">
774     <xsd:sequence>
775         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
776     </xsd:sequence>
777     <xsd:attributeGroup ref="AG_Locking"/>
778     <xsd:attribute name="noCrop" type="xsd:boolean" use="optional" default="false"/>
779 </xsd:complexType>
780 <xsd:complexType name="CT_GroupLocking">
781     <xsd:sequence>
782         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
783     </xsd:sequence>
784     <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
785     <xsd:attribute name="noUngrp" type="xsd:boolean" use="optional" default="false"/>
786     <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
787     <xsd:attribute name="noRot" type="xsd:boolean" use="optional" default="false"/>
788     <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
789     <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
790     <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
791 </xsd:complexType>
792 <xsd:complexType name="CT_GraphicalObjectFrameLocking">
793     <xsd:sequence>
794         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
795     </xsd:sequence>
796     <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
797     <xsd:attribute name="noDrilldown" type="xsd:boolean" use="optional" default="false"/>
798     <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
799     <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
800     <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
801     <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
802 </xsd:complexType>
803 <xsd:complexType name="CT_ContentPartLocking">
804     <xsd:sequence>
805         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
806     </xsd:sequence>
807     <xsd:attributeGroup ref="AG_Locking"/>
808 </xsd:complexType>
809 <xsd:complexType name="CT_NonVisualDrawingProps">
810     <xsd:sequence>
811         <xsd:element name="hlinkClick" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
812         <xsd:element name="hlinkHover" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
813         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
814     </xsd:sequence>
815     <xsd:attribute name="id" type="ST_DrawingElementId" use="required"/>
816     <xsd:attribute name="name" type="xsd:string" use="required"/>
817     <xsd:attribute name="descr" type="xsd:string" use="optional" default=""/>
818     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
819     <xsd:attribute name="title" type="xsd:string" use="optional" default=""/>
820 </xsd:complexType>
821 <xsd:complexType name="CT_NonVisualDrawingShapeProps">

```

```

822     <xsd:sequence>
823         <xsd:element name="spLocks" type="CT_ShapeLocking" minOccurs="0" maxOccurs="1"/>
824         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
825     </xsd:sequence>
826     <xsd:attribute name="txBox" type="xsd:boolean" use="optional" default="false"/>
827 </xsd:complexType>
828 <xsd:complexType name="CT_NonVisualConnectorProperties">
829     <xsd:sequence>
830         <xsd:element name="cxnSpLocks" type="CT_ConnectorLocking" minOccurs="0" maxOccurs="1"/>
831         <xsd:element name="stCxn" type="CT_Connection" minOccurs="0" maxOccurs="1"/>
832         <xsd:element name="endCxn" type="CT_Connection" minOccurs="0" maxOccurs="1"/>
833         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
834     </xsd:sequence>
835 </xsd:complexType>
836 <xsd:complexType name="CT_NonVisualPictureProperties">
837     <xsd:sequence>
838         <xsd:element name="picLocks" type="CT_PictureLocking" minOccurs="0" maxOccurs="1"/>
839         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
840     </xsd:sequence>
841     <xsd:attribute name="preferRelativeResize" type="xsd:boolean" use="optional" default="true"/>
842 </xsd:complexType>
843 <xsd:complexType name="CT_NonVisualGroupDrawingShapeProps">
844     <xsd:sequence>
845         <xsd:element name="grpSpLocks" type="CT_GroupLocking" minOccurs="0" maxOccurs="1"/>
846         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
847     </xsd:sequence>
848 </xsd:complexType>
849 <xsd:complexType name="CT_NonVisualGraphicFrameProperties">
850     <xsd:sequence>
851         <xsd:element name="graphicFrameLocks" type="CT_GraphicalObjectFrameLocking" minOccurs="0"
852             maxOccurs="1"/>
853         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
854     </xsd:sequence>
855 </xsd:complexType>
856 <xsd:complexType name="CT_NonVisualContentPartProperties">
857     <xsd:sequence>
858         <xsd:element name="cpLocks" type="CT_ContentPartLocking" minOccurs="0" maxOccurs="1"/>
859         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
860     </xsd:sequence>
861     <xsd:attribute name="isComment" type="xsd:boolean" use="optional" default="true"/>
862 </xsd:complexType>
863 <xsd:complexType name="CT_GraphicalObjectData">
864     <xsd:sequence>
865         <xsd:any minOccurs="0" maxOccurs="unbounded" processContents="strict"/>
866     </xsd:sequence>
867     <xsd:attribute name="uri" type="xsd:token" use="required"/>
868 </xsd:complexType>
869 <xsd:complexType name="CT_GraphicalObject">
870     <xsd:sequence>
871         <xsd:element name="graphicData" type="CT_GraphicalObjectData"/>
872     </xsd:sequence>
873 </xsd:complexType>
874 <xsd:element name="graphic" type="CT_GraphicalObject"/>

```

```

875 <xsd:simpleType name="ST_ChartBuildStep">
876   <xsd:restriction base="xsd:token">
877     <xsd:enumeration value="category"/>
878     <xsd:enumeration value="ptInCategory"/>
879     <xsd:enumeration value="series"/>
880     <xsd:enumeration value="ptInSeries"/>
881     <xsd:enumeration value="allPts"/>
882     <xsd:enumeration value="gridLegend"/>
883   </xsd:restriction>
884 </xsd:simpleType>
885 <xsd:simpleType name="ST_DgmBuildStep">
886   <xsd:restriction base="xsd:token">
887     <xsd:enumeration value="sp"/>
888     <xsd:enumeration value="bg"/>
889   </xsd:restriction>
890 </xsd:simpleType>
891 <xsd:complexType name="CT_AnimationDgmElement">
892   <xsd:attribute name="id" type="s:ST_Guid" use="optional" default="{00000000-0000-0000-0000-
893     000000000000}"/>
894   <xsd:attribute name="bldStep" type="ST_DgmBuildStep" use="optional" default="sp"/>
895 </xsd:complexType>
896 <xsd:complexType name="CT_AnimationChartElement">
897   <xsd:attribute name="seriesIdx" type="xsd:int" use="optional" default="-1"/>
898   <xsd:attribute name="categoryIdx" type="xsd:int" use="optional" default="-1"/>
899   <xsd:attribute name="bldStep" type="ST_ChartBuildStep" use="required"/>
900 </xsd:complexType>
901 <xsd:complexType name="CT_AnimationElementChoice">
902   <xsd:choice minOccurs="1" maxOccurs="1">
903     <xsd:element name="dgm" type="CT_AnimationDgmElement"/>
904     <xsd:element name="chart" type="CT_AnimationChartElement"/>
905   </xsd:choice>
906 </xsd:complexType>
907 <xsd:simpleType name="ST_AnimationBuildType">
908   <xsd:restriction base="xsd:token">
909     <xsd:enumeration value="allAtOnce"/>
910   </xsd:restriction>
911 </xsd:simpleType>
912 <xsd:simpleType name="ST_AnimationDgmOnlyBuildType">
913   <xsd:restriction base="xsd:token">
914     <xsd:enumeration value="one"/>
915     <xsd:enumeration value="lvlOne"/>
916     <xsd:enumeration value="lvlAtOnce"/>
917   </xsd:restriction>
918 </xsd:simpleType>
919 <xsd:simpleType name="ST_AnimationDgmBuildType">
920   <xsd:union memberTypes="ST_AnimationBuildType ST_AnimationDgmOnlyBuildType"/>
921 </xsd:simpleType>
922 <xsd:complexType name="CT_AnimationDgmBuildProperties">
923   <xsd:attribute name="bld" type="ST_AnimationDgmBuildType" use="optional" default="allAtOnce"/>
924   <xsd:attribute name="rev" type="xsd:boolean" use="optional" default="false"/>
925 </xsd:complexType>
926 <xsd:simpleType name="ST_AnimationChartOnlyBuildType">
927   <xsd:restriction base="xsd:token">

```



```

928         <xsd:enumeration value="series"/>
929         <xsd:enumeration value="category"/>
930         <xsd:enumeration value="seriesEl"/>
931         <xsd:enumeration value="categoryEl"/>
932     </xsd:restriction>
933 </xsd:simpleType>
934 <xsd:simpleType name="ST_AnimationChartBuildType">
935     <xsd:union memberTypes="ST_AnimationBuildType ST_AnimationChartOnlyBuildType"/>
936 </xsd:simpleType>
937 <xsd:complexType name="CT_AnimationChartBuildProperties">
938     <xsd:attribute name="bld" type="ST_AnimationChartBuildType" use="optional"
939         default="allAtOnce"/>
940     <xsd:attribute name="animBg" type="xsd:boolean" use="optional" default="true"/>
941 </xsd:complexType>
942 <xsd:complexType name="CT_AnimationGraphicalObjectBuildProperties">
943     <xsd:choice>
944         <xsd:element name="bldDgm" type="CT_AnimationDgmBuildProperties"/>
945         <xsd:element name="bldChart" type="CT_AnimationChartBuildProperties"/>
946     </xsd:choice>
947 </xsd:complexType>
948 <xsd:complexType name="CT_BackgroundFormatting">
949     <xsd:sequence>
950         <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
951         <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
952     </xsd:sequence>
953 </xsd:complexType>
954 <xsd:complexType name="CT_WholeE2oFormatting">
955     <xsd:sequence>
956         <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
957         <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
958     </xsd:sequence>
959 </xsd:complexType>
960 <xsd:complexType name="CT_GvmlUseShapeRectangle"/>
961 <xsd:complexType name="CT_GvmlTextShape">
962     <xsd:sequence>
963         <xsd:element name="txBody" type="CT_TextBody" minOccurs="1" maxOccurs="1"/>
964         <xsd:choice>
965             <xsd:element name="useSpRect" type="CT_GvmlUseShapeRectangle" minOccurs="1"
966                 maxOccurs="1"/>
967             <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="1" maxOccurs="1"/>
968         </xsd:choice>
969         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
970     </xsd:sequence>
971 </xsd:complexType>
972 <xsd:complexType name="CT_GvmlShapeNonVisual">
973     <xsd:sequence>
974         <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
975         <xsd:element name="cNvSpPr" type="CT_NonVisualDrawingShapeProps" minOccurs="1"
976             maxOccurs="1"/>
977     </xsd:sequence>
978 </xsd:complexType>
979 <xsd:complexType name="CT_GvmlShape">
980     <xsd:sequence>

```

```

981     <xsd:element name="nvSpPr" type="CT_GvmlShapeNonVisual" minOccurs="1" maxOccurs="1"/>
982     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
983     <xsd:element name="txSp" type="CT_GvmlTextShape" minOccurs="0" maxOccurs="1"/>
984     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
985     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
986   </xsd:sequence>
987 </xsd:complexType>
988 <xsd:complexType name="CT_GvmlConnectorNonVisual">
989   <xsd:sequence>
990     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
991     <xsd:element name="cNvCxnSpPr" type="CT_NonVisualConnectorProperties" minOccurs="1"
992       maxOccurs="1"/>
993   </xsd:sequence>
994 </xsd:complexType>
995 <xsd:complexType name="CT_GvmlConnector">
996   <xsd:sequence>
997     <xsd:element name="nvCxnSpPr" type="CT_GvmlConnectorNonVisual" minOccurs="1"
998       maxOccurs="1"/>
999     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1000     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1001     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1002   </xsd:sequence>
1003 </xsd:complexType>
1004 <xsd:complexType name="CT_GvmlPictureNonVisual">
1005   <xsd:sequence>
1006     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1007     <xsd:element name="cNvPicPr" type="CT_NonVisualPictureProperties" minOccurs="1"
1008       maxOccurs="1"/>
1009   </xsd:sequence>
1010 </xsd:complexType>
1011 <xsd:complexType name="CT_GvmlPicture">
1012   <xsd:sequence>
1013     <xsd:element name="nvPicPr" type="CT_GvmlPictureNonVisual" minOccurs="1" maxOccurs="1"/>
1014     <xsd:element name="blipFill" type="CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
1015     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1016     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1017     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1018   </xsd:sequence>
1019 </xsd:complexType>
1020 <xsd:complexType name="CT_GvmlGraphicFrameNonVisual">
1021   <xsd:sequence>
1022     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1023     <xsd:element name="cNvGraphicFramePr" type="CT_NonVisualGraphicFrameProperties"
1024       minOccurs="1" maxOccurs="1"/>
1025   </xsd:sequence>
1026 </xsd:complexType>
1027 <xsd:complexType name="CT_GvmlGraphicalObjectFrame">
1028   <xsd:sequence>
1029     <xsd:element name="nvGraphicFramePr" type="CT_GvmlGraphicFrameNonVisual" minOccurs="1"
1030       maxOccurs="1"/>
1031     <xsd:element ref="graphic" minOccurs="1" maxOccurs="1"/>
1032     <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="1" maxOccurs="1"/>
1033     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

1034     </xsd:sequence>
1035 </xsd:complexType>
1036 <xsd:complexType name="CT_GvmlGroupShapeNonVisual">
1037     <xsd:sequence>
1038         <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1039         <xsd:element name="cNvGrpSpPr" type="CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
1040             maxOccurs="1"/>
1041     </xsd:sequence>
1042 </xsd:complexType>
1043 <xsd:complexType name="CT_GvmlGroupShape">
1044     <xsd:sequence>
1045         <xsd:element name="nvGrpSpPr" type="CT_GvmlGroupShapeNonVisual" minOccurs="1"
1046             maxOccurs="1"/>
1047         <xsd:element name="grpSpPr" type="CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
1048         <xsd:choice minOccurs="0" maxOccurs="unbounded">
1049             <xsd:element name="txSp" type="CT_GvmlTextShape"/>
1050             <xsd:element name="sp" type="CT_GvmlShape"/>
1051             <xsd:element name="cxnSp" type="CT_GvmlConnector"/>
1052             <xsd:element name="pic" type="CT_GvmlPicture"/>
1053             <xsd:element name="graphicFrame" type="CT_GvmlGraphicalObjectFrame"/>
1054             <xsd:element name="grpSp" type="CT_GvmlGroupShape"/>
1055         </xsd:choice>
1056         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1057     </xsd:sequence>
1058 </xsd:complexType>
1059 <xsd:simpleType name="ST_PresetCameraType">
1060     <xsd:restriction base="xsd:token">
1061         <xsd:enumeration value="legacyObliqueTopLeft"/>
1062         <xsd:enumeration value="legacyObliqueTop"/>
1063         <xsd:enumeration value="legacyObliqueTopRight"/>
1064         <xsd:enumeration value="legacyObliqueLeft"/>
1065         <xsd:enumeration value="legacyObliqueFront"/>
1066         <xsd:enumeration value="legacyObliqueRight"/>
1067         <xsd:enumeration value="legacyObliqueBottomLeft"/>
1068         <xsd:enumeration value="legacyObliqueBottom"/>
1069         <xsd:enumeration value="legacyObliqueBottomRight"/>
1070         <xsd:enumeration value="legacyPerspectiveTopLeft"/>
1071         <xsd:enumeration value="legacyPerspectiveTop"/>
1072         <xsd:enumeration value="legacyPerspectiveTopRight"/>
1073         <xsd:enumeration value="legacyPerspectiveLeft"/>
1074         <xsd:enumeration value="legacyPerspectiveFront"/>
1075         <xsd:enumeration value="legacyPerspectiveRight"/>
1076         <xsd:enumeration value="legacyPerspectiveBottomLeft"/>
1077         <xsd:enumeration value="legacyPerspectiveBottom"/>
1078         <xsd:enumeration value="legacyPerspectiveBottomRight"/>
1079         <xsd:enumeration value="orthographicFront"/>
1080         <xsd:enumeration value="isometricTopUp"/>
1081         <xsd:enumeration value="isometricTopDown"/>
1082         <xsd:enumeration value="isometricBottomUp"/>
1083         <xsd:enumeration value="isometricBottomDown"/>
1084         <xsd:enumeration value="isometricLeftUp"/>
1085         <xsd:enumeration value="isometricLeftDown"/>
1086         <xsd:enumeration value="isometricRightUp"/>

```

```

1087     <xsd:enumeration value="isometricRightDown"/>
1088     <xsd:enumeration value="isometricOffAxis1Left"/>
1089     <xsd:enumeration value="isometricOffAxis1Right"/>
1090     <xsd:enumeration value="isometricOffAxis1Top"/>
1091     <xsd:enumeration value="isometricOffAxis2Left"/>
1092     <xsd:enumeration value="isometricOffAxis2Right"/>
1093     <xsd:enumeration value="isometricOffAxis2Top"/>
1094     <xsd:enumeration value="isometricOffAxis3Left"/>
1095     <xsd:enumeration value="isometricOffAxis3Right"/>
1096     <xsd:enumeration value="isometricOffAxis3Bottom"/>
1097     <xsd:enumeration value="isometricOffAxis4Left"/>
1098     <xsd:enumeration value="isometricOffAxis4Right"/>
1099     <xsd:enumeration value="isometricOffAxis4Bottom"/>
1100     <xsd:enumeration value="obliqueTopLeft"/>
1101     <xsd:enumeration value="obliqueTop"/>
1102     <xsd:enumeration value="obliqueTopRight"/>
1103     <xsd:enumeration value="obliqueLeft"/>
1104     <xsd:enumeration value="obliqueRight"/>
1105     <xsd:enumeration value="obliqueBottomLeft"/>
1106     <xsd:enumeration value="obliqueBottom"/>
1107     <xsd:enumeration value="obliqueBottomRight"/>
1108     <xsd:enumeration value="perspectiveFront"/>
1109     <xsd:enumeration value="perspectiveLeft"/>
1110     <xsd:enumeration value="perspectiveRight"/>
1111     <xsd:enumeration value="perspectiveAbove"/>
1112     <xsd:enumeration value="perspectiveBelow"/>
1113     <xsd:enumeration value="perspectiveAboveLeftFacing"/>
1114     <xsd:enumeration value="perspectiveAboveRightFacing"/>
1115     <xsd:enumeration value="perspectiveContrastingLeftFacing"/>
1116     <xsd:enumeration value="perspectiveContrastingRightFacing"/>
1117     <xsd:enumeration value="perspectiveHeroicLeftFacing"/>
1118     <xsd:enumeration value="perspectiveHeroicRightFacing"/>
1119     <xsd:enumeration value="perspectiveHeroicExtremeLeftFacing"/>
1120     <xsd:enumeration value="perspectiveHeroicExtremeRightFacing"/>
1121     <xsd:enumeration value="perspectiveRelaxed"/>
1122     <xsd:enumeration value="perspectiveRelaxedModerately"/>
1123   </xsd:restriction>
1124 </xsd:simpleType>
1125 <xsd:simpleType name="ST_FOVAngle">
1126   <xsd:restriction base="ST_Angle">
1127     <xsd:minInclusive value="0"/>
1128     <xsd:maxInclusive value="10800000"/>
1129   </xsd:restriction>
1130 </xsd:simpleType>
1131 <xsd:complexType name="CT_Camera">
1132   <xsd:sequence>
1133     <xsd:element name="rot" type="CT_SphereCoords" minOccurs="0" maxOccurs="1"/>
1134   </xsd:sequence>
1135   <xsd:attribute name="prst" type="ST_PresetCameraType" use="required"/>
1136   <xsd:attribute name="fov" type="ST_FOVAngle" use="optional"/>
1137   <xsd:attribute name="zoom" type="ST_PositivePercentage" use="optional" default="100%"/>
1138 </xsd:complexType>
1139 <xsd:simpleType name="ST_LightRigDirection">

```

```

1140     <xsd:restriction base="xsd:token">
1141         <xsd:enumeration value="tl"/>
1142         <xsd:enumeration value="t"/>
1143         <xsd:enumeration value="tr"/>
1144         <xsd:enumeration value="l"/>
1145         <xsd:enumeration value="r"/>
1146         <xsd:enumeration value="bl"/>
1147         <xsd:enumeration value="b"/>
1148         <xsd:enumeration value="br"/>
1149     </xsd:restriction>
1150 </xsd:simpleType>
1151 <xsd:simpleType name="ST_LightRigType">
1152     <xsd:restriction base="xsd:token">
1153         <xsd:enumeration value="legacyFlat1"/>
1154         <xsd:enumeration value="legacyFlat2"/>
1155         <xsd:enumeration value="legacyFlat3"/>
1156         <xsd:enumeration value="legacyFlat4"/>
1157         <xsd:enumeration value="legacyNormal1"/>
1158         <xsd:enumeration value="legacyNormal2"/>
1159         <xsd:enumeration value="legacyNormal3"/>
1160         <xsd:enumeration value="legacyNormal4"/>
1161         <xsd:enumeration value="legacyHarsh1"/>
1162         <xsd:enumeration value="legacyHarsh2"/>
1163         <xsd:enumeration value="legacyHarsh3"/>
1164         <xsd:enumeration value="legacyHarsh4"/>
1165         <xsd:enumeration value="threePt"/>
1166         <xsd:enumeration value="balanced"/>
1167         <xsd:enumeration value="soft"/>
1168         <xsd:enumeration value="harsh"/>
1169         <xsd:enumeration value="flood"/>
1170         <xsd:enumeration value="contrasting"/>
1171         <xsd:enumeration value="morning"/>
1172         <xsd:enumeration value="sunrise"/>
1173         <xsd:enumeration value="sunset"/>
1174         <xsd:enumeration value="chilly"/>
1175         <xsd:enumeration value="freezing"/>
1176         <xsd:enumeration value="flat"/>
1177         <xsd:enumeration value="twoPt"/>
1178         <xsd:enumeration value="glow"/>
1179         <xsd:enumeration value="brightRoom"/>
1180     </xsd:restriction>
1181 </xsd:simpleType>
1182 <xsd:complexType name="CT_LightRig">
1183     <xsd:sequence>
1184         <xsd:element name="rot" type="CT_SphereCoords" minOccurs="0" maxOccurs="1"/>
1185     </xsd:sequence>
1186     <xsd:attribute name="rig" type="ST_LightRigType" use="required"/>
1187     <xsd:attribute name="dir" type="ST_LightRigDirection" use="required"/>
1188 </xsd:complexType>
1189 <xsd:complexType name="CT_Scene3D">
1190     <xsd:sequence>
1191         <xsd:element name="camera" type="CT_Camera" minOccurs="1" maxOccurs="1"/>
1192         <xsd:element name="lightRig" type="CT_LightRig" minOccurs="1" maxOccurs="1"/>

```

```

1193     <xsd:element name="backdrop" type="CT_Backdrop" minOccurs="0" maxOccurs="1"/>
1194     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1195   </xsd:sequence>
1196 </xsd:complexType>
1197 <xsd:complexType name="CT_Backdrop">
1198   <xsd:sequence>
1199     <xsd:element name="anchor" type="CT_Point3D" minOccurs="1" maxOccurs="1"/>
1200     <xsd:element name="norm" type="CT_Vector3D" minOccurs="1" maxOccurs="1"/>
1201     <xsd:element name="up" type="CT_Vector3D" minOccurs="1" maxOccurs="1"/>
1202     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1203   </xsd:sequence>
1204 </xsd:complexType>
1205 <xsd:simpleType name="ST_BevelPresetType">
1206   <xsd:restriction base="xsd:token">
1207     <xsd:enumeration value="relaxedInset"/>
1208     <xsd:enumeration value="circle"/>
1209     <xsd:enumeration value="slope"/>
1210     <xsd:enumeration value="cross"/>
1211     <xsd:enumeration value="angle"/>
1212     <xsd:enumeration value="softRound"/>
1213     <xsd:enumeration value="convex"/>
1214     <xsd:enumeration value="coolSlant"/>
1215     <xsd:enumeration value="divot"/>
1216     <xsd:enumeration value="riblet"/>
1217     <xsd:enumeration value="hardEdge"/>
1218     <xsd:enumeration value="artDeco"/>
1219   </xsd:restriction>
1220 </xsd:simpleType>
1221 <xsd:complexType name="CT_Bevel">
1222   <xsd:attribute name="w" type="ST_PositiveCoordinate" use="optional" default="76200"/>
1223   <xsd:attribute name="h" type="ST_PositiveCoordinate" use="optional" default="76200"/>
1224   <xsd:attribute name="prst" type="ST_BevelPresetType" use="optional" default="circle"/>
1225 </xsd:complexType>
1226 <xsd:simpleType name="ST_PresetMaterialType">
1227   <xsd:restriction base="xsd:token">
1228     <xsd:enumeration value="legacyMatte"/>
1229     <xsd:enumeration value="legacyPlastic"/>
1230     <xsd:enumeration value="legacyMetal"/>
1231     <xsd:enumeration value="legacyWireframe"/>
1232     <xsd:enumeration value="matte"/>
1233     <xsd:enumeration value="plastic"/>
1234     <xsd:enumeration value="metal"/>
1235     <xsd:enumeration value="warmMatte"/>
1236     <xsd:enumeration value="translucentPowder"/>
1237     <xsd:enumeration value="powder"/>
1238     <xsd:enumeration value="dkEdge"/>
1239     <xsd:enumeration value="softEdge"/>
1240     <xsd:enumeration value="clear"/>
1241     <xsd:enumeration value="flat"/>
1242     <xsd:enumeration value="softmetal"/>
1243   </xsd:restriction>
1244 </xsd:simpleType>
1245 <xsd:complexType name="CT_Shape3D">

```

```

1246     <xsd:sequence>
1247         <xsd:element name="bevelT" type="CT Bevel" minOccurs="0" maxOccurs="1"/>
1248         <xsd:element name="bevelB" type="CT Bevel" minOccurs="0" maxOccurs="1"/>
1249         <xsd:element name="extrusionClr" type="CT Color" minOccurs="0" maxOccurs="1"/>
1250         <xsd:element name="contourClr" type="CT Color" minOccurs="0" maxOccurs="1"/>
1251         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1252     </xsd:sequence>
1253     <xsd:attribute name="z" type="ST Coordinate" use="optional" default="0"/>
1254     <xsd:attribute name="extrusionH" type="ST PositiveCoordinate" use="optional" default="0"/>
1255     <xsd:attribute name="contourW" type="ST PositiveCoordinate" use="optional" default="0"/>
1256     <xsd:attribute name="prstMaterial" type="ST PresetMaterialType" use="optional"
1257         default="warmMatte"/>
1258 </xsd:complexType>
1259 <xsd:complexType name="CT_FlatText">
1260     <xsd:attribute name="z" type="ST Coordinate" use="optional" default="0"/>
1261 </xsd:complexType>
1262 <xsd:group name="EG_Text3D">
1263     <xsd:choice>
1264         <xsd:element name="sp3d" type="CT Shape3D" minOccurs="1" maxOccurs="1"/>
1265         <xsd:element name="flatTx" type="CT FlatText" minOccurs="1" maxOccurs="1"/>
1266     </xsd:choice>
1267 </xsd:group>
1268 <xsd:complexType name="CT_AlphaBiLevelEffect">
1269     <xsd:attribute name="thresh" type="ST PositiveFixedPercentage" use="required"/>
1270 </xsd:complexType>
1271 <xsd:complexType name="CT_AlphaCeilingEffect"/>
1272 <xsd:complexType name="CT_AlphaFloorEffect"/>
1273 <xsd:complexType name="CT_AlphaInverseEffect">
1274     <xsd:sequence>
1275         <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
1276     </xsd:sequence>
1277 </xsd:complexType>
1278 <xsd:complexType name="CT_AlphaModulateFixedEffect">
1279     <xsd:attribute name="amt" type="ST PositivePercentage" use="optional" default="100%"/>
1280 </xsd:complexType>
1281 <xsd:complexType name="CT_AlphaOutsetEffect">
1282     <xsd:attribute name="rad" type="ST Coordinate" use="optional" default="0"/>
1283 </xsd:complexType>
1284 <xsd:complexType name="CT_AlphaReplaceEffect">
1285     <xsd:attribute name="a" type="ST PositiveFixedPercentage" use="required"/>
1286 </xsd:complexType>
1287 <xsd:complexType name="CT_BiLevelEffect">
1288     <xsd:attribute name="thresh" type="ST PositiveFixedPercentage" use="required"/>
1289 </xsd:complexType>
1290 <xsd:complexType name="CT_BlurEffect">
1291     <xsd:attribute name="rad" type="ST PositiveCoordinate" use="optional" default="0"/>
1292     <xsd:attribute name="grow" type="xsd:boolean" use="optional" default="true"/>
1293 </xsd:complexType>
1294 <xsd:complexType name="CT_ColorChangeEffect">
1295     <xsd:sequence>
1296         <xsd:element name="clrFrom" type="CT Color" minOccurs="1" maxOccurs="1"/>
1297         <xsd:element name="clrTo" type="CT Color" minOccurs="1" maxOccurs="1"/>
1298     </xsd:sequence>

```

```

1299     <xsd:attribute name="useA" type="xsd:boolean" use="optional" default="true"/>
1300 </xsd:complexType>
1301 <xsd:complexType name="CT_ColorReplaceEffect">
1302     <xsd:sequence>
1303         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1304     </xsd:sequence>
1305 </xsd:complexType>
1306 <xsd:complexType name="CT_DuotoneEffect">
1307     <xsd:sequence>
1308         <xsd:group ref="EG_ColorChoice" minOccurs="2" maxOccurs="2"/>
1309     </xsd:sequence>
1310 </xsd:complexType>
1311 <xsd:complexType name="CT_GlowEffect">
1312     <xsd:sequence>
1313         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1314     </xsd:sequence>
1315     <xsd:attribute name="rad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1316 </xsd:complexType>
1317 <xsd:complexType name="CT_GrayscaleEffect"/>
1318 <xsd:complexType name="CT_HSLEffect">
1319     <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1320     <xsd:attribute name="sat" type="ST_FixedPercentage" use="optional" default="0%"/>
1321     <xsd:attribute name="lum" type="ST_FixedPercentage" use="optional" default="0%"/>
1322 </xsd:complexType>
1323 <xsd:complexType name="CT_InnerShadowEffect">
1324     <xsd:sequence>
1325         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1326     </xsd:sequence>
1327     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1328     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1329     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1330 </xsd:complexType>
1331 <xsd:complexType name="CT_LuminanceEffect">
1332     <xsd:attribute name="bright" type="ST_FixedPercentage" use="optional" default="0%"/>
1333     <xsd:attribute name="contrast" type="ST_FixedPercentage" use="optional" default="0%"/>
1334 </xsd:complexType>
1335 <xsd:complexType name="CT_OuterShadowEffect">
1336     <xsd:sequence>
1337         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1338     </xsd:sequence>
1339     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1340     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1341     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1342     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1343     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1344     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1345     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1346     <xsd:attribute name="align" type="ST_RectAlignment" use="optional" default="b"/>
1347     <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional" default="true"/>
1348 </xsd:complexType>
1349 <xsd:simpleType name="ST_PresetShadowVal">
1350     <xsd:restriction base="xsd:token">
1351         <xsd:enumeration value="shdw1"/>

```



```

1352     <xsd:enumeration value="shdw2"/>
1353     <xsd:enumeration value="shdw3"/>
1354     <xsd:enumeration value="shdw4"/>
1355     <xsd:enumeration value="shdw5"/>
1356     <xsd:enumeration value="shdw6"/>
1357     <xsd:enumeration value="shdw7"/>
1358     <xsd:enumeration value="shdw8"/>
1359     <xsd:enumeration value="shdw9"/>
1360     <xsd:enumeration value="shdw10"/>
1361     <xsd:enumeration value="shdw11"/>
1362     <xsd:enumeration value="shdw12"/>
1363     <xsd:enumeration value="shdw13"/>
1364     <xsd:enumeration value="shdw14"/>
1365     <xsd:enumeration value="shdw15"/>
1366     <xsd:enumeration value="shdw16"/>
1367     <xsd:enumeration value="shdw17"/>
1368     <xsd:enumeration value="shdw18"/>
1369     <xsd:enumeration value="shdw19"/>
1370     <xsd:enumeration value="shdw20"/>
1371 </xsd:restriction>
1372 </xsd:simpleType>
1373 <xsd:complexType name="CT_PresetShadowEffect">
1374     <xsd:sequence>
1375         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1376     </xsd:sequence>
1377     <xsd:attribute name="prst" type="ST_PresetShadowVal" use="required"/>
1378     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1379     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1380 </xsd:complexType>
1381 <xsd:complexType name="CT_ReflectionEffect">
1382     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1383     <xsd:attribute name="stA" type="ST_PositiveFixedPercentage" use="optional" default="100%"/>
1384     <xsd:attribute name="stPos" type="ST_PositiveFixedPercentage" use="optional" default="0%"/>
1385     <xsd:attribute name="endA" type="ST_PositiveFixedPercentage" use="optional" default="0%"/>
1386     <xsd:attribute name="endPos" type="ST_PositiveFixedPercentage" use="optional" default="100%"/>
1387     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1388     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1389     <xsd:attribute name="fadeDir" type="ST_PositiveFixedAngle" use="optional" default="5400000"/>
1390     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1391     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1392     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1393     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1394     <xsd:attribute name="align" type="ST_RectAlignment" use="optional" default="b"/>
1395     <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional" default="true"/>
1396 </xsd:complexType>
1397 <xsd:complexType name="CT_RelativeOffsetEffect">
1398     <xsd:attribute name="tx" type="ST_Percentage" use="optional" default="0%"/>
1399     <xsd:attribute name="ty" type="ST_Percentage" use="optional" default="0%"/>
1400 </xsd:complexType>
1401 <xsd:complexType name="CT_SoftEdgesEffect">
1402     <xsd:attribute name="rad" type="ST_PositiveCoordinate" use="required"/>
1403 </xsd:complexType>
1404 <xsd:complexType name="CT_TintEffect">

```

```

1405     <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1406     <xsd:attribute name="amt" type="ST_FixedPercentage" use="optional" default="0%"/>
1407 </xsd:complexType>
1408 <xsd:complexType name="CT_TransformEffect">
1409     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1410     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1411     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1412     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1413     <xsd:attribute name="tx" type="ST_Coordinate" use="optional" default="0"/>
1414     <xsd:attribute name="ty" type="ST_Coordinate" use="optional" default="0"/>
1415 </xsd:complexType>
1416 <xsd:complexType name="CT_NoFillProperties"/>
1417 <xsd:complexType name="CT_SolidColorFillProperties">
1418     <xsd:sequence>
1419         <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
1420     </xsd:sequence>
1421 </xsd:complexType>
1422 <xsd:complexType name="CT_LinearShadeProperties">
1423     <xsd:attribute name="ang" type="ST_PositiveFixedAngle" use="optional"/>
1424     <xsd:attribute name="scaled" type="xsd:boolean" use="optional"/>
1425 </xsd:complexType>
1426 <xsd:simpleType name="ST_PathShadeType">
1427     <xsd:restriction base="xsd:token">
1428         <xsd:enumeration value="shape"/>
1429         <xsd:enumeration value="circle"/>
1430         <xsd:enumeration value="rect"/>
1431     </xsd:restriction>
1432 </xsd:simpleType>
1433 <xsd:complexType name="CT_PathShadeProperties">
1434     <xsd:sequence>
1435         <xsd:element name="fillToRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1436     </xsd:sequence>
1437     <xsd:attribute name="path" type="ST_PathShadeType" use="optional"/>
1438 </xsd:complexType>
1439 <xsd:group name="EG_ShadeProperties">
1440     <xsd:choice>
1441         <xsd:element name="lin" type="CT_LinearShadeProperties" minOccurs="1" maxOccurs="1"/>
1442         <xsd:element name="path" type="CT_PathShadeProperties" minOccurs="1" maxOccurs="1"/>
1443     </xsd:choice>
1444 </xsd:group>
1445 <xsd:simpleType name="ST_TileFlipMode">
1446     <xsd:restriction base="xsd:token">
1447         <xsd:enumeration value="none"/>
1448         <xsd:enumeration value="x"/>
1449         <xsd:enumeration value="y"/>
1450         <xsd:enumeration value="xy"/>
1451     </xsd:restriction>
1452 </xsd:simpleType>
1453 <xsd:complexType name="CT_GradientStop">
1454     <xsd:sequence>
1455         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1456     </xsd:sequence>
1457     <xsd:attribute name="pos" type="ST_PositiveFixedPercentage" use="required"/>

```

```

1458 </xsd:complexType>
1459 <xsd:complexType name="CT_GradientStopList">
1460   <xsd:sequence>
1461     <xsd:element name="gs" type="CT_GradientStop" minOccurs="2" maxOccurs="unbounded"/>
1462   </xsd:sequence>
1463 </xsd:complexType>
1464 <xsd:complexType name="CT_GradientFillProperties">
1465   <xsd:sequence>
1466     <xsd:element name="gsLst" type="CT_GradientStopList" minOccurs="0" maxOccurs="1"/>
1467     <xsd:group ref="EG_ShadeProperties" minOccurs="0" maxOccurs="1"/>
1468     <xsd:element name="tileRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1469   </xsd:sequence>
1470   <xsd:attribute name="flip" type="ST_TileFlipMode" use="optional"/>
1471   <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional"/>
1472 </xsd:complexType>
1473 <xsd:complexType name="CT_TileInfoProperties">
1474   <xsd:attribute name="tx" type="ST_Coordinate" use="optional"/>
1475   <xsd:attribute name="ty" type="ST_Coordinate" use="optional"/>
1476   <xsd:attribute name="sx" type="ST_Percentage" use="optional"/>
1477   <xsd:attribute name="sy" type="ST_Percentage" use="optional"/>
1478   <xsd:attribute name="flip" type="ST_TileFlipMode" use="optional"/>
1479   <xsd:attribute name="align" type="ST_RectAlignment" use="optional"/>
1480 </xsd:complexType>
1481 <xsd:complexType name="CT_StretchInfoProperties">
1482   <xsd:sequence>
1483     <xsd:element name="fillRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1484   </xsd:sequence>
1485 </xsd:complexType>
1486 <xsd:group name="EG_FillModeProperties">
1487   <xsd:choice>
1488     <xsd:element name="tile" type="CT_TileInfoProperties" minOccurs="1" maxOccurs="1"/>
1489     <xsd:element name="stretch" type="CT_StretchInfoProperties" minOccurs="1" maxOccurs="1"/>
1490   </xsd:choice>
1491 </xsd:group>
1492 <xsd:simpleType name="ST_BlipCompression">
1493   <xsd:restriction base="xsd:token">
1494     <xsd:enumeration value="email"/>
1495     <xsd:enumeration value="screen"/>
1496     <xsd:enumeration value="print"/>
1497     <xsd:enumeration value="hqprint"/>
1498     <xsd:enumeration value="none"/>
1499   </xsd:restriction>
1500 </xsd:simpleType>
1501 <xsd:complexType name="CT_Blip">
1502   <xsd:sequence>
1503     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1504       <xsd:element name="alphaBiLevel" type="CT_AlphaBiLevelEffect" minOccurs="1"
1505         maxOccurs="1"/>
1506       <xsd:element name="alphaCeiling" type="CT_AlphaCeilingEffect" minOccurs="1"
1507         maxOccurs="1"/>
1508       <xsd:element name="alphaFloor" type="CT_AlphaFloorEffect" minOccurs="1" maxOccurs="1"/>
1509       <xsd:element name="alphaInv" type="CT_AlphaInverseEffect" minOccurs="1" maxOccurs="1"/>

```

```

1510     <xsd:element name="alphaMod" type="CT_AlphaModulateEffect" minOccurs="1"
1511       maxOccurs="1"/>
1512     <xsd:element name="alphaModFix" type="CT_AlphaModulateFixedEffect" minOccurs="1"
1513       maxOccurs="1"/>
1514     <xsd:element name="alphaRepl" type="CT_AlphaReplaceEffect" minOccurs="1"
1515       maxOccurs="1"/>
1516     <xsd:element name="biLevel" type="CT_BiLevelEffect" minOccurs="1" maxOccurs="1"/>
1517     <xsd:element name="blur" type="CT_BlurEffect" minOccurs="1" maxOccurs="1"/>
1518     <xsd:element name="clrChange" type="CT_ColorChangeEffect" minOccurs="1" maxOccurs="1"/>
1519     <xsd:element name="clrRepl" type="CT_ColorReplaceEffect" minOccurs="1" maxOccurs="1"/>
1520     <xsd:element name="duotone" type="CT_DuotoneEffect" minOccurs="1" maxOccurs="1"/>
1521     <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="1"
1522       maxOccurs="1"/>
1523     <xsd:element name="grayscale" type="CT_GrayscaleEffect" minOccurs="1" maxOccurs="1"/>
1524     <xsd:element name="hsl" type="CT_HSLEffect" minOccurs="1" maxOccurs="1"/>
1525     <xsd:element name="lum" type="CT_LuminanceEffect" minOccurs="1" maxOccurs="1"/>
1526     <xsd:element name="tint" type="CT_TintEffect" minOccurs="1" maxOccurs="1"/>
1527   </xsd:choice>
1528   <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1529 </xsd:sequence>
1530 <xsd:attributeGroup ref="AG_Blob"/>
1531 <xsd:attribute name="cstate" type="ST_BlipCompression" use="optional" default="none"/>
1532 </xsd:complexType>
1533 <xsd:complexType name="CT_BlipFillProperties">
1534   <xsd:sequence>
1535     <xsd:element name="blip" type="CT_Blip" minOccurs="0" maxOccurs="1"/>
1536     <xsd:element name="srcRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1537     <xsd:group ref="EG_FillModeProperties" minOccurs="0" maxOccurs="1"/>
1538   </xsd:sequence>
1539   <xsd:attribute name="dpi" type="xsd:unsignedInt" use="optional"/>
1540   <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional"/>
1541 </xsd:complexType>
1542 <xsd:simpleType name="ST_PresetPatternVal">
1543   <xsd:restriction base="xsd:token">
1544     <xsd:enumeration value="pct5"/>
1545     <xsd:enumeration value="pct10"/>
1546     <xsd:enumeration value="pct20"/>
1547     <xsd:enumeration value="pct25"/>
1548     <xsd:enumeration value="pct30"/>
1549     <xsd:enumeration value="pct40"/>
1550     <xsd:enumeration value="pct50"/>
1551     <xsd:enumeration value="pct60"/>
1552     <xsd:enumeration value="pct70"/>
1553     <xsd:enumeration value="pct75"/>
1554     <xsd:enumeration value="pct80"/>
1555     <xsd:enumeration value="pct90"/>
1556     <xsd:enumeration value="horz"/>
1557     <xsd:enumeration value="vert"/>
1558     <xsd:enumeration value="ltHorz"/>
1559     <xsd:enumeration value="ltVert"/>
1560     <xsd:enumeration value="dkHorz"/>
1561     <xsd:enumeration value="dkVert"/>
1562     <xsd:enumeration value="narHorz"/>

```

```

1563     <xsd:enumeration value="narVert"/>
1564     <xsd:enumeration value="dashHorz"/>
1565     <xsd:enumeration value="dashVert"/>
1566     <xsd:enumeration value="cross"/>
1567     <xsd:enumeration value="dnDiag"/>
1568     <xsd:enumeration value="upDiag"/>
1569     <xsd:enumeration value="ltDnDiag"/>
1570     <xsd:enumeration value="ltUpDiag"/>
1571     <xsd:enumeration value="dkDnDiag"/>
1572     <xsd:enumeration value="dkUpDiag"/>
1573     <xsd:enumeration value="wdDnDiag"/>
1574     <xsd:enumeration value="wdUpDiag"/>
1575     <xsd:enumeration value="dashDnDiag"/>
1576     <xsd:enumeration value="dashUpDiag"/>
1577     <xsd:enumeration value="diagCross"/>
1578     <xsd:enumeration value="smCheck"/>
1579     <xsd:enumeration value="lgCheck"/>
1580     <xsd:enumeration value="smGrid"/>
1581     <xsd:enumeration value="lgGrid"/>
1582     <xsd:enumeration value="dotGrid"/>
1583     <xsd:enumeration value="smConfetti"/>
1584     <xsd:enumeration value="lgConfetti"/>
1585     <xsd:enumeration value="horzBrick"/>
1586     <xsd:enumeration value="diagBrick"/>
1587     <xsd:enumeration value="solidDmnd"/>
1588     <xsd:enumeration value="openDmnd"/>
1589     <xsd:enumeration value="dotDmnd"/>
1590     <xsd:enumeration value="plaid"/>
1591     <xsd:enumeration value="sphere"/>
1592     <xsd:enumeration value="weave"/>
1593     <xsd:enumeration value="divot"/>
1594     <xsd:enumeration value="shingle"/>
1595     <xsd:enumeration value="wave"/>
1596     <xsd:enumeration value="trellis"/>
1597     <xsd:enumeration value="zigZag"/>
1598   </xsd:restriction>
1599 </xsd:simpleType>
1600 <xsd:complexType name="CT_PatternFillProperties">
1601   <xsd:sequence>
1602     <xsd:element name="fgClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1603     <xsd:element name="bgClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1604   </xsd:sequence>
1605   <xsd:attribute name="prst" type="ST_PresetPatternVal" use="optional"/>
1606 </xsd:complexType>
1607 <xsd:complexType name="CT_GroupFillProperties"/>
1608 <xsd:group name="EG_FillProperties">
1609   <xsd:choice>
1610     <xsd:element name="noFill" type="CT_NoFillProperties" minOccurs="1" maxOccurs="1"/>
1611     <xsd:element name="solidFill" type="CT_SolidColorFillProperties" minOccurs="1"
1612       maxOccurs="1"/>
1613     <xsd:element name="gradFill" type="CT_GradientFillProperties" minOccurs="1"
1614       maxOccurs="1"/>
1615     <xsd:element name="blipFill" type="CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>

```

```

1616         <xsd:element name="pattFill" type="CT_PatternFillProperties" minOccurs="1" maxOccurs="1"/>
1617         <xsd:element name="grpFill" type="CT_GroupFillProperties" minOccurs="1" maxOccurs="1"/>
1618     </xsd:choice>
1619 </xsd:group>
1620 <xsd:complexType name="CT_FillProperties">
1621     <xsd:sequence>
1622         <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1623     </xsd:sequence>
1624 </xsd:complexType>
1625 <xsd:complexType name="CT_FillEffect">
1626     <xsd:sequence>
1627         <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1628     </xsd:sequence>
1629 </xsd:complexType>
1630 <xsd:simpleType name="ST_BlendMode">
1631     <xsd:restriction base="xsd:token">
1632         <xsd:enumeration value="over"/>
1633         <xsd:enumeration value="mult"/>
1634         <xsd:enumeration value="screen"/>
1635         <xsd:enumeration value="darken"/>
1636         <xsd:enumeration value="lighten"/>
1637     </xsd:restriction>
1638 </xsd:simpleType>
1639 <xsd:complexType name="CT_FillOverlayEffect">
1640     <xsd:sequence>
1641         <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1642     </xsd:sequence>
1643     <xsd:attribute name="blend" type="ST_BlendMode" use="required"/>
1644 </xsd:complexType>
1645 <xsd:complexType name="CT_EffectReference">
1646     <xsd:attribute name="ref" type="xsd:token" use="required"/>
1647 </xsd:complexType>
1648 <xsd:group name="EG_Effect">
1649     <xsd:choice>
1650         <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1651         <xsd:element name="effect" type="CT_EffectReference" minOccurs="1" maxOccurs="1"/>
1652         <xsd:element name="alphaBiLevel" type="CT_AlphaBiLevelEffect" minOccurs="1"
1653             maxOccurs="1"/>
1654         <xsd:element name="alphaCeiling" type="CT_AlphaCeilingEffect" minOccurs="1"
1655             maxOccurs="1"/>
1656         <xsd:element name="alphaFloor" type="CT_AlphaFloorEffect" minOccurs="1" maxOccurs="1"/>
1657         <xsd:element name="alphaInv" type="CT_AlphaInverseEffect" minOccurs="1" maxOccurs="1"/>
1658         <xsd:element name="alphaMod" type="CT_AlphaModulateEffect" minOccurs="1" maxOccurs="1"/>
1659         <xsd:element name="alphaModFix" type="CT_AlphaModulateFixedEffect" minOccurs="1"
1660             maxOccurs="1"/>
1661         <xsd:element name="alphaOutset" type="CT_AlphaOutsetEffect" minOccurs="1" maxOccurs="1"/>
1662         <xsd:element name="alphaRepl" type="CT_AlphaReplaceEffect" minOccurs="1" maxOccurs="1"/>
1663         <xsd:element name="biLevel" type="CT_BiLevelEffect" minOccurs="1" maxOccurs="1"/>
1664         <xsd:element name="blend" type="CT_BlendEffect" minOccurs="1" maxOccurs="1"/>
1665         <xsd:element name="blur" type="CT_BlurEffect" minOccurs="1" maxOccurs="1"/>
1666         <xsd:element name="clrChange" type="CT_ColorChangeEffect" minOccurs="1" maxOccurs="1"/>
1667         <xsd:element name="clrRepl" type="CT_ColorReplaceEffect" minOccurs="1" maxOccurs="1"/>
1668         <xsd:element name="duotone" type="CT_DuotoneEffect" minOccurs="1" maxOccurs="1"/>

```

```

1669     <xsd:element name="fill" type="CT_FillEffect" minOccurs="1" maxOccurs="1"/>
1670     <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="1" maxOccurs="1"/>
1671     <xsd:element name="glow" type="CT_GlowEffect" minOccurs="1" maxOccurs="1"/>
1672     <xsd:element name="grayscale" type="CT_GrayscaleEffect" minOccurs="1" maxOccurs="1"/>
1673     <xsd:element name="hsl" type="CT_HSLEffect" minOccurs="1" maxOccurs="1"/>
1674     <xsd:element name="innerShdw" type="CT_InnerShadowEffect" minOccurs="1" maxOccurs="1"/>
1675     <xsd:element name="lum" type="CT_LuminanceEffect" minOccurs="1" maxOccurs="1"/>
1676     <xsd:element name="outerShdw" type="CT_OuterShadowEffect" minOccurs="1" maxOccurs="1"/>
1677     <xsd:element name="prstShdw" type="CT_PresetShadowEffect" minOccurs="1" maxOccurs="1"/>
1678     <xsd:element name="reflection" type="CT_ReflectionEffect" minOccurs="1" maxOccurs="1"/>
1679     <xsd:element name="relOff" type="CT_RelativeOffsetEffect" minOccurs="1" maxOccurs="1"/>
1680     <xsd:element name="softEdge" type="CT_SoftEdgesEffect" minOccurs="1" maxOccurs="1"/>
1681     <xsd:element name="tint" type="CT_TintEffect" minOccurs="1" maxOccurs="1"/>
1682     <xsd:element name="xfrm" type="CT_TransformEffect" minOccurs="1" maxOccurs="1"/>
1683   </xsd:choice>
1684 </xsd:group>
1685 <xsd:simpleType name="ST_EffectContainerType">
1686   <xsd:restriction base="xsd:token">
1687     <xsd:enumeration value="sib"/>
1688     <xsd:enumeration value="tree"/>
1689   </xsd:restriction>
1690 </xsd:simpleType>
1691 <xsd:complexType name="CT_EffectContainer">
1692   <xsd:group ref="EG_Effect" minOccurs="0" maxOccurs="unbounded"/>
1693   <xsd:attribute name="type" type="ST_EffectContainerType" use="optional" default="sib"/>
1694   <xsd:attribute name="name" type="xsd:token" use="optional"/>
1695 </xsd:complexType>
1696 <xsd:complexType name="CT_AlphaModulateEffect">
1697   <xsd:sequence>
1698     <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1699   </xsd:sequence>
1700 </xsd:complexType>
1701 <xsd:complexType name="CT_BlendEffect">
1702   <xsd:sequence>
1703     <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1704   </xsd:sequence>
1705   <xsd:attribute name="blend" type="ST_BlendMode" use="required"/>
1706 </xsd:complexType>
1707 <xsd:complexType name="CT_EffectList">
1708   <xsd:sequence>
1709     <xsd:element name="blur" type="CT_BlurEffect" minOccurs="0" maxOccurs="1"/>
1710     <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="0" maxOccurs="1"/>
1711     <xsd:element name="glow" type="CT_GlowEffect" minOccurs="0" maxOccurs="1"/>
1712     <xsd:element name="innerShdw" type="CT_InnerShadowEffect" minOccurs="0" maxOccurs="1"/>
1713     <xsd:element name="outerShdw" type="CT_OuterShadowEffect" minOccurs="0" maxOccurs="1"/>
1714     <xsd:element name="prstShdw" type="CT_PresetShadowEffect" minOccurs="0" maxOccurs="1"/>
1715     <xsd:element name="reflection" type="CT_ReflectionEffect" minOccurs="0" maxOccurs="1"/>
1716     <xsd:element name="softEdge" type="CT_SoftEdgesEffect" minOccurs="0" maxOccurs="1"/>
1717   </xsd:sequence>
1718 </xsd:complexType>
1719 <xsd:group name="EG_EffectProperties">
1720   <xsd:choice>
1721     <xsd:element name="effectList" type="CT_EffectList" minOccurs="1" maxOccurs="1"/>

```

```

1722     <xsd:element name="effectDag" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1723   </xsd:choice>
1724 </xsd:group>
1725 <xsd:complexType name="CT_EffectProperties">
1726   <xsd:sequence>
1727     <xsd:group ref="EG_EffectProperties" minOccurs="1" maxOccurs="1"/>
1728   </xsd:sequence>
1729 </xsd:complexType>
1730 <xsd:element name="blip" type="CT_Blip"/>
1731 <xsd:simpleType name="ST_ShapeType">
1732   <xsd:restriction base="xsd:token">
1733     <xsd:enumeration value="line"/>
1734     <xsd:enumeration value="lineInv"/>
1735     <xsd:enumeration value="triangle"/>
1736     <xsd:enumeration value="rtTriangle"/>
1737     <xsd:enumeration value="rect"/>
1738     <xsd:enumeration value="diamond"/>
1739     <xsd:enumeration value="parallelogram"/>
1740     <xsd:enumeration value="trapezoid"/>
1741     <xsd:enumeration value="nonIsoscelesTrapezoid"/>
1742     <xsd:enumeration value="pentagon"/>
1743     <xsd:enumeration value="hexagon"/>
1744     <xsd:enumeration value="heptagon"/>
1745     <xsd:enumeration value="octagon"/>
1746     <xsd:enumeration value="decagon"/>
1747     <xsd:enumeration value="dodecagon"/>
1748     <xsd:enumeration value="star4"/>
1749     <xsd:enumeration value="star5"/>
1750     <xsd:enumeration value="star6"/>
1751     <xsd:enumeration value="star7"/>
1752     <xsd:enumeration value="star8"/>
1753     <xsd:enumeration value="star10"/>
1754     <xsd:enumeration value="star12"/>
1755     <xsd:enumeration value="star16"/>
1756     <xsd:enumeration value="star24"/>
1757     <xsd:enumeration value="star32"/>
1758     <xsd:enumeration value="roundRect"/>
1759     <xsd:enumeration value="round1Rect"/>
1760     <xsd:enumeration value="round2SameRect"/>
1761     <xsd:enumeration value="round2DiagRect"/>
1762     <xsd:enumeration value="snipRoundRect"/>
1763     <xsd:enumeration value="snip1Rect"/>
1764     <xsd:enumeration value="snip2SameRect"/>
1765     <xsd:enumeration value="snip2DiagRect"/>
1766     <xsd:enumeration value="plaque"/>
1767     <xsd:enumeration value="ellipse"/>
1768     <xsd:enumeration value="teardrop"/>
1769     <xsd:enumeration value="homePlate"/>
1770     <xsd:enumeration value="chevron"/>
1771     <xsd:enumeration value="pieWedge"/>
1772     <xsd:enumeration value="pie"/>
1773     <xsd:enumeration value="blockArc"/>
1774     <xsd:enumeration value="donut"/>

```



```

1775 <xsd:enumeration value="noSmoking"/>
1776 <xsd:enumeration value="rightArrow"/>
1777 <xsd:enumeration value="leftArrow"/>
1778 <xsd:enumeration value="upArrow"/>
1779 <xsd:enumeration value="downArrow"/>
1780 <xsd:enumeration value="stripedRightArrow"/>
1781 <xsd:enumeration value="notchedRightArrow"/>
1782 <xsd:enumeration value="bentUpArrow"/>
1783 <xsd:enumeration value="leftRightArrow"/>
1784 <xsd:enumeration value="upDownArrow"/>
1785 <xsd:enumeration value="leftUpArrow"/>
1786 <xsd:enumeration value="leftRightUpArrow"/>
1787 <xsd:enumeration value="quadArrow"/>
1788 <xsd:enumeration value="leftArrowCallout"/>
1789 <xsd:enumeration value="rightArrowCallout"/>
1790 <xsd:enumeration value="upArrowCallout"/>
1791 <xsd:enumeration value="downArrowCallout"/>
1792 <xsd:enumeration value="leftRightArrowCallout"/>
1793 <xsd:enumeration value="upDownArrowCallout"/>
1794 <xsd:enumeration value="quadArrowCallout"/>
1795 <xsd:enumeration value="bentArrow"/>
1796 <xsd:enumeration value="uturnArrow"/>
1797 <xsd:enumeration value="circularArrow"/>
1798 <xsd:enumeration value="leftCircularArrow"/>
1799 <xsd:enumeration value="leftRightCircularArrow"/>
1800 <xsd:enumeration value="curvedRightArrow"/>
1801 <xsd:enumeration value="curvedLeftArrow"/>
1802 <xsd:enumeration value="curvedUpArrow"/>
1803 <xsd:enumeration value="curvedDownArrow"/>
1804 <xsd:enumeration value="swooshArrow"/>
1805 <xsd:enumeration value="cube"/>
1806 <xsd:enumeration value="can"/>
1807 <xsd:enumeration value="lightningBolt"/>
1808 <xsd:enumeration value="heart"/>
1809 <xsd:enumeration value="sun"/>
1810 <xsd:enumeration value="moon"/>
1811 <xsd:enumeration value="smileyFace"/>
1812 <xsd:enumeration value="irregularSeal1"/>
1813 <xsd:enumeration value="irregularSeal2"/>
1814 <xsd:enumeration value="foldedCorner"/>
1815 <xsd:enumeration value="bevel"/>
1816 <xsd:enumeration value="frame"/>
1817 <xsd:enumeration value="halfFrame"/>
1818 <xsd:enumeration value="corner"/>
1819 <xsd:enumeration value="diagStripe"/>
1820 <xsd:enumeration value="chord"/>
1821 <xsd:enumeration value="arc"/>
1822 <xsd:enumeration value="leftBracket"/>
1823 <xsd:enumeration value="rightBracket"/>
1824 <xsd:enumeration value="leftBrace"/>
1825 <xsd:enumeration value="rightBrace"/>
1826 <xsd:enumeration value="bracketPair"/>
1827 <xsd:enumeration value="bracePair"/>

```

```

1828     <xsd:enumeration value="straightConnector1"/>
1829     <xsd:enumeration value="bentConnector2"/>
1830     <xsd:enumeration value="bentConnector3"/>
1831     <xsd:enumeration value="bentConnector4"/>
1832     <xsd:enumeration value="bentConnector5"/>
1833     <xsd:enumeration value="curvedConnector2"/>
1834     <xsd:enumeration value="curvedConnector3"/>
1835     <xsd:enumeration value="curvedConnector4"/>
1836     <xsd:enumeration value="curvedConnector5"/>
1837     <xsd:enumeration value="callout1"/>
1838     <xsd:enumeration value="callout2"/>
1839     <xsd:enumeration value="callout3"/>
1840     <xsd:enumeration value="accentCallout1"/>
1841     <xsd:enumeration value="accentCallout2"/>
1842     <xsd:enumeration value="accentCallout3"/>
1843     <xsd:enumeration value="borderCallout1"/>
1844     <xsd:enumeration value="borderCallout2"/>
1845     <xsd:enumeration value="borderCallout3"/>
1846     <xsd:enumeration value="accentBorderCallout1"/>
1847     <xsd:enumeration value="accentBorderCallout2"/>
1848     <xsd:enumeration value="accentBorderCallout3"/>
1849     <xsd:enumeration value="wedgeRectCallout"/>
1850     <xsd:enumeration value="wedgeRoundRectCallout"/>
1851     <xsd:enumeration value="wedgeEllipseCallout"/>
1852     <xsd:enumeration value="cloudCallout"/>
1853     <xsd:enumeration value="cloud"/>
1854     <xsd:enumeration value="ribbon"/>
1855     <xsd:enumeration value="ribbon2"/>
1856     <xsd:enumeration value="ellipseRibbon"/>
1857     <xsd:enumeration value="ellipseRibbon2"/>
1858     <xsd:enumeration value="leftRightRibbon"/>
1859     <xsd:enumeration value="verticalScroll"/>
1860     <xsd:enumeration value="horizontalScroll"/>
1861     <xsd:enumeration value="wave"/>
1862     <xsd:enumeration value="doubleWave"/>
1863     <xsd:enumeration value="plus"/>
1864     <xsd:enumeration value="flowChartProcess"/>
1865     <xsd:enumeration value="flowChartDecision"/>
1866     <xsd:enumeration value="flowChartInputOutput"/>
1867     <xsd:enumeration value="flowChartPredefinedProcess"/>
1868     <xsd:enumeration value="flowChartInternalStorage"/>
1869     <xsd:enumeration value="flowChartDocument"/>
1870     <xsd:enumeration value="flowChartMultidocument"/>
1871     <xsd:enumeration value="flowChartTerminator"/>
1872     <xsd:enumeration value="flowChartPreparation"/>
1873     <xsd:enumeration value="flowChartManualInput"/>
1874     <xsd:enumeration value="flowChartManualOperation"/>
1875     <xsd:enumeration value="flowChartConnector"/>
1876     <xsd:enumeration value="flowChartPunchedCard"/>
1877     <xsd:enumeration value="flowChartPunchedTape"/>
1878     <xsd:enumeration value="flowChartSummingJunction"/>
1879     <xsd:enumeration value="flowChartOr"/>
1880     <xsd:enumeration value="flowChartCollate"/>

```

```

1881     <xsd:enumeration value="flowChartSort"/>
1882     <xsd:enumeration value="flowChartExtract"/>
1883     <xsd:enumeration value="flowChartMerge"/>
1884     <xsd:enumeration value="flowChartOfflineStorage"/>
1885     <xsd:enumeration value="flowChartOnlineStorage"/>
1886     <xsd:enumeration value="flowChartMagneticTape"/>
1887     <xsd:enumeration value="flowChartMagneticDisk"/>
1888     <xsd:enumeration value="flowChartMagneticDrum"/>
1889     <xsd:enumeration value="flowChartDisplay"/>
1890     <xsd:enumeration value="flowChartDelay"/>
1891     <xsd:enumeration value="flowChartAlternateProcess"/>
1892     <xsd:enumeration value="flowChartOffpageConnector"/>
1893     <xsd:enumeration value="actionButtonBlank"/>
1894     <xsd:enumeration value="actionButtonHome"/>
1895     <xsd:enumeration value="actionButtonHelp"/>
1896     <xsd:enumeration value="actionButtonInformation"/>
1897     <xsd:enumeration value="actionButtonForwardNext"/>
1898     <xsd:enumeration value="actionButtonBackPrevious"/>
1899     <xsd:enumeration value="actionButtonEnd"/>
1900     <xsd:enumeration value="actionButtonBeginning"/>
1901     <xsd:enumeration value="actionButtonReturn"/>
1902     <xsd:enumeration value="actionButtonDocument"/>
1903     <xsd:enumeration value="actionButtonSound"/>
1904     <xsd:enumeration value="actionButtonMovie"/>
1905     <xsd:enumeration value="gear6"/>
1906     <xsd:enumeration value="gear9"/>
1907     <xsd:enumeration value="funnel"/>
1908     <xsd:enumeration value="mathPlus"/>
1909     <xsd:enumeration value="mathMinus"/>
1910     <xsd:enumeration value="mathMultiply"/>
1911     <xsd:enumeration value="mathDivide"/>
1912     <xsd:enumeration value="mathEqual"/>
1913     <xsd:enumeration value="mathNotEqual"/>
1914     <xsd:enumeration value="cornerTabs"/>
1915     <xsd:enumeration value="squareTabs"/>
1916     <xsd:enumeration value="plaqueTabs"/>
1917     <xsd:enumeration value="chartX"/>
1918     <xsd:enumeration value="chartStar"/>
1919     <xsd:enumeration value="chartPlus"/>
1920 </xsd:restriction>
1921 </xsd:simpleType>
1922 <xsd:simpleType name="ST_TextShapeType">
1923     <xsd:restriction base="xsd:token">
1924         <xsd:enumeration value="textNoShape"/>
1925         <xsd:enumeration value="textPlain"/>
1926         <xsd:enumeration value="textStop"/>
1927         <xsd:enumeration value="textTriangle"/>
1928         <xsd:enumeration value="textTriangleInverted"/>
1929         <xsd:enumeration value="textChevron"/>
1930         <xsd:enumeration value="textChevronInverted"/>
1931         <xsd:enumeration value="textRingInside"/>
1932         <xsd:enumeration value="textRingOutside"/>
1933         <xsd:enumeration value="textArchUp"/>

```

```

1934     <xsd:enumeration value="textArchDown"/>
1935     <xsd:enumeration value="textCircle"/>
1936     <xsd:enumeration value="textButton"/>
1937     <xsd:enumeration value="textArchUpPour"/>
1938     <xsd:enumeration value="textArchDownPour"/>
1939     <xsd:enumeration value="textCirclePour"/>
1940     <xsd:enumeration value="textButtonPour"/>
1941     <xsd:enumeration value="textCurveUp"/>
1942     <xsd:enumeration value="textCurveDown"/>
1943     <xsd:enumeration value="textCanUp"/>
1944     <xsd:enumeration value="textCanDown"/>
1945     <xsd:enumeration value="textWave1"/>
1946     <xsd:enumeration value="textWave2"/>
1947     <xsd:enumeration value="textDoubleWave1"/>
1948     <xsd:enumeration value="textWave4"/>
1949     <xsd:enumeration value="textInflate"/>
1950     <xsd:enumeration value="textDeflate"/>
1951     <xsd:enumeration value="textInflateBottom"/>
1952     <xsd:enumeration value="textDeflateBottom"/>
1953     <xsd:enumeration value="textInflateTop"/>
1954     <xsd:enumeration value="textDeflateTop"/>
1955     <xsd:enumeration value="textDeflateInflate"/>
1956     <xsd:enumeration value="textDeflateInflateDeflate"/>
1957     <xsd:enumeration value="textFadeRight"/>
1958     <xsd:enumeration value="textFadeLeft"/>
1959     <xsd:enumeration value="textFadeUp"/>
1960     <xsd:enumeration value="textFadeDown"/>
1961     <xsd:enumeration value="textSlantUp"/>
1962     <xsd:enumeration value="textSlantDown"/>
1963     <xsd:enumeration value="textCascadeUp"/>
1964     <xsd:enumeration value="textCascadeDown"/>
1965 </xsd:restriction>
1966 </xsd:simpleType>
1967 <xsd:simpleType name="ST_GeomGuideName">
1968     <xsd:restriction base="xsd:token"/>
1969 </xsd:simpleType>
1970 <xsd:simpleType name="ST_GeomGuideFormula">
1971     <xsd:restriction base="xsd:string"/>
1972 </xsd:simpleType>
1973 <xsd:complexType name="CT_GeomGuide">
1974     <xsd:attribute name="name" type="ST_GeomGuideName" use="required"/>
1975     <xsd:attribute name="fmla" type="ST_GeomGuideFormula" use="required"/>
1976 </xsd:complexType>
1977 <xsd:complexType name="CT_GeomGuideList">
1978     <xsd:sequence>
1979         <xsd:element name="gd" type="CT_GeomGuide" minOccurs="0" maxOccurs="unbounded"/>
1980     </xsd:sequence>
1981 </xsd:complexType>
1982 <xsd:simpleType name="ST_AdjCoordinate">
1983     <xsd:union memberTypes="ST_Coordinate ST_GeomGuideName"/>
1984 </xsd:simpleType>
1985 <xsd:simpleType name="ST_AdjAngle">
1986     <xsd:union memberTypes="ST_Angle ST_GeomGuideName"/>

```

```

1987 </xsd:simpleType>
1988 <xsd:complexType name="CT_AdjPoint2D">
1989     <xsd:attribute name="x" type="ST_AdjCoordinate" use="required"/>
1990     <xsd:attribute name="y" type="ST_AdjCoordinate" use="required"/>
1991 </xsd:complexType>
1992 <xsd:complexType name="CT_GeomRect">
1993     <xsd:attribute name="l" type="ST_AdjCoordinate" use="required"/>
1994     <xsd:attribute name="t" type="ST_AdjCoordinate" use="required"/>
1995     <xsd:attribute name="r" type="ST_AdjCoordinate" use="required"/>
1996     <xsd:attribute name="b" type="ST_AdjCoordinate" use="required"/>
1997 </xsd:complexType>
1998 <xsd:complexType name="CT_XYAdjustHandle">
1999     <xsd:sequence>
2000         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2001     </xsd:sequence>
2002     <xsd:attribute name="gdRefX" type="ST_GeomGuideName" use="optional"/>
2003     <xsd:attribute name="minX" type="ST_AdjCoordinate" use="optional"/>
2004     <xsd:attribute name="maxX" type="ST_AdjCoordinate" use="optional"/>
2005     <xsd:attribute name="gdRefY" type="ST_GeomGuideName" use="optional"/>
2006     <xsd:attribute name="minY" type="ST_AdjCoordinate" use="optional"/>
2007     <xsd:attribute name="maxY" type="ST_AdjCoordinate" use="optional"/>
2008 </xsd:complexType>
2009 <xsd:complexType name="CT_PolarAdjustHandle">
2010     <xsd:sequence>
2011         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2012     </xsd:sequence>
2013     <xsd:attribute name="gdRefR" type="ST_GeomGuideName" use="optional"/>
2014     <xsd:attribute name="minR" type="ST_AdjCoordinate" use="optional"/>
2015     <xsd:attribute name="maxR" type="ST_AdjCoordinate" use="optional"/>
2016     <xsd:attribute name="gdRefAng" type="ST_GeomGuideName" use="optional"/>
2017     <xsd:attribute name="minAng" type="ST_AdjAngle" use="optional"/>
2018     <xsd:attribute name="maxAng" type="ST_AdjAngle" use="optional"/>
2019 </xsd:complexType>
2020 <xsd:complexType name="CT_ConnectionSite">
2021     <xsd:sequence>
2022         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2023     </xsd:sequence>
2024     <xsd:attribute name="ang" type="ST_AdjAngle" use="required"/>
2025 </xsd:complexType>
2026 <xsd:complexType name="CT_AdjustHandleList">
2027     <xsd:choice minOccurs="0" maxOccurs="unbounded">
2028         <xsd:element name="ahXY" type="CT_XYAdjustHandle" minOccurs="1" maxOccurs="1"/>
2029         <xsd:element name="ahPolar" type="CT_PolarAdjustHandle" minOccurs="1" maxOccurs="1"/>
2030     </xsd:choice>
2031 </xsd:complexType>
2032 <xsd:complexType name="CT_ConnectionSiteList">
2033     <xsd:sequence>
2034         <xsd:element name="cxn" type="CT_ConnectionSite" minOccurs="0" maxOccurs="unbounded"/>
2035     </xsd:sequence>
2036 </xsd:complexType>
2037 <xsd:complexType name="CT_Connection">
2038     <xsd:attribute name="id" type="ST_DrawingElementId" use="required"/>
2039     <xsd:attribute name="idx" type="xsd:unsignedInt" use="required"/>

```

```

2040 </xsd:complexType>
2041 <xsd:complexType name="CT_Path2DMoveTo">
2042   <xsd:sequence>
2043     <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2044   </xsd:sequence>
2045 </xsd:complexType>
2046 <xsd:complexType name="CT_Path2DLineTo">
2047   <xsd:sequence>
2048     <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2049   </xsd:sequence>
2050 </xsd:complexType>
2051 <xsd:complexType name="CT_Path2DArcTo">
2052   <xsd:attribute name="wR" type="ST_AdjCoordinate" use="required"/>
2053   <xsd:attribute name="hR" type="ST_AdjCoordinate" use="required"/>
2054   <xsd:attribute name="stAng" type="ST_AdjAngle" use="required"/>
2055   <xsd:attribute name="swAng" type="ST_AdjAngle" use="required"/>
2056 </xsd:complexType>
2057 <xsd:complexType name="CT_Path2DQuadBezierTo">
2058   <xsd:sequence>
2059     <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="2" maxOccurs="2"/>
2060   </xsd:sequence>
2061 </xsd:complexType>
2062 <xsd:complexType name="CT_Path2DCubicBezierTo">
2063   <xsd:sequence>
2064     <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="3" maxOccurs="3"/>
2065   </xsd:sequence>
2066 </xsd:complexType>
2067 <xsd:complexType name="CT_Path2DClose"/>
2068 <xsd:simpleType name="ST_PathFillMode">
2069   <xsd:restriction base="xsd:token">
2070     <xsd:enumeration value="none"/>
2071     <xsd:enumeration value="norm"/>
2072     <xsd:enumeration value="lighten"/>
2073     <xsd:enumeration value="lightenLess"/>
2074     <xsd:enumeration value="darken"/>
2075     <xsd:enumeration value="darkenLess"/>
2076   </xsd:restriction>
2077 </xsd:simpleType>
2078 <xsd:complexType name="CT_Path2D">
2079   <xsd:choice minOccurs="0" maxOccurs="unbounded">
2080     <xsd:element name="close" type="CT_Path2DClose" minOccurs="1" maxOccurs="1"/>
2081     <xsd:element name="moveTo" type="CT_Path2DMoveTo" minOccurs="1" maxOccurs="1"/>
2082     <xsd:element name="lnTo" type="CT_Path2DLineTo" minOccurs="1" maxOccurs="1"/>
2083     <xsd:element name="arcTo" type="CT_Path2DArcTo" minOccurs="1" maxOccurs="1"/>
2084     <xsd:element name="quadBezTo" type="CT_Path2DQuadBezierTo" minOccurs="1" maxOccurs="1"/>
2085     <xsd:element name="cubicBezTo" type="CT_Path2DCubicBezierTo" minOccurs="1" maxOccurs="1"/>
2086   </xsd:choice>
2087   <xsd:attribute name="w" type="ST_PositiveCoordinate" use="optional" default="0"/>
2088   <xsd:attribute name="h" type="ST_PositiveCoordinate" use="optional" default="0"/>
2089   <xsd:attribute name="fill" type="ST_PathFillMode" use="optional" default="norm"/>
2090   <xsd:attribute name="stroke" type="xsd:boolean" use="optional" default="true"/>
2091   <xsd:attribute name="extrusionOk" type="xsd:boolean" use="optional" default="true"/>
2092 </xsd:complexType>

```

```

2093 <xsd:complexType name="CT_Path2DList">
2094   <xsd:sequence>
2095     <xsd:element name="path" type="CT_Path2D" minOccurs="0" maxOccurs="unbounded"/>
2096   </xsd:sequence>
2097 </xsd:complexType>
2098 <xsd:complexType name="CT_PresetGeometry2D">
2099   <xsd:sequence>
2100     <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2101   </xsd:sequence>
2102   <xsd:attribute name="prst" type="ST_ShapeType" use="required"/>
2103 </xsd:complexType>
2104 <xsd:complexType name="CT_PresetTextShape">
2105   <xsd:sequence>
2106     <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2107   </xsd:sequence>
2108   <xsd:attribute name="prst" type="ST_TextShapeType" use="required"/>
2109 </xsd:complexType>
2110 <xsd:complexType name="CT_CustomGeometry2D">
2111   <xsd:sequence>
2112     <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2113     <xsd:element name="gdLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2114     <xsd:element name="ahLst" type="CT_AdjustHandleList" minOccurs="0" maxOccurs="1"/>
2115     <xsd:element name="cxnLst" type="CT_ConnectionSiteList" minOccurs="0" maxOccurs="1"/>
2116     <xsd:element name="rect" type="CT_GeomRect" minOccurs="0" maxOccurs="1"/>
2117     <xsd:element name="pathLst" type="CT_Path2DList" minOccurs="1" maxOccurs="1"/>
2118   </xsd:sequence>
2119 </xsd:complexType>
2120 <xsd:group name="EG_Geometry">
2121   <xsd:choice>
2122     <xsd:element name="custGeom" type="CT_CustomGeometry2D" minOccurs="1" maxOccurs="1"/>
2123     <xsd:element name="prstGeom" type="CT_PresetGeometry2D" minOccurs="1" maxOccurs="1"/>
2124   </xsd:choice>
2125 </xsd:group>
2126 <xsd:group name="EG_TextGeometry">
2127   <xsd:choice>
2128     <xsd:element name="custGeom" type="CT_CustomGeometry2D" minOccurs="1" maxOccurs="1"/>
2129     <xsd:element name="prstTxWarp" type="CT_PresetTextShape" minOccurs="1" maxOccurs="1"/>
2130   </xsd:choice>
2131 </xsd:group>
2132 <xsd:simpleType name="ST_LineEndType">
2133   <xsd:restriction base="xsd:token">
2134     <xsd:enumeration value="none"/>
2135     <xsd:enumeration value="triangle"/>
2136     <xsd:enumeration value="stealth"/>
2137     <xsd:enumeration value="diamond"/>
2138     <xsd:enumeration value="oval"/>
2139     <xsd:enumeration value="arrow"/>
2140   </xsd:restriction>
2141 </xsd:simpleType>
2142 <xsd:simpleType name="ST_LineEndWidth">
2143   <xsd:restriction base="xsd:token">
2144     <xsd:enumeration value="sm"/>
2145     <xsd:enumeration value="med"/>

```

```

2146         <xsd:enumeration value="lg"/>
2147     </xsd:restriction>
2148 </xsd:simpleType>
2149 <xsd:simpleType name="ST_LineEndLength">
2150     <xsd:restriction base="xsd:token">
2151         <xsd:enumeration value="sm"/>
2152         <xsd:enumeration value="med"/>
2153         <xsd:enumeration value="lg"/>
2154     </xsd:restriction>
2155 </xsd:simpleType>
2156 <xsd:complexType name="CT_LineEndProperties">
2157     <xsd:attribute name="type" type="ST_LineEndType" use="optional"/>
2158     <xsd:attribute name="w" type="ST_LineEndWidth" use="optional"/>
2159     <xsd:attribute name="len" type="ST_LineEndLength" use="optional"/>
2160 </xsd:complexType>
2161 <xsd:group name="EG_LineFillProperties">
2162     <xsd:choice>
2163         <xsd:element name="noFill" type="CT_NoFillProperties" minOccurs="1" maxOccurs="1"/>
2164         <xsd:element name="solidFill" type="CT_SolidColorFillProperties" minOccurs="1"
2165             maxOccurs="1"/>
2166         <xsd:element name="gradFill" type="CT_GradientFillProperties" minOccurs="1"
2167             maxOccurs="1"/>
2168         <xsd:element name="pattFill" type="CT_PatternFillProperties" minOccurs="1" maxOccurs="1"/>
2169     </xsd:choice>
2170 </xsd:group>
2171 <xsd:complexType name="CT_LineJoinBevel"/>
2172 <xsd:complexType name="CT_LineJoinRound"/>
2173 <xsd:complexType name="CT_LineJoinMiterProperties">
2174     <xsd:attribute name="lim" type="ST_PositivePercentage" use="optional"/>
2175 </xsd:complexType>
2176 <xsd:group name="EG_LineJoinProperties">
2177     <xsd:choice>
2178         <xsd:element name="round" type="CT_LineJoinRound" minOccurs="1" maxOccurs="1"/>
2179         <xsd:element name="bevel" type="CT_LineJoinBevel" minOccurs="1" maxOccurs="1"/>
2180         <xsd:element name="miter" type="CT_LineJoinMiterProperties" minOccurs="1" maxOccurs="1"/>
2181     </xsd:choice>
2182 </xsd:group>
2183 <xsd:simpleType name="ST_PresetLineDashVal">
2184     <xsd:restriction base="xsd:token">
2185         <xsd:enumeration value="solid"/>
2186         <xsd:enumeration value="dot"/>
2187         <xsd:enumeration value="dash"/>
2188         <xsd:enumeration value="lgDash"/>
2189         <xsd:enumeration value="dashDot"/>
2190         <xsd:enumeration value="lgDashDot"/>
2191         <xsd:enumeration value="lgDashDotDot"/>
2192         <xsd:enumeration value="sysDash"/>
2193         <xsd:enumeration value="sysDot"/>
2194         <xsd:enumeration value="sysDashDot"/>
2195         <xsd:enumeration value="sysDashDotDot"/>
2196     </xsd:restriction>
2197 </xsd:simpleType>
2198 <xsd:complexType name="CT_PresetLineDashProperties">

```



```

2199     <xsd:attribute name="val" type="ST_PresetLineDashVal" use="optional"/>
2200 </xsd:complexType>
2201 <xsd:complexType name="CT_DashStop">
2202     <xsd:attribute name="d" type="ST_PositivePercentage" use="required"/>
2203     <xsd:attribute name="sp" type="ST_PositivePercentage" use="required"/>
2204 </xsd:complexType>
2205 <xsd:complexType name="CT_DashStopList">
2206     <xsd:sequence>
2207         <xsd:element name="ds" type="CT_DashStop" minOccurs="0" maxOccurs="unbounded"/>
2208     </xsd:sequence>
2209 </xsd:complexType>
2210 <xsd:group name="EG_LineDashProperties">
2211     <xsd:choice>
2212         <xsd:element name="prstDash" type="CT_PresetLineDashProperties" minOccurs="1"
2213             maxOccurs="1"/>
2214         <xsd:element name="custDash" type="CT_DashStopList" minOccurs="1" maxOccurs="1"/>
2215     </xsd:choice>
2216 </xsd:group>
2217 <xsd:simpleType name="ST_LineCap">
2218     <xsd:restriction base="xsd:token">
2219         <xsd:enumeration value="rnd"/>
2220         <xsd:enumeration value="sq"/>
2221         <xsd:enumeration value="flat"/>
2222     </xsd:restriction>
2223 </xsd:simpleType>
2224 <xsd:simpleType name="ST_LineWidth">
2225     <xsd:restriction base="ST_Coordinate32Unqualified">
2226         <xsd:minInclusive value="0"/>
2227         <xsd:maxInclusive value="20116800"/>
2228     </xsd:restriction>
2229 </xsd:simpleType>
2230 <xsd:simpleType name="ST_PenAlignment">
2231     <xsd:restriction base="xsd:token">
2232         <xsd:enumeration value="ctr"/>
2233         <xsd:enumeration value="in"/>
2234     </xsd:restriction>
2235 </xsd:simpleType>
2236 <xsd:simpleType name="ST_CompoundLine">
2237     <xsd:restriction base="xsd:token">
2238         <xsd:enumeration value="sng"/>
2239         <xsd:enumeration value="dbl"/>
2240         <xsd:enumeration value="thickThin"/>
2241         <xsd:enumeration value="thinThick"/>
2242         <xsd:enumeration value="tri"/>
2243     </xsd:restriction>
2244 </xsd:simpleType>
2245 <xsd:complexType name="CT_LineProperties">
2246     <xsd:sequence>
2247         <xsd:group ref="EG_LineFillProperties" minOccurs="0" maxOccurs="1"/>
2248         <xsd:group ref="EG_LineDashProperties" minOccurs="0" maxOccurs="1"/>
2249         <xsd:group ref="EG_LineJoinProperties" minOccurs="0" maxOccurs="1"/>
2250         <xsd:element name="headEnd" type="CT_LineEndProperties" minOccurs="0" maxOccurs="1"/>
2251         <xsd:element name="tailEnd" type="CT_LineEndProperties" minOccurs="0" maxOccurs="1"/>

```

```

2252     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2253   </xsd:sequence>
2254   <xsd:attribute name="w" type="ST_LineWidth" use="optional"/>
2255   <xsd:attribute name="cap" type="ST_LineCap" use="optional"/>
2256   <xsd:attribute name="cmpd" type="ST_CompoundLine" use="optional"/>
2257   <xsd:attribute name="align" type="ST_PenAlignment" use="optional"/>
2258 </xsd:complexType>
2259 <xsd:simpleType name="ST_ShapeID">
2260   <xsd:restriction base="xsd:token"/>
2261 </xsd:simpleType>
2262 <xsd:complexType name="CT_ShapeProperties">
2263   <xsd:sequence>
2264     <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="0" maxOccurs="1"/>
2265     <xsd:group ref="EG_Geometry" minOccurs="0" maxOccurs="1"/>
2266     <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2267     <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2268     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2269     <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
2270     <xsd:element name="sp3d" type="CT_Shape3D" minOccurs="0" maxOccurs="1"/>
2271     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2272   </xsd:sequence>
2273   <xsd:attribute name="bwMode" type="ST_BlackWhiteMode" use="optional"/>
2274 </xsd:complexType>
2275 <xsd:complexType name="CT_GroupShapeProperties">
2276   <xsd:sequence>
2277     <xsd:element name="xfrm" type="CT_GroupTransform2D" minOccurs="0" maxOccurs="1"/>
2278     <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2279     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2280     <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
2281     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2282   </xsd:sequence>
2283   <xsd:attribute name="bwMode" type="ST_BlackWhiteMode" use="optional"/>
2284 </xsd:complexType>
2285 <xsd:complexType name="CT_StyleMatrixReference">
2286   <xsd:sequence>
2287     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2288   </xsd:sequence>
2289   <xsd:attribute name="idx" type="ST_StyleMatrixColumnIndex" use="required"/>
2290 </xsd:complexType>
2291 <xsd:complexType name="CT_FontReference">
2292   <xsd:sequence>
2293     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2294   </xsd:sequence>
2295   <xsd:attribute name="idx" type="ST_FontCollectionIndex" use="required"/>
2296 </xsd:complexType>
2297 <xsd:complexType name="CT_ShapeStyle">
2298   <xsd:sequence>
2299     <xsd:element name="lnRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2300     <xsd:element name="fillRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2301     <xsd:element name="effectRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2302     <xsd:element name="fontRef" type="CT_FontReference" minOccurs="1" maxOccurs="1"/>
2303   </xsd:sequence>
2304 </xsd:complexType>

```

```

2305 <xsd:complexType name="CT_DefaultShapeDefinition">
2306   <xsd:sequence>
2307     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
2308     <xsd:element name="bodyPr" type="CT_TextBodyProperties" minOccurs="1" maxOccurs="1"/>
2309     <xsd:element name="lstStyle" type="CT_TextListStyle" minOccurs="1" maxOccurs="1"/>
2310     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
2311     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2312   </xsd:sequence>
2313 </xsd:complexType>
2314 <xsd:complexType name="CT_ObjectStyleDefaults">
2315   <xsd:sequence>
2316     <xsd:element name="spDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2317     <xsd:element name="lnDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2318     <xsd:element name="txDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2319     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2320   </xsd:sequence>
2321 </xsd:complexType>
2322 <xsd:complexType name="CT_EmptyElement"/>
2323 <xsd:complexType name="CT_ColorMapping">
2324   <xsd:sequence>
2325     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2326   </xsd:sequence>
2327   <xsd:attribute name="bg1" type="ST_ColorSchemeIndex" use="required"/>
2328   <xsd:attribute name="tx1" type="ST_ColorSchemeIndex" use="required"/>
2329   <xsd:attribute name="bg2" type="ST_ColorSchemeIndex" use="required"/>
2330   <xsd:attribute name="tx2" type="ST_ColorSchemeIndex" use="required"/>
2331   <xsd:attribute name="accent1" type="ST_ColorSchemeIndex" use="required"/>
2332   <xsd:attribute name="accent2" type="ST_ColorSchemeIndex" use="required"/>
2333   <xsd:attribute name="accent3" type="ST_ColorSchemeIndex" use="required"/>
2334   <xsd:attribute name="accent4" type="ST_ColorSchemeIndex" use="required"/>
2335   <xsd:attribute name="accent5" type="ST_ColorSchemeIndex" use="required"/>
2336   <xsd:attribute name="accent6" type="ST_ColorSchemeIndex" use="required"/>
2337   <xsd:attribute name="hlink" type="ST_ColorSchemeIndex" use="required"/>
2338   <xsd:attribute name="folHlink" type="ST_ColorSchemeIndex" use="required"/>
2339 </xsd:complexType>
2340 <xsd:complexType name="CT_ColorMappingOverride">
2341   <xsd:sequence>
2342     <xsd:choice minOccurs="1" maxOccurs="1">
2343       <xsd:element name="masterClrMapping" type="CT_EmptyElement"/>
2344       <xsd:element name="overrideClrMapping" type="CT_ColorMapping"/>
2345     </xsd:choice>
2346   </xsd:sequence>
2347 </xsd:complexType>
2348 <xsd:complexType name="CT_ColorSchemeAndMapping">
2349   <xsd:sequence>
2350     <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="1" maxOccurs="1"/>
2351     <xsd:element name="clrMap" type="CT_ColorMapping" minOccurs="0" maxOccurs="1"/>
2352   </xsd:sequence>
2353 </xsd:complexType>
2354 <xsd:complexType name="CT_ColorSchemeList">
2355   <xsd:sequence>
2356     <xsd:element name="extraClrScheme" type="CT_ColorSchemeAndMapping" minOccurs="0"
2357       maxOccurs="unbounded"/>

```

```

2358     </xsd:sequence>
2359 </xsd:complexType>
2360 <xsd:complexType name="CT_OfficeStyleSheet">
2361     <xsd:sequence>
2362         <xsd:element name="themeElements" type="CT_BaseStyles" minOccurs="1" maxOccurs="1"/>
2363         <xsd:element name="objectDefaults" type="CT_ObjectStyleDefaults" minOccurs="0"
2364             maxOccurs="1"/>
2365         <xsd:element name="extraClrSchemeLst" type="CT_ColorSchemeList" minOccurs="0"
2366             maxOccurs="1"/>
2367         <xsd:element name="custClrLst" type="CT_CustomColorList" minOccurs="0" maxOccurs="1"/>
2368         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2369     </xsd:sequence>
2370     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
2371 </xsd:complexType>
2372 <xsd:complexType name="CT_BaseStylesOverride">
2373     <xsd:sequence>
2374         <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="0" maxOccurs="1"/>
2375         <xsd:element name="fontScheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
2376         <xsd:element name="fmtScheme" type="CT_StyleMatrix" minOccurs="0" maxOccurs="1"/>
2377     </xsd:sequence>
2378 </xsd:complexType>
2379 <xsd:complexType name="CT_ClipboardStyleSheet">
2380     <xsd:sequence>
2381         <xsd:element name="themeElements" type="CT_BaseStyles" minOccurs="1" maxOccurs="1"/>
2382         <xsd:element name="clrMap" type="CT_ColorMapping" minOccurs="1" maxOccurs="1"/>
2383     </xsd:sequence>
2384 </xsd:complexType>
2385 <xsd:element name="theme" type="CT_OfficeStyleSheet"/>
2386 <xsd:element name="themeOverride" type="CT_BaseStylesOverride"/>
2387 <xsd:element name="themeManager" type="CT_EmptyElement"/>
2388 <xsd:complexType name="CT_TableCellProperties">
2389     <xsd:sequence>
2390         <xsd:element name="lnL" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2391         <xsd:element name="lnR" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2392         <xsd:element name="lnT" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2393         <xsd:element name="lnB" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2394         <xsd:element name="lnTlToBr" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2395         <xsd:element name="lnBlToTr" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2396         <xsd:element name="cell3D" type="CT_Cell3D" minOccurs="0" maxOccurs="1"/>
2397         <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2398         <xsd:element name="headers" type="CT_Headers" minOccurs="0"/>
2399         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2400     </xsd:sequence>
2401     <xsd:attribute name="marL" type="ST_Coordinate32" use="optional" default="91440"/>
2402     <xsd:attribute name="marR" type="ST_Coordinate32" use="optional" default="91440"/>
2403     <xsd:attribute name="marT" type="ST_Coordinate32" use="optional" default="45720"/>
2404     <xsd:attribute name="marB" type="ST_Coordinate32" use="optional" default="45720"/>
2405     <xsd:attribute name="vert" type="ST_TextVerticalType" use="optional" default="horz"/>
2406     <xsd:attribute name="anchor" type="ST_TextAnchoringType" use="optional" default="t"/>
2407     <xsd:attribute name="anchorCtr" type="xsd:boolean" use="optional" default="false"/>
2408     <xsd:attribute name="horzOverflow" type="ST_TextHorzOverflowType" use="optional"
2409         default="clip"/>
2410 </xsd:complexType>

```

```

2411 <xsd:complexType name="CT_Headers">
2412     <xsd:sequence minOccurs="0" maxOccurs="unbounded">
2413         <xsd:element name="header" type="xsd:string"/>
2414     </xsd:sequence>
2415 </xsd:complexType>
2416 <xsd:complexType name="CT_TableCol">
2417     <xsd:sequence>
2418         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2419     </xsd:sequence>
2420     <xsd:attribute name="w" type="ST Coordinate" use="required"/>
2421 </xsd:complexType>
2422 <xsd:complexType name="CT_TableGrid">
2423     <xsd:sequence>
2424         <xsd:element name="gridCol" type="CT TableCol" minOccurs="0" maxOccurs="unbounded"/>
2425     </xsd:sequence>
2426 </xsd:complexType>
2427 <xsd:complexType name="CT_TableCell">
2428     <xsd:sequence>
2429         <xsd:element name="txBody" type="CT TextBody" minOccurs="0" maxOccurs="1"/>
2430         <xsd:element name="tcPr" type="CT TableCellProperties" minOccurs="0" maxOccurs="1"/>
2431         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2432     </xsd:sequence>
2433     <xsd:attribute name="rowSpan" type="xsd:int" use="optional" default="1"/>
2434     <xsd:attribute name="gridSpan" type="xsd:int" use="optional" default="1"/>
2435     <xsd:attribute name="hMerge" type="xsd:boolean" use="optional" default="false"/>
2436     <xsd:attribute name="vMerge" type="xsd:boolean" use="optional" default="false"/>
2437     <xsd:attribute name="id" type="xsd:string" use="optional"/>
2438 </xsd:complexType>
2439 <xsd:complexType name="CT_TableRow">
2440     <xsd:sequence>
2441         <xsd:element name="tc" type="CT TableCell" minOccurs="0" maxOccurs="unbounded"/>
2442         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2443     </xsd:sequence>
2444     <xsd:attribute name="h" type="ST Coordinate" use="required"/>
2445 </xsd:complexType>
2446 <xsd:complexType name="CT_TableProperties">
2447     <xsd:sequence>
2448         <xsd:group ref="EG FillProperties" minOccurs="0" maxOccurs="1"/>
2449         <xsd:group ref="EG EffectProperties" minOccurs="0" maxOccurs="1"/>
2450         <xsd:choice minOccurs="0" maxOccurs="1">
2451             <xsd:element name="tableStyle" type="CT TableStyle"/>
2452             <xsd:element name="tableStyleId" type="s:ST Guid"/>
2453         </xsd:choice>
2454         <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2455     </xsd:sequence>
2456     <xsd:attribute name="rtl" type="xsd:boolean" use="optional" default="false"/>
2457     <xsd:attribute name="firstRow" type="xsd:boolean" use="optional" default="false"/>
2458     <xsd:attribute name="firstCol" type="xsd:boolean" use="optional" default="false"/>
2459     <xsd:attribute name="lastRow" type="xsd:boolean" use="optional" default="false"/>
2460     <xsd:attribute name="lastCol" type="xsd:boolean" use="optional" default="false"/>
2461     <xsd:attribute name="bandRow" type="xsd:boolean" use="optional" default="false"/>
2462     <xsd:attribute name="bandCol" type="xsd:boolean" use="optional" default="false"/>
2463 </xsd:complexType>

```

```

2464 <xsd:complexType name="CT_Table">
2465   <xsd:sequence>
2466     <xsd:element name="tblPr" type="CT_TableProperties" minOccurs="0" maxOccurs="1"/>
2467     <xsd:element name="tblGrid" type="CT_TableGrid" minOccurs="1" maxOccurs="1"/>
2468     <xsd:element name="tr" type="CT_TableRow" minOccurs="0" maxOccurs="unbounded"/>
2469   </xsd:sequence>
2470 </xsd:complexType>
2471 <xsd:element name="tbl" type="CT_Table"/>
2472 <xsd:complexType name="CT_Cell3D">
2473   <xsd:sequence>
2474     <xsd:element name="bevel" type="CT_Bevel" minOccurs="1" maxOccurs="1"/>
2475     <xsd:element name="lightRig" type="CT_LightRig" minOccurs="0" maxOccurs="1"/>
2476     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2477   </xsd:sequence>
2478   <xsd:attribute name="prstMaterial" type="ST_PresetMaterialType" use="optional"
2479     default="plastic"/>
2480 </xsd:complexType>
2481 <xsd:group name="EG_ThemeableFillStyle">
2482   <xsd:choice>
2483     <xsd:element name="fill" type="CT_FillProperties" minOccurs="1" maxOccurs="1"/>
2484     <xsd:element name="fillRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2485   </xsd:choice>
2486 </xsd:group>
2487 <xsd:complexType name="CT_ThemeableLineStyle">
2488   <xsd:choice>
2489     <xsd:element name="ln" type="CT_LineProperties" minOccurs="1" maxOccurs="1"/>
2490     <xsd:element name="lnRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2491   </xsd:choice>
2492 </xsd:complexType>
2493 <xsd:group name="EG_ThemeableEffectStyle">
2494   <xsd:choice>
2495     <xsd:element name="effect" type="CT_EffectProperties" minOccurs="1" maxOccurs="1"/>
2496     <xsd:element name="effectRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2497   </xsd:choice>
2498 </xsd:group>
2499 <xsd:group name="EG_ThemeableFontStyles">
2500   <xsd:choice>
2501     <xsd:element name="font" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
2502     <xsd:element name="fontRef" type="CT_FontReference" minOccurs="1" maxOccurs="1"/>
2503   </xsd:choice>
2504 </xsd:group>
2505 <xsd:simpleType name="ST_OnOffStyleType">
2506   <xsd:restriction base="xsd:token">
2507     <xsd:enumeration value="on"/>
2508     <xsd:enumeration value="off"/>
2509     <xsd:enumeration value="def"/>
2510   </xsd:restriction>
2511 </xsd:simpleType>
2512 <xsd:complexType name="CT_TableStyleTextStyle">
2513   <xsd:sequence>
2514     <xsd:group ref="EG_ThemeableFontStyles" minOccurs="0" maxOccurs="1"/>
2515     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2516     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

2517     </xsd:sequence>
2518     <xsd:attribute name="b" type="ST_OnOffStyleType" use="optional" default="def"/>
2519     <xsd:attribute name="i" type="ST_OnOffStyleType" use="optional" default="def"/>
2520 </xsd:complexType>
2521 <xsd:complexType name="CT_TableCellBorderStyle">
2522     <xsd:sequence>
2523         <xsd:element name="left" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2524         <xsd:element name="right" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2525         <xsd:element name="top" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2526         <xsd:element name="bottom" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2527         <xsd:element name="insideH" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2528         <xsd:element name="insideV" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2529         <xsd:element name="tl2br" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2530         <xsd:element name="tr2bl" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2531         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2532     </xsd:sequence>
2533 </xsd:complexType>
2534 <xsd:complexType name="CT_TableBackgroundStyle">
2535     <xsd:sequence>
2536         <xsd:group ref="EG_ThemeableFillStyle" minOccurs="0" maxOccurs="1"/>
2537         <xsd:group ref="EG_ThemeableEffectStyle" minOccurs="0" maxOccurs="1"/>
2538     </xsd:sequence>
2539 </xsd:complexType>
2540 <xsd:complexType name="CT_TableStyleCellStyle">
2541     <xsd:sequence>
2542         <xsd:element name="tcBdr" type="CT_TableCellBorderStyle" minOccurs="0" maxOccurs="1"/>
2543         <xsd:group ref="EG_ThemeableFillStyle" minOccurs="0" maxOccurs="1"/>
2544         <xsd:element name="cell3D" type="CT_Cell3D" minOccurs="0" maxOccurs="1"/>
2545     </xsd:sequence>
2546 </xsd:complexType>
2547 <xsd:complexType name="CT_TablePartStyle">
2548     <xsd:sequence>
2549         <xsd:element name="tcTxStyle" type="CT_TableStyleTextStyle" minOccurs="0" maxOccurs="1"/>
2550         <xsd:element name="tcStyle" type="CT_TableStyleCellStyle" minOccurs="0" maxOccurs="1"/>
2551     </xsd:sequence>
2552 </xsd:complexType>
2553 <xsd:complexType name="CT_TableStyle">
2554     <xsd:sequence>
2555         <xsd:element name="tblBg" type="CT_TableBackgroundStyle" minOccurs="0" maxOccurs="1"/>
2556         <xsd:element name="wholeTbl" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2557         <xsd:element name="band1H" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2558         <xsd:element name="band2H" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2559         <xsd:element name="band1V" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2560         <xsd:element name="band2V" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2561         <xsd:element name="lastCol" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2562         <xsd:element name="firstCol" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2563         <xsd:element name="lastRow" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2564         <xsd:element name="seCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2565         <xsd:element name="swCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2566         <xsd:element name="firstRow" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2567         <xsd:element name="neCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2568         <xsd:element name="nwCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2569         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

2570     </xsd:sequence>
2571     <xsd:attribute name="styleId" type="s:ST_Guid" use="required"/>
2572     <xsd:attribute name="styleName" type="xsd:string" use="required"/>
2573 </xsd:complexType>
2574 <xsd:complexType name="CT_TableStyleList">
2575     <xsd:sequence>
2576         <xsd:element name="tblStyle" type="CT_TableStyle" minOccurs="0" maxOccurs="unbounded"/>
2577     </xsd:sequence>
2578     <xsd:attribute name="def" type="s:ST_Guid" use="required"/>
2579 </xsd:complexType>
2580 <xsd:element name="tblStyleLst" type="CT_TableStyleList"/>
2581 <xsd:complexType name="CT_TextParagraph">
2582     <xsd:sequence>
2583         <xsd:element name="pPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
2584         <xsd:group ref="EG_TextRun" minOccurs="0" maxOccurs="unbounded"/>
2585         <xsd:element name="endParaRPr" type="CT_TextCharacterProperties" minOccurs="0"
2586             maxOccurs="1"/>
2587     </xsd:sequence>
2588 </xsd:complexType>
2589 <xsd:simpleType name="ST_TextAnchoringType">
2590     <xsd:restriction base="xsd:token">
2591         <xsd:enumeration value="t"/>
2592         <xsd:enumeration value="ctr"/>
2593         <xsd:enumeration value="b"/>
2594         <xsd:enumeration value="just"/>
2595         <xsd:enumeration value="dist"/>
2596     </xsd:restriction>
2597 </xsd:simpleType>
2598 <xsd:simpleType name="ST_TextVertOverflowType">
2599     <xsd:restriction base="xsd:token">
2600         <xsd:enumeration value="overflow"/>
2601         <xsd:enumeration value="ellipsis"/>
2602         <xsd:enumeration value="clip"/>
2603     </xsd:restriction>
2604 </xsd:simpleType>
2605 <xsd:simpleType name="ST_TextHorzOverflowType">
2606     <xsd:restriction base="xsd:token">
2607         <xsd:enumeration value="overflow"/>
2608         <xsd:enumeration value="clip"/>
2609     </xsd:restriction>
2610 </xsd:simpleType>
2611 <xsd:simpleType name="ST_TextVerticalType">
2612     <xsd:restriction base="xsd:token">
2613         <xsd:enumeration value="horz"/>
2614         <xsd:enumeration value="vert"/>
2615         <xsd:enumeration value="vert270"/>
2616         <xsd:enumeration value="wordArtVert"/>
2617         <xsd:enumeration value="eaVert"/>
2618         <xsd:enumeration value="mongolianVert"/>
2619         <xsd:enumeration value="wordArtVertRtl"/>
2620     </xsd:restriction>
2621 </xsd:simpleType>
2622 <xsd:simpleType name="ST_TextWrappingType">

```



```

2623     <xsd:restriction base="xsd:token">
2624         <xsd:enumeration value="none"/>
2625         <xsd:enumeration value="square"/>
2626     </xsd:restriction>
2627 </xsd:simpleType>
2628 <xsd:simpleType name="ST_TextColumnCount">
2629     <xsd:restriction base="xsd:int">
2630         <xsd:minInclusive value="1"/>
2631         <xsd:maxInclusive value="16"/>
2632     </xsd:restriction>
2633 </xsd:simpleType>
2634 <xsd:complexType name="CT_TextListStyle">
2635     <xsd:sequence>
2636         <xsd:element name="defPPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
2637         <xsd:element name="lv11pPr" type="CT_TextParagraphProperties" minOccurs="0"
2638             maxOccurs="1"/>
2639         <xsd:element name="lv12pPr" type="CT_TextParagraphProperties" minOccurs="0"
2640             maxOccurs="1"/>
2641         <xsd:element name="lv13pPr" type="CT_TextParagraphProperties" minOccurs="0"
2642             maxOccurs="1"/>
2643         <xsd:element name="lv14pPr" type="CT_TextParagraphProperties" minOccurs="0"
2644             maxOccurs="1"/>
2645         <xsd:element name="lv15pPr" type="CT_TextParagraphProperties" minOccurs="0"
2646             maxOccurs="1"/>
2647         <xsd:element name="lv16pPr" type="CT_TextParagraphProperties" minOccurs="0"
2648             maxOccurs="1"/>
2649         <xsd:element name="lv17pPr" type="CT_TextParagraphProperties" minOccurs="0"
2650             maxOccurs="1"/>
2651         <xsd:element name="lv18pPr" type="CT_TextParagraphProperties" minOccurs="0"
2652             maxOccurs="1"/>
2653         <xsd:element name="lv19pPr" type="CT_TextParagraphProperties" minOccurs="0"
2654             maxOccurs="1"/>
2655         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2656     </xsd:sequence>
2657 </xsd:complexType>
2658 <xsd:simpleType name="ST_TextFontScalePercentOrPercentString">
2659     <xsd:union memberTypes="ST_TextFontScalePercent s:ST_Percentage"/>
2660 </xsd:simpleType>
2661 <xsd:simpleType name="ST_TextFontScalePercent">
2662     <xsd:restriction base="ST_PercentageDecimal">
2663         <xsd:minInclusive value="1000"/>
2664         <xsd:maxInclusive value="100000"/>
2665     </xsd:restriction>
2666 </xsd:simpleType>
2667 <xsd:complexType name="CT_TextNormalAutofit">
2668     <xsd:attribute name="fontScale" type="ST_TextFontScalePercentOrPercentString" use="optional"
2669         default="100%"/>
2670     <xsd:attribute name="lnSpcReduction" type="ST_TextSpacingPercentOrPercentString"
2671         use="optional" default="0%"/>
2672 </xsd:complexType>
2673 <xsd:complexType name="CT_TextShapeAutofit"/>
2674 <xsd:complexType name="CT_TextNoAutofit"/>
2675 <xsd:group name="EG_TextAutofit">

```

```

2676     <xsd:choice>
2677         <xsd:element name="noAutofit" type="CT_TextNoAutofit"/>
2678         <xsd:element name="normAutofit" type="CT_TextNormalAutofit"/>
2679         <xsd:element name="spAutoFit" type="CT_TextShapeAutofit"/>
2680     </xsd:choice>
2681 </xsd:group>
2682 <xsd:complexType name="CT_TextBodyProperties">
2683     <xsd:sequence>
2684         <xsd:element name="prstTxWarp" type="CT_PresetTextShape" minOccurs="0" maxOccurs="1"/>
2685         <xsd:group ref="EG_TextAutofit" minOccurs="0" maxOccurs="1"/>
2686         <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
2687         <xsd:group ref="EG_Text3D" minOccurs="0" maxOccurs="1"/>
2688         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2689     </xsd:sequence>
2690     <xsd:attribute name="rot" type="ST_Angle" use="optional"/>
2691     <xsd:attribute name="spcFirstLastPara" type="xsd:boolean" use="optional"/>
2692     <xsd:attribute name="vertOverflow" type="ST_TextVertOverflowType" use="optional"/>
2693     <xsd:attribute name="horzOverflow" type="ST_TextHorzOverflowType" use="optional"/>
2694     <xsd:attribute name="vert" type="ST_TextVerticalType" use="optional"/>
2695     <xsd:attribute name="wrap" type="ST_TextWrappingType" use="optional"/>
2696     <xsd:attribute name="lIns" type="ST_Coordinate32" use="optional"/>
2697     <xsd:attribute name="tIns" type="ST_Coordinate32" use="optional"/>
2698     <xsd:attribute name="rIns" type="ST_Coordinate32" use="optional"/>
2699     <xsd:attribute name="bIns" type="ST_Coordinate32" use="optional"/>
2700     <xsd:attribute name="numCol" type="ST_TextColumnCount" use="optional"/>
2701     <xsd:attribute name="spcCol" type="ST_PositiveCoordinate32" use="optional"/>
2702     <xsd:attribute name="rtlCol" type="xsd:boolean" use="optional"/>
2703     <xsd:attribute name="fromWordArt" type="xsd:boolean" use="optional"/>
2704     <xsd:attribute name="anchor" type="ST_TextAnchoringType" use="optional"/>
2705     <xsd:attribute name="anchorCtr" type="xsd:boolean" use="optional"/>
2706     <xsd:attribute name="forceAA" type="xsd:boolean" use="optional"/>
2707     <xsd:attribute name="upright" type="xsd:boolean" use="optional" default="false"/>
2708     <xsd:attribute name="compatLnSpc" type="xsd:boolean" use="optional"/>
2709 </xsd:complexType>
2710 <xsd:complexType name="CT_TextBody">
2711     <xsd:sequence>
2712         <xsd:element name="bodyPr" type="CT_TextBodyProperties" minOccurs="1" maxOccurs="1"/>
2713         <xsd:element name="lstStyle" type="CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
2714         <xsd:element name="p" type="CT_TextParagraph" minOccurs="1" maxOccurs="unbounded"/>
2715     </xsd:sequence>
2716 </xsd:complexType>
2717 <xsd:simpleType name="ST_TextBulletStartAtNum">
2718     <xsd:restriction base="xsd:int">
2719         <xsd:minInclusive value="1"/>
2720         <xsd:maxInclusive value="32767"/>
2721     </xsd:restriction>
2722 </xsd:simpleType>
2723 <xsd:simpleType name="ST_TextAutonumberScheme">
2724     <xsd:restriction base="xsd:token">
2725         <xsd:enumeration value="alphaLcParenBoth"/>
2726         <xsd:enumeration value="alphaUcParenBoth"/>
2727         <xsd:enumeration value="alphaLcParenR"/>
2728         <xsd:enumeration value="alphaUcParenR"/>

```

```

2729     <xsd:enumeration value="alphaLcPeriod"/>
2730     <xsd:enumeration value="alphaUcPeriod"/>
2731     <xsd:enumeration value="arabicParenBoth"/>
2732     <xsd:enumeration value="arabicParenR"/>
2733     <xsd:enumeration value="arabicPeriod"/>
2734     <xsd:enumeration value="arabicPlain"/>
2735     <xsd:enumeration value="romanLcParenBoth"/>
2736     <xsd:enumeration value="romanUcParenBoth"/>
2737     <xsd:enumeration value="romanLcParenR"/>
2738     <xsd:enumeration value="romanUcParenR"/>
2739     <xsd:enumeration value="romanLcPeriod"/>
2740     <xsd:enumeration value="romanUcPeriod"/>
2741     <xsd:enumeration value="circleNumDbPlain"/>
2742     <xsd:enumeration value="circleNumWdBlackPlain"/>
2743     <xsd:enumeration value="circleNumWdWhitePlain"/>
2744     <xsd:enumeration value="arabicDbPeriod"/>
2745     <xsd:enumeration value="arabicDbPlain"/>
2746     <xsd:enumeration value="ea1ChsPeriod"/>
2747     <xsd:enumeration value="ea1ChsPlain"/>
2748     <xsd:enumeration value="ea1ChtPeriod"/>
2749     <xsd:enumeration value="ea1ChtPlain"/>
2750     <xsd:enumeration value="ea1JpnChsDbPeriod"/>
2751     <xsd:enumeration value="ea1JpnKorPlain"/>
2752     <xsd:enumeration value="ea1JpnKorPeriod"/>
2753     <xsd:enumeration value="arabic1Minus"/>
2754     <xsd:enumeration value="arabic2Minus"/>
2755     <xsd:enumeration value="hebrew2Minus"/>
2756     <xsd:enumeration value="thaiAlphaPeriod"/>
2757     <xsd:enumeration value="thaiAlphaParenR"/>
2758     <xsd:enumeration value="thaiAlphaParenBoth"/>
2759     <xsd:enumeration value="thaiNumPeriod"/>
2760     <xsd:enumeration value="thaiNumParenR"/>
2761     <xsd:enumeration value="thaiNumParenBoth"/>
2762     <xsd:enumeration value="hindiAlphaPeriod"/>
2763     <xsd:enumeration value="hindiNumPeriod"/>
2764     <xsd:enumeration value="hindiNumParenR"/>
2765     <xsd:enumeration value="hindiAlpha1Period"/>
2766   </xsd:restriction>
2767 </xsd:simpleType>
2768 <xsd:complexType name="CT_TextBulletColorFollowText"/>
2769 <xsd:group name="EG_TextBulletColor">
2770   <xsd:choice>
2771     <xsd:element name="buClrTx" type="CT_TextBulletColorFollowText" minOccurs="1"
2772       maxOccurs="1"/>
2773     <xsd:element name="buClr" type="CT_Color" minOccurs="1" maxOccurs="1"/>
2774   </xsd:choice>
2775 </xsd:group>
2776 <xsd:simpleType name="ST_TextBulletSize">
2777   <xsd:union memberTypes="ST_TextBulletSizePercent ST_TextBulletSizeDecimal"/>
2778 </xsd:simpleType>
2779 <xsd:simpleType name="ST_TextBulletSizePercent">
2780   <xsd:restriction base="xsd:string">
2781     <xsd:pattern value="0*(([2-5-9])|([3-9][0-9])|([1-3][0-9][0-9])|400)%"/>

```

```

2782     </xsd:restriction>
2783 </xsd:simpleType>
2784 <xsd:simpleType name="ST_TextBulletSizeDecimal">
2785     <xsd:restriction base="ST_PercentageDecimal">
2786         <xsd:minInclusive value="25000"/>
2787         <xsd:maxInclusive value="400000"/>
2788     </xsd:restriction>
2789 </xsd:simpleType>
2790 <xsd:complexType name="CT_TextBulletSizeFollowText"/>
2791 <xsd:complexType name="CT_TextBulletSizePercent">
2792     <xsd:attribute name="val" type="ST_TextBulletSizePercent" use="required"/>
2793 </xsd:complexType>
2794 <xsd:complexType name="CT_TextBulletSizePoint">
2795     <xsd:attribute name="val" type="ST_TextFontSize" use="required"/>
2796 </xsd:complexType>
2797 <xsd:group name="EG_TextBulletSize">
2798     <xsd:choice>
2799         <xsd:element name="buSzTx" type="CT_TextBulletSizeFollowText"/>
2800         <xsd:element name="buSzPct" type="CT_TextBulletSizePercent"/>
2801         <xsd:element name="buSzPts" type="CT_TextBulletSizePoint"/>
2802     </xsd:choice>
2803 </xsd:group>
2804 <xsd:complexType name="CT_TextBulletTypefaceFollowText"/>
2805 <xsd:group name="EG_TextBulletTypeface">
2806     <xsd:choice>
2807         <xsd:element name="buFontTx" type="CT_TextBulletTypefaceFollowText"/>
2808         <xsd:element name="buFont" type="CT_TextFont"/>
2809     </xsd:choice>
2810 </xsd:group>
2811 <xsd:complexType name="CT_TextAutonumberBullet">
2812     <xsd:attribute name="type" type="ST_TextAutonumberScheme" use="required"/>
2813     <xsd:attribute name="startAt" type="ST_TextBulletStartAtNum" use="optional" default="1"/>
2814 </xsd:complexType>
2815 <xsd:complexType name="CT_TextCharBullet">
2816     <xsd:attribute name="char" type="xsd:string" use="required"/>
2817 </xsd:complexType>
2818 <xsd:complexType name="CT_TextBlipBullet">
2819     <xsd:sequence>
2820         <xsd:element name="blip" type="CT_Blip" minOccurs="1" maxOccurs="1"/>
2821     </xsd:sequence>
2822 </xsd:complexType>
2823 <xsd:complexType name="CT_TextNoBullet"/>
2824 <xsd:group name="EG_TextBullet">
2825     <xsd:choice>
2826         <xsd:element name="buNone" type="CT_TextNoBullet"/>
2827         <xsd:element name="buAutoNum" type="CT_TextAutonumberBullet"/>
2828         <xsd:element name="buChar" type="CT_TextCharBullet"/>
2829         <xsd:element name="buBlip" type="CT_TextBlipBullet"/>
2830     </xsd:choice>
2831 </xsd:group>
2832 <xsd:simpleType name="ST_TextPoint">
2833     <xsd:union memberTypes="ST_TextPointUnqualified s:ST_UniversalMeasure"/>
2834 </xsd:simpleType>

```

```

2835 <xsd:simpleType name="ST_TextPointUnqualified">
2836   <xsd:restriction base="xsd:int">
2837     <xsd:minInclusive value="-400000"/>
2838     <xsd:maxInclusive value="400000"/>
2839   </xsd:restriction>
2840 </xsd:simpleType>
2841 <xsd:simpleType name="ST_TextNonNegativePoint">
2842   <xsd:restriction base="xsd:int">
2843     <xsd:minInclusive value="0"/>
2844     <xsd:maxInclusive value="400000"/>
2845   </xsd:restriction>
2846 </xsd:simpleType>
2847 <xsd:simpleType name="ST_TextFontSize">
2848   <xsd:restriction base="xsd:int">
2849     <xsd:minInclusive value="100"/>
2850     <xsd:maxInclusive value="400000"/>
2851   </xsd:restriction>
2852 </xsd:simpleType>
2853 <xsd:simpleType name="ST_TextTypeface">
2854   <xsd:restriction base="xsd:string"/>
2855 </xsd:simpleType>
2856 <xsd:simpleType name="ST_PitchFamily">
2857   <xsd:restriction base="xsd:byte">
2858     <xsd:enumeration value="00"/>
2859     <xsd:enumeration value="01"/>
2860     <xsd:enumeration value="02"/>
2861     <xsd:enumeration value="16"/>
2862     <xsd:enumeration value="17"/>
2863     <xsd:enumeration value="18"/>
2864     <xsd:enumeration value="32"/>
2865     <xsd:enumeration value="33"/>
2866     <xsd:enumeration value="34"/>
2867     <xsd:enumeration value="48"/>
2868     <xsd:enumeration value="49"/>
2869     <xsd:enumeration value="50"/>
2870     <xsd:enumeration value="64"/>
2871     <xsd:enumeration value="65"/>
2872     <xsd:enumeration value="66"/>
2873     <xsd:enumeration value="80"/>
2874     <xsd:enumeration value="81"/>
2875     <xsd:enumeration value="82"/>
2876   </xsd:restriction>
2877 </xsd:simpleType>
2878 <xsd:complexType name="CT_TextFont">
2879   <xsd:attribute name="typeface" type="ST_TextTypeface" use="required"/>
2880   <xsd:attribute name="panose" type="s:ST_Panose" use="optional"/>
2881   <xsd:attribute name="pitchFamily" type="ST_PitchFamily" use="optional" default="0"/>
2882   <xsd:attribute name="charset" type="xsd:byte" use="optional" default="1"/>
2883 </xsd:complexType>
2884 <xsd:simpleType name="ST_TextUnderlineType">
2885   <xsd:restriction base="xsd:token">
2886     <xsd:enumeration value="none"/>
2887     <xsd:enumeration value="words"/>

```

```

2888         <xsd:enumeration value="sng"/>
2889         <xsd:enumeration value="dbl"/>
2890         <xsd:enumeration value="heavy"/>
2891         <xsd:enumeration value="dotted"/>
2892         <xsd:enumeration value="dottedHeavy"/>
2893         <xsd:enumeration value="dash"/>
2894         <xsd:enumeration value="dashHeavy"/>
2895         <xsd:enumeration value="dashLong"/>
2896         <xsd:enumeration value="dashLongHeavy"/>
2897         <xsd:enumeration value="dotDash"/>
2898         <xsd:enumeration value="dotDashHeavy"/>
2899         <xsd:enumeration value="dotDotDash"/>
2900         <xsd:enumeration value="dotDotDashHeavy"/>
2901         <xsd:enumeration value="wavy"/>
2902         <xsd:enumeration value="wavyHeavy"/>
2903         <xsd:enumeration value="wavyDbl"/>
2904     </xsd:restriction>
2905 </xsd:simpleType>
2906 <xsd:complexType name="CT_TextUnderlineLineFollowText"/>
2907 <xsd:complexType name="CT_TextUnderlineFillFollowText"/>
2908 <xsd:complexType name="CT_TextUnderlineFillGroupWrapper">
2909     <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
2910 </xsd:complexType>
2911 <xsd:group name="EG_TextUnderlineLine">
2912     <xsd:choice>
2913         <xsd:element name="uLnTx" type="CT_TextUnderlineLineFollowText"/>
2914         <xsd:element name="uLn" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2915     </xsd:choice>
2916 </xsd:group>
2917 <xsd:group name="EG_TextUnderlineFill">
2918     <xsd:choice>
2919         <xsd:element name="uFillTx" type="CT_TextUnderlineFillFollowText"/>
2920         <xsd:element name="uFill" type="CT_TextUnderlineFillGroupWrapper"/>
2921     </xsd:choice>
2922 </xsd:group>
2923 <xsd:simpleType name="ST_TextStrikeType">
2924     <xsd:restriction base="xsd:token">
2925         <xsd:enumeration value="noStrike"/>
2926         <xsd:enumeration value="sngStrike"/>
2927         <xsd:enumeration value="dblStrike"/>
2928     </xsd:restriction>
2929 </xsd:simpleType>
2930 <xsd:simpleType name="ST_TextCapsType">
2931     <xsd:restriction base="xsd:token">
2932         <xsd:enumeration value="none"/>
2933         <xsd:enumeration value="small"/>
2934         <xsd:enumeration value="all"/>
2935     </xsd:restriction>
2936 </xsd:simpleType>
2937 <xsd:complexType name="CT_TextCharacterProperties">
2938     <xsd:sequence>
2939         <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2940         <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>

```

```

2941     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2942     <xsd:element name="highlight" type="CT_Color" minOccurs="0" maxOccurs="1"/>
2943     <xsd:group ref="EG_TextUnderlineLine" minOccurs="0" maxOccurs="1"/>
2944     <xsd:group ref="EG_TextUnderlineFill" minOccurs="0" maxOccurs="1"/>
2945     <xsd:element name="latin" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2946     <xsd:element name="ea" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2947     <xsd:element name="cs" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2948     <xsd:element name="sym" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2949     <xsd:element name="hlinkClick" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
2950     <xsd:element name="hlinkMouseOver" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
2951     <xsd:element name="rtl" type="CT_Boolean" minOccurs="0"/>
2952     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2953 </xsd:sequence>
2954 <xsd:attribute name="kumimoji" type="xsd:boolean" use="optional"/>
2955 <xsd:attribute name="lang" type="s:ST_Lang" use="optional"/>
2956 <xsd:attribute name="altLang" type="s:ST_Lang" use="optional"/>
2957 <xsd:attribute name="sz" type="ST_TextFontSize" use="optional"/>
2958 <xsd:attribute name="b" type="xsd:boolean" use="optional"/>
2959 <xsd:attribute name="i" type="xsd:boolean" use="optional"/>
2960 <xsd:attribute name="u" type="ST_TextUnderlineType" use="optional"/>
2961 <xsd:attribute name="strike" type="ST_TextStrikeType" use="optional"/>
2962 <xsd:attribute name="kern" type="ST_TextNonNegativePoint" use="optional"/>
2963 <xsd:attribute name="cap" type="ST_TextCapsType" use="optional"/>
2964 <xsd:attribute name="spc" type="ST_TextPoint" use="optional"/>
2965 <xsd:attribute name="normalizeH" type="xsd:boolean" use="optional"/>
2966 <xsd:attribute name="baseline" type="ST_Percentage" use="optional"/>
2967 <xsd:attribute name="noProof" type="xsd:boolean" use="optional"/>
2968 <xsd:attribute name="dirty" type="xsd:boolean" use="optional" default="true"/>
2969 <xsd:attribute name="err" type="xsd:boolean" use="optional" default="false"/>
2970 <xsd:attribute name="smtClean" type="xsd:boolean" use="optional" default="true"/>
2971 <xsd:attribute name="smtId" type="xsd:unsignedInt" use="optional" default="0"/>
2972 <xsd:attribute name="bmk" type="xsd:string" use="optional"/>
2973 </xsd:complexType>
2974 <xsd:complexType name="CT_Boolean">
2975     <xsd:attribute name="val" type="s:ST_OnOff" default="0"/>
2976 </xsd:complexType>
2977 <xsd:simpleType name="ST_TextSpacingPoint">
2978     <xsd:restriction base="xsd:int">
2979         <xsd:minInclusive value="0"/>
2980         <xsd:maxInclusive value="158400"/>
2981     </xsd:restriction>
2982 </xsd:simpleType>
2983 <xsd:simpleType name="ST_TextSpacingPercentOrPercentString">
2984     <xsd:union memberTypes="ST_TextSpacingPercent s:ST_Percentage"/>
2985 </xsd:simpleType>
2986 <xsd:simpleType name="ST_TextSpacingPercent">
2987     <xsd:restriction base="ST_PercentageDecimal">
2988         <xsd:minInclusive value="0"/>
2989         <xsd:maxInclusive value="13200000"/>
2990     </xsd:restriction>
2991 </xsd:simpleType>
2992 <xsd:complexType name="CT_TextSpacingPercent">
2993     <xsd:attribute name="val" type="ST_TextSpacingPercentOrPercentString" use="required"/>

```

```

2994 </xsd:complexType>
2995 <xsd:complexType name="CT_TextSpacingPoint">
2996   <xsd:attribute name="val" type="ST_TextSpacingPoint" use="required"/>
2997 </xsd:complexType>
2998 <xsd:simpleType name="ST_TextMargin">
2999   <xsd:restriction base="ST_Coordinate32Unqualified">
3000     <xsd:minInclusive value="0"/>
3001     <xsd:maxInclusive value="51206400"/>
3002   </xsd:restriction>
3003 </xsd:simpleType>
3004 <xsd:simpleType name="ST_TextIndent">
3005   <xsd:restriction base="ST_Coordinate32Unqualified">
3006     <xsd:minInclusive value="-51206400"/>
3007     <xsd:maxInclusive value="51206400"/>
3008   </xsd:restriction>
3009 </xsd:simpleType>
3010 <xsd:simpleType name="ST_TextTabAlignType">
3011   <xsd:restriction base="xsd:token">
3012     <xsd:enumeration value="l"/>
3013     <xsd:enumeration value="ctr"/>
3014     <xsd:enumeration value="r"/>
3015     <xsd:enumeration value="dec"/>
3016   </xsd:restriction>
3017 </xsd:simpleType>
3018 <xsd:complexType name="CT_TextTabStop">
3019   <xsd:attribute name="pos" type="ST_Coordinate32" use="optional"/>
3020   <xsd:attribute name="align" type="ST_TextTabAlignType" use="optional"/>
3021 </xsd:complexType>
3022 <xsd:complexType name="CT_TextTabStopList">
3023   <xsd:sequence>
3024     <xsd:element name="tab" type="CT_TextTabStop" minOccurs="0" maxOccurs="32"/>
3025   </xsd:sequence>
3026 </xsd:complexType>
3027 <xsd:complexType name="CT_TextLineBreak">
3028   <xsd:sequence>
3029     <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3030   </xsd:sequence>
3031 </xsd:complexType>
3032 <xsd:complexType name="CT_TextSpacing">
3033   <xsd:choice>
3034     <xsd:element name="spcPct" type="CT_TextSpacingPercent"/>
3035     <xsd:element name="spcPts" type="CT_TextSpacingPoint"/>
3036   </xsd:choice>
3037 </xsd:complexType>
3038 <xsd:simpleType name="ST_TextAlignType">
3039   <xsd:restriction base="xsd:token">
3040     <xsd:enumeration value="l"/>
3041     <xsd:enumeration value="ctr"/>
3042     <xsd:enumeration value="r"/>
3043     <xsd:enumeration value="just"/>
3044     <xsd:enumeration value="justLow"/>
3045     <xsd:enumeration value="dist"/>
3046     <xsd:enumeration value="thaiDist"/>

```



```

3047     </xsd:restriction>
3048 </xsd:simpleType>
3049 <xsd:simpleType name="ST_TextFontAlignType">
3050     <xsd:restriction base="xsd:token">
3051         <xsd:enumeration value="auto"/>
3052         <xsd:enumeration value="t"/>
3053         <xsd:enumeration value="ctr"/>
3054         <xsd:enumeration value="base"/>
3055         <xsd:enumeration value="b"/>
3056     </xsd:restriction>
3057 </xsd:simpleType>
3058 <xsd:simpleType name="ST_TextIndentLevelType">
3059     <xsd:restriction base="xsd:int">
3060         <xsd:minInclusive value="0"/>
3061         <xsd:maxInclusive value="8"/>
3062     </xsd:restriction>
3063 </xsd:simpleType>
3064 <xsd:complexType name="CT_TextParagraphProperties">
3065     <xsd:sequence>
3066         <xsd:element name="lnSp" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3067         <xsd:element name="spcBef" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3068         <xsd:element name="spcAft" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3069         <xsd:group ref="EG_TextBulletColor" minOccurs="0" maxOccurs="1"/>
3070         <xsd:group ref="EG_TextBulletSize" minOccurs="0" maxOccurs="1"/>
3071         <xsd:group ref="EG_TextBulletTypeface" minOccurs="0" maxOccurs="1"/>
3072         <xsd:group ref="EG_TextBullet" minOccurs="0" maxOccurs="1"/>
3073         <xsd:element name="tabLst" type="CT_TextTabStopList" minOccurs="0" maxOccurs="1"/>
3074         <xsd:element name="defRPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3075         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
3076     </xsd:sequence>
3077     <xsd:attribute name="marL" type="ST_TextMargin" use="optional"/>
3078     <xsd:attribute name="marR" type="ST_TextMargin" use="optional"/>
3079     <xsd:attribute name="lvl" type="ST_TextIndentLevelType" use="optional"/>
3080     <xsd:attribute name="indent" type="ST_TextIndent" use="optional"/>
3081     <xsd:attribute name="algn" type="ST_TextAlignType" use="optional"/>
3082     <xsd:attribute name="defTabSz" type="ST_Coordinate32" use="optional"/>
3083     <xsd:attribute name="rtl" type="xsd:boolean" use="optional"/>
3084     <xsd:attribute name="eaLnBrk" type="xsd:boolean" use="optional"/>
3085     <xsd:attribute name="fontAlgn" type="ST_TextFontAlignType" use="optional"/>
3086     <xsd:attribute name="latinLnBrk" type="xsd:boolean" use="optional"/>
3087     <xsd:attribute name="hangingPunct" type="xsd:boolean" use="optional"/>
3088 </xsd:complexType>
3089 <xsd:complexType name="CT_TextField">
3090     <xsd:sequence>
3091         <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3092         <xsd:element name="pPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
3093         <xsd:element name="t" type="xsd:string" minOccurs="0" maxOccurs="1"/>
3094     </xsd:sequence>
3095     <xsd:attribute name="id" type="s:ST_Guid" use="required"/>
3096     <xsd:attribute name="type" type="xsd:string" use="optional"/>
3097 </xsd:complexType>
3098 <xsd:group name="EG_TextRun">
3099     <xsd:choice>

```

```

3100     <xsd:element name="r" type="CT_RegularTextRun"/>
3101     <xsd:element name="br" type="CT_TextLineBreak"/>
3102     <xsd:element name="fld" type="CT_TextField"/>
3103   </xsd:choice>
3104 </xsd:group>
3105 <xsd:complexType name="CT_RegularTextRun">
3106   <xsd:sequence>
3107     <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3108     <xsd:element name="t" type="xsd:string" minOccurs="1" maxOccurs="1"/>
3109   </xsd:sequence>
3110 </xsd:complexType>
3111 </xsd:schema>

```

A.4.2 DrawingML - Picture

This schema is available in the file dml-picture.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/drawingml/2006/picture"
3   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main" elementFormDefault="qualified"
4   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/picture">
5   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
6     main.xsd"/>
7   <xsd:complexType name="CT_PictureNonVisual">
8     <xsd:sequence>
9       <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
10      <xsd:element name="cNvPicPr" type="a:CT_NonVisualPictureProperties" minOccurs="1"
11        maxOccurs="1"/>
12    </xsd:sequence>
13  </xsd:complexType>
14  <xsd:complexType name="CT_Picture">
15    <xsd:sequence minOccurs="1" maxOccurs="1">
16      <xsd:element name="nvPicPr" type="CT_PictureNonVisual" minOccurs="1" maxOccurs="1"/>
17      <xsd:element name="blipFill" type="a:CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
18      <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
19    </xsd:sequence>
20  </xsd:complexType>
21  <xsd:element name="pic" type="CT_Picture"/>
22 </xsd:schema>

```

A.4.3 DrawingML - Locked Canvas

This schema is available in the file dml-lockedCanvas.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas"
3   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
4   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
5   elementFormDefault="qualified"
6   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas">
7   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
8     main.xsd"/>
9   <xsd:element name="lockedCanvas" type="a:CT_GvmlGroupShape"/>

```

```
</xsd:schema>
```

A.4.4 DrawingML - WordprocessingML Drawing

This schema is available in the file dml-wordprocessingDrawing.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
3   xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
4   xmlns:dpct="http://schemas.openxmlformats.org/drawingml/2006/picture"
5   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
6   xmlns="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
7   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
8   elementFormDefault="qualified">
9
10   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
11     main.xsd"/>
12   <xsd:import schemaLocation="wml.xsd"
13     namespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main"/>
14   <xsd:import
15     namespace="http://schemas.openxmlformats.org/drawingml/2006/picture"
16     schemaLocation="dml-picture.xsd"/>
17   <xsd:import
18     namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
19     schemaLocation="shared-relationshipReference.xsd"/>
20     <xsd:complexType name="CT_EffectExtent">
21       <xsd:attribute name="l" type="a:ST_Coordinate" use="required"/>
22       <xsd:attribute name="t" type="a:ST_Coordinate" use="required"/>
23       <xsd:attribute name="r" type="a:ST_Coordinate" use="required"/>
24       <xsd:attribute name="b" type="a:ST_Coordinate" use="required"/>
25     </xsd:complexType>
26     <xsd:simpleType name="ST_WrapDistance">
27       <xsd:restriction base="xsd:unsignedInt"/>
28     </xsd:simpleType>
29     <xsd:complexType name="CT_Inline">
30       <xsd:sequence>
31         <xsd:element name="extent" type="a:CT_PositiveSize2D"/>
32         <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
33         <xsd:element name="docPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
34         <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
35           minOccurs="0" maxOccurs="1"/>
36         <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
37       </xsd:sequence>
38       <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
39       <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
40       <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
41       <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
42     </xsd:complexType>
43     <xsd:simpleType name="ST_WrapText">
44       <xsd:restriction base="xsd:token">
45         <xsd:enumeration value="bothSides"/>
46         <xsd:enumeration value="left"/>
47         <xsd:enumeration value="right"/>

```

```

47     <xsd:enumeration value="largest"/>
48   </xsd:restriction>
49 </xsd:simpleType>
50 <xsd:complexType name="CT_WrapPath">
51   <xsd:sequence>
52     <xsd:element name="start" type="a:CT_Point2D" minOccurs="1" maxOccurs="1"/>
53     <xsd:element name="lineTo" type="a:CT_Point2D" minOccurs="2" maxOccurs="unbounded"/>
54   </xsd:sequence>
55   <xsd:attribute name="edited" type="xsd:boolean" use="optional"/>
56 </xsd:complexType>
57 <xsd:complexType name="CT_WrapNone"/>
58 <xsd:complexType name="CT_WrapSquare">
59   <xsd:sequence>
60     <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
61   </xsd:sequence>
62   <xsd:attribute name="wrapText" type="ST_WrapText" use="required"/>
63   <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
64   <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
65   <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
66   <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
67 </xsd:complexType>
68 <xsd:complexType name="CT_WrapTight">
69   <xsd:sequence>
70     <xsd:element name="wrapPolygon" type="CT_WrapPath" minOccurs="1" maxOccurs="1"/>
71   </xsd:sequence>
72   <xsd:attribute name="wrapText" type="ST_WrapText" use="required"/>
73   <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
74   <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
75 </xsd:complexType>
76 <xsd:complexType name="CT_WrapThrough">
77   <xsd:sequence>
78     <xsd:element name="wrapPolygon" type="CT_WrapPath" minOccurs="1" maxOccurs="1"/>
79   </xsd:sequence>
80   <xsd:attribute name="wrapText" type="ST_WrapText" use="required"/>
81   <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
82   <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
83 </xsd:complexType>
84 <xsd:complexType name="CT_WrapTopBottom">
85   <xsd:sequence>
86     <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
87   </xsd:sequence>
88   <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
89   <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
90 </xsd:complexType>
91 <xsd:group name="EG_WrapType">
92   <xsd:sequence>
93     <xsd:choice minOccurs="1" maxOccurs="1">
94       <xsd:element name="wrapNone" type="CT_WrapNone" minOccurs="1" maxOccurs="1"/>
95       <xsd:element name="wrapSquare" type="CT_WrapSquare" minOccurs="1" maxOccurs="1"/>
96       <xsd:element name="wrapTight" type="CT_WrapTight" minOccurs="1" maxOccurs="1"/>
97       <xsd:element name="wrapThrough" type="CT_WrapThrough" minOccurs="1" maxOccurs="1"/>
98       <xsd:element name="wrapTopAndBottom" type="CT_WrapTopBottom" minOccurs="1"
99         maxOccurs="1"/>

```

```

100         </xsd:choice>
101     </xsd:sequence>
102 </xsd:group>
103 <xsd:simpleType name="ST_PositionOffset">
104     <xsd:restriction base="xsd:int"/>
105 </xsd:simpleType>
106 <xsd:simpleType name="ST_AlignH">
107     <xsd:restriction base="xsd:token">
108         <xsd:enumeration value="left"/>
109         <xsd:enumeration value="right"/>
110         <xsd:enumeration value="center"/>
111         <xsd:enumeration value="inside"/>
112         <xsd:enumeration value="outside"/>
113     </xsd:restriction>
114 </xsd:simpleType>
115 <xsd:simpleType name="ST_RelFromH">
116     <xsd:restriction base="xsd:token">
117         <xsd:enumeration value="margin"/>
118         <xsd:enumeration value="page"/>
119         <xsd:enumeration value="column"/>
120         <xsd:enumeration value="character"/>
121         <xsd:enumeration value="leftMargin"/>
122         <xsd:enumeration value="rightMargin"/>
123         <xsd:enumeration value="insideMargin"/>
124         <xsd:enumeration value="outsideMargin"/>
125     </xsd:restriction>
126 </xsd:simpleType>
127 <xsd:complexType name="CT_PosH">
128     <xsd:sequence>
129         <xsd:choice minOccurs="1" maxOccurs="1">
130             <xsd:element name="align" type="ST_AlignH" minOccurs="1" maxOccurs="1"/>
131             <xsd:element name="posOffset" type="ST_PositionOffset" minOccurs="1" maxOccurs="1"/>
132         </xsd:choice>
133     </xsd:sequence>
134     <xsd:attribute name="relativeFrom" type="ST_RelFromH" use="required"/>
135 </xsd:complexType>
136 <xsd:simpleType name="ST_AlignV">
137     <xsd:restriction base="xsd:token">
138         <xsd:enumeration value="top"/>
139         <xsd:enumeration value="bottom"/>
140         <xsd:enumeration value="center"/>
141         <xsd:enumeration value="inside"/>
142         <xsd:enumeration value="outside"/>
143     </xsd:restriction>
144 </xsd:simpleType>
145 <xsd:simpleType name="ST_RelFromV">
146     <xsd:restriction base="xsd:token">
147         <xsd:enumeration value="margin"/>
148         <xsd:enumeration value="page"/>
149         <xsd:enumeration value="paragraph"/>
150         <xsd:enumeration value="line"/>
151         <xsd:enumeration value="topMargin"/>
152         <xsd:enumeration value="bottomMargin"/>

```

```

153     <xsd:enumeration value="insideMargin"/>
154     <xsd:enumeration value="outsideMargin"/>
155 </xsd:restriction>
156 </xsd:simpleType>
157 <xsd:complexType name="CT_PosV">
158     <xsd:sequence>
159         <xsd:choice minOccurs="1" maxOccurs="1">
160             <xsd:element name="align" type="ST_AlignV" minOccurs="1" maxOccurs="1"/>
161             <xsd:element name="posOffset" type="ST_PositionOffset" minOccurs="1" maxOccurs="1"/>
162         </xsd:choice>
163     </xsd:sequence>
164     <xsd:attribute name="relativeFrom" type="ST_RelFromV" use="required"/>
165 </xsd:complexType>
166 <xsd:complexType name="CT_Anchor">
167     <xsd:sequence>
168         <xsd:element name="simplePos" type="a:CT_Point2D"/>
169         <xsd:element name="positionH" type="CT_PosH"/>
170         <xsd:element name="positionV" type="CT_PosV"/>
171         <xsd:element name="extent" type="a:CT_PositiveSize2D"/>
172         <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
173         <xsd:group ref="EG_WrapType"/>
174         <xsd:element name="docPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
175         <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
176             minOccurs="0" maxOccurs="1"/>
177         <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
178     </xsd:sequence>
179     <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
180     <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
181     <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
182     <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
183     <xsd:attribute name="simplePos" type="xsd:boolean"/>
184     <xsd:attribute name="relativeHeight" type="xsd:unsignedInt" use="required"/>
185     <xsd:attribute name="behindDoc" type="xsd:boolean" use="required"/>
186     <xsd:attribute name="locked" type="xsd:boolean" use="required"/>
187     <xsd:attribute name="layoutInCell" type="xsd:boolean" use="required"/>
188     <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
189     <xsd:attribute name="allowOverlap" type="xsd:boolean" use="required"/>
190 </xsd:complexType>
191 <xsd:complexType name="CT_TxbxContent">
192     <xsd:group ref="w:EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
193 </xsd:complexType>
194 <xsd:complexType name="CT_TextboxInfo">
195     <xsd:sequence>
196         <xsd:element name="txbxContent" type="CT_TxbxContent" minOccurs="1" maxOccurs="1"/>
197         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
198             maxOccurs="1"/>
199     </xsd:sequence>
200     <xsd:attribute name="id" type="xsd:unsignedShort" use="optional" default="0"/>
201 </xsd:complexType>
202 <xsd:complexType name="CT_LinkedTextboxInformation">
203     <xsd:sequence>
204         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
205             maxOccurs="1"/>

```

```

206     </xsd:sequence>
207     <xsd:attribute name="id" type="xsd:unsignedShort" use="required"/>
208     <xsd:attribute name="seq" type="xsd:unsignedShort" use="required"/>
209 </xsd:complexType>
210     <xsd:complexType name="CT_WordprocessingShape">
211     <xsd:sequence minOccurs="1" maxOccurs="1">
212         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="0" maxOccurs="1"/>
213         <xsd:choice minOccurs="1" maxOccurs="1">
214             <xsd:element name="cNvSpPr" type="a:CT_NonVisualDrawingShapeProps" minOccurs="1"
215                 maxOccurs="1"/>
216             <xsd:element name="cNvCnPr" type="a:CT_NonVisualConnectorProperties" minOccurs="1"
217                 maxOccurs="1"/>
218         </xsd:choice>
219         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
220         <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
221         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
222             maxOccurs="1"/>
223         <xsd:choice minOccurs="0" maxOccurs="1">
224             <xsd:element name="txbx" type="CT_TextboxInfo" minOccurs="1" maxOccurs="1"/>
225             <xsd:element name="linkedTxbx" type="CT_LinkedTextboxInformation" minOccurs="1"
226                 maxOccurs="1"/>
227         </xsd:choice>
228         <xsd:element name="bodyPr" type="a:CT_TextBodyProperties" minOccurs="1" maxOccurs="1"/>
229     </xsd:sequence>
230     <xsd:attribute name="normalEastAsianFlow" type="xsd:boolean" use="optional" default="false"/>
231 </xsd:complexType>
232 <xsd:complexType name="CT_GraphicFrame">
233     <xsd:sequence>
234         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
235         <xsd:element name="cNvFrPr" type="a:CT_NonVisualGraphicFrameProperties" minOccurs="1"
236             maxOccurs="1"/>
237         <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="1" maxOccurs="1"/>
238         <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
239         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
240             maxOccurs="1"/>
241     </xsd:sequence>
242 </xsd:complexType>
243 <xsd:complexType name="CT_WordprocessingContentPartNonVisual">
244     <xsd:sequence>
245         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="0" maxOccurs="1"/>
246         <xsd:element name="cNvContentPartPr" type="a:CT_NonVisualContentPartProperties"
247             minOccurs="0" maxOccurs="1"/>
248     </xsd:sequence>
249 </xsd:complexType>
250 <xsd:complexType name="CT_WordprocessingContentPart">
251     <xsd:sequence>
252         <xsd:element name="nvContentPartPr" type="CT_WordprocessingContentPartNonVisual"
253             minOccurs="0" maxOccurs="1"/>
254         <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="0" maxOccurs="1"/>
255         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
256             maxOccurs="1"/>
257     </xsd:sequence>
258     <xsd:attribute name="bwMode" type="a:ST_BlackWhiteMode" use="optional"/>

```

```

259     <xsd:attribute ref="r:id" use="required"/>
260 </xsd:complexType>
261 <xsd:complexType name="CT_WordprocessingGroup">
262     <xsd:sequence minOccurs="1" maxOccurs="1">
263         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="0" maxOccurs="1"/>
264         <xsd:element name="cNvGrpSpPr" type="a:CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
265             maxOccurs="1"/>
266         <xsd:element name="grpSpPr" type="a:CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
267         <xsd:choice minOccurs="0" maxOccurs="unbounded">
268             <xsd:element ref="wsp"/>
269             <xsd:element name="grpSp" type="CT_WordprocessingGroup"/>
270             <xsd:element name="graphicFrame" type="CT_GraphicFrame"/>
271             <xsd:element ref="dpct:pic"/>
272             <xsd:element name="contentPart" type="CT_WordprocessingContentPart"/>
273         </xsd:choice>
274         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
275             maxOccurs="1"/>
276     </xsd:sequence>
277 </xsd:complexType>
278 <xsd:complexType name="CT_WordprocessingCanvas">
279     <xsd:sequence minOccurs="1" maxOccurs="1">
280         <xsd:element name="bg" type="a:CT_BackgroundFormatting" minOccurs="0" maxOccurs="1"/>
281         <xsd:element name="whole" type="a:CT_WholeE2oFormatting" minOccurs="0" maxOccurs="1"/>
282         <xsd:choice minOccurs="0" maxOccurs="unbounded">
283             <xsd:element ref="wsp"/>
284             <xsd:element ref="dpct:pic"/>
285             <xsd:element name="contentPart" type="CT_WordprocessingContentPart"/>
286             <xsd:element ref="wgp"/>
287             <xsd:element name="graphicFrame" type="CT_GraphicFrame"/>
288         </xsd:choice>
289         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
290             maxOccurs="1"/>
291     </xsd:sequence>
292 </xsd:complexType>
293 <xsd:element name="wpc" type="CT_WordprocessingCanvas"/>
294 <xsd:element name="wgp" type="CT_WordprocessingGroup"/>
295 <xsd:element name="wsp" type="CT_WordprocessingShape"/>
296 <xsd:element name="inline" type="CT Inline"/>
297 <xsd:element name="anchor" type="CT Anchor"/>
298 </xsd:schema>

```

A.4.5 DrawingML - SpreadsheetML Drawing

This schema is available in the file dml-spreadsheetDrawing.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
3   xmlns="http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
4   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
5   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
6   elementFormDefault="qualified">
7   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
8     main.xsd"/>

```



```

9      <xsd:import schemaLocation="shared-relationshipReference.xsd"
10      namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"/>
11      <xsd:element name="from" type="CT_Marker"/>
12      <xsd:element name="to" type="CT_Marker"/>
13      <xsd:complexType name="CT_AnchorClientData">
14          <xsd:attribute name="fLocksWithSheet" type="xsd:boolean" use="optional" default="true"/>
15          <xsd:attribute name="fPrintsWithSheet" type="xsd:boolean" use="optional" default="true"/>
16      </xsd:complexType>
17      <xsd:complexType name="CT_ShapeNonVisual">
18          <xsd:sequence>
19              <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
20              <xsd:element name="cNvSpPr" type="a:CT_NonVisualDrawingShapeProps" minOccurs="1"
21              maxOccurs="1"/>
22          </xsd:sequence>
23      </xsd:complexType>
24      <xsd:complexType name="CT_Shape">
25          <xsd:sequence>
26              <xsd:element name="nvSpPr" type="CT_ShapeNonVisual" minOccurs="1" maxOccurs="1"/>
27              <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
28              <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
29              <xsd:element name="txBody" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
30          </xsd:sequence>
31          <xsd:attribute name="macro" type="xsd:string" use="optional"/>
32          <xsd:attribute name="textlink" type="xsd:string" use="optional"/>
33          <xsd:attribute name="fLocksText" type="xsd:boolean" use="optional" default="true"/>
34          <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
35      </xsd:complexType>
36      <xsd:complexType name="CT_ConnectorNonVisual">
37          <xsd:sequence>
38              <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
39              <xsd:element name="cNvCxnSpPr" type="a:CT_NonVisualConnectorProperties" minOccurs="1"
40              maxOccurs="1"/>
41          </xsd:sequence>
42      </xsd:complexType>
43      <xsd:complexType name="CT_Connector">
44          <xsd:sequence>
45              <xsd:element name="nvCxnSpPr" type="CT_ConnectorNonVisual" minOccurs="1" maxOccurs="1"/>
46              <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
47              <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
48          </xsd:sequence>
49          <xsd:attribute name="macro" type="xsd:string" use="optional"/>
50          <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
51      </xsd:complexType>
52      <xsd:complexType name="CT_PictureNonVisual">
53          <xsd:sequence>
54              <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
55              <xsd:element name="cNvPicPr" type="a:CT_NonVisualPictureProperties" minOccurs="1"
56              maxOccurs="1"/>
57          </xsd:sequence>
58      </xsd:complexType>
59      <xsd:complexType name="CT_Picture">
60          <xsd:sequence>
61              <xsd:element name="nvPicPr" type="CT_PictureNonVisual" minOccurs="1" maxOccurs="1"/>

```

```

62      <xsd:element name="blipFill" type="a:CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
63      <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
64      <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
65    </xsd:sequence>
66    <xsd:attribute name="macro" type="xsd:string" use="optional" default=""/>
67    <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
68  </xsd:complexType>
69  <xsd:complexType name="CT_GraphicalObjectFrameNonVisual">
70    <xsd:sequence>
71      <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
72      <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
73        minOccurs="1" maxOccurs="1"/>
74    </xsd:sequence>
75  </xsd:complexType>
76  <xsd:complexType name="CT_GraphicalObjectFrame">
77    <xsd:sequence>
78      <xsd:element name="nvGraphicFramePr" type="CT_GraphicalObjectFrameNonVisual" minOccurs="1"
79        maxOccurs="1"/>
80      <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="1" maxOccurs="1"/>
81      <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
82    </xsd:sequence>
83    <xsd:attribute name="macro" type="xsd:string" use="optional"/>
84    <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
85  </xsd:complexType>
86  <xsd:complexType name="CT_GroupShapeNonVisual">
87    <xsd:sequence>
88      <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
89      <xsd:element name="cNvGrpSpPr" type="a:CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
90        maxOccurs="1"/>
91    </xsd:sequence>
92  </xsd:complexType>
93  <xsd:complexType name="CT_GroupShape">
94    <xsd:sequence>
95      <xsd:element name="nvGrpSpPr" type="CT_GroupShapeNonVisual" minOccurs="1" maxOccurs="1"/>
96      <xsd:element name="grpSpPr" type="a:CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
97      <xsd:choice minOccurs="0" maxOccurs="unbounded">
98        <xsd:element name="sp" type="CT_Shape"/>
99        <xsd:element name="grpSp" type="CT_GroupShape"/>
100       <xsd:element name="graphicFrame" type="CT_GraphicalObjectFrame"/>
101       <xsd:element name="cxnSp" type="CT_Connector"/>
102       <xsd:element name="pic" type="CT_Picture"/>
103     </xsd:choice>
104    </xsd:sequence>
105  </xsd:complexType>
106  <xsd:group name="EG_ObjectChoices">
107    <xsd:sequence>
108      <xsd:choice minOccurs="1" maxOccurs="1">
109        <xsd:element name="sp" type="CT_Shape"/>
110        <xsd:element name="grpSp" type="CT_GroupShape"/>
111        <xsd:element name="graphicFrame" type="CT_GraphicalObjectFrame"/>
112        <xsd:element name="cxnSp" type="CT_Connector"/>
113        <xsd:element name="pic" type="CT_Picture"/>
114        <xsd:element name="contentPart" type="CT_Rel"/>

```

```

115         </xsd:choice>
116     </xsd:sequence>
117 </xsd:group>
118 <xsd:complexType name="CT_Rel">
119     <xsd:attribute ref="r:id" use="required"/>
120 </xsd:complexType>
121 <xsd:simpleType name="ST_ColID">
122     <xsd:restriction base="xsd:int">
123         <xsd:minInclusive value="0"/>
124     </xsd:restriction>
125 </xsd:simpleType>
126 <xsd:simpleType name="ST_RowID">
127     <xsd:restriction base="xsd:int">
128         <xsd:minInclusive value="0"/>
129     </xsd:restriction>
130 </xsd:simpleType>
131 <xsd:complexType name="CT_Marker">
132     <xsd:sequence>
133         <xsd:element name="col" type="ST_ColID"/>
134         <xsd:element name="colOff" type="a:ST_Coordinate"/>
135         <xsd:element name="row" type="ST_RowID"/>
136         <xsd:element name="rowOff" type="a:ST_Coordinate"/>
137     </xsd:sequence>
138 </xsd:complexType>
139 <xsd:simpleType name="ST_EditAs">
140     <xsd:restriction base="xsd:token">
141         <xsd:enumeration value="twoCell"/>
142         <xsd:enumeration value="oneCell"/>
143         <xsd:enumeration value="absolute"/>
144     </xsd:restriction>
145 </xsd:simpleType>
146 <xsd:complexType name="CT_TwoCellAnchor">
147     <xsd:sequence>
148         <xsd:element name="from" type="CT_Marker"/>
149         <xsd:element name="to" type="CT_Marker"/>
150         <xsd:group ref="EG_ObjectChoices"/>
151         <xsd:element name="clientData" type="CT_AnchorClientData" minOccurs="1" maxOccurs="1"/>
152     </xsd:sequence>
153     <xsd:attribute name="editAs" type="ST_EditAs" use="optional" default="twoCell"/>
154 </xsd:complexType>
155 <xsd:complexType name="CT_OneCellAnchor">
156     <xsd:sequence>
157         <xsd:element name="from" type="CT_Marker"/>
158         <xsd:element name="ext" type="a:CT_PositiveSize2D"/>
159         <xsd:group ref="EG_ObjectChoices"/>
160         <xsd:element name="clientData" type="CT_AnchorClientData" minOccurs="1" maxOccurs="1"/>
161     </xsd:sequence>
162 </xsd:complexType>
163 <xsd:complexType name="CT_AbsoluteAnchor">
164     <xsd:sequence>
165         <xsd:element name="pos" type="a:CT_Point2D"/>
166         <xsd:element name="ext" type="a:CT_PositiveSize2D"/>
167         <xsd:group ref="EG_ObjectChoices"/>

```

```

168     <xsd:element name="clientData" type="CT_AnchorClientData" minOccurs="1" maxOccurs="1"/>
169   </xsd:sequence>
170 </xsd:complexType>
171 <xsd:group name="EG_Anchor">
172   <xsd:choice>
173     <xsd:element name="twoCellAnchor" type="CT_TwoCellAnchor"/>
174     <xsd:element name="oneCellAnchor" type="CT_OneCellAnchor"/>
175     <xsd:element name="absoluteAnchor" type="CT_AbsoluteAnchor"/>
176   </xsd:choice>
177 </xsd:group>
178 <xsd:complexType name="CT_Drawing">
179   <xsd:sequence>
180     <xsd:group ref="EG_Anchor" minOccurs="0" maxOccurs="unbounded"/>
181   </xsd:sequence>
182 </xsd:complexType>
183 <xsd:element name="wsDr" type="CT_Drawing"/>
184 </xsd:schema>

```

A.5 DrawingML - Components

A.5.1 DrawingML - Charts

This schema is available in the file dml-chart.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
3   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4   xmlns="http://schemas.openxmlformats.org/drawingml/2006/chart"
5   xmlns:cd="http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
6   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/chart"
8   elementFormDefault="qualified" attributeFormDefault="unqualified" blockDefault="#all">
9   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
10     schemaLocation="shared-relationshipReference.xsd"/>
11   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
12     main.xsd"/>
13   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
14     schemaLocation="dml-chartDrawing.xsd"/>
15   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
16     schemaLocation="shared-commonSimpleTypes.xsd"/>
17   <xsd:complexType name="CT_Boolean">
18     <xsd:attribute name="val" type="xsd:boolean" use="optional" default="true"/>
19   </xsd:complexType>
20   <xsd:complexType name="CT_Double">
21     <xsd:attribute name="val" type="xsd:double" use="required"/>
22   </xsd:complexType>
23   <xsd:complexType name="CT_UnsignedInt">
24     <xsd:attribute name="val" type="xsd:unsignedInt" use="required"/>
25   </xsd:complexType>
26   <xsd:complexType name="CT_RelId">
27     <xsd:attribute ref="r:id" use="required"/>
28   </xsd:complexType>
29   <xsd:complexType name="CT_Extension">

```

```

30     <xsd:sequence>
31         <xsd:any processContents="lax"/>
32     </xsd:sequence>
33     <xsd:attribute name="uri" type="xsd:token"/>
34 </xsd:complexType>
35 <xsd:complexType name="CT_ExtensionList">
36     <xsd:sequence>
37         <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
38     </xsd:sequence>
39 </xsd:complexType>
40 <xsd:complexType name="CT_NumVal">
41     <xsd:sequence>
42         <xsd:element name="v" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
43     </xsd:sequence>
44     <xsd:attribute name="idx" type="xsd:unsignedInt" use="required"/>
45     <xsd:attribute name="formatCode" type="s:ST_Xstring" use="optional"/>
46 </xsd:complexType>
47 <xsd:complexType name="CT_NumData">
48     <xsd:sequence>
49         <xsd:element name="formatCode" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
50         <xsd:element name="ptCount" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
51         <xsd:element name="pt" type="CT_NumVal" minOccurs="0" maxOccurs="unbounded"/>
52         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
53     </xsd:sequence>
54 </xsd:complexType>
55 <xsd:complexType name="CT_NumRef">
56     <xsd:sequence>
57         <xsd:element name="f" type="xsd:string" minOccurs="1" maxOccurs="1"/>
58         <xsd:element name="numCache" type="CT_NumData" minOccurs="0" maxOccurs="1"/>
59         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
60     </xsd:sequence>
61 </xsd:complexType>
62 <xsd:complexType name="CT_NumDataSource">
63     <xsd:sequence>
64         <xsd:choice minOccurs="1" maxOccurs="1">
65             <xsd:element name="numRef" type="CT_NumRef" minOccurs="1" maxOccurs="1"/>
66             <xsd:element name="numLit" type="CT_NumData" minOccurs="1" maxOccurs="1"/>
67         </xsd:choice>
68     </xsd:sequence>
69 </xsd:complexType>
70 <xsd:complexType name="CT_StrVal">
71     <xsd:sequence>
72         <xsd:element name="v" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
73     </xsd:sequence>
74     <xsd:attribute name="idx" type="xsd:unsignedInt" use="required"/>
75 </xsd:complexType>
76 <xsd:complexType name="CT_StrData">
77     <xsd:sequence>
78         <xsd:element name="ptCount" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
79         <xsd:element name="pt" type="CT_StrVal" minOccurs="0" maxOccurs="unbounded"/>
80         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
81     </xsd:sequence>
82 </xsd:complexType>

```

```

83 <xsd:complexType name="CT_StrRef">
84   <xsd:sequence>
85     <xsd:element name="f" type="xsd:string" minOccurs="1" maxOccurs="1"/>
86     <xsd:element name="strCache" type="CT_StrData" minOccurs="0" maxOccurs="1"/>
87     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
88   </xsd:sequence>
89 </xsd:complexType>
90 <xsd:complexType name="CT_Tx">
91   <xsd:sequence>
92     <xsd:choice minOccurs="1" maxOccurs="1">
93       <xsd:element name="strRef" type="CT_StrRef" minOccurs="1" maxOccurs="1"/>
94       <xsd:element name="rich" type="a:CT_TextBody" minOccurs="1" maxOccurs="1"/>
95     </xsd:choice>
96   </xsd:sequence>
97 </xsd:complexType>
98 <xsd:complexType name="CT_TextLanguageID">
99   <xsd:attribute name="val" type="s:ST_Lang" use="required"/>
100 </xsd:complexType>
101 <xsd:complexType name="CT_Lvl">
102   <xsd:sequence>
103     <xsd:element name="pt" type="CT_StrVal" minOccurs="0" maxOccurs="unbounded"/>
104   </xsd:sequence>
105 </xsd:complexType>
106 <xsd:complexType name="CT_MultiLvlStrData">
107   <xsd:sequence>
108     <xsd:element name="ptCount" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
109     <xsd:element name="lvl" type="CT_Lvl" minOccurs="0" maxOccurs="unbounded"/>
110     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
111   </xsd:sequence>
112 </xsd:complexType>
113 <xsd:complexType name="CT_MultiLvlStrRef">
114   <xsd:sequence>
115     <xsd:element name="f" type="xsd:string" minOccurs="1" maxOccurs="1"/>
116     <xsd:element name="multiLvlStrCache" type="CT_MultiLvlStrData" minOccurs="0"
117       maxOccurs="1"/>
118     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
119   </xsd:sequence>
120 </xsd:complexType>
121 <xsd:complexType name="CT_AxDataSource">
122   <xsd:sequence>
123     <xsd:choice minOccurs="1" maxOccurs="1">
124       <xsd:element name="multiLvlStrRef" type="CT_MultiLvlStrRef" minOccurs="1"
125         maxOccurs="1"/>
126       <xsd:element name="numRef" type="CT_NumRef" minOccurs="1" maxOccurs="1"/>
127       <xsd:element name="numLit" type="CT_NumData" minOccurs="1" maxOccurs="1"/>
128       <xsd:element name="strRef" type="CT_StrRef" minOccurs="1" maxOccurs="1"/>
129       <xsd:element name="strLit" type="CT_StrData" minOccurs="1" maxOccurs="1"/>
130     </xsd:choice>
131   </xsd:sequence>
132 </xsd:complexType>
133 <xsd:complexType name="CT_SerTx">
134   <xsd:sequence>
135     <xsd:choice minOccurs="1" maxOccurs="1">

```

```

136         <xsd:element name="strRef" type="CT_StrRef" minOccurs="1" maxOccurs="1"/>
137         <xsd:element name="v" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
138     </xsd:choice>
139 </xsd:sequence>
140 </xsd:complexType>
141 <xsd:simpleType name="ST_LayoutTarget">
142     <xsd:restriction base="xsd:string">
143         <xsd:enumeration value="inner"/>
144         <xsd:enumeration value="outer"/>
145     </xsd:restriction>
146 </xsd:simpleType>
147 <xsd:complexType name="CT_LayoutTarget">
148     <xsd:attribute name="val" type="ST_LayoutTarget" default="outer"/>
149 </xsd:complexType>
150 <xsd:simpleType name="ST_LayoutMode">
151     <xsd:restriction base="xsd:string">
152         <xsd:enumeration value="edge"/>
153         <xsd:enumeration value="factor"/>
154     </xsd:restriction>
155 </xsd:simpleType>
156 <xsd:complexType name="CT_LayoutMode">
157     <xsd:attribute name="val" type="ST_LayoutMode" default="factor"/>
158 </xsd:complexType>
159 <xsd:complexType name="CT_ManualLayout">
160     <xsd:sequence>
161         <xsd:element name="layoutTarget" type="CT_LayoutTarget" minOccurs="0" maxOccurs="1"/>
162         <xsd:element name="xMode" type="CT_LayoutMode" minOccurs="0" maxOccurs="1"/>
163         <xsd:element name="yMode" type="CT_LayoutMode" minOccurs="0" maxOccurs="1"/>
164         <xsd:element name="wMode" type="CT_LayoutMode" minOccurs="0" maxOccurs="1"/>
165         <xsd:element name="hMode" type="CT_LayoutMode" minOccurs="0" maxOccurs="1"/>
166         <xsd:element name="x" type="CT_Double" minOccurs="0" maxOccurs="1"/>
167         <xsd:element name="y" type="CT_Double" minOccurs="0" maxOccurs="1"/>
168         <xsd:element name="w" type="CT_Double" minOccurs="0" maxOccurs="1"/>
169         <xsd:element name="h" type="CT_Double" minOccurs="0" maxOccurs="1"/>
170         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
171     </xsd:sequence>
172 </xsd:complexType>
173 <xsd:complexType name="CT_Layout">
174     <xsd:sequence>
175         <xsd:element name="manualLayout" type="CT_ManualLayout" minOccurs="0" maxOccurs="1"/>
176         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
177     </xsd:sequence>
178 </xsd:complexType>
179 <xsd:complexType name="CT_Title">
180     <xsd:sequence>
181         <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
182         <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
183         <xsd:element name="overlay" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
184         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
185         <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
186         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
187     </xsd:sequence>
188 </xsd:complexType>

```

```

189 <xsd:simpleType name="ST_RotX">
190   <xsd:restriction base="xsd:byte">
191     <xsd:minInclusive value="-90"/>
192     <xsd:maxInclusive value="90"/>
193   </xsd:restriction>
194 </xsd:simpleType>
195 <xsd:complexType name="CT_RotX">
196   <xsd:attribute name="val" type="ST_RotX" default="0"/>
197 </xsd:complexType>
198 <xsd:simpleType name="ST_HPercent">
199   <xsd:restriction base="xsd:unsignedShort">
200     <xsd:minInclusive value="5"/>
201     <xsd:maxInclusive value="500"/>
202   </xsd:restriction>
203   <xsd:union memberTypes="ST_HPercentWithSymbol ST_HPercentUShort"/>
204 </xsd:simpleType>
205 <xsd:simpleType name="ST_HPercentWithSymbol">
206   <xsd:restriction base="xsd:string">
207     <xsd:pattern value="0*(([5-9])|([1-9][0-9])|([1-4][0-9][0-9])|500)%"/>
208   </xsd:restriction>
209 </xsd:simpleType>
210 <xsd:simpleType name="ST_HPercentUShort">
211   <xsd:restriction base="xsd:unsignedShort">
212     <xsd:minInclusive value="5"/>
213     <xsd:maxInclusive value="500"/>
214   </xsd:restriction>
215 </xsd:simpleType>
216 <xsd:complexType name="CT_HPercent">
217   <xsd:attribute name="val" type="ST_HPercent" default="100%"/>
218 </xsd:complexType>
219 <xsd:simpleType name="ST_RotY">
220   <xsd:restriction base="xsd:unsignedShort">
221     <xsd:minInclusive value="0"/>
222     <xsd:maxInclusive value="360"/>
223   </xsd:restriction>
224 </xsd:simpleType>
225 <xsd:complexType name="CT_RotY">
226   <xsd:attribute name="val" type="ST_RotY" default="0"/>
227 </xsd:complexType>
228 <xsd:simpleType name="ST_DepthPercent">
229   <xsd:union memberTypes="ST_DepthPercentWithSymbol ST_DepthPercentUShort"/>
230 </xsd:simpleType>
231 <xsd:simpleType name="ST_DepthPercentWithSymbol">
232   <xsd:restriction base="xsd:string">
233     <xsd:pattern value="0*([2-9][0-9])|([1-9][0-9][0-9])|(1[0-9][0-9][0-9])|2000)%"/>
234   </xsd:restriction>
235 </xsd:simpleType>
236 <xsd:simpleType name="ST_DepthPercentUShort">
237   <xsd:restriction base="xsd:unsignedShort">
238     <xsd:minInclusive value="20"/>
239     <xsd:maxInclusive value="2000"/>
240   </xsd:restriction>
241 </xsd:simpleType>

```



```

242 <xsd:complexType name="CT_DepthPercent">
243   <xsd:attribute name="val" type="ST_DepthPercent" default="100%"/>
244 </xsd:complexType>
245 <xsd:simpleType name="ST_Perspective">
246   <xsd:restriction base="xsd:unsignedByte">
247     <xsd:minInclusive value="0"/>
248     <xsd:maxInclusive value="240"/>
249   </xsd:restriction>
250 </xsd:simpleType>
251 <xsd:complexType name="CT_Perspective">
252   <xsd:attribute name="val" type="ST_Perspective" default="30"/>
253 </xsd:complexType>
254 <xsd:complexType name="CT_View3D">
255   <xsd:sequence>
256     <xsd:element name="rotX" type="CT_RotX" minOccurs="0" maxOccurs="1"/>
257     <xsd:element name="hPercent" type="CT_HPercent" minOccurs="0" maxOccurs="1"/>
258     <xsd:element name="rotY" type="CT_RotY" minOccurs="0" maxOccurs="1"/>
259     <xsd:element name="depthPercent" type="CT_DepthPercent" minOccurs="0" maxOccurs="1"/>
260     <xsd:element name="rAngAx" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
261     <xsd:element name="perspective" type="CT_Perspective" minOccurs="0" maxOccurs="1"/>
262     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
263   </xsd:sequence>
264 </xsd:complexType>
265 <xsd:complexType name="CT_Surface">
266   <xsd:sequence>
267     <xsd:element name="thickness" type="CT_Thickness" minOccurs="0" maxOccurs="1"/>
268     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
269     <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
270     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
271   </xsd:sequence>
272 </xsd:complexType>
273 <xsd:simpleType name="ST_Thickness">
274   <xsd:union memberTypes="ST_ThicknessPercent xsd:unsignedInt"/>
275 </xsd:simpleType>
276 <xsd:simpleType name="ST_ThicknessPercent">
277   <xsd:restriction base="xsd:string">
278     <xsd:pattern value="([0-9]+)%"/>
279   </xsd:restriction>
280 </xsd:simpleType>
281 <xsd:complexType name="CT_Thickness">
282   <xsd:attribute name="val" type="ST_Thickness" use="required"/>
283 </xsd:complexType>
284 <xsd:complexType name="CT_DTable">
285   <xsd:sequence>
286     <xsd:element name="showHorzBorder" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
287     <xsd:element name="showVertBorder" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
288     <xsd:element name="showOutline" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
289     <xsd:element name="showKeys" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
290     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
291     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
292     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
293   </xsd:sequence>
294 </xsd:complexType>

```

```

295 <xsd:simpleType name="ST_GapAmount">
296   <xsd:union memberTypes="ST_GapAmountPercent ST_GapAmountUShort"/>
297 </xsd:simpleType>
298 <xsd:simpleType name="ST_GapAmountPercent">
299   <xsd:restriction base="xsd:string">
300     <xsd:pattern value="0*(([0-9])|([1-9][0-9])|([1-4][0-9][0-9])|500)%"/>
301   </xsd:restriction>
302 </xsd:simpleType>
303 <xsd:simpleType name="ST_GapAmountUShort">
304   <xsd:restriction base="xsd:unsignedShort">
305     <xsd:minInclusive value="0"/>
306     <xsd:maxInclusive value="500"/>
307   </xsd:restriction>
308 </xsd:simpleType>
309 <xsd:complexType name="CT_GapAmount">
310   <xsd:attribute name="val" type="ST_GapAmount" default="150%"/>
311 </xsd:complexType>
312 <xsd:simpleType name="ST_Overlap">
313   <xsd:union memberTypes="ST_OverlapPercent ST_OverlapByte"/>
314 </xsd:simpleType>
315 <xsd:simpleType name="ST_OverlapPercent">
316   <xsd:restriction base="xsd:string">
317     <xsd:pattern value="(-?0*(([0-9])|([1-9][0-9])|100))%"/>
318   </xsd:restriction>
319 </xsd:simpleType>
320 <xsd:simpleType name="ST_OverlapByte">
321   <xsd:restriction base="xsd:byte">
322     <xsd:minInclusive value="-100"/>
323     <xsd:maxInclusive value="100"/>
324   </xsd:restriction>
325 </xsd:simpleType>
326 <xsd:complexType name="CT_Overlap">
327   <xsd:attribute name="val" type="ST_Overlap" default="0%"/>
328 </xsd:complexType>
329 <xsd:simpleType name="ST_BubbleScale">
330   <xsd:union memberTypes="ST_BubbleScalePercent ST_BubbleScaleUInt"/>
331 </xsd:simpleType>
332 <xsd:simpleType name="ST_BubbleScalePercent">
333   <xsd:restriction base="xsd:string">
334     <xsd:pattern value="0*(([0-9])|([1-9][0-9])|([1-2][0-9][0-9])|300)%"/>
335   </xsd:restriction>
336 </xsd:simpleType>
337 <xsd:simpleType name="ST_BubbleScaleUInt">
338   <xsd:restriction base="xsd:unsignedInt">
339     <xsd:minInclusive value="0"/>
340     <xsd:maxInclusive value="300"/>
341   </xsd:restriction>
342 </xsd:simpleType>
343 <xsd:complexType name="CT_BubbleScale">
344   <xsd:attribute name="val" type="ST_BubbleScale" default="100%"/>
345 </xsd:complexType>
346 <xsd:simpleType name="ST_SizeRepresents">
347   <xsd:restriction base="xsd:string">

```

```

348         <xsd:enumeration value="area"/>
349         <xsd:enumeration value="w"/>
350     </xsd:restriction>
351 </xsd:simpleType>
352 <xsd:complexType name="CT_SizeRepresents">
353     <xsd:attribute name="val" type="ST_SizeRepresents" default="area"/>
354 </xsd:complexType>
355 <xsd:simpleType name="ST_FirstSliceAng">
356     <xsd:restriction base="xsd:unsignedShort">
357         <xsd:minInclusive value="0"/>
358         <xsd:maxInclusive value="360"/>
359     </xsd:restriction>
360 </xsd:simpleType>
361 <xsd:complexType name="CT_FirstSliceAng">
362     <xsd:attribute name="val" type="ST_FirstSliceAng" default="0"/>
363 </xsd:complexType>
364 <xsd:simpleType name="ST_HoleSize">
365     <xsd:union memberTypes="ST_HoleSizePercent ST_HoleSizeUByte"/>
366 </xsd:simpleType>
367 <xsd:simpleType name="ST_HoleSizePercent">
368     <xsd:restriction base="xsd:string">
369         <xsd:pattern value="0*([1-9]|([1-8][0-9])|90)%"/>
370     </xsd:restriction>
371 </xsd:simpleType>
372 <xsd:simpleType name="ST_HoleSizeUByte">
373     <xsd:restriction base="xsd:unsignedByte">
374         <xsd:minInclusive value="10"/>
375         <xsd:maxInclusive value="90"/>
376     </xsd:restriction>
377 </xsd:simpleType>
378 <xsd:complexType name="CT_HoleSize">
379     <xsd:attribute name="val" type="ST_HoleSize" default="10%"/>
380 </xsd:complexType>
381 <xsd:simpleType name="ST_SplitType">
382     <xsd:restriction base="xsd:string">
383         <xsd:enumeration value="auto"/>
384         <xsd:enumeration value="cust"/>
385         <xsd:enumeration value="percent"/>
386         <xsd:enumeration value="pos"/>
387         <xsd:enumeration value="val"/>
388     </xsd:restriction>
389 </xsd:simpleType>
390 <xsd:complexType name="CT_SplitType">
391     <xsd:attribute name="val" type="ST_SplitType" default="auto"/>
392 </xsd:complexType>
393 <xsd:complexType name="CT_CustSplit">
394     <xsd:sequence>
395         <xsd:element name="secondPiePt" type="CT_UnsignedInt" minOccurs="0"
396             maxOccurs="unbounded"/>
397     </xsd:sequence>
398 </xsd:complexType>
399 <xsd:simpleType name="ST_SecondPieSize">
400     <xsd:union memberTypes="ST_SecondPieSizePercent ST_SecondPieSizeUShort"/>

```

```

401 </xsd:simpleType>
402 <xsd:simpleType name="ST_SecondPieSizePercent">
403   <xsd:restriction base="xsd:string">
404     <xsd:pattern value="0*(([5-9])|([1-9][0-9])|(1[0-9][0-9])|200)%"/>
405   </xsd:restriction>
406 </xsd:simpleType>
407 <xsd:simpleType name="ST_SecondPieSizeUShort">
408   <xsd:restriction base="xsd:unsignedShort">
409     <xsd:minInclusive value="5"/>
410     <xsd:maxInclusive value="200"/>
411   </xsd:restriction>
412 </xsd:simpleType>
413 <xsd:complexType name="CT_SecondPieSize">
414   <xsd:attribute name="val" type="ST_SecondPieSize" default="75%"/>
415 </xsd:complexType>
416 <xsd:complexType name="CT_NumFmt">
417   <xsd:attribute name="formatCode" type="s:ST_Xstring" use="required"/>
418   <xsd:attribute name="sourceLinked" type="xsd:boolean"/>
419 </xsd:complexType>
420 <xsd:simpleType name="ST_LblAlgn">
421   <xsd:restriction base="xsd:string">
422     <xsd:enumeration value="ctr"/>
423     <xsd:enumeration value="l"/>
424     <xsd:enumeration value="r"/>
425   </xsd:restriction>
426 </xsd:simpleType>
427 <xsd:complexType name="CT_LblAlgn">
428   <xsd:attribute name="val" type="ST_LblAlgn" use="required"/>
429 </xsd:complexType>
430 <xsd:simpleType name="ST_DLblPos">
431   <xsd:restriction base="xsd:string">
432     <xsd:enumeration value="bestFit"/>
433     <xsd:enumeration value="b"/>
434     <xsd:enumeration value="ctr"/>
435     <xsd:enumeration value="inBase"/>
436     <xsd:enumeration value="inEnd"/>
437     <xsd:enumeration value="l"/>
438     <xsd:enumeration value="outEnd"/>
439     <xsd:enumeration value="r"/>
440     <xsd:enumeration value="t"/>
441   </xsd:restriction>
442 </xsd:simpleType>
443 <xsd:complexType name="CT_DLblPos">
444   <xsd:attribute name="val" type="ST_DLblPos" use="required"/>
445 </xsd:complexType>
446 <xsd:group name="EG_DLblShared">
447   <xsd:sequence>
448     <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
449     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
450     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
451     <xsd:element name="dLblPos" type="CT_DLblPos" minOccurs="0" maxOccurs="1"/>
452     <xsd:element name="showLegendKey" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
453     <xsd:element name="showVal" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>

```

```

454     <xsd:element name="showCatName" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
455     <xsd:element name="showSerName" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
456     <xsd:element name="showPercent" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
457     <xsd:element name="showBubbleSize" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
458     <xsd:element name="separator" type="xsd:string" minOccurs="0" maxOccurs="1"/>
459   </xsd:sequence>
460 </xsd:group>
461 <xsd:group name="Group_DLbl">
462   <xsd:sequence>
463     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
464     <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
465     <xsd:group ref="EG_DLblShared" minOccurs="1" maxOccurs="1"/>
466   </xsd:sequence>
467 </xsd:group>
468 <xsd:complexType name="CT_DLbl">
469   <xsd:sequence>
470     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
471     <xsd:choice>
472       <xsd:element name="delete" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
473       <xsd:group ref="Group_DLbl" minOccurs="1" maxOccurs="1"/>
474     </xsd:choice>
475     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
476   </xsd:sequence>
477 </xsd:complexType>
478 <xsd:group name="Group_DLbls">
479   <xsd:sequence>
480     <xsd:group ref="EG_DLblShared" minOccurs="1" maxOccurs="1"/>
481     <xsd:element name="showLeaderLines" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
482     <xsd:element name="leaderLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
483   </xsd:sequence>
484 </xsd:group>
485 <xsd:complexType name="CT_DLbls">
486   <xsd:sequence>
487     <xsd:element name="dLbl" type="CT_DLbl" minOccurs="0" maxOccurs="unbounded"/>
488     <xsd:choice>
489       <xsd:element name="delete" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
490       <xsd:group ref="Group_DLbls" minOccurs="1" maxOccurs="1"/>
491     </xsd:choice>
492     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
493   </xsd:sequence>
494 </xsd:complexType>
495 <xsd:simpleType name="ST_MarkerStyle">
496   <xsd:restriction base="xsd:string">
497     <xsd:enumeration value="circle"/>
498     <xsd:enumeration value="dash"/>
499     <xsd:enumeration value="diamond"/>
500     <xsd:enumeration value="dot"/>
501     <xsd:enumeration value="none"/>
502     <xsd:enumeration value="picture"/>
503     <xsd:enumeration value="plus"/>
504     <xsd:enumeration value="square"/>
505     <xsd:enumeration value="star"/>
506     <xsd:enumeration value="triangle"/>

```

```

507         <xsd:enumeration value="x"/>
508         <xsd:enumeration value="auto"/>
509     </xsd:restriction>
510 </xsd:simpleType>
511 <xsd:complexType name="CT_MarkerStyle">
512     <xsd:attribute name="val" type="ST_MarkerStyle" use="required"/>
513 </xsd:complexType>
514 <xsd:simpleType name="ST_MarkerSize">
515     <xsd:restriction base="xsd:unsignedByte">
516         <xsd:minInclusive value="2"/>
517         <xsd:maxInclusive value="72"/>
518     </xsd:restriction>
519 </xsd:simpleType>
520 <xsd:complexType name="CT_MarkerSize">
521     <xsd:attribute name="val" type="ST_MarkerSize" default="5"/>
522 </xsd:complexType>
523 <xsd:complexType name="CT_Marker">
524     <xsd:sequence>
525         <xsd:element name="symbol" type="CT_MarkerStyle" minOccurs="0" maxOccurs="1"/>
526         <xsd:element name="size" type="CT_MarkerSize" minOccurs="0" maxOccurs="1"/>
527         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
528         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
529     </xsd:sequence>
530 </xsd:complexType>
531 <xsd:complexType name="CT_DPt">
532     <xsd:sequence>
533         <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
534         <xsd:element name="invertIfNegative" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
535         <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
536         <xsd:element name="bubble3D" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
537         <xsd:element name="explosion" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
538         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
539         <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
540         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
541     </xsd:sequence>
542 </xsd:complexType>
543 <xsd:simpleType name="ST_TrendlineType">
544     <xsd:restriction base="xsd:string">
545         <xsd:enumeration value="exp"/>
546         <xsd:enumeration value="linear"/>
547         <xsd:enumeration value="log"/>
548         <xsd:enumeration value="movingAvg"/>
549         <xsd:enumeration value="poly"/>
550         <xsd:enumeration value="power"/>
551     </xsd:restriction>
552 </xsd:simpleType>
553 <xsd:complexType name="CT_TrendlineType">
554     <xsd:attribute name="val" type="ST_TrendlineType" default="linear"/>
555 </xsd:complexType>
556 <xsd:simpleType name="ST_Order">
557     <xsd:restriction base="xsd:unsignedByte">
558         <xsd:minInclusive value="2"/>
559         <xsd:maxInclusive value="6"/>

```

```

560     </xsd:restriction>
561 </xsd:simpleType>
562 <xsd:complexType name="CT_Order">
563     <xsd:attribute name="val" type="ST_Order" default="2"/>
564 </xsd:complexType>
565 <xsd:simpleType name="ST_Period">
566     <xsd:restriction base="xsd:unsignedInt">
567         <xsd:minInclusive value="2"/>
568     </xsd:restriction>
569 </xsd:simpleType>
570 <xsd:complexType name="CT_Period">
571     <xsd:attribute name="val" type="ST_Period" default="2"/>
572 </xsd:complexType>
573 <xsd:complexType name="CT_TrendlineLbl">
574     <xsd:sequence>
575         <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
576         <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
577         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
578         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
579         <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
580         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
581     </xsd:sequence>
582 </xsd:complexType>
583 <xsd:complexType name="CT_Trendline">
584     <xsd:sequence>
585         <xsd:element name="name" type="xsd:string" minOccurs="0" maxOccurs="1"/>
586         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
587         <xsd:element name="trendlineType" type="CT_TrendlineType" minOccurs="1" maxOccurs="1"/>
588         <xsd:element name="order" type="CT_Order" minOccurs="0" maxOccurs="1"/>
589         <xsd:element name="period" type="CT_Period" minOccurs="0" maxOccurs="1"/>
590         <xsd:element name="forward" type="CT_Double" minOccurs="0" maxOccurs="1"/>
591         <xsd:element name="backward" type="CT_Double" minOccurs="0" maxOccurs="1"/>
592         <xsd:element name="intercept" type="CT_Double" minOccurs="0" maxOccurs="1"/>
593         <xsd:element name="dispRSqr" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
594         <xsd:element name="dispEq" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
595         <xsd:element name="trendlineLbl" type="CT_TrendlineLbl" minOccurs="0" maxOccurs="1"/>
596         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
597     </xsd:sequence>
598 </xsd:complexType>
599 <xsd:simpleType name="ST_ErrDir">
600     <xsd:restriction base="xsd:string">
601         <xsd:enumeration value="x"/>
602         <xsd:enumeration value="y"/>
603     </xsd:restriction>
604 </xsd:simpleType>
605 <xsd:complexType name="CT_ErrDir">
606     <xsd:attribute name="val" type="ST_ErrDir" use="required"/>
607 </xsd:complexType>
608 <xsd:simpleType name="ST_ErrBarType">
609     <xsd:restriction base="xsd:string">
610         <xsd:enumeration value="both"/>
611         <xsd:enumeration value="minus"/>
612         <xsd:enumeration value="plus"/>

```

```

613     </xsd:restriction>
614 </xsd:simpleType>
615 <xsd:complexType name="CT_ErrBarType">
616     <xsd:attribute name="val" type="ST_ErrBarType" default="both"/>
617 </xsd:complexType>
618 <xsd:simpleType name="ST_ErrValType">
619     <xsd:restriction base="xsd:string">
620         <xsd:enumeration value="cust"/>
621         <xsd:enumeration value="fixedVal"/>
622         <xsd:enumeration value="percentage"/>
623         <xsd:enumeration value="stdDev"/>
624         <xsd:enumeration value="stdErr"/>
625     </xsd:restriction>
626 </xsd:simpleType>
627 <xsd:complexType name="CT_ErrValType">
628     <xsd:attribute name="val" type="ST_ErrValType" default="fixedVal"/>
629 </xsd:complexType>
630 <xsd:complexType name="CT_ErrBars">
631     <xsd:sequence>
632         <xsd:element name="errDir" type="CT_ErrDir" minOccurs="0" maxOccurs="1"/>
633         <xsd:element name="errBarType" type="CT_ErrBarType" minOccurs="1" maxOccurs="1"/>
634         <xsd:element name="errValType" type="CT_ErrValType" minOccurs="1" maxOccurs="1"/>
635         <xsd:element name="noEndCap" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
636         <xsd:element name="plus" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
637         <xsd:element name="minus" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
638         <xsd:element name="val" type="CT_Double" minOccurs="0" maxOccurs="1"/>
639         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
640         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
641     </xsd:sequence>
642 </xsd:complexType>
643 <xsd:complexType name="CT_UpDownBar">
644     <xsd:sequence>
645         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
646     </xsd:sequence>
647 </xsd:complexType>
648 <xsd:complexType name="CT_UpDownBars">
649     <xsd:sequence>
650         <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
651         <xsd:element name="upBars" type="CT_UpDownBar" minOccurs="0" maxOccurs="1"/>
652         <xsd:element name="downBars" type="CT_UpDownBar" minOccurs="0" maxOccurs="1"/>
653         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
654     </xsd:sequence>
655 </xsd:complexType>
656 <xsd:group name="EG_SerShared">
657     <xsd:sequence>
658         <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
659         <xsd:element name="order" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
660         <xsd:element name="tx" type="CT_SerTx" minOccurs="0" maxOccurs="1"/>
661         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
662     </xsd:sequence>
663 </xsd:group>
664 <xsd:complexType name="CT_LineSer">
665     <xsd:sequence>

```



```

666     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
667     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
668     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
669     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
670     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
671     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="1"/>
672     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
673     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
674     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
675     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
676   </xsd:sequence>
677 </xsd:complexType>
678 <xsd:complexType name="CT_ScatterSer">
679   <xsd:sequence>
680     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
681     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
682     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
683     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
684     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
685     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
686     <xsd:element name="xVal" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
687     <xsd:element name="yVal" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
688     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
689     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
690   </xsd:sequence>
691 </xsd:complexType>
692 <xsd:complexType name="CT_RadarSer">
693   <xsd:sequence>
694     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
695     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
696     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
697     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
698     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
699     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
700     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
701   </xsd:sequence>
702 </xsd:complexType>
703 <xsd:complexType name="CT_BarSer">
704   <xsd:sequence>
705     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
706     <xsd:element name="invertIfNegative" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
707     <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
708     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
709     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
710     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
711     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="1"/>
712     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
713     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
714     <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
715     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
716   </xsd:sequence>
717 </xsd:complexType>
718 <xsd:complexType name="CT_AreaSer">

```

```

719     <xsd:sequence>
720       <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
721       <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
722       <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
723       <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
724       <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
725       <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
726       <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
727       <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
728       <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
729     </xsd:sequence>
730   </xsd:complexType>
731   <xsd:complexType name="CT_PieSer">
732     <xsd:sequence>
733       <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
734       <xsd:element name="explosion" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
735       <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
736       <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
737       <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
738       <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
739       <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
740     </xsd:sequence>
741   </xsd:complexType>
742   <xsd:complexType name="CT_BubbleSer">
743     <xsd:sequence>
744       <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
745       <xsd:element name="invertIfNegative" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
746       <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
747       <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
748       <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
749       <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
750       <xsd:element name="xVal" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
751       <xsd:element name="yVal" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
752       <xsd:element name="bubbleSize" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
753       <xsd:element name="bubble3D" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
754       <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
755     </xsd:sequence>
756   </xsd:complexType>
757   <xsd:complexType name="CT_SurfaceSer">
758     <xsd:sequence>
759       <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
760       <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
761       <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
762       <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
763     </xsd:sequence>
764   </xsd:complexType>
765   <xsd:simpleType name="ST_Grouping">
766     <xsd:restriction base="xsd:string">
767       <xsd:enumeration value="percentStacked"/>
768       <xsd:enumeration value="standard"/>
769       <xsd:enumeration value="stacked"/>
770     </xsd:restriction>
771   </xsd:simpleType>

```

```

772 <xsd:complexType name="CT_Grouping">
773   <xsd:attribute name="val" type="ST_Grouping" default="standard"/>
774 </xsd:complexType>
775 <xsd:complexType name="CT_ChartLines">
776   <xsd:sequence>
777     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
778   </xsd:sequence>
779 </xsd:complexType>
780 <xsd:group name="EG_LineChartShared">
781   <xsd:sequence>
782     <xsd:element name="grouping" type="CT_Grouping" minOccurs="1" maxOccurs="1"/>
783     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
784     <xsd:element name="ser" type="CT_LineSer" minOccurs="0" maxOccurs="unbounded"/>
785     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
786     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
787   </xsd:sequence>
788 </xsd:group>
789 <xsd:complexType name="CT_LineChart">
790   <xsd:sequence>
791     <xsd:group ref="EG_LineChartShared" minOccurs="1" maxOccurs="1"/>
792     <xsd:element name="hiLowLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
793     <xsd:element name="upDownBars" type="CT_UpDownBars" minOccurs="0" maxOccurs="1"/>
794     <xsd:element name="marker" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
795     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
796     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
797     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
798   </xsd:sequence>
799 </xsd:complexType>
800 <xsd:complexType name="CT_Line3DChart">
801   <xsd:sequence>
802     <xsd:group ref="EG_LineChartShared" minOccurs="1" maxOccurs="1"/>
803     <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
804     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="3" maxOccurs="3"/>
805     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
806   </xsd:sequence>
807 </xsd:complexType>
808 <xsd:complexType name="CT_StockChart">
809   <xsd:sequence>
810     <xsd:element name="ser" type="CT_LineSer" minOccurs="3" maxOccurs="4"/>
811     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
812     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
813     <xsd:element name="hiLowLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
814     <xsd:element name="upDownBars" type="CT_UpDownBars" minOccurs="0" maxOccurs="1"/>
815     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
816     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
817   </xsd:sequence>
818 </xsd:complexType>
819 <xsd:simpleType name="ST_ScatterStyle">
820   <xsd:restriction base="xsd:string">
821     <xsd:enumeration value="none"/>
822     <xsd:enumeration value="line"/>
823     <xsd:enumeration value="lineMarker"/>
824     <xsd:enumeration value="marker"/>

```

```

825         <xsd:enumeration value="smooth"/>
826         <xsd:enumeration value="smoothMarker"/>
827     </xsd:restriction>
828 </xsd:simpleType>
829 <xsd:complexType name="CT_ScatterStyle">
830     <xsd:attribute name="val" type="ST_ScatterStyle" default="marker"/>
831 </xsd:complexType>
832 <xsd:complexType name="CT_ScatterChart">
833     <xsd:sequence>
834         <xsd:element name="scatterStyle" type="CT_ScatterStyle" minOccurs="1" maxOccurs="1"/>
835         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
836         <xsd:element name="ser" type="CT_ScatterSer" minOccurs="0" maxOccurs="unbounded"/>
837         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
838         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
839         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
840     </xsd:sequence>
841 </xsd:complexType>
842 <xsd:simpleType name="ST_RadarStyle">
843     <xsd:restriction base="xsd:string">
844         <xsd:enumeration value="standard"/>
845         <xsd:enumeration value="marker"/>
846         <xsd:enumeration value="filled"/>
847     </xsd:restriction>
848 </xsd:simpleType>
849 <xsd:complexType name="CT_RadarStyle">
850     <xsd:attribute name="val" type="ST_RadarStyle" default="standard"/>
851 </xsd:complexType>
852 <xsd:complexType name="CT_RadarChart">
853     <xsd:sequence>
854         <xsd:element name="radarStyle" type="CT_RadarStyle" minOccurs="1" maxOccurs="1"/>
855         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
856         <xsd:element name="ser" type="CT_RadarSer" minOccurs="0" maxOccurs="unbounded"/>
857         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
858         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
859         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
860     </xsd:sequence>
861 </xsd:complexType>
862 <xsd:simpleType name="ST_BarGrouping">
863     <xsd:restriction base="xsd:string">
864         <xsd:enumeration value="percentStacked"/>
865         <xsd:enumeration value="clustered"/>
866         <xsd:enumeration value="standard"/>
867         <xsd:enumeration value="stacked"/>
868     </xsd:restriction>
869 </xsd:simpleType>
870 <xsd:complexType name="CT_BarGrouping">
871     <xsd:attribute name="val" type="ST_BarGrouping" default="clustered"/>
872 </xsd:complexType>
873 <xsd:simpleType name="ST_BarDir">
874     <xsd:restriction base="xsd:string">
875         <xsd:enumeration value="bar"/>
876         <xsd:enumeration value="col"/>
877     </xsd:restriction>

```

```

878 </xsd:simpleType>
879 <xsd:complexType name="CT_BarDir">
880   <xsd:attribute name="val" type="ST_BarDir" default="col"/>
881 </xsd:complexType>
882 <xsd:simpleType name="ST_Shape">
883   <xsd:restriction base="xsd:string">
884     <xsd:enumeration value="cone"/>
885     <xsd:enumeration value="coneToMax"/>
886     <xsd:enumeration value="box"/>
887     <xsd:enumeration value="cylinder"/>
888     <xsd:enumeration value="pyramid"/>
889     <xsd:enumeration value="pyramidToMax"/>
890   </xsd:restriction>
891 </xsd:simpleType>
892 <xsd:complexType name="CT_Shape">
893   <xsd:attribute name="val" type="ST_Shape" default="box"/>
894 </xsd:complexType>
895 <xsd:group name="EG_BarChartShared">
896   <xsd:sequence>
897     <xsd:element name="barDir" type="CT_BarDir" minOccurs="1" maxOccurs="1"/>
898     <xsd:element name="grouping" type="CT_BarGrouping" minOccurs="0" maxOccurs="1"/>
899     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
900     <xsd:element name="ser" type="CT_BarSer" minOccurs="0" maxOccurs="unbounded"/>
901     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
902   </xsd:sequence>
903 </xsd:group>
904 <xsd:complexType name="CT_BarChart">
905   <xsd:sequence>
906     <xsd:group ref="EG_BarChartShared" minOccurs="1" maxOccurs="1"/>
907     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
908     <xsd:element name="overlap" type="CT_Overlap" minOccurs="0" maxOccurs="1"/>
909     <xsd:element name="serLines" type="CT_ChartLines" minOccurs="0" maxOccurs="unbounded"/>
910     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
911     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
912   </xsd:sequence>
913 </xsd:complexType>
914 <xsd:complexType name="CT_Bar3DChart">
915   <xsd:sequence>
916     <xsd:group ref="EG_BarChartShared" minOccurs="1" maxOccurs="1"/>
917     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
918     <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
919     <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
920     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="3"/>
921     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
922   </xsd:sequence>
923 </xsd:complexType>
924 <xsd:group name="EG_AreaChartShared">
925   <xsd:sequence>
926     <xsd:element name="grouping" type="CT_Grouping" minOccurs="0" maxOccurs="1"/>
927     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
928     <xsd:element name="ser" type="CT_AreaSer" minOccurs="0" maxOccurs="unbounded"/>
929     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
930     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>

```

```

931     </xsd:sequence>
932 </xsd:group>
933 <xsd:complexType name="CT_AreaChart">
934     <xsd:sequence>
935         <xsd:group ref="EG_AreaChartShared" minOccurs="1" maxOccurs="1"/>
936         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
937         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
938     </xsd:sequence>
939 </xsd:complexType>
940 <xsd:complexType name="CT_Area3DChart">
941     <xsd:sequence>
942         <xsd:group ref="EG_AreaChartShared" minOccurs="1" maxOccurs="1"/>
943         <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
944         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="3"/>
945         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
946     </xsd:sequence>
947 </xsd:complexType>
948 <xsd:group name="EG_PieChartShared">
949     <xsd:sequence>
950         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
951         <xsd:element name="ser" type="CT_PieSer" minOccurs="0" maxOccurs="unbounded"/>
952         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
953     </xsd:sequence>
954 </xsd:group>
955 <xsd:complexType name="CT_PieChart">
956     <xsd:sequence>
957         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
958         <xsd:element name="firstSliceAng" type="CT_FirstSliceAng" minOccurs="0" maxOccurs="1"/>
959         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
960     </xsd:sequence>
961 </xsd:complexType>
962 <xsd:complexType name="CT_Pie3DChart">
963     <xsd:sequence>
964         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
965         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
966     </xsd:sequence>
967 </xsd:complexType>
968 <xsd:complexType name="CT_DoughnutChart">
969     <xsd:sequence>
970         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
971         <xsd:element name="firstSliceAng" type="CT_FirstSliceAng" minOccurs="0" maxOccurs="1"/>
972         <xsd:element name="holeSize" type="CT_HoleSize" minOccurs="0" maxOccurs="1"/>
973         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
974     </xsd:sequence>
975 </xsd:complexType>
976 <xsd:simpleType name="ST_OfPieType">
977     <xsd:restriction base="xsd:string">
978         <xsd:enumeration value="pie"/>
979         <xsd:enumeration value="bar"/>
980     </xsd:restriction>
981 </xsd:simpleType>
982 <xsd:complexType name="CT_OfPieType">
983     <xsd:attribute name="val" type="ST_OfPieType" default="pie"/>

```

```

984 </xsd:complexType>
985 <xsd:complexType name="CT_OfPieChart">
986   <xsd:sequence>
987     <xsd:element name="ofPieType" type="CT_OfPieType" minOccurs="1" maxOccurs="1"/>
988     <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
989     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
990     <xsd:element name="splitType" type="CT_SplitType" minOccurs="0" maxOccurs="1"/>
991     <xsd:element name="splitPos" type="CT_Double" minOccurs="0" maxOccurs="1"/>
992     <xsd:element name="custSplit" type="CT_CustSplit" minOccurs="0" maxOccurs="1"/>
993     <xsd:element name="secondPieSize" type="CT_SecondPieSize" minOccurs="0" maxOccurs="1"/>
994     <xsd:element name="serLines" type="CT_ChartLines" minOccurs="0" maxOccurs="unbounded"/>
995     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
996   </xsd:sequence>
997 </xsd:complexType>
998 <xsd:complexType name="CT_BubbleChart">
999   <xsd:sequence>
1000     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1001     <xsd:element name="ser" type="CT_BubbleSer" minOccurs="0" maxOccurs="unbounded"/>
1002     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
1003     <xsd:element name="bubble3D" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1004     <xsd:element name="bubbleScale" type="CT_BubbleScale" minOccurs="0" maxOccurs="1"/>
1005     <xsd:element name="showNegBubbles" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1006     <xsd:element name="sizeRepresents" type="CT_SizeRepresents" minOccurs="0" maxOccurs="1"/>
1007     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
1008     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1009   </xsd:sequence>
1010 </xsd:complexType>
1011 <xsd:complexType name="CT_BandFmt">
1012   <xsd:sequence>
1013     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1014     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1015   </xsd:sequence>
1016 </xsd:complexType>
1017 <xsd:complexType name="CT_BandFmts">
1018   <xsd:sequence>
1019     <xsd:element name="bandFmt" type="CT_BandFmt" minOccurs="0" maxOccurs="unbounded"/>
1020   </xsd:sequence>
1021 </xsd:complexType>
1022 <xsd:group name="EG_SurfaceChartShared">
1023   <xsd:sequence>
1024     <xsd:element name="wireframe" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1025     <xsd:element name="ser" type="CT_SurfaceSer" minOccurs="0" maxOccurs="unbounded"/>
1026     <xsd:element name="bandFmts" type="CT_BandFmts" minOccurs="0" maxOccurs="1"/>
1027   </xsd:sequence>
1028 </xsd:group>
1029 <xsd:complexType name="CT_SurfaceChart">
1030   <xsd:sequence>
1031     <xsd:group ref="EG_SurfaceChartShared" minOccurs="1" maxOccurs="1"/>
1032     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="3"/>
1033     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1034   </xsd:sequence>
1035 </xsd:complexType>
1036 <xsd:complexType name="CT_Surface3DChart">

```

```

1037     <xsd:sequence>
1038         <xsd:group ref="EG_SurfaceChartShared" minOccurs="1" maxOccurs="1"/>
1039         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="3" maxOccurs="3"/>
1040         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1041     </xsd:sequence>
1042 </xsd:complexType>
1043 <xsd:simpleType name="ST_AxPos">
1044     <xsd:restriction base="xsd:string">
1045         <xsd:enumeration value="b"/>
1046         <xsd:enumeration value="l"/>
1047         <xsd:enumeration value="r"/>
1048         <xsd:enumeration value="t"/>
1049     </xsd:restriction>
1050 </xsd:simpleType>
1051 <xsd:complexType name="CT_AxPos">
1052     <xsd:attribute name="val" type="ST_AxPos" use="required"/>
1053 </xsd:complexType>
1054 <xsd:simpleType name="ST_Crosses">
1055     <xsd:restriction base="xsd:string">
1056         <xsd:enumeration value="autoZero"/>
1057         <xsd:enumeration value="max"/>
1058         <xsd:enumeration value="min"/>
1059     </xsd:restriction>
1060 </xsd:simpleType>
1061 <xsd:complexType name="CT_Crosses">
1062     <xsd:attribute name="val" type="ST_Crosses" use="required"/>
1063 </xsd:complexType>
1064 <xsd:simpleType name="ST_CrossBetween">
1065     <xsd:restriction base="xsd:string">
1066         <xsd:enumeration value="between"/>
1067         <xsd:enumeration value="midCat"/>
1068     </xsd:restriction>
1069 </xsd:simpleType>
1070 <xsd:complexType name="CT_CrossBetween">
1071     <xsd:attribute name="val" type="ST_CrossBetween" use="required"/>
1072 </xsd:complexType>
1073 <xsd:simpleType name="ST_TickMark">
1074     <xsd:restriction base="xsd:string">
1075         <xsd:enumeration value="cross"/>
1076         <xsd:enumeration value="in"/>
1077         <xsd:enumeration value="none"/>
1078         <xsd:enumeration value="out"/>
1079     </xsd:restriction>
1080 </xsd:simpleType>
1081 <xsd:complexType name="CT_TickMark">
1082     <xsd:attribute name="val" type="ST_TickMark" default="cross"/>
1083 </xsd:complexType>
1084 <xsd:simpleType name="ST_TickLblPos">
1085     <xsd:restriction base="xsd:string">
1086         <xsd:enumeration value="high"/>
1087         <xsd:enumeration value="low"/>
1088         <xsd:enumeration value="nextTo"/>
1089         <xsd:enumeration value="none"/>

```



```

1090     </xsd:restriction>
1091 </xsd:simpleType>
1092 <xsd:complexType name="CT_TickLblPos">
1093     <xsd:attribute name="val" type="ST_TickLblPos" default="nextTo"/>
1094 </xsd:complexType>
1095 <xsd:simpleType name="ST_Skip">
1096     <xsd:restriction base="xsd:unsignedInt">
1097         <xsd:minInclusive value="1"/>
1098     </xsd:restriction>
1099 </xsd:simpleType>
1100 <xsd:complexType name="CT_Skip">
1101     <xsd:attribute name="val" type="ST_Skip" use="required"/>
1102 </xsd:complexType>
1103 <xsd:simpleType name="ST_TimeUnit">
1104     <xsd:restriction base="xsd:string">
1105         <xsd:enumeration value="days"/>
1106         <xsd:enumeration value="months"/>
1107         <xsd:enumeration value="years"/>
1108     </xsd:restriction>
1109 </xsd:simpleType>
1110 <xsd:complexType name="CT_TimeUnit">
1111     <xsd:attribute name="val" type="ST_TimeUnit" default="days"/>
1112 </xsd:complexType>
1113 <xsd:simpleType name="ST_AxisUnit">
1114     <xsd:restriction base="xsd:double">
1115         <xsd:minExclusive value="0"/>
1116     </xsd:restriction>
1117 </xsd:simpleType>
1118 <xsd:complexType name="CT_AxisUnit">
1119     <xsd:attribute name="val" type="ST_AxisUnit" use="required"/>
1120 </xsd:complexType>
1121 <xsd:simpleType name="ST_BuiltInUnit">
1122     <xsd:restriction base="xsd:string">
1123         <xsd:enumeration value="hundreds"/>
1124         <xsd:enumeration value="thousands"/>
1125         <xsd:enumeration value="tenThousands"/>
1126         <xsd:enumeration value="hundredThousands"/>
1127         <xsd:enumeration value="millions"/>
1128         <xsd:enumeration value="tenMillions"/>
1129         <xsd:enumeration value="hundredMillions"/>
1130         <xsd:enumeration value="billions"/>
1131         <xsd:enumeration value="trillions"/>
1132     </xsd:restriction>
1133 </xsd:simpleType>
1134 <xsd:complexType name="CT_BuiltInUnit">
1135     <xsd:attribute name="val" type="ST_BuiltInUnit" default="thousands"/>
1136 </xsd:complexType>
1137 <xsd:simpleType name="ST_PictureFormat">
1138     <xsd:restriction base="xsd:string">
1139         <xsd:enumeration value="stretch"/>
1140         <xsd:enumeration value="stack"/>
1141         <xsd:enumeration value="stackScale"/>
1142     </xsd:restriction>

```

```

1143 </xsd:simpleType>
1144 <xsd:complexType name="CT_PictureFormat">
1145   <xsd:attribute name="val" type="ST_PictureFormat" use="required"/>
1146 </xsd:complexType>
1147 <xsd:simpleType name="ST_PictureStackUnit">
1148   <xsd:restriction base="xsd:double">
1149     <xsd:minExclusive value="0"/>
1150   </xsd:restriction>
1151 </xsd:simpleType>
1152 <xsd:complexType name="CT_PictureStackUnit">
1153   <xsd:attribute name="val" type="ST_PictureStackUnit" use="required"/>
1154 </xsd:complexType>
1155 <xsd:complexType name="CT_PictureOptions">
1156   <xsd:sequence>
1157     <xsd:element name="applyToFront" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1158     <xsd:element name="applyToSides" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1159     <xsd:element name="applyToEnd" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1160     <xsd:element name="pictureFormat" type="CT_PictureFormat" minOccurs="0" maxOccurs="1"/>
1161     <xsd:element name="pictureStackUnit" type="CT_PictureStackUnit" minOccurs="0"
1162       maxOccurs="1"/>
1163   </xsd:sequence>
1164 </xsd:complexType>
1165 <xsd:complexType name="CT_DispUnitsLbl">
1166   <xsd:sequence>
1167     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
1168     <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
1169     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1170     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1171   </xsd:sequence>
1172 </xsd:complexType>
1173 <xsd:complexType name="CT_DispUnits">
1174   <xsd:sequence>
1175     <xsd:choice>
1176       <xsd:element name="custUnit" type="CT_Double" minOccurs="1" maxOccurs="1"/>
1177       <xsd:element name="builtInUnit" type="CT_BuiltInUnit" minOccurs="1" maxOccurs="1"/>
1178     </xsd:choice>
1179     <xsd:element name="dispUnitsLbl" type="CT_DispUnitsLbl" minOccurs="0" maxOccurs="1"/>
1180     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1181   </xsd:sequence>
1182 </xsd:complexType>
1183 <xsd:simpleType name="ST_Orientation">
1184   <xsd:restriction base="xsd:string">
1185     <xsd:enumeration value="maxMin"/>
1186     <xsd:enumeration value="minMax"/>
1187   </xsd:restriction>
1188 </xsd:simpleType>
1189 <xsd:complexType name="CT_Orientation">
1190   <xsd:attribute name="val" type="ST_Orientation" default="minMax"/>
1191 </xsd:complexType>
1192 <xsd:simpleType name="ST_LogBase">
1193   <xsd:restriction base="xsd:double">
1194     <xsd:minInclusive value="2"/>
1195     <xsd:maxInclusive value="1000"/>

```

```

1196     </xsd:restriction>
1197 </xsd:simpleType>
1198 <xsd:complexType name="CT_LogBase">
1199     <xsd:attribute name="val" type="ST_LogBase" use="required"/>
1200 </xsd:complexType>
1201 <xsd:complexType name="CT_Scaling">
1202     <xsd:sequence>
1203         <xsd:element name="logBase" type="CT_LogBase" minOccurs="0" maxOccurs="1"/>
1204         <xsd:element name="orientation" type="CT_Orientation" minOccurs="0" maxOccurs="1"/>
1205         <xsd:element name="max" type="CT_Double" minOccurs="0" maxOccurs="1"/>
1206         <xsd:element name="min" type="CT_Double" minOccurs="0" maxOccurs="1"/>
1207         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1208     </xsd:sequence>
1209 </xsd:complexType>
1210 <xsd:simpleType name="ST_LblOffset">
1211     <xsd:union memberTypes="ST_LblOffsetPercent ST_LblOffsetUShort"/>
1212 </xsd:simpleType>
1213 <xsd:simpleType name="ST_LblOffsetPercent">
1214     <xsd:restriction base="xsd:string">
1215         <xsd:pattern value="0*(([0-9])|([1-9][0-9])|([1-9][0-9][0-9])|1000)%"/>
1216     </xsd:restriction>
1217 </xsd:simpleType>
1218 <xsd:simpleType name="ST_LblOffsetUShort">
1219     <xsd:restriction base="xsd:unsignedShort">
1220         <xsd:minInclusive value="0"/>
1221         <xsd:maxInclusive value="1000"/>
1222     </xsd:restriction>
1223 </xsd:simpleType>
1224 <xsd:complexType name="CT_LblOffset">
1225     <xsd:attribute name="val" type="ST_LblOffset" default="100%"/>
1226 </xsd:complexType>
1227 <xsd:group name="EG_AxShared">
1228     <xsd:sequence>
1229         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1230         <xsd:element name="scaling" type="CT_Scaling" minOccurs="1" maxOccurs="1"/>
1231         <xsd:element name="delete" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1232         <xsd:element name="axPos" type="CT_AxPos" minOccurs="1" maxOccurs="1"/>
1233         <xsd:element name="majorGridlines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
1234         <xsd:element name="minorGridlines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
1235         <xsd:element name="title" type="CT_Title" minOccurs="0" maxOccurs="1"/>
1236         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
1237         <xsd:element name="majorTickMark" type="CT_TickMark" minOccurs="0" maxOccurs="1"/>
1238         <xsd:element name="minorTickMark" type="CT_TickMark" minOccurs="0" maxOccurs="1"/>
1239         <xsd:element name="tickLblPos" type="CT_TickLblPos" minOccurs="0" maxOccurs="1"/>
1240         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1241         <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1242         <xsd:element name="crossAx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1243         <xsd:choice minOccurs="0" maxOccurs="1">
1244             <xsd:element name="crosses" type="CT_Crosses" minOccurs="1" maxOccurs="1"/>
1245             <xsd:element name="crossesAt" type="CT_Double" minOccurs="1" maxOccurs="1"/>
1246         </xsd:choice>
1247     </xsd:sequence>
1248 </xsd:group>

```

```

1249 <xsd:complexType name="CT_CatAx">
1250   <xsd:sequence>
1251     <xsd:group ref="EG_AxShared" minOccurs="1" maxOccurs="1"/>
1252     <xsd:element name="auto" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1253     <xsd:element name="lblAlign" type="CT_LblAlign" minOccurs="0" maxOccurs="1"/>
1254     <xsd:element name="lblOffset" type="CT_LblOffset" minOccurs="0" maxOccurs="1"/>
1255     <xsd:element name="tickLblSkip" type="CT_Skip" minOccurs="0" maxOccurs="1"/>
1256     <xsd:element name="tickMarkSkip" type="CT_Skip" minOccurs="0" maxOccurs="1"/>
1257     <xsd:element name="noMultiLvlLbl" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1258     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1259   </xsd:sequence>
1260 </xsd:complexType>
1261 <xsd:complexType name="CT_DateAx">
1262   <xsd:sequence>
1263     <xsd:group ref="EG_AxShared" minOccurs="1" maxOccurs="1"/>
1264     <xsd:element name="auto" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1265     <xsd:element name="lblOffset" type="CT_LblOffset" minOccurs="0" maxOccurs="1"/>
1266     <xsd:element name="baseTimeUnit" type="CT_TimeUnit" minOccurs="0" maxOccurs="1"/>
1267     <xsd:element name="majorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1268     <xsd:element name="majorTimeUnit" type="CT_TimeUnit" minOccurs="0" maxOccurs="1"/>
1269     <xsd:element name="minorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1270     <xsd:element name="minorTimeUnit" type="CT_TimeUnit" minOccurs="0" maxOccurs="1"/>
1271     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1272   </xsd:sequence>
1273 </xsd:complexType>
1274 <xsd:complexType name="CT_SerAx">
1275   <xsd:sequence>
1276     <xsd:group ref="EG_AxShared" minOccurs="1" maxOccurs="1"/>
1277     <xsd:element name="tickLblSkip" type="CT_Skip" minOccurs="0" maxOccurs="1"/>
1278     <xsd:element name="tickMarkSkip" type="CT_Skip" minOccurs="0" maxOccurs="1"/>
1279     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1280   </xsd:sequence>
1281 </xsd:complexType>
1282 <xsd:complexType name="CT_ValAx">
1283   <xsd:sequence>
1284     <xsd:group ref="EG_AxShared" minOccurs="1" maxOccurs="1"/>
1285     <xsd:element name="crossBetween" type="CT_CrossBetween" minOccurs="0" maxOccurs="1"/>
1286     <xsd:element name="majorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1287     <xsd:element name="minorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1288     <xsd:element name="dispUnits" type="CT_DispUnits" minOccurs="0" maxOccurs="1"/>
1289     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1290   </xsd:sequence>
1291 </xsd:complexType>
1292 <xsd:complexType name="CT_PlotArea">
1293   <xsd:sequence>
1294     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
1295     <xsd:choice minOccurs="1" maxOccurs="unbounded">
1296       <xsd:element name="areaChart" type="CT_AreaChart" minOccurs="1" maxOccurs="1"/>
1297       <xsd:element name="area3DChart" type="CT_Area3DChart" minOccurs="1" maxOccurs="1"/>
1298       <xsd:element name="lineChart" type="CT_LineChart" minOccurs="1" maxOccurs="1"/>
1299       <xsd:element name="line3DChart" type="CT_Line3DChart" minOccurs="1" maxOccurs="1"/>
1300       <xsd:element name="stockChart" type="CT_StockChart" minOccurs="1" maxOccurs="1"/>
1301       <xsd:element name="radarChart" type="CT_RadarChart" minOccurs="1" maxOccurs="1"/>

```

```

1302     <xsd:element name="scatterChart" type="CT_ScatterChart" minOccurs="1" maxOccurs="1"/>
1303     <xsd:element name="pieChart" type="CT_PieChart" minOccurs="1" maxOccurs="1"/>
1304     <xsd:element name="pie3DChart" type="CT_Pie3DChart" minOccurs="1" maxOccurs="1"/>
1305     <xsd:element name="doughnutChart" type="CT_DoughnutChart" minOccurs="1" maxOccurs="1"/>
1306     <xsd:element name="barChart" type="CT_BarChart" minOccurs="1" maxOccurs="1"/>
1307     <xsd:element name="bar3DChart" type="CT_Bar3DChart" minOccurs="1" maxOccurs="1"/>
1308     <xsd:element name="ofPieChart" type="CT_OfPieChart" minOccurs="1" maxOccurs="1"/>
1309     <xsd:element name="surfaceChart" type="CT_SurfaceChart" minOccurs="1" maxOccurs="1"/>
1310     <xsd:element name="surface3DChart" type="CT_Surface3DChart" minOccurs="1"
1311         maxOccurs="1"/>
1312     <xsd:element name="bubbleChart" type="CT_BubbleChart" minOccurs="1" maxOccurs="1"/>
1313 </xsd:choice>
1314 <xsd:choice minOccurs="0" maxOccurs="unbounded">
1315     <xsd:element name="valAx" type="CT_ValAx" minOccurs="1" maxOccurs="1"/>
1316     <xsd:element name="catAx" type="CT_CatAx" minOccurs="1" maxOccurs="1"/>
1317     <xsd:element name="dateAx" type="CT_DateAx" minOccurs="1" maxOccurs="1"/>
1318     <xsd:element name="serAx" type="CT_SerAx" minOccurs="1" maxOccurs="1"/>
1319 </xsd:choice>
1320 <xsd:element name="dTable" type="CT_DTable" minOccurs="0" maxOccurs="1"/>
1321 <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1322 <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1323 </xsd:sequence>
1324 </xsd:complexType>
1325 <xsd:complexType name="CT_PivotFmt">
1326     <xsd:sequence>
1327         <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1328         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1329         <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1330         <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
1331         <xsd:element name="dLbl" type="CT_DLbl" minOccurs="0" maxOccurs="1"/>
1332         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1333     </xsd:sequence>
1334 </xsd:complexType>
1335 <xsd:complexType name="CT_PivotFmts">
1336     <xsd:sequence>
1337         <xsd:element name="pivotFmt" type="CT_PivotFmt" minOccurs="0" maxOccurs="unbounded"/>
1338     </xsd:sequence>
1339 </xsd:complexType>
1340 <xsd:simpleType name="ST_LegendPos">
1341     <xsd:restriction base="xsd:string">
1342         <xsd:enumeration value="b"/>
1343         <xsd:enumeration value="tr"/>
1344         <xsd:enumeration value="l"/>
1345         <xsd:enumeration value="r"/>
1346         <xsd:enumeration value="t"/>
1347     </xsd:restriction>
1348 </xsd:simpleType>
1349 <xsd:complexType name="CT_LegendPos">
1350     <xsd:attribute name="val" type="ST_LegendPos" default="r"/>
1351 </xsd:complexType>
1352 <xsd:group name="EG_LegendEntryData">
1353     <xsd:sequence>
1354         <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>

```

```

1355     </xsd:sequence>
1356 </xsd:group>
1357 <xsd:complexType name="CT_LegendEntry">
1358     <xsd:sequence>
1359         <xsd:element name="idx" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
1360         <xsd:choice>
1361             <xsd:element name="delete" type="CT Boolean" minOccurs="1" maxOccurs="1"/>
1362             <xsd:group ref="EG_LegendEntryData" minOccurs="1" maxOccurs="1"/>
1363         </xsd:choice>
1364         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1365     </xsd:sequence>
1366 </xsd:complexType>
1367 <xsd:complexType name="CT_Legend">
1368     <xsd:sequence>
1369         <xsd:element name="legendPos" type="CT LegendPos" minOccurs="0" maxOccurs="1"/>
1370         <xsd:element name="legendEntry" type="CT LegendEntry" minOccurs="0"
1371             maxOccurs="unbounded"/>
1372         <xsd:element name="layout" type="CT Layout" minOccurs="0" maxOccurs="1"/>
1373         <xsd:element name="overlay" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1374         <xsd:element name="spPr" type="a:CT ShapeProperties" minOccurs="0" maxOccurs="1"/>
1375         <xsd:element name="txPr" type="a:CT TextBody" minOccurs="0" maxOccurs="1"/>
1376         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1377     </xsd:sequence>
1378 </xsd:complexType>
1379 <xsd:simpleType name="ST_DisbBlanksAs">
1380     <xsd:restriction base="xsd:string">
1381         <xsd:enumeration value="span"/>
1382         <xsd:enumeration value="gap"/>
1383         <xsd:enumeration value="zero"/>
1384     </xsd:restriction>
1385 </xsd:simpleType>
1386 <xsd:complexType name="CT_DisbBlanksAs">
1387     <xsd:attribute name="val" type="ST_DisbBlanksAs" default="zero"/>
1388 </xsd:complexType>
1389 <xsd:complexType name="CT_Chart">
1390     <xsd:sequence>
1391         <xsd:element name="title" type="CT Title" minOccurs="0" maxOccurs="1"/>
1392         <xsd:element name="autoTitleDeleted" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1393         <xsd:element name="pivotFmts" type="CT PivotFmts" minOccurs="0" maxOccurs="1"/>
1394         <xsd:element name="view3D" type="CT View3D" minOccurs="0" maxOccurs="1"/>
1395         <xsd:element name="floor" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1396         <xsd:element name="sideWall" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1397         <xsd:element name="backWall" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1398         <xsd:element name="plotArea" type="CT PlotArea" minOccurs="1" maxOccurs="1"/>
1399         <xsd:element name="legend" type="CT Legend" minOccurs="0" maxOccurs="1"/>
1400         <xsd:element name="plotVisOnly" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1401         <xsd:element name="disbBlanksAs" type="CT DisbBlanksAs" minOccurs="0" maxOccurs="1"/>
1402         <xsd:element name="showDLblsOverMax" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1403         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1404     </xsd:sequence>
1405 </xsd:complexType>
1406 <xsd:simpleType name="ST_Style">
1407     <xsd:restriction base="xsd:unsignedByte">

```

```

1408         <xsd:minInclusive value="1"/>
1409         <xsd:maxInclusive value="48"/>
1410     </xsd:restriction>
1411 </xsd:simpleType>
1412 <xsd:complexType name="CT_Style">
1413     <xsd:attribute name="val" type="ST_Style" use="required"/>
1414 </xsd:complexType>
1415 <xsd:complexType name="CT_PivotSource">
1416     <xsd:sequence>
1417         <xsd:element name="name" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
1418         <xsd:element name="fmtId" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1419         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="unbounded"/>
1420     </xsd:sequence>
1421 </xsd:complexType>
1422 <xsd:complexType name="CT_Protection">
1423     <xsd:sequence>
1424         <xsd:element name="chartObject" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1425         <xsd:element name="data" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1426         <xsd:element name="formatting" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1427         <xsd:element name="selection" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1428         <xsd:element name="userInterface" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1429     </xsd:sequence>
1430 </xsd:complexType>
1431 <xsd:complexType name="CT_HeaderFooter">
1432     <xsd:sequence>
1433         <xsd:element name="oddHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1434         <xsd:element name="oddFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1435         <xsd:element name="evenHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1436         <xsd:element name="evenFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1437         <xsd:element name="firstHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1438         <xsd:element name="firstFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1439     </xsd:sequence>
1440     <xsd:attribute name="alignWithMargins" type="xsd:boolean" default="true"/>
1441     <xsd:attribute name="differentOddEven" type="xsd:boolean" default="false"/>
1442     <xsd:attribute name="differentFirst" type="xsd:boolean" default="false"/>
1443 </xsd:complexType>
1444 <xsd:complexType name="CT_PageMargins">
1445     <xsd:attribute name="l" type="xsd:double" use="required"/>
1446     <xsd:attribute name="r" type="xsd:double" use="required"/>
1447     <xsd:attribute name="t" type="xsd:double" use="required"/>
1448     <xsd:attribute name="b" type="xsd:double" use="required"/>
1449     <xsd:attribute name="header" type="xsd:double" use="required"/>
1450     <xsd:attribute name="footer" type="xsd:double" use="required"/>
1451 </xsd:complexType>
1452 <xsd:simpleType name="ST_PageSetupOrientation">
1453     <xsd:restriction base="xsd:string">
1454         <xsd:enumeration value="default"/>
1455         <xsd:enumeration value="portrait"/>
1456         <xsd:enumeration value="landscape"/>
1457     </xsd:restriction>
1458 </xsd:simpleType>
1459 <xsd:complexType name="CT_ExternalData">
1460     <xsd:sequence>

```

```

1461     <xsd:element name="autoUpdate" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1462   </xsd:sequence>
1463   <xsd:attribute ref="r:id" use="required"/>
1464 </xsd:complexType>
1465 <xsd:complexType name="CT_PageSetup">
1466   <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>
1467   <xsd:attribute name="paperHeight" type="s:ST_PositiveUniversalMeasure" use="optional"/>
1468   <xsd:attribute name="paperWidth" type="s:ST_PositiveUniversalMeasure" use="optional"/>
1469   <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
1470   <xsd:attribute name="orientation" type="ST_PageSetupOrientation" use="optional"
1471     default="default"/>
1472   <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
1473   <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
1474   <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
1475   <xsd:attribute name="horizontalDpi" type="xsd:int" use="optional" default="600"/>
1476   <xsd:attribute name="verticalDpi" type="xsd:int" use="optional" default="600"/>
1477   <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
1478 </xsd:complexType>
1479 <xsd:complexType name="CT_PrintSettings">
1480   <xsd:sequence>
1481     <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1482     <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
1483     <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
1484     <xsd:element name="legacyDrawingHF" type="CT_RelId" minOccurs="0" maxOccurs="1"/>
1485   </xsd:sequence>
1486 </xsd:complexType>
1487 <xsd:complexType name="CT_ChartSpace">
1488   <xsd:sequence>
1489     <xsd:element name="date1904" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1490     <xsd:element name="lang" type="CT_TextLanguageID" minOccurs="0" maxOccurs="1"/>
1491     <xsd:element name="roundedCorners" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1492     <xsd:element name="style" type="CT_Style" minOccurs="0" maxOccurs="1"/>
1493     <xsd:element name="clrMapOvr" type="a:CT_ColorMapping" minOccurs="0" maxOccurs="1"/>
1494     <xsd:element name="pivotSource" type="CT_PivotSource" minOccurs="0" maxOccurs="1"/>
1495     <xsd:element name="protection" type="CT_Protection" minOccurs="0" maxOccurs="1"/>
1496     <xsd:element name="chart" type="CT_Chart" minOccurs="1" maxOccurs="1"/>
1497     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1498     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1499     <xsd:element name="externalData" type="CT_ExternalData" minOccurs="0" maxOccurs="1"/>
1500     <xsd:element name="printSettings" type="CT_PrintSettings" minOccurs="0" maxOccurs="1"/>
1501     <xsd:element name="userShapes" type="CT_RelId" minOccurs="0" maxOccurs="1"/>
1502     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1503   </xsd:sequence>
1504 </xsd:complexType>
1505 <xsd:element name="chartSpace" type="CT_ChartSpace"/>
1506 <xsd:element name="userShapes" type="cdr:CT_Drawing"/>
1507 <xsd:element name="chart" type="CT_RelId"/>
1508 </xsd:schema>

```

A.5.2 DrawingML - Chart Drawings

This schema is available in the file dml-chartDrawing.xsd.


```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
3   xmlns="http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
4   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
5   elementFormDefault="qualified">
6   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
7     main.xsd"/>
8   <xsd:complexType name="CT_ShapeNonVisual">
9     <xsd:sequence>
10       <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
11       <xsd:element name="cNvSpPr" type="a:CT_NonVisualDrawingShapeProps" minOccurs="1"
12         maxOccurs="1"/>
13     </xsd:sequence>
14   </xsd:complexType>
15   <xsd:complexType name="CT_Shape">
16     <xsd:sequence>
17       <xsd:element name="nvSpPr" type="CT_ShapeNonVisual" minOccurs="1" maxOccurs="1"/>
18       <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
19       <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
20       <xsd:element name="txBody" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
21     </xsd:sequence>
22     <xsd:attribute name="macro" type="xsd:string" use="optional"/>
23     <xsd:attribute name="textlink" type="xsd:string" use="optional"/>
24     <xsd:attribute name="fLocksText" type="xsd:boolean" use="optional" default="true"/>
25     <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
26   </xsd:complexType>
27   <xsd:complexType name="CT_ConnectorNonVisual">
28     <xsd:sequence>
29       <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
30       <xsd:element name="cNvCxnSpPr" type="a:CT_NonVisualConnectorProperties" minOccurs="1"
31         maxOccurs="1"/>
32     </xsd:sequence>
33   </xsd:complexType>
34   <xsd:complexType name="CT_Connector">
35     <xsd:sequence>
36       <xsd:element name="nvCxnSpPr" type="CT_ConnectorNonVisual" minOccurs="1" maxOccurs="1"/>
37       <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
38       <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
39     </xsd:sequence>
40     <xsd:attribute name="macro" type="xsd:string" use="optional"/>
41     <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
42   </xsd:complexType>
43   <xsd:complexType name="CT_PictureNonVisual">
44     <xsd:sequence>
45       <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
46       <xsd:element name="cNvPicPr" type="a:CT_NonVisualPictureProperties" minOccurs="1"
47         maxOccurs="1"/>
48     </xsd:sequence>
49   </xsd:complexType>
50   <xsd:complexType name="CT_Picture">
51     <xsd:sequence>
52       <xsd:element name="nvPicPr" type="CT_PictureNonVisual" minOccurs="1" maxOccurs="1"/>
53       <xsd:element name="blipFill" type="a:CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>

```

```

54      <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
55      <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
56    </xsd:sequence>
57    <xsd:attribute name="macro" type="xsd:string" use="optional" default=""/>
58    <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
59  </xsd:complexType>
60  <xsd:complexType name="CT_GraphicFrameNonVisual">
61    <xsd:sequence>
62      <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
63      <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
64        minOccurs="1" maxOccurs="1"/>
65    </xsd:sequence>
66  </xsd:complexType>
67  <xsd:complexType name="CT_GraphicFrame">
68    <xsd:sequence>
69      <xsd:element name="nvGraphicFramePr" type="CT_GraphicFrameNonVisual" minOccurs="1"
70        maxOccurs="1"/>
71      <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="1" maxOccurs="1"/>
72      <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
73    </xsd:sequence>
74    <xsd:attribute name="macro" type="xsd:string" use="optional"/>
75    <xsd:attribute name="fPublished" type="xsd:boolean" use="optional" default="false"/>
76  </xsd:complexType>
77  <xsd:complexType name="CT_GroupShapeNonVisual">
78    <xsd:sequence>
79      <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
80      <xsd:element name="cNvGrpSpPr" type="a:CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
81        maxOccurs="1"/>
82    </xsd:sequence>
83  </xsd:complexType>
84  <xsd:complexType name="CT_GroupShape">
85    <xsd:sequence>
86      <xsd:element name="nvGrpSpPr" type="CT_GroupShapeNonVisual" minOccurs="1" maxOccurs="1"/>
87      <xsd:element name="grpSpPr" type="a:CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
88      <xsd:choice minOccurs="0" maxOccurs="unbounded">
89        <xsd:element name="sp" type="CT_Shape"/>
90        <xsd:element name="grpSp" type="CT_GroupShape"/>
91        <xsd:element name="graphicFrame" type="CT_GraphicFrame"/>
92        <xsd:element name="cxnSp" type="CT_Connector"/>
93        <xsd:element name="pic" type="CT_Picture"/>
94      </xsd:choice>
95    </xsd:sequence>
96  </xsd:complexType>
97  <xsd:group name="EG_ObjectChoices">
98    <xsd:sequence>
99      <xsd:choice minOccurs="1" maxOccurs="1">
100        <xsd:element name="sp" type="CT_Shape"/>
101        <xsd:element name="grpSp" type="CT_GroupShape"/>
102        <xsd:element name="graphicFrame" type="CT_GraphicFrame"/>
103        <xsd:element name="cxnSp" type="CT_Connector"/>
104        <xsd:element name="pic" type="CT_Picture"/>
105      </xsd:choice>
106    </xsd:sequence>

```

```

107 </xsd:group>
108 <xsd:simpleType name="ST_MarkerCoordinate">
109   <xsd:restriction base="xsd:double">
110     <xsd:minInclusive value="0.0"/>
111     <xsd:maxInclusive value="1.0"/>
112   </xsd:restriction>
113 </xsd:simpleType>
114 <xsd:complexType name="CT_Marker">
115   <xsd:sequence>
116     <xsd:element name="x" type="ST_MarkerCoordinate" minOccurs="1" maxOccurs="1"/>
117     <xsd:element name="y" type="ST_MarkerCoordinate" minOccurs="1" maxOccurs="1"/>
118   </xsd:sequence>
119 </xsd:complexType>
120 <xsd:complexType name="CT_RelSizeAnchor">
121   <xsd:sequence>
122     <xsd:element name="from" type="CT_Marker"/>
123     <xsd:element name="to" type="CT_Marker"/>
124     <xsd:group ref="EG_ObjectChoices"/>
125   </xsd:sequence>
126 </xsd:complexType>
127 <xsd:complexType name="CT_AbsSizeAnchor">
128   <xsd:sequence>
129     <xsd:element name="from" type="CT_Marker"/>
130     <xsd:element name="ext" type="a:CT_PositiveSize2D"/>
131     <xsd:group ref="EG_ObjectChoices"/>
132   </xsd:sequence>
133 </xsd:complexType>
134 <xsd:group name="EG_Anchor">
135   <xsd:choice>
136     <xsd:element name="relSizeAnchor" type="CT_RelSizeAnchor"/>
137     <xsd:element name="absSizeAnchor" type="CT_AbsSizeAnchor"/>
138   </xsd:choice>
139 </xsd:group>
140 <xsd:complexType name="CT_Drawing">
141   <xsd:sequence>
142     <xsd:group ref="EG_Anchor" minOccurs="0" maxOccurs="unbounded"/>
143   </xsd:sequence>
144 </xsd:complexType>
145 </xsd:schema>

```

A.5.3 DrawingML - Diagrams

This schema is available in the file dml-diagram.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/drawingml/2006/diagram"
3   xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main"
4   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
5   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6   targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/diagram"
7   elementFormDefault="qualified" attributeFormDefault="unqualified">
8   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9     schemaLocation="shared-relationshipReference.xsd"/>

```

```

10 <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
11   main.xsd"/>
12 <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
13   schemaLocation="shared-commonSimpleTypes.xsd"/>
14 <xsd:complexType name="CT_CTName">
15   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
16   <xsd:attribute name="val" type="xsd:string" use="required"/>
17 </xsd:complexType>
18 <xsd:complexType name="CT_CTDescription">
19   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
20   <xsd:attribute name="val" type="xsd:string" use="required"/>
21 </xsd:complexType>
22 <xsd:complexType name="CT_CTCategory">
23   <xsd:attribute name="type" type="xsd:anyURI" use="required"/>
24   <xsd:attribute name="pri" type="xsd:unsignedInt" use="required"/>
25 </xsd:complexType>
26 <xsd:complexType name="CT_CTCategories">
27   <xsd:sequence minOccurs="0" maxOccurs="unbounded">
28     <xsd:element name="cat" type="CT_CTCategory" minOccurs="0" maxOccurs="unbounded"/>
29   </xsd:sequence>
30 </xsd:complexType>
31 <xsd:simpleType name="ST_ClrAppMethod">
32   <xsd:restriction base="xsd:token">
33     <xsd:enumeration value="span"/>
34     <xsd:enumeration value="cycle"/>
35     <xsd:enumeration value="repeat"/>
36   </xsd:restriction>
37 </xsd:simpleType>
38 <xsd:simpleType name="ST_HueDir">
39   <xsd:restriction base="xsd:token">
40     <xsd:enumeration value="cw"/>
41     <xsd:enumeration value="ccw"/>
42   </xsd:restriction>
43 </xsd:simpleType>
44 <xsd:complexType name="CT_Colors">
45   <xsd:sequence>
46     <xsd:group ref="a:EG_ColorChoice" minOccurs="0" maxOccurs="unbounded"/>
47   </xsd:sequence>
48   <xsd:attribute name="meth" type="ST_ClrAppMethod" use="optional" default="span"/>
49   <xsd:attribute name="hueDir" type="ST_HueDir" use="optional" default="cw"/>
50 </xsd:complexType>
51 <xsd:complexType name="CT_CTStyleLabel">
52   <xsd:sequence>
53     <xsd:element name="fillClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
54     <xsd:element name="linClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
55     <xsd:element name="effectClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
56     <xsd:element name="txLinClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
57     <xsd:element name="txFillClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
58     <xsd:element name="txEffectClrLst" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
59     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
60       maxOccurs="1"/>
61   </xsd:sequence>
62   <xsd:attribute name="name" type="xsd:string" use="required"/>

```

```

63 </xsd:complexType>
64 <xsd:complexType name="CT_ColorTransform">
65   <xsd:sequence>
66     <xsd:element name="title" type="CT_CTName" minOccurs="0" maxOccurs="unbounded"/>
67     <xsd:element name="desc" type="CT_CTDescription" minOccurs="0" maxOccurs="unbounded"/>
68     <xsd:element name="catLst" type="CT_CTCategories" minOccurs="0"/>
69     <xsd:element name="styleLbl" type="CT_CTStyleLabel" minOccurs="0" maxOccurs="unbounded"/>
70     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
71       maxOccurs="1"/>
72   </xsd:sequence>
73   <xsd:attribute name="uniqueId" type="xsd:string" use="optional" default=""/>
74   <xsd:attribute name="minVer" type="xsd:string" use="optional" />
75 </xsd:complexType>
76 <xsd:element name="colorsDef" type="CT_ColorTransform"/>
77 <xsd:complexType name="CT_ColorTransformHeader">
78   <xsd:sequence>
79     <xsd:element name="title" type="CT_CTName" minOccurs="1" maxOccurs="unbounded"/>
80     <xsd:element name="desc" type="CT_CTDescription" minOccurs="1" maxOccurs="unbounded"/>
81     <xsd:element name="catLst" type="CT_CTCategories" minOccurs="0"/>
82     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
83       maxOccurs="1"/>
84   </xsd:sequence>
85   <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
86   <xsd:attribute name="minVer" type="xsd:string" use="optional" />
87   <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
88 </xsd:complexType>
89 <xsd:element name="colorsDefHdr" type="CT_ColorTransformHeader"/>
90 <xsd:complexType name="CT_ColorTransformHeaderLst">
91   <xsd:sequence>
92     <xsd:element name="colorsDefHdr" type="CT_ColorTransformHeader" minOccurs="0"
93       maxOccurs="unbounded"/>
94   </xsd:sequence>
95 </xsd:complexType>
96 <xsd:element name="colorsDefHdrLst" type="CT_ColorTransformHeaderLst"/>
97 <xsd:simpleType name="ST_PtType">
98   <xsd:restriction base="xsd:token">
99     <xsd:enumeration value="node"/>
100    <xsd:enumeration value="asst"/>
101    <xsd:enumeration value="doc"/>
102    <xsd:enumeration value="pres"/>
103    <xsd:enumeration value="parTrans"/>
104    <xsd:enumeration value="sibTrans"/>
105  </xsd:restriction>
106 </xsd:simpleType>
107 <xsd:complexType name="CT_Pt">
108   <xsd:sequence>
109     <xsd:element name="prSet" type="CT_ElemPropSet" minOccurs="0" maxOccurs="1"/>
110     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
111     <xsd:element name="t" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
112     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
113       maxOccurs="1"/>
114   </xsd:sequence>
115   <xsd:attribute name="modelId" type="ST_ModelId" use="required"/>

```

```

116     <xsd:attribute name="type" type="ST PtType" use="optional" default="node"/>
117     <xsd:attribute name="cxnId" type="ST ModelId" use="optional" default="0"/>
118 </xsd:complexType>
119 <xsd:complexType name="CT_PtList">
120     <xsd:sequence>
121         <xsd:element name="pt" type="CT Pt" minOccurs="0" maxOccurs="unbounded"/>
122     </xsd:sequence>
123 </xsd:complexType>
124 <xsd:simpleType name="ST_CxnType">
125     <xsd:restriction base="xsd:token">
126         <xsd:enumeration value="parOf"/>
127         <xsd:enumeration value="presOf"/>
128         <xsd:enumeration value="presParOf"/>
129         <xsd:enumeration value="unknownRelationship"/>
130     </xsd:restriction>
131 </xsd:simpleType>
132 <xsd:complexType name="CT_Cxn">
133     <xsd:sequence>
134         <xsd:element name="extLst" type="a:CT OfficeArtExtensionList" minOccurs="0"
135             maxOccurs="1"/>
136     </xsd:sequence>
137     <xsd:attribute name="modelId" type="ST ModelId" use="required"/>
138     <xsd:attribute name="type" type="ST CxnType" use="optional" default="parOf"/>
139     <xsd:attribute name="srcId" type="ST ModelId" use="required"/>
140     <xsd:attribute name="destId" type="ST ModelId" use="required"/>
141     <xsd:attribute name="srcOrd" type="xsd:unsignedInt" use="required"/>
142     <xsd:attribute name="destOrd" type="xsd:unsignedInt" use="required"/>
143     <xsd:attribute name="parTransId" type="ST ModelId" use="optional" default="0"/>
144     <xsd:attribute name="sibTransId" type="ST ModelId" use="optional" default="0"/>
145     <xsd:attribute name="presId" type="xsd:string" use="optional" default=""/>
146 </xsd:complexType>
147 <xsd:complexType name="CT_CxnList">
148     <xsd:sequence>
149         <xsd:element name="cxn" type="CT Cxn" minOccurs="0" maxOccurs="unbounded"/>
150     </xsd:sequence>
151 </xsd:complexType>
152 <xsd:complexType name="CT_DataModel">
153     <xsd:sequence>
154         <xsd:element name="ptLst" type="CT PtList"/>
155         <xsd:element name="cxnLst" type="CT CxnList" minOccurs="0" maxOccurs="1"/>
156         <xsd:element name="bg" type="a:CT BackgroundFormatting" minOccurs="0"/>
157         <xsd:element name="whole" type="a:CT WholeE2oFormatting" minOccurs="0"/>
158         <xsd:element name="extLst" type="a:CT OfficeArtExtensionList" minOccurs="0"
159             maxOccurs="1"/>
160     </xsd:sequence>
161 </xsd:complexType>
162 <xsd:element name="dataModel" type="CT_DataModel"/>
163 <xsd:attributeGroup name="AG_IteratorAttributes">
164     <xsd:attribute name="axis" type="ST AxisTypes" use="optional" default="none"/>
165     <xsd:attribute name="ptType" type="ST ElementTypes" use="optional" default="all"/>
166     <xsd:attribute name="hideLastTrans" type="ST Booleans" use="optional" default="true"/>
167     <xsd:attribute name="st" type="ST Ints" use="optional" default="1"/>
168     <xsd:attribute name="cnt" type="ST UnsignedInts" use="optional" default="0"/>

```

```

169     <xsd:attribute name="step" type="ST_Ints" use="optional" default="1"/>
170 </xsd:attributeGroup>
171 <xsd:attributeGroup name="AG_ConstraintAttributes">
172     <xsd:attribute name="type" type="ST_ConstraintType" use="required"/>
173     <xsd:attribute name="for" type="ST_ConstraintRelationship" use="optional" default="self"/>
174     <xsd:attribute name="forName" type="xsd:string" use="optional" default=""/>
175     <xsd:attribute name="ptType" type="ST_ElementType" use="optional" default="all"/>
176 </xsd:attributeGroup>
177 <xsd:attributeGroup name="AG_ConstraintRefAttributes">
178     <xsd:attribute name="refType" type="ST_ConstraintType" use="optional" default="none"/>
179     <xsd:attribute name="refFor" type="ST_ConstraintRelationship" use="optional" default="self"/>
180     <xsd:attribute name="refForName" type="xsd:string" use="optional" default=""/>
181     <xsd:attribute name="refPtType" type="ST_ElementType" use="optional" default="all"/>
182 </xsd:attributeGroup>
183 <xsd:complexType name="CT_Constraint">
184     <xsd:sequence>
185         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
186             maxOccurs="1"/>
187     </xsd:sequence>
188     <xsd:attributeGroup ref="AG_ConstraintAttributes"/>
189     <xsd:attributeGroup ref="AG_ConstraintRefAttributes"/>
190     <xsd:attribute name="op" type="ST_BoolOperator" use="optional" default="none"/>
191     <xsd:attribute name="val" type="xsd:double" use="optional" default="0"/>
192     <xsd:attribute name="fact" type="xsd:double" use="optional" default="1"/>
193 </xsd:complexType>
194 <xsd:complexType name="CT_Constraints">
195     <xsd:sequence>
196         <xsd:element name="constr" type="CT_Constraint" minOccurs="0" maxOccurs="unbounded"/>
197     </xsd:sequence>
198 </xsd:complexType>
199 <xsd:complexType name="CT_NumericRule">
200     <xsd:sequence>
201         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
202             maxOccurs="1"/>
203     </xsd:sequence>
204     <xsd:attributeGroup ref="AG_ConstraintAttributes"/>
205     <xsd:attribute name="val" type="xsd:double" use="optional" default="NaN"/>
206     <xsd:attribute name="fact" type="xsd:double" use="optional" default="NaN"/>
207     <xsd:attribute name="max" type="xsd:double" use="optional" default="NaN"/>
208 </xsd:complexType>
209 <xsd:complexType name="CT_Rules">
210     <xsd:sequence>
211         <xsd:element name="rule" type="CT_NumericRule" minOccurs="0" maxOccurs="unbounded"/>
212     </xsd:sequence>
213 </xsd:complexType>
214 <xsd:complexType name="CT_PresentationOf">
215     <xsd:sequence>
216         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
217             maxOccurs="1"/>
218     </xsd:sequence>
219     <xsd:attributeGroup ref="AG_IteratorAttributes"/>
220 </xsd:complexType>
221 <xsd:simpleType name="ST_LayoutShapeType" final="restriction">

```

```

222     <xsd:union memberTypes="a:ST_ShapeType ST_OutputShapeType"/>
223   </xsd:simpleType>
224   <xsd:simpleType name="ST_Index1">
225     <xsd:restriction base="xsd:unsignedInt">
226       <xsd:minInclusive value="1"/>
227     </xsd:restriction>
228   </xsd:simpleType>
229   <xsd:complexType name="CT_Adj">
230     <xsd:attribute name="idx" type="ST_Index1" use="required"/>
231     <xsd:attribute name="val" type="xsd:double" use="required"/>
232   </xsd:complexType>
233   <xsd:complexType name="CT_AdjLst">
234     <xsd:sequence>
235       <xsd:element name="adj" type="CT_Adj" minOccurs="0" maxOccurs="unbounded"/>
236     </xsd:sequence>
237   </xsd:complexType>
238   <xsd:complexType name="CT_Shape">
239     <xsd:sequence>
240       <xsd:element name="adjLst" type="CT_AdjLst" minOccurs="0" maxOccurs="1"/>
241       <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
242         maxOccurs="1"/>
243     </xsd:sequence>
244     <xsd:attribute name="rot" type="xsd:double" use="optional" default="0"/>
245     <xsd:attribute name="type" type="ST_LayoutShapeType" use="optional" default="none"/>
246     <xsd:attribute ref="r:blip" use="optional"/>
247     <xsd:attribute name="zOrderOff" type="xsd:int" use="optional" default="0"/>
248     <xsd:attribute name="hideGeom" type="xsd:boolean" use="optional" default="false"/>
249     <xsd:attribute name="lkTxEntry" type="xsd:boolean" use="optional" default="false"/>
250     <xsd:attribute name="blipPhldr" type="xsd:boolean" use="optional" default="false"/>
251   </xsd:complexType>
252   <xsd:complexType name="CT_Parameter">
253     <xsd:attribute name="type" type="ST_ParameterId" use="required"/>
254     <xsd:attribute name="val" type="ST_ParameterVal" use="required"/>
255   </xsd:complexType>
256   <xsd:complexType name="CT_Algorithm">
257     <xsd:sequence>
258       <xsd:element name="param" type="CT_Parameter" minOccurs="0" maxOccurs="unbounded"/>
259       <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
260         maxOccurs="1"/>
261     </xsd:sequence>
262     <xsd:attribute name="type" type="ST_AlgorithmType" use="required"/>
263     <xsd:attribute name="rev" type="xsd:unsignedInt" use="optional" default="0"/>
264   </xsd:complexType>
265   <xsd:complexType name="CT_LayoutNode">
266     <xsd:choice minOccurs="0" maxOccurs="unbounded">
267       <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
268       <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
269       <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
270       <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
271       <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
272       <xsd:element name="varLst" type="CT_LayoutVariablePropertySet" minOccurs="0"
273         maxOccurs="1"/>
274       <xsd:element name="forEach" type="CT_ForEach"/>

```



```

275     <xsd:element name="layoutNode" type="CT_LayoutNode"/>
276     <xsd:element name="choose" type="CT_Choose"/>
277     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
278         maxOccurs="1"/>
279 </xsd:choice>
280 <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
281 <xsd:attribute name="styleLbl" type="xsd:string" use="optional" default=""/>
282 <xsd:attribute name="chOrder" type="ST_ChildOrderType" use="optional" default="b"/>
283 <xsd:attribute name="moveWith" type="xsd:string" use="optional" default=""/>
284 </xsd:complexType>
285 <xsd:complexType name="CT_ForEach">
286     <xsd:choice minOccurs="0" maxOccurs="unbounded">
287         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
288         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
289         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
290         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
291         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
292         <xsd:element name="forEach" type="CT_ForEach"/>
293         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
294         <xsd:element name="choose" type="CT_Choose"/>
295         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
296             maxOccurs="1"/>
297     </xsd:choice>
298     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
299     <xsd:attribute name="ref" type="xsd:string" use="optional" default=""/>
300     <xsd:attributeGroup ref="AG_IteratorAttributes"/>
301 </xsd:complexType>
302 <xsd:complexType name="CT_When">
303     <xsd:choice minOccurs="0" maxOccurs="unbounded">
304         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
305         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
306         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
307         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
308         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
309         <xsd:element name="forEach" type="CT_ForEach"/>
310         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
311         <xsd:element name="choose" type="CT_Choose"/>
312         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
313             maxOccurs="1"/>
314     </xsd:choice>
315     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
316     <xsd:attributeGroup ref="AG_IteratorAttributes"/>
317     <xsd:attribute name="func" type="ST_FunctionType" use="required"/>
318     <xsd:attribute name="arg" type="ST_FunctionArgument" use="optional" default="none"/>
319     <xsd:attribute name="op" type="ST_FunctionOperator" use="required"/>
320     <xsd:attribute name="val" type="ST_FunctionValue" use="required"/>
321 </xsd:complexType>
322 <xsd:complexType name="CT_Otherwise">
323     <xsd:choice minOccurs="0" maxOccurs="unbounded">
324         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
325         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
326         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
327         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>

```

```

328     <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
329     <xsd:element name="forEach" type="CT_ForEach"/>
330     <xsd:element name="layoutNode" type="CT_LayoutNode"/>
331     <xsd:element name="choose" type="CT_Choose"/>
332     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
333         maxOccurs="1"/>
334 </xsd:choice>
335     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
336 </xsd:complexType>
337 <xsd:complexType name="CT_Choose">
338     <xsd:sequence>
339         <xsd:element name="if" type="CT_When" maxOccurs="unbounded"/>
340         <xsd:element name="else" type="CT_Otherwise" minOccurs="0"/>
341     </xsd:sequence>
342     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
343 </xsd:complexType>
344 <xsd:complexType name="CT_SampleData">
345     <xsd:sequence>
346         <xsd:element name="dataModel" type="CT_DataModel" minOccurs="0"/>
347     </xsd:sequence>
348     <xsd:attribute name="useDef" type="xsd:boolean" use="optional" default="false"/>
349 </xsd:complexType>
350 <xsd:complexType name="CT_Category">
351     <xsd:attribute name="type" type="xsd:anyURI" use="required"/>
352     <xsd:attribute name="pri" type="xsd:unsignedInt" use="required"/>
353 </xsd:complexType>
354 <xsd:complexType name="CT_Categories">
355     <xsd:sequence>
356         <xsd:element name="cat" type="CT_Category" minOccurs="0" maxOccurs="unbounded"/>
357     </xsd:sequence>
358 </xsd:complexType>
359 <xsd:complexType name="CT_Name">
360     <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
361     <xsd:attribute name="val" type="xsd:string" use="required"/>
362 </xsd:complexType>
363 <xsd:complexType name="CT_Description">
364     <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
365     <xsd:attribute name="val" type="xsd:string" use="required"/>
366 </xsd:complexType>
367 <xsd:complexType name="CT_DiagramDefinition">
368     <xsd:sequence>
369         <xsd:element name="title" type="CT_Name" minOccurs="0" maxOccurs="unbounded"/>
370         <xsd:element name="desc" type="CT_Description" minOccurs="0" maxOccurs="unbounded"/>
371         <xsd:element name="catLst" type="CT_Categories" minOccurs="0"/>
372         <xsd:element name="sampData" type="CT_SampleData" minOccurs="0"/>
373         <xsd:element name="styleData" type="CT_SampleData" minOccurs="0"/>
374         <xsd:element name="clrData" type="CT_SampleData" minOccurs="0"/>
375         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
376         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
377             maxOccurs="1"/>
378     </xsd:sequence>
379     <xsd:attribute name="uniqueId" type="xsd:string" use="optional" default=""/>
380     <xsd:attribute name="minVer" type="xsd:string" use="optional" />

```

```

381     <xsd:attribute name="defStyle" type="xsd:string" use="optional" default=""/>
382 </xsd:complexType>
383 <xsd:element name="layoutDef" type="CT_DiagramDefinition"/>
384 <xsd:complexType name="CT_DiagramDefinitionHeader">
385     <xsd:sequence>
386         <xsd:element name="title" type="CT_Name" minOccurs="1" maxOccurs="unbounded"/>
387         <xsd:element name="desc" type="CT_Description" minOccurs="1" maxOccurs="unbounded"/>
388         <xsd:element name="catLst" type="CT_Categories" minOccurs="0"/>
389         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
390             maxOccurs="1"/>
391     </xsd:sequence>
392     <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
393     <xsd:attribute name="minVer" type="xsd:string" use="optional" />
394     <xsd:attribute name="defStyle" type="xsd:string" use="optional" default=""/>
395     <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
396 </xsd:complexType>
397 <xsd:element name="layoutDefHdr" type="CT_DiagramDefinitionHeader"/>
398 <xsd:complexType name="CT_DiagramDefinitionHeaderLst">
399     <xsd:sequence>
400         <xsd:element name="layoutDefHdr" type="CT_DiagramDefinitionHeader" minOccurs="0"
401             maxOccurs="unbounded"/>
402     </xsd:sequence>
403 </xsd:complexType>
404 <xsd:element name="layoutDefHdrLst" type="CT_DiagramDefinitionHeaderLst"/>
405 <xsd:complexType name="CT_RelIds">
406     <xsd:attribute ref="r:dm" use="required"/>
407     <xsd:attribute ref="r:lo" use="required"/>
408     <xsd:attribute ref="r:qs" use="required"/>
409     <xsd:attribute ref="r:cs" use="required"/>
410 </xsd:complexType>
411 <xsd:element name="relIds" type="CT_RelIds"/>
412 <xsd:simpleType name="ST_ParameterVal">
413     <xsd:union memberTypes="ST_DiagramHorizontalAlignment ST_VerticalAlignment ST_ChildDirection
414         ST_ChildAlignment ST_SecondaryChildAlignment ST_LinearDirection ST_SecondaryLinearDirection
415         ST_StartingElement ST_BendPoint ST_ConnectorRouting ST_ArrowheadStyle ST_ConnectorDimension
416         ST_RotationPath ST_CenterShapeMapping ST_NodeHorizontalAlignment ST_NodeVerticalAlignment
417         ST_FallbackDimension ST_TextDirection ST_PyramidAccentPosition ST_PyramidAccentTextMargin
418         ST_TextBlockDirection ST_TextAnchorHorizontal ST_TextAnchorVertical ST_DiagramTextAlignment
419         ST_AutoTextRotation ST_GrowDirection ST_FlowDirection ST_ContinueDirection ST_Breakpoint
420         ST_Offset ST_HierarchyAlignment xsd:int xsd:double xsd:boolean xsd:string
421         ST_ConnectorPoint"/>
422 </xsd:simpleType>
423 <xsd:simpleType name="ST_ModelId">
424     <xsd:union memberTypes="xsd:int s:ST_Guid"/>
425 </xsd:simpleType>
426 <xsd:simpleType name="ST_PrSetCustVal">
427     <xsd:union memberTypes="s:ST_Percentage xsd:int"/>
428 </xsd:simpleType>
429 <xsd:complexType name="CT_ElemPropSet">
430     <xsd:sequence>
431         <xsd:element name="presLayoutVars" type="CT_LayoutVariablePropertySet" minOccurs="0"
432             maxOccurs="1"/>
433         <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>

```

```

434     </xsd:sequence>
435     <xsd:attribute name="presAssocID" type="ST_ModelId" use="optional"/>
436     <xsd:attribute name="presName" type="xsd:string" use="optional"/>
437     <xsd:attribute name="presStyleLbl" type="xsd:string" use="optional"/>
438     <xsd:attribute name="presStyleIdx" type="xsd:int" use="optional"/>
439     <xsd:attribute name="presStyleCnt" type="xsd:int" use="optional"/>
440     <xsd:attribute name="loTypeId" type="xsd:string" use="optional"/>
441     <xsd:attribute name="loCatId" type="xsd:string" use="optional"/>
442     <xsd:attribute name="qsTypeId" type="xsd:string" use="optional"/>
443     <xsd:attribute name="qsCatId" type="xsd:string" use="optional"/>
444     <xsd:attribute name="csTypeId" type="xsd:string" use="optional"/>
445     <xsd:attribute name="csCatId" type="xsd:string" use="optional"/>
446     <xsd:attribute name="coherent3DOff" type="xsd:boolean" use="optional"/>
447     <xsd:attribute name="phldrT" type="xsd:string" use="optional"/>
448     <xsd:attribute name="phldr" type="xsd:boolean" use="optional"/>
449     <xsd:attribute name="custAng" type="xsd:int" use="optional"/>
450     <xsd:attribute name="custFlipVert" type="xsd:boolean" use="optional"/>
451     <xsd:attribute name="custFlipHor" type="xsd:boolean" use="optional"/>
452     <xsd:attribute name="custSzX" type="xsd:int" use="optional"/>
453     <xsd:attribute name="custSzY" type="xsd:int" use="optional"/>
454     <xsd:attribute name="custScaleX" type="ST_PrSetCustVal" use="optional"/>
455     <xsd:attribute name="custScaleY" type="ST_PrSetCustVal" use="optional"/>
456     <xsd:attribute name="custT" type="xsd:boolean" use="optional"/>
457     <xsd:attribute name="custLinFactX" type="ST_PrSetCustVal" use="optional"/>
458     <xsd:attribute name="custLinFactY" type="ST_PrSetCustVal" use="optional"/>
459     <xsd:attribute name="custLinFactNeighborX" type="ST_PrSetCustVal" use="optional"/>
460     <xsd:attribute name="custLinFactNeighborY" type="ST_PrSetCustVal" use="optional"/>
461     <xsd:attribute name="custRadScaleRad" type="ST_PrSetCustVal" use="optional"/>
462     <xsd:attribute name="custRadScaleInc" type="ST_PrSetCustVal" use="optional"/>
463 </xsd:complexType>
464 <xsd:simpleType name="ST_Direction" final="restriction">
465     <xsd:restriction base="xsd:token">
466         <xsd:enumeration value="norm"/>
467         <xsd:enumeration value="rev"/>
468     </xsd:restriction>
469 </xsd:simpleType>
470 <xsd:simpleType name="ST_HierBranchStyle" final="restriction">
471     <xsd:restriction base="xsd:token">
472         <xsd:enumeration value="l"/>
473         <xsd:enumeration value="r"/>
474         <xsd:enumeration value="hang"/>
475         <xsd:enumeration value="std"/>
476         <xsd:enumeration value="init"/>
477     </xsd:restriction>
478 </xsd:simpleType>
479 <xsd:simpleType name="ST_AnimOneStr" final="restriction">
480     <xsd:restriction base="xsd:token">
481         <xsd:enumeration value="none"/>
482         <xsd:enumeration value="one"/>
483         <xsd:enumeration value="branch"/>
484     </xsd:restriction>
485 </xsd:simpleType>
486 <xsd:simpleType name="ST_AnimLvlStr" final="restriction">

```

```

487     <xsd:restriction base="xsd:token">
488         <xsd:enumeration value="none"/>
489         <xsd:enumeration value="lvl"/>
490         <xsd:enumeration value="ctr"/>
491     </xsd:restriction>
492 </xsd:simpleType>
493 <xsd:complexType name="CT_OrgChart">
494     <xsd:attribute name="val" type="xsd:boolean" default="false" use="optional"/>
495 </xsd:complexType>
496 <xsd:simpleType name="ST_NodeCount">
497     <xsd:restriction base="xsd:int">
498         <xsd:minInclusive value="-1"/>
499     </xsd:restriction>
500 </xsd:simpleType>
501 <xsd:complexType name="CT_ChildMax">
502     <xsd:attribute name="val" type="ST_NodeCount" default="-1" use="optional"/>
503 </xsd:complexType>
504 <xsd:complexType name="CT_ChildPref">
505     <xsd:attribute name="val" type="ST_NodeCount" default="-1" use="optional"/>
506 </xsd:complexType>
507 <xsd:complexType name="CT_BulletEnabled">
508     <xsd:attribute name="val" type="xsd:boolean" default="false" use="optional"/>
509 </xsd:complexType>
510 <xsd:complexType name="CT_Direction">
511     <xsd:attribute name="val" type="ST_Direction" default="norm" use="optional"/>
512 </xsd:complexType>
513 <xsd:complexType name="CT_HierBranchStyle">
514     <xsd:attribute name="val" type="ST_HierBranchStyle" default="std" use="optional"/>
515 </xsd:complexType>
516 <xsd:complexType name="CT_AnimOne">
517     <xsd:attribute name="val" type="ST_AnimOneStr" default="one" use="optional"/>
518 </xsd:complexType>
519 <xsd:complexType name="CT_AnimLvl">
520     <xsd:attribute name="val" type="ST_AnimLvlStr" default="none" use="optional"/>
521 </xsd:complexType>
522 <xsd:simpleType name="ST_ResizeHandlesStr" final="restriction">
523     <xsd:restriction base="xsd:token">
524         <xsd:enumeration value="exact"/>
525         <xsd:enumeration value="rel"/>
526     </xsd:restriction>
527 </xsd:simpleType>
528 <xsd:complexType name="CT_ResizeHandles">
529     <xsd:attribute name="val" type="ST_ResizeHandlesStr" default="rel" use="optional"/>
530 </xsd:complexType>
531 <xsd:complexType name="CT_LayoutVariablePropertySet">
532     <xsd:sequence>
533         <xsd:element name="orgChart" type="CT_OrgChart" minOccurs="0" maxOccurs="1"/>
534         <xsd:element name="chMax" type="CT_ChildMax" minOccurs="0" maxOccurs="1"/>
535         <xsd:element name="chPref" type="CT_ChildPref" minOccurs="0" maxOccurs="1"/>
536         <xsd:element name="bulletEnabled" type="CT_BulletEnabled" minOccurs="0" maxOccurs="1"/>
537         <xsd:element name="dir" type="CT_Direction" minOccurs="0" maxOccurs="1"/>
538         <xsd:element name="hierBranch" type="CT_HierBranchStyle" minOccurs="0" maxOccurs="1"/>
539         <xsd:element name="animOne" type="CT_AnimOne" minOccurs="0" maxOccurs="1"/>

```

```

540     <xsd:element name="animLvl" type="CT_AnimLvl" minOccurs="0" maxOccurs="1"/>
541     <xsd:element name="resizeHandles" type="CT_ResizeHandles" minOccurs="0" maxOccurs="1"/>
542   </xsd:sequence>
543 </xsd:complexType>
544 <xsd:complexType name="CT_SDName">
545   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
546   <xsd:attribute name="val" type="xsd:string" use="required"/>
547 </xsd:complexType>
548 <xsd:complexType name="CT_SDDescription">
549   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
550   <xsd:attribute name="val" type="xsd:string" use="required"/>
551 </xsd:complexType>
552 <xsd:complexType name="CT_SDCategory">
553   <xsd:attribute name="type" type="xsd:anyURI" use="required"/>
554   <xsd:attribute name="pri" type="xsd:unsignedInt" use="required"/>
555 </xsd:complexType>
556 <xsd:complexType name="CT_SDCategories">
557   <xsd:sequence minOccurs="0" maxOccurs="unbounded">
558     <xsd:element name="cat" type="CT_SDCategory" minOccurs="0" maxOccurs="unbounded"/>
559   </xsd:sequence>
560 </xsd:complexType>
561 <xsd:complexType name="CT_TextProps">
562   <xsd:sequence>
563     <xsd:group ref="a:EG_Text3D" minOccurs="0" maxOccurs="1"/>
564   </xsd:sequence>
565 </xsd:complexType>
566 <xsd:complexType name="CT_StyleLabel">
567   <xsd:sequence>
568     <xsd:element name="scene3d" type="a:CT_Scene3D" minOccurs="0" maxOccurs="1"/>
569     <xsd:element name="sp3d" type="a:CT_Shape3D" minOccurs="0" maxOccurs="1"/>
570     <xsd:element name="txPr" type="CT_TextProps" minOccurs="0" maxOccurs="1"/>
571     <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
572     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
573       maxOccurs="1"/>
574   </xsd:sequence>
575   <xsd:attribute name="name" type="xsd:string" use="required"/>
576 </xsd:complexType>
577 <xsd:complexType name="CT_StyleDefinition">
578   <xsd:sequence>
579     <xsd:element name="title" type="CT_SDName" minOccurs="0" maxOccurs="unbounded"/>
580     <xsd:element name="desc" type="CT_SDDescription" minOccurs="0" maxOccurs="unbounded"/>
581     <xsd:element name="catLst" type="CT_SDCategories" minOccurs="0"/>
582     <xsd:element name="scene3d" type="a:CT_Scene3D" minOccurs="0" maxOccurs="1"/>
583     <xsd:element name="styleLbl" type="CT_StyleLabel" minOccurs="1" maxOccurs="unbounded"/>
584     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
585       maxOccurs="1"/>
586   </xsd:sequence>
587   <xsd:attribute name="uniqueId" type="xsd:string" use="optional" default=""/>
588   <xsd:attribute name="minVer" type="xsd:string" use="optional" />
589 </xsd:complexType>
590 <xsd:element name="styleDef" type="CT_StyleDefinition"/>
591 <xsd:complexType name="CT_StyleDefinitionHeader">
592   <xsd:sequence>

```

```

593     <xsd:element name="title" type="CT_SDName" minOccurs="1" maxOccurs="unbounded"/>
594     <xsd:element name="desc" type="CT_SDDescription" minOccurs="1" maxOccurs="unbounded"/>
595     <xsd:element name="catLst" type="CT_SDCategories" minOccurs="0"/>
596     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
597         maxOccurs="1"/>
598     </xsd:sequence>
599     <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
600     <xsd:attribute name="minVer" type="xsd:string" use="optional" />
601     <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
602 </xsd:complexType>
603 <xsd:element name="styleDefHdr" type="CT_StyleDefinitionHeader"/>
604 <xsd:complexType name="CT_StyleDefinitionHeaderLst">
605     <xsd:sequence>
606         <xsd:element name="styleDefHdr" type="CT_StyleDefinitionHeader" minOccurs="0"
607             maxOccurs="unbounded"/>
608     </xsd:sequence>
609 </xsd:complexType>
610 <xsd:element name="styleDefHdrLst" type="CT_StyleDefinitionHeaderLst"/>
611 <xsd:simpleType name="ST_AlgorithmType" final="restriction">
612     <xsd:restriction base="xsd:token">
613         <xsd:enumeration value="composite"/>
614         <xsd:enumeration value="conn"/>
615         <xsd:enumeration value="cycle"/>
616         <xsd:enumeration value="hierChild"/>
617         <xsd:enumeration value="hierRoot"/>
618         <xsd:enumeration value="pyra"/>
619         <xsd:enumeration value="lin"/>
620         <xsd:enumeration value="sp"/>
621         <xsd:enumeration value="tx"/>
622         <xsd:enumeration value="snake"/>
623     </xsd:restriction>
624 </xsd:simpleType>
625 <xsd:simpleType name="ST_AxisType" final="restriction">
626     <xsd:restriction base="xsd:token">
627         <xsd:enumeration value="self"/>
628         <xsd:enumeration value="ch"/>
629         <xsd:enumeration value="des"/>
630         <xsd:enumeration value="desOrSelf"/>
631         <xsd:enumeration value="par"/>
632         <xsd:enumeration value="ancst"/>
633         <xsd:enumeration value="ancstOrSelf"/>
634         <xsd:enumeration value="followSib"/>
635         <xsd:enumeration value="precedSib"/>
636         <xsd:enumeration value="follow"/>
637         <xsd:enumeration value="preced"/>
638         <xsd:enumeration value="root"/>
639         <xsd:enumeration value="none"/>
640     </xsd:restriction>
641 </xsd:simpleType>
642 <xsd:simpleType name="ST_AxisTypes">
643     <xsd:list itemType="ST_AxisType"/>
644 </xsd:simpleType>
645 <xsd:simpleType name="ST_BoolOperator" final="restriction">

```

```

646     <xsd:restriction base="xsd:token">
647         <xsd:enumeration value="none"/>
648         <xsd:enumeration value="equ"/>
649         <xsd:enumeration value="gte"/>
650         <xsd:enumeration value="lte"/>
651     </xsd:restriction>
652 </xsd:simpleType>
653 <xsd:simpleType name="ST_ChildOrderType" final="restriction">
654     <xsd:restriction base="xsd:token">
655         <xsd:enumeration value="b"/>
656         <xsd:enumeration value="t"/>
657     </xsd:restriction>
658 </xsd:simpleType>
659 <xsd:simpleType name="ST_ConstraintType" final="restriction">
660     <xsd:restriction base="xsd:token">
661         <xsd:enumeration value="none"/>
662         <xsd:enumeration value="alignOff"/>
663         <xsd:enumeration value="begMarg"/>
664         <xsd:enumeration value="bendDist"/>
665         <xsd:enumeration value="begPad"/>
666         <xsd:enumeration value="b"/>
667         <xsd:enumeration value="bMarg"/>
668         <xsd:enumeration value="bOff"/>
669         <xsd:enumeration value="ctrX"/>
670         <xsd:enumeration value="ctrXOff"/>
671         <xsd:enumeration value="ctrY"/>
672         <xsd:enumeration value="ctrYOff"/>
673         <xsd:enumeration value="connDist"/>
674         <xsd:enumeration value="diam"/>
675         <xsd:enumeration value="endMarg"/>
676         <xsd:enumeration value="endPad"/>
677         <xsd:enumeration value="h"/>
678         <xsd:enumeration value="hArH"/>
679         <xsd:enumeration value="hOff"/>
680         <xsd:enumeration value="l"/>
681         <xsd:enumeration value="lMarg"/>
682         <xsd:enumeration value="lOff"/>
683         <xsd:enumeration value="r"/>
684         <xsd:enumeration value="rMarg"/>
685         <xsd:enumeration value="rOff"/>
686         <xsd:enumeration value="primFontSz"/>
687         <xsd:enumeration value="pyraAcctRatio"/>
688         <xsd:enumeration value="secFontSz"/>
689         <xsd:enumeration value="sibSp"/>
690         <xsd:enumeration value="secSibSp"/>
691         <xsd:enumeration value="sp"/>
692         <xsd:enumeration value="stemThick"/>
693         <xsd:enumeration value="t"/>
694         <xsd:enumeration value="tMarg"/>
695         <xsd:enumeration value="tOff"/>
696         <xsd:enumeration value="userA"/>
697         <xsd:enumeration value="userB"/>
698         <xsd:enumeration value="userC"/>

```



```

699         <xsd:enumeration value="userD"/>
700         <xsd:enumeration value="userE"/>
701         <xsd:enumeration value="userF"/>
702         <xsd:enumeration value="userG"/>
703         <xsd:enumeration value="userH"/>
704         <xsd:enumeration value="userI"/>
705         <xsd:enumeration value="userJ"/>
706         <xsd:enumeration value="userK"/>
707         <xsd:enumeration value="userL"/>
708         <xsd:enumeration value="userM"/>
709         <xsd:enumeration value="userN"/>
710         <xsd:enumeration value="userO"/>
711         <xsd:enumeration value="userP"/>
712         <xsd:enumeration value="userQ"/>
713         <xsd:enumeration value="userR"/>
714         <xsd:enumeration value="userS"/>
715         <xsd:enumeration value="userT"/>
716         <xsd:enumeration value="userU"/>
717         <xsd:enumeration value="userV"/>
718         <xsd:enumeration value="userW"/>
719         <xsd:enumeration value="userX"/>
720         <xsd:enumeration value="userY"/>
721         <xsd:enumeration value="userZ"/>
722         <xsd:enumeration value="w"/>
723         <xsd:enumeration value="wArH"/>
724         <xsd:enumeration value="wOff"/>
725     </xsd:restriction>
726 </xsd:simpleType>
727 <xsd:simpleType name="ST_ConstraintRelationship" final="restriction">
728     <xsd:restriction base="xsd:token">
729         <xsd:enumeration value="self"/>
730         <xsd:enumeration value="ch"/>
731         <xsd:enumeration value="des"/>
732     </xsd:restriction>
733 </xsd:simpleType>
734 <xsd:simpleType name="ST_ElementType" final="restriction">
735     <xsd:restriction base="xsd:token">
736         <xsd:enumeration value="all"/>
737         <xsd:enumeration value="doc"/>
738         <xsd:enumeration value="node"/>
739         <xsd:enumeration value="norm"/>
740         <xsd:enumeration value="nonNorm"/>
741         <xsd:enumeration value="asst"/>
742         <xsd:enumeration value="nonAsst"/>
743         <xsd:enumeration value="parTrans"/>
744         <xsd:enumeration value="pres"/>
745         <xsd:enumeration value="sibTrans"/>
746     </xsd:restriction>
747 </xsd:simpleType>
748 <xsd:simpleType name="ST_ElementTypes">
749     <xsd:list itemType="ST_ElementType"/>
750 </xsd:simpleType>
751 <xsd:simpleType name="ST_ParameterId" final="restriction">

```

```

752     <xsd:restriction base="xsd:token">
753         <xsd:enumeration value="horzAlign"/>
754         <xsd:enumeration value="vertAlign"/>
755         <xsd:enumeration value="chDir"/>
756         <xsd:enumeration value="chAlign"/>
757         <xsd:enumeration value="secChAlign"/>
758         <xsd:enumeration value="linDir"/>
759         <xsd:enumeration value="secLinDir"/>
760         <xsd:enumeration value="stElem"/>
761         <xsd:enumeration value="bendPt"/>
762         <xsd:enumeration value="connRout"/>
763         <xsd:enumeration value="begSty"/>
764         <xsd:enumeration value="endSty"/>
765         <xsd:enumeration value="dim"/>
766         <xsd:enumeration value="rotPath"/>
767         <xsd:enumeration value="ctrShpMap"/>
768         <xsd:enumeration value="nodeHorzAlign"/>
769         <xsd:enumeration value="nodeVertAlign"/>
770         <xsd:enumeration value="fallback"/>
771         <xsd:enumeration value="txDir"/>
772         <xsd:enumeration value="pyraAcctPos"/>
773         <xsd:enumeration value="pyraAcctTxMar"/>
774         <xsd:enumeration value="txBldir"/>
775         <xsd:enumeration value="txAnchorHorz"/>
776         <xsd:enumeration value="txAnchorVert"/>
777         <xsd:enumeration value="txAnchorHorzCh"/>
778         <xsd:enumeration value="txAnchorVertCh"/>
779         <xsd:enumeration value="parTxLTRAlign"/>
780         <xsd:enumeration value="parTxRTLAlign"/>
781         <xsd:enumeration value="shpTxLTRAlignCh"/>
782         <xsd:enumeration value="shpTxRTLAlignCh"/>
783         <xsd:enumeration value="autoTxRot"/>
784         <xsd:enumeration value="grDir"/>
785         <xsd:enumeration value="flowDir"/>
786         <xsd:enumeration value="contDir"/>
787         <xsd:enumeration value="bkpt"/>
788         <xsd:enumeration value="off"/>
789         <xsd:enumeration value="hierAlign"/>
790         <xsd:enumeration value="bkPtFixedVal"/>
791         <xsd:enumeration value="stBulletLvl"/>
792         <xsd:enumeration value="stAng"/>
793         <xsd:enumeration value="spanAng"/>
794         <xsd:enumeration value="ar"/>
795         <xsd:enumeration value="lnSpPar"/>
796         <xsd:enumeration value="lnSpAfParP"/>
797         <xsd:enumeration value="lnSpCh"/>
798         <xsd:enumeration value="lnSpAfChP"/>
799         <xsd:enumeration value="rtShortDist"/>
800         <xsd:enumeration value="alignTx"/>
801         <xsd:enumeration value="pyraLvlNode"/>
802         <xsd:enumeration value="pyraAcctBkgdNode"/>
803         <xsd:enumeration value="pyraAcctTxNode"/>
804         <xsd:enumeration value="srcNode"/>

```

```

805         <xsd:enumeration value="dstNode"/>
806         <xsd:enumeration value="begPts"/>
807         <xsd:enumeration value="endPts"/>
808     </xsd:restriction>
809 </xsd:simpleType>
810 <xsd:simpleType name="ST_Ints">
811     <xsd:list itemType="xsd:int"/>
812 </xsd:simpleType>
813 <xsd:simpleType name="ST_UnsignedInts">
814     <xsd:list itemType="xsd:unsignedInt"/>
815 </xsd:simpleType>
816 <xsd:simpleType name="ST_Booleans">
817     <xsd:list itemType="xsd:boolean"/>
818 </xsd:simpleType>
819 <xsd:simpleType name="ST_FunctionType" final="restriction">
820     <xsd:restriction base="xsd:token">
821         <xsd:enumeration value="cnt"/>
822         <xsd:enumeration value="pos"/>
823         <xsd:enumeration value="revPos"/>
824         <xsd:enumeration value="posEven"/>
825         <xsd:enumeration value="posOdd"/>
826         <xsd:enumeration value="var"/>
827         <xsd:enumeration value="depth"/>
828         <xsd:enumeration value="maxDepth"/>
829     </xsd:restriction>
830 </xsd:simpleType>
831 <xsd:simpleType name="ST_FunctionOperator" final="restriction">
832     <xsd:restriction base="xsd:token">
833         <xsd:enumeration value="equ"/>
834         <xsd:enumeration value="neq"/>
835         <xsd:enumeration value="gt"/>
836         <xsd:enumeration value="lt"/>
837         <xsd:enumeration value="gte"/>
838         <xsd:enumeration value="lte"/>
839     </xsd:restriction>
840 </xsd:simpleType>
841 <xsd:simpleType name="ST_DiagramHorizontalAlignment" final="restriction">
842     <xsd:restriction base="xsd:token">
843         <xsd:enumeration value="l"/>
844         <xsd:enumeration value="ctr"/>
845         <xsd:enumeration value="r"/>
846         <xsd:enumeration value="none"/>
847     </xsd:restriction>
848 </xsd:simpleType>
849 <xsd:simpleType name="ST_VerticalAlignment" final="restriction">
850     <xsd:restriction base="xsd:token">
851         <xsd:enumeration value="t"/>
852         <xsd:enumeration value="mid"/>
853         <xsd:enumeration value="b"/>
854         <xsd:enumeration value="none"/>
855     </xsd:restriction>
856 </xsd:simpleType>
857 <xsd:simpleType name="ST_ChildDirection" final="restriction">

```

```

858     <xsd:restriction base="xsd:token">
859         <xsd:enumeration value="horz"/>
860         <xsd:enumeration value="vert"/>
861     </xsd:restriction>
862 </xsd:simpleType>
863 <xsd:simpleType name="ST_ChildAlignment" final="restriction">
864     <xsd:restriction base="xsd:token">
865         <xsd:enumeration value="t"/>
866         <xsd:enumeration value="b"/>
867         <xsd:enumeration value="l"/>
868         <xsd:enumeration value="r"/>
869     </xsd:restriction>
870 </xsd:simpleType>
871 <xsd:simpleType name="ST_SecondaryChildAlignment" final="restriction">
872     <xsd:restriction base="xsd:token">
873         <xsd:enumeration value="none"/>
874         <xsd:enumeration value="t"/>
875         <xsd:enumeration value="b"/>
876         <xsd:enumeration value="l"/>
877         <xsd:enumeration value="r"/>
878     </xsd:restriction>
879 </xsd:simpleType>
880 <xsd:simpleType name="ST_LinearDirection" final="restriction">
881     <xsd:restriction base="xsd:token">
882         <xsd:enumeration value="fromL"/>
883         <xsd:enumeration value="fromR"/>
884         <xsd:enumeration value="fromT"/>
885         <xsd:enumeration value="fromB"/>
886     </xsd:restriction>
887 </xsd:simpleType>
888 <xsd:simpleType name="ST_SecondaryLinearDirection" final="restriction">
889     <xsd:restriction base="xsd:token">
890         <xsd:enumeration value="none"/>
891         <xsd:enumeration value="fromL"/>
892         <xsd:enumeration value="fromR"/>
893         <xsd:enumeration value="fromT"/>
894         <xsd:enumeration value="fromB"/>
895     </xsd:restriction>
896 </xsd:simpleType>
897 <xsd:simpleType name="ST_StartingElement" final="restriction">
898     <xsd:restriction base="xsd:token">
899         <xsd:enumeration value="node"/>
900         <xsd:enumeration value="trans"/>
901     </xsd:restriction>
902 </xsd:simpleType>
903 <xsd:simpleType name="ST_RotationPath" final="restriction">
904     <xsd:restriction base="xsd:token">
905         <xsd:enumeration value="none"/>
906         <xsd:enumeration value="alongPath"/>
907     </xsd:restriction>
908 </xsd:simpleType>
909 <xsd:simpleType name="ST_CenterShapeMapping" final="restriction">
910     <xsd:restriction base="xsd:token">

```

```

911         <xsd:enumeration value="none"/>
912         <xsd:enumeration value="fNode"/>
913     </xsd:restriction>
914 </xsd:simpleType>
915 <xsd:simpleType name="ST_BendPoint" final="restriction">
916     <xsd:restriction base="xsd:token">
917         <xsd:enumeration value="beg"/>
918         <xsd:enumeration value="def"/>
919         <xsd:enumeration value="end"/>
920     </xsd:restriction>
921 </xsd:simpleType>
922 <xsd:simpleType name="ST_ConnectorRouting" final="restriction">
923     <xsd:restriction base="xsd:token">
924         <xsd:enumeration value="stra"/>
925         <xsd:enumeration value="bend"/>
926         <xsd:enumeration value="curve"/>
927         <xsd:enumeration value="longCurve"/>
928     </xsd:restriction>
929 </xsd:simpleType>
930 <xsd:simpleType name="ST_ArrowheadStyle" final="restriction">
931     <xsd:restriction base="xsd:token">
932         <xsd:enumeration value="auto"/>
933         <xsd:enumeration value="arr"/>
934         <xsd:enumeration value="noArr"/>
935     </xsd:restriction>
936 </xsd:simpleType>
937 <xsd:simpleType name="ST_ConnectorDimension" final="restriction">
938     <xsd:restriction base="xsd:token">
939         <xsd:enumeration value="1D"/>
940         <xsd:enumeration value="2D"/>
941         <xsd:enumeration value="cust"/>
942     </xsd:restriction>
943 </xsd:simpleType>
944 <xsd:simpleType name="ST_ConnectorPoint" final="restriction">
945     <xsd:restriction base="xsd:token">
946         <xsd:enumeration value="auto"/>
947         <xsd:enumeration value="bCtr"/>
948         <xsd:enumeration value="ctr"/>
949         <xsd:enumeration value="midL"/>
950         <xsd:enumeration value="midR"/>
951         <xsd:enumeration value="tCtr"/>
952         <xsd:enumeration value="bL"/>
953         <xsd:enumeration value="bR"/>
954         <xsd:enumeration value="tL"/>
955         <xsd:enumeration value="tR"/>
956         <xsd:enumeration value="radial"/>
957     </xsd:restriction>
958 </xsd:simpleType>
959 <xsd:simpleType name="ST_NodeHorizontalAlignment" final="restriction">
960     <xsd:restriction base="xsd:token">
961         <xsd:enumeration value="l"/>
962         <xsd:enumeration value="ctr"/>
963         <xsd:enumeration value="r"/>

```

```

964     </xsd:restriction>
965 </xsd:simpleType>
966 <xsd:simpleType name="ST_NodeVerticalAlignment" final="restriction">
967     <xsd:restriction base="xsd:token">
968         <xsd:enumeration value="t"/>
969         <xsd:enumeration value="mid"/>
970         <xsd:enumeration value="b"/>
971     </xsd:restriction>
972 </xsd:simpleType>
973 <xsd:simpleType name="ST_FallbackDimension" final="restriction">
974     <xsd:restriction base="xsd:token">
975         <xsd:enumeration value="1D"/>
976         <xsd:enumeration value="2D"/>
977     </xsd:restriction>
978 </xsd:simpleType>
979 <xsd:simpleType name="ST_TextDirection" final="restriction">
980     <xsd:restriction base="xsd:token">
981         <xsd:enumeration value="fromT"/>
982         <xsd:enumeration value="fromB"/>
983     </xsd:restriction>
984 </xsd:simpleType>
985 <xsd:simpleType name="ST_PyramidAccentPosition" final="restriction">
986     <xsd:restriction base="xsd:token">
987         <xsd:enumeration value="bef"/>
988         <xsd:enumeration value="aft"/>
989     </xsd:restriction>
990 </xsd:simpleType>
991 <xsd:simpleType name="ST_PyramidAccentTextMargin" final="restriction">
992     <xsd:restriction base="xsd:token">
993         <xsd:enumeration value="step"/>
994         <xsd:enumeration value="stack"/>
995     </xsd:restriction>
996 </xsd:simpleType>
997 <xsd:simpleType name="ST_TextBlockDirection" final="restriction">
998     <xsd:restriction base="xsd:token">
999         <xsd:enumeration value="horz"/>
1000         <xsd:enumeration value="vert"/>
1001     </xsd:restriction>
1002 </xsd:simpleType>
1003 <xsd:simpleType name="ST_TextAnchorHorizontal" final="restriction">
1004     <xsd:restriction base="xsd:token">
1005         <xsd:enumeration value="none"/>
1006         <xsd:enumeration value="ctr"/>
1007     </xsd:restriction>
1008 </xsd:simpleType>
1009 <xsd:simpleType name="ST_TextAnchorVertical" final="restriction">
1010     <xsd:restriction base="xsd:token">
1011         <xsd:enumeration value="t"/>
1012         <xsd:enumeration value="mid"/>
1013         <xsd:enumeration value="b"/>
1014     </xsd:restriction>
1015 </xsd:simpleType>
1016 <xsd:simpleType name="ST_DiagramTextAlignment" final="restriction">

```

```

1017     <xsd:restriction base="xsd:token">
1018         <xsd:enumeration value="l"/>
1019         <xsd:enumeration value="ctr"/>
1020         <xsd:enumeration value="r"/>
1021     </xsd:restriction>
1022 </xsd:simpleType>
1023 <xsd:simpleType name="ST_AutoTextRotation" final="restriction">
1024     <xsd:restriction base="xsd:token">
1025         <xsd:enumeration value="none"/>
1026         <xsd:enumeration value="upr"/>
1027         <xsd:enumeration value="grav"/>
1028     </xsd:restriction>
1029 </xsd:simpleType>
1030 <xsd:simpleType name="ST_GrowDirection" final="restriction">
1031     <xsd:restriction base="xsd:token">
1032         <xsd:enumeration value="tL"/>
1033         <xsd:enumeration value="tR"/>
1034         <xsd:enumeration value="bL"/>
1035         <xsd:enumeration value="bR"/>
1036     </xsd:restriction>
1037 </xsd:simpleType>
1038 <xsd:simpleType name="ST_FlowDirection" final="restriction">
1039     <xsd:restriction base="xsd:token">
1040         <xsd:enumeration value="row"/>
1041         <xsd:enumeration value="col"/>
1042     </xsd:restriction>
1043 </xsd:simpleType>
1044 <xsd:simpleType name="ST_ContinueDirection" final="restriction">
1045     <xsd:restriction base="xsd:token">
1046         <xsd:enumeration value="revDir"/>
1047         <xsd:enumeration value="sameDir"/>
1048     </xsd:restriction>
1049 </xsd:simpleType>
1050 <xsd:simpleType name="ST_Breakpoint" final="restriction">
1051     <xsd:restriction base="xsd:token">
1052         <xsd:enumeration value="endCnv"/>
1053         <xsd:enumeration value="bal"/>
1054         <xsd:enumeration value="fixed"/>
1055     </xsd:restriction>
1056 </xsd:simpleType>
1057 <xsd:simpleType name="ST_Offset" final="restriction">
1058     <xsd:restriction base="xsd:token">
1059         <xsd:enumeration value="ctr"/>
1060         <xsd:enumeration value="off"/>
1061     </xsd:restriction>
1062 </xsd:simpleType>
1063 <xsd:simpleType name="ST_HierarchyAlignment" final="restriction">
1064     <xsd:restriction base="xsd:token">
1065         <xsd:enumeration value="tL"/>
1066         <xsd:enumeration value="tR"/>
1067         <xsd:enumeration value="tCtrCh"/>
1068         <xsd:enumeration value="tCtrDes"/>
1069         <xsd:enumeration value="bL"/>

```

```

1070     <xsd:enumeration value="bR"/>
1071     <xsd:enumeration value="bCtrCh"/>
1072     <xsd:enumeration value="bCtrDes"/>
1073     <xsd:enumeration value="lT"/>
1074     <xsd:enumeration value="lB"/>
1075     <xsd:enumeration value="lCtrCh"/>
1076     <xsd:enumeration value="lCtrDes"/>
1077     <xsd:enumeration value="rT"/>
1078     <xsd:enumeration value="rB"/>
1079     <xsd:enumeration value="rCtrCh"/>
1080     <xsd:enumeration value="rCtrDes"/>
1081   </xsd:restriction>
1082 </xsd:simpleType>
1083 <xsd:simpleType name="ST_FunctionValue" final="restriction">
1084   <xsd:union memberTypes="xsd:int xsd:boolean ST_Direction ST_HierBranchStyle ST_AnimOneStr
1085     ST_AnimLvlStr ST_ResizeHandlesStr"/>
1086 </xsd:simpleType>
1087 <xsd:simpleType name="ST_VariableType" final="restriction">
1088   <xsd:restriction base="xsd:token">
1089     <xsd:enumeration value="none"/>
1090     <xsd:enumeration value="orgChart"/>
1091     <xsd:enumeration value="chMax"/>
1092     <xsd:enumeration value="chPref"/>
1093     <xsd:enumeration value="bulEnabled"/>
1094     <xsd:enumeration value="dir"/>
1095     <xsd:enumeration value="hierBranch"/>
1096     <xsd:enumeration value="animOne"/>
1097     <xsd:enumeration value="animLvl"/>
1098     <xsd:enumeration value="resizeHandles"/>
1099   </xsd:restriction>
1100 </xsd:simpleType>
1101 <xsd:simpleType name="ST_FunctionArgument" final="restriction">
1102   <xsd:union memberTypes="ST_VariableType"/>
1103 </xsd:simpleType>
1104 <xsd:simpleType name="ST_OutputShapeType" final="restriction">
1105   <xsd:restriction base="xsd:token">
1106     <xsd:enumeration value="none"/>
1107     <xsd:enumeration value="conn"/>
1108   </xsd:restriction>
1109 </xsd:simpleType>
1110 </xsd:schema>

```

A.6 VML

A.6.1 VML

This schema is available in the file vml-main.xsd.

```

1 <xsd:schema xmlns="urn:schemas-microsoft-com:vml" xmlns:pvm1="urn:schemas-microsoft-
2   com:office:powerpoint" xmlns:o="urn:schemas-microsoft-com:office:office"
3   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
4   xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main" xmlns:w10="urn:schemas-
5   microsoft-com:office:word"

```



```

6  xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships" xmlns:x="urn:schemas-
7  microsoft-com:office:excel"
8  xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
9  targetNamespace="urn:schemas-microsoft-com:vm1" elementFormDefault="qualified"
10 attributeFormDefault="unqualified">
11   <xsd:import namespace="urn:schemas-microsoft-com:office:office" schemaLocation="vm1-
12     officeDrawing.xsd"/>
13   <xsd:import namespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
14     schemaLocation="wml.xsd"/>
15   <xsd:import namespace="urn:schemas-microsoft-com:office:word" schemaLocation="vm1-
16     wordprocessingDrawing.xsd"/>
17   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
18     schemaLocation="shared-relationshipReference.xsd"/>
19   <xsd:import namespace="urn:schemas-microsoft-com:office:excel" schemaLocation="vm1-
20     spreadsheetDrawing.xsd"/>
21   <xsd:import namespace="urn:schemas-microsoft-com:office:powerpoint" schemaLocation="vm1-
22     presentationDrawing.xsd"/>
23   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
24     schemaLocation="shared-commonSimpleTypes.xsd"/>
25   <xsd:attributeGroup name="AG_Id">
26     <xsd:attribute name="id" type="xsd:string" use="optional"/>
27   </xsd:attributeGroup>
28   <xsd:attributeGroup name="AG_Style">
29     <xsd:attribute name="style" type="xsd:string" use="optional"/>
30   </xsd:attributeGroup>
31   <xsd:attributeGroup name="AG_Type">
32     <xsd:attribute name="type" type="xsd:string" use="optional"/>
33   </xsd:attributeGroup>
34   <xsd:attributeGroup name="AG_Adj">
35     <xsd:attribute name="adj" type="xsd:string" use="optional"/>
36   </xsd:attributeGroup>
37   <xsd:attributeGroup name="AG_Path">
38     <xsd:attribute name="path" type="xsd:string" use="optional"/>
39   </xsd:attributeGroup>
40   <xsd:attributeGroup name="AG_Fill">
41     <xsd:attribute name="filled" type="s:ST TrueFalse" use="optional"/>
42     <xsd:attribute name="fillcolor" type="s:ST ColorType" use="optional"/>
43   </xsd:attributeGroup>
44   <xsd:attributeGroup name="AG_ChromaKey">
45     <xsd:attribute name="chromaKey" type="s:ST ColorType" use="optional"/>
46   </xsd:attributeGroup>
47   <xsd:attributeGroup name="AG_Ext">
48     <xsd:attribute name="ext" form="qualified" type="ST_Ext"/>
49   </xsd:attributeGroup>
50   <xsd:attributeGroup name="AG_CoreAttributes">
51     <xsd:attributeGroup ref="AG_Id"/>
52     <xsd:attributeGroup ref="AG_Style"/>
53     <xsd:attribute name="href" type="xsd:string" use="optional"/>
54     <xsd:attribute name="target" type="xsd:string" use="optional"/>
55     <xsd:attribute name="class" type="xsd:string" use="optional"/>
56     <xsd:attribute name="title" type="xsd:string" use="optional"/>
57     <xsd:attribute name="alt" type="xsd:string" use="optional"/>
58     <xsd:attribute name="coordsize" type="xsd:string" use="optional"/>

```

```

59     <xsd:attribute name="coordorigin" type="xsd:string" use="optional"/>
60     <xsd:attribute name="wrapcoords" type="xsd:string" use="optional"/>
61     <xsd:attribute name="print" type="s:ST TrueFalse" use="optional"/>
62 </xsd:attributeGroup>
63 <xsd:attributeGroup name="AG_ShapeAttributes">
64     <xsd:attributeGroup ref="AG_ChromaKey"/>
65     <xsd:attributeGroup ref="AG_Fill"/>
66     <xsd:attribute name="opacity" type="xsd:string" use="optional"/>
67     <xsd:attribute name="stroked" type="s:ST TrueFalse" use="optional"/>
68     <xsd:attribute name="strokecolor" type="s:ST ColorType" use="optional"/>
69     <xsd:attribute name="strokeweight" type="xsd:string" use="optional"/>
70     <xsd:attribute name="insetpen" type="s:ST TrueFalse" use="optional"/>
71 </xsd:attributeGroup>
72 <xsd:attributeGroup name="AG_OfficeCoreAttributes">
73     <xsd:attribute ref="o:spid"/>
74     <xsd:attribute ref="o:oned"/>
75     <xsd:attribute ref="o:regroupid"/>
76     <xsd:attribute ref="o:doubleclicknotify"/>
77     <xsd:attribute ref="o:button"/>
78     <xsd:attribute ref="o:userhidden"/>
79     <xsd:attribute ref="o:bullet"/>
80     <xsd:attribute ref="o:hr"/>
81     <xsd:attribute ref="o:hrrstd"/>
82     <xsd:attribute ref="o:hrrnoshade"/>
83     <xsd:attribute ref="o:hrrpct"/>
84     <xsd:attribute ref="o:hrralign"/>
85     <xsd:attribute ref="o:allowincell"/>
86     <xsd:attribute ref="o:allowoverlap"/>
87     <xsd:attribute ref="o:userdrawn"/>
88     <xsd:attribute ref="o:bordertopcolor"/>
89     <xsd:attribute ref="o:borderleftcolor"/>
90     <xsd:attribute ref="o:borderbottomcolor"/>
91     <xsd:attribute ref="o:borderrightcolor"/>
92     <xsd:attribute ref="o:dgmLayout"/>
93     <xsd:attribute ref="o:dgmnodekind"/>
94     <xsd:attribute ref="o:dgmLayoutmru"/>
95     <xsd:attribute ref="o:insetmode"/>
96 </xsd:attributeGroup>
97 <xsd:attributeGroup name="AG_OfficeShapeAttributes">
98     <xsd:attribute ref="o:spt"/>
99     <xsd:attribute ref="o:connectortype"/>
100    <xsd:attribute ref="o:bwmode"/>
101    <xsd:attribute ref="o:bwpure"/>
102    <xsd:attribute ref="o:bwnormal"/>
103    <xsd:attribute ref="o:forcedash"/>
104    <xsd:attribute ref="o:oleicon"/>
105    <xsd:attribute ref="o:ole"/>
106    <xsd:attribute ref="o:preferrelative"/>
107    <xsd:attribute ref="o:cliptowrap"/>
108    <xsd:attribute ref="o:clip"/>
109 </xsd:attributeGroup>
110 <xsd:attributeGroup name="AG_AllCoreAttributes">
111     <xsd:attributeGroup ref="AG_CoreAttributes"/>

```

```

112     <xsd:attributeGroup ref="AG_OfficeCoreAttributes"/>
113 </xsd:attributeGroup>
114 <xsd:attributeGroup name="AG_AllShapeAttributes">
115     <xsd:attributeGroup ref="AG_ShapeAttributes"/>
116     <xsd:attributeGroup ref="AG_OfficeShapeAttributes"/>
117 </xsd:attributeGroup>
118 <xsd:attributeGroup name="AG_ImageAttributes">
119     <xsd:attribute name="src" type="xsd:string" use="optional"/>
120     <xsd:attribute name="cropleft" type="xsd:string" use="optional"/>
121     <xsd:attribute name="croptop" type="xsd:string" use="optional"/>
122     <xsd:attribute name="cropright" type="xsd:string" use="optional"/>
123     <xsd:attribute name="cropbottom" type="xsd:string" use="optional"/>
124     <xsd:attribute name="gain" type="xsd:string" use="optional"/>
125     <xsd:attribute name="blacklevel" type="xsd:string" use="optional"/>
126     <xsd:attribute name="gamma" type="xsd:string" use="optional"/>
127     <xsd:attribute name="grayscale" type="s:ST TrueFalse" use="optional"/>
128     <xsd:attribute name="bilevel" type="s:ST TrueFalse" use="optional"/>
129 </xsd:attributeGroup>
130 <xsd:attributeGroup name="AG_StrokeAttributes">
131     <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
132     <xsd:attribute name="weight" type="xsd:string" use="optional"/>
133     <xsd:attribute name="color" type="s:ST ColorType" use="optional"/>
134     <xsd:attribute name="opacity" type="xsd:string" use="optional"/>
135     <xsd:attribute name="linestyle" type="ST StrokeLineStyle" use="optional"/>
136     <xsd:attribute name="miterlimit" type="xsd:decimal" use="optional"/>
137     <xsd:attribute name="joinstyle" type="ST StrokeJoinStyle" use="optional"/>
138     <xsd:attribute name="endcap" type="ST StrokeEndCap" use="optional"/>
139     <xsd:attribute name="dashstyle" type="xsd:string" use="optional"/>
140     <xsd:attribute name="filltype" type="ST FillType" use="optional"/>
141     <xsd:attribute name="src" type="xsd:string" use="optional"/>
142     <xsd:attribute name="imageaspect" type="ST ImageAspect" use="optional"/>
143     <xsd:attribute name="imagesize" type="xsd:string" use="optional"/>
144     <xsd:attribute name="imagealignshape" type="s:ST TrueFalse" use="optional"/>
145     <xsd:attribute name="color2" type="s:ST ColorType" use="optional"/>
146     <xsd:attribute name="startarrow" type="ST StrokeArrowType" use="optional"/>
147     <xsd:attribute name="startarrowwidth" type="ST StrokeArrowWidth" use="optional"/>
148     <xsd:attribute name="startarrowlength" type="ST StrokeArrowLength" use="optional"/>
149     <xsd:attribute name="endarrow" type="ST StrokeArrowType" use="optional"/>
150     <xsd:attribute name="endarrowwidth" type="ST StrokeArrowWidth" use="optional"/>
151     <xsd:attribute name="endarrowlength" type="ST StrokeArrowLength" use="optional"/>
152     <xsd:attribute ref="o:href"/>
153     <xsd:attribute ref="o:althref"/>
154     <xsd:attribute ref="o:title"/>
155     <xsd:attribute ref="o:forcedash"/>
156     <xsd:attribute ref="r:id" use="optional"/>
157     <xsd:attribute name="insetpen" type="s:ST TrueFalse" use="optional"/>
158     <xsd:attribute ref="o:relid"/>
159 </xsd:attributeGroup>
160 <xsd:group name="EG_ShapeElements">
161     <xsd:choice>
162         <xsd:element ref="path"/>
163         <xsd:element ref="formulas"/>
164         <xsd:element ref="handles"/>

```

```

165     <xsd:element ref="fill"/>
166     <xsd:element ref="stroke"/>
167     <xsd:element ref="shadow"/>
168     <xsd:element ref="textbox"/>
169     <xsd:element ref="textpath"/>
170     <xsd:element ref="imagedata"/>
171     <xsd:element ref="o:skew"/>
172     <xsd:element ref="o:extrusion"/>
173     <xsd:element ref="o:callout"/>
174     <xsd:element ref="o:lock"/>
175     <xsd:element ref="o:clippath"/>
176     <xsd:element ref="o:signatureline"/>
177     <xsd:element ref="w10:wrap"/>
178     <xsd:element ref="w10:anchorlock"/>
179     <xsd:element ref="w10:bordertop"/>
180     <xsd:element ref="w10:borderbottom"/>
181     <xsd:element ref="w10:borderleft"/>
182     <xsd:element ref="w10:borderright"/>
183     <xsd:element ref="x:ClientData" minOccurs="0"/>
184     <xsd:element ref="pvm1:textdata" minOccurs="0"/>
185   </xsd:choice>
186 </xsd:group>
187 <xsd:element name="shape" type="CT_Shape"/>
188 <xsd:element name="shapetype" type="CT_Shapetype"/>
189 <xsd:element name="group" type="CT_Group"/>
190 <xsd:element name="background" type="CT_Background"/>
191 <xsd:complexType name="CT_Shape">
192   <xsd:choice maxOccurs="unbounded">
193     <xsd:group ref="EG_ShapeElements"/>
194     <xsd:element ref="o:ink"/>
195     <xsd:element ref="pvm1:iscomment"/>
196     <xsd:element ref="o:equationxml"/>
197   </xsd:choice>
198   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
199   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
200   <xsd:attributeGroup ref="AG_Type"/>
201   <xsd:attributeGroup ref="AG_Adj"/>
202   <xsd:attributeGroup ref="AG_Path"/>
203   <xsd:attribute ref="o:gfxdata"/>
204   <xsd:attribute name="equationxml" type="xsd:string" use="optional"/>
205 </xsd:complexType>
206 <xsd:complexType name="CT_Shapetype">
207   <xsd:sequence>
208     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
209     <xsd:element ref="o:complex" minOccurs="0"/>
210   </xsd:sequence>
211   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
212   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
213   <xsd:attributeGroup ref="AG_Adj"/>
214   <xsd:attributeGroup ref="AG_Path"/>
215   <xsd:attribute ref="o:master"/>
216 </xsd:complexType>
217 <xsd:complexType name="CT_Group">

```

```

218     <xsd:choice maxOccurs="unbounded">
219         <xsd:group ref="EG_ShapeElements"/>
220         <xsd:element ref="group"/>
221         <xsd:element ref="shape"/>
222         <xsd:element ref="shapetype"/>
223         <xsd:element ref="arc"/>
224         <xsd:element ref="curve"/>
225         <xsd:element ref="image"/>
226         <xsd:element ref="line"/>
227         <xsd:element ref="oval"/>
228         <xsd:element ref="polyline"/>
229         <xsd:element ref="rect"/>
230         <xsd:element ref="roundrect"/>
231         <xsd:element ref="o:diagram"/>
232     </xsd:choice>
233     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
234     <xsd:attributeGroup ref="AG_Fill"/>
235     <xsd:attribute name="editas" type="ST_EditAs" use="optional"/>
236     <xsd:attribute ref="o:tableproperties"/>
237     <xsd:attribute ref="o:tablelimits"/>
238 </xsd:complexType>
239 <xsd:complexType name="CT_Background">
240     <xsd:sequence>
241         <xsd:element ref="fill" minOccurs="0"/>
242     </xsd:sequence>
243     <xsd:attributeGroup ref="AG_Id"/>
244     <xsd:attributeGroup ref="AG_Fill"/>
245     <xsd:attribute ref="o:bwmode"/>
246     <xsd:attribute ref="o:bwpure"/>
247     <xsd:attribute ref="o:bwnormal"/>
248     <xsd:attribute ref="o:targetscreenSize"/>
249 </xsd:complexType>
250 <xsd:element name="fill" type="CT_Fill"/>
251 <xsd:element name="formulas" type="CT_Formulas"/>
252 <xsd:element name="handles" type="CT_Handles"/>
253 <xsd:element name="imagedata" type="CT_ImageData"/>
254 <xsd:element name="path" type="CT_Path"/>
255 <xsd:element name="textbox" type="CT_Textbox"/>
256 <xsd:element name="shadow" type="CT_Shadow"/>
257 <xsd:element name="stroke" type="CT_Stroke"/>
258 <xsd:element name="textpath" type="CT_TextPath"/>
259 <xsd:complexType name="CT_Fill">
260     <xsd:sequence>
261         <xsd:element ref="o:fill" minOccurs="0"/>
262     </xsd:sequence>
263     <xsd:attributeGroup ref="AG_Id"/>
264     <xsd:attribute name="type" type="ST_FillType" use="optional"/>
265     <xsd:attribute name="on" type="s:ST_TrueFalse" use="optional"/>
266     <xsd:attribute name="color" type="s:ST_ColorType" use="optional"/>
267     <xsd:attribute name="opacity" type="xsd:string" use="optional"/>
268     <xsd:attribute name="color2" type="s:ST_ColorType" use="optional"/>
269     <xsd:attribute name="src" type="xsd:string" use="optional"/>
270     <xsd:attribute ref="o:href"/>

```

```

271     <xsd:attribute ref="o:althref"/>
272     <xsd:attribute name="size" type="xsd:string" use="optional"/>
273     <xsd:attribute name="origin" type="xsd:string" use="optional"/>
274     <xsd:attribute name="position" type="xsd:string" use="optional"/>
275     <xsd:attribute name="aspect" type="ST ImageAspect" use="optional"/>
276     <xsd:attribute name="colors" type="xsd:string" use="optional"/>
277     <xsd:attribute name="angle" type="xsd:decimal" use="optional"/>
278     <xsd:attribute name="alignshape" type="s:ST TrueFalse" use="optional"/>
279     <xsd:attribute name="focus" type="xsd:string" use="optional"/>
280     <xsd:attribute name="focussize" type="xsd:string" use="optional"/>
281     <xsd:attribute name="focusposition" type="xsd:string" use="optional"/>
282     <xsd:attribute name="method" type="ST FillMethod" use="optional"/>
283     <xsd:attribute ref="o:detectmouseclick"/>
284     <xsd:attribute ref="o:title"/>
285     <xsd:attribute ref="o:opacity2"/>
286     <xsd:attribute name="recolor" type="s:ST TrueFalse" use="optional"/>
287     <xsd:attribute name="rotate" type="s:ST TrueFalse" use="optional"/>
288     <xsd:attribute ref="r:id" use="optional"/>
289     <xsd:attribute ref="o:relid" use="optional"/>
290 </xsd:complexType>
291 <xsd:complexType name="CT_Formulas">
292     <xsd:sequence>
293         <xsd:element name="f" type="CT_F" minOccurs="0" maxOccurs="unbounded"/>
294     </xsd:sequence>
295 </xsd:complexType>
296 <xsd:complexType name="CT_F">
297     <xsd:attribute name="eqn" type="xsd:string"/>
298 </xsd:complexType>
299 <xsd:complexType name="CT_Handles">
300     <xsd:sequence>
301         <xsd:element name="h" type="CT_H" minOccurs="0" maxOccurs="unbounded"/>
302     </xsd:sequence>
303 </xsd:complexType>
304 <xsd:complexType name="CT_H">
305     <xsd:attribute name="position" type="xsd:string"/>
306     <xsd:attribute name="polar" type="xsd:string"/>
307     <xsd:attribute name="map" type="xsd:string"/>
308     <xsd:attribute name="invx" type="s:ST TrueFalse"/>
309     <xsd:attribute name="invy" type="s:ST TrueFalse"/>
310     <xsd:attribute name="switch" type="s:ST TrueFalseBlank"/>
311     <xsd:attribute name="xrange" type="xsd:string"/>
312     <xsd:attribute name="yrange" type="xsd:string"/>
313     <xsd:attribute name="radiusrange" type="xsd:string"/>
314 </xsd:complexType>
315 <xsd:complexType name="CT_ImageData">
316     <xsd:attributeGroup ref="AG_Id"/>
317     <xsd:attributeGroup ref="AG_ImageAttributes"/>
318     <xsd:attributeGroup ref="AG_Chromakey"/>
319     <xsd:attribute name="embosscolor" type="s:ST ColorType" use="optional"/>
320     <xsd:attribute name="recolortarget" type="s:ST ColorType"/>
321     <xsd:attribute ref="o:href"/>
322     <xsd:attribute ref="o:althref"/>
323     <xsd:attribute ref="o:title"/>

```

```

324     <xsd:attribute ref="o:oleid"/>
325     <xsd:attribute ref="o:detectmouseclick"/>
326     <xsd:attribute ref="o:movie"/>
327     <xsd:attribute ref="o:relid"/>
328     <xsd:attribute ref="r:id"/>
329     <xsd:attribute ref="r:pict"/>
330     <xsd:attribute ref="r:href"/>
331 </xsd:complexType>
332 <xsd:complexType name="CT_Path">
333     <xsd:attributeGroup ref="AG_Id"/>
334     <xsd:attribute name="v" type="xsd:string" use="optional"/>
335     <xsd:attribute name="limo" type="xsd:string" use="optional"/>
336     <xsd:attribute name="textboxrect" type="xsd:string" use="optional"/>
337     <xsd:attribute name="fillok" type="s:ST TrueFalse" use="optional"/>
338     <xsd:attribute name="strokeok" type="s:ST TrueFalse" use="optional"/>
339     <xsd:attribute name="shadowok" type="s:ST TrueFalse" use="optional"/>
340     <xsd:attribute name="arrowok" type="s:ST TrueFalse" use="optional"/>
341     <xsd:attribute name="gradientshapeok" type="s:ST TrueFalse" use="optional"/>
342     <xsd:attribute name="textpathok" type="s:ST TrueFalse" use="optional"/>
343     <xsd:attribute name="insetpenok" type="s:ST TrueFalse" use="optional"/>
344     <xsd:attribute ref="o:connecttype"/>
345     <xsd:attribute ref="o:connectlocs"/>
346     <xsd:attribute ref="o:connectangles"/>
347     <xsd:attribute ref="o:extrusionok"/>
348 </xsd:complexType>
349 <xsd:complexType name="CT_Shadow">
350     <xsd:attributeGroup ref="AG_Id"/>
351     <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
352     <xsd:attribute name="type" type="ST ShadowType" use="optional"/>
353     <xsd:attribute name="obscured" type="s:ST TrueFalse" use="optional"/>
354     <xsd:attribute name="color" type="s:ST ColorType" use="optional"/>
355     <xsd:attribute name="opacity" type="xsd:string" use="optional"/>
356     <xsd:attribute name="offset" type="xsd:string" use="optional"/>
357     <xsd:attribute name="color2" type="s:ST ColorType" use="optional"/>
358     <xsd:attribute name="offset2" type="xsd:string" use="optional"/>
359     <xsd:attribute name="origin" type="xsd:string" use="optional"/>
360     <xsd:attribute name="matrix" type="xsd:string" use="optional"/>
361 </xsd:complexType>
362 <xsd:complexType name="CT_Stroke">
363     <xsd:sequence>
364         <xsd:element ref="o:left" minOccurs="0"/>
365         <xsd:element ref="o:top" minOccurs="0"/>
366         <xsd:element ref="o:right" minOccurs="0"/>
367         <xsd:element ref="o:bottom" minOccurs="0"/>
368         <xsd:element ref="o:column" minOccurs="0"/>
369     </xsd:sequence>
370     <xsd:attributeGroup ref="AG_Id"/>
371     <xsd:attributeGroup ref="AG_StrokeAttributes"/>
372 </xsd:complexType>
373 <xsd:complexType name="CT_Textbox">
374     <xsd:choice>
375         <xsd:element ref="w:txbxContent" minOccurs="0"/>
376         <xsd:any namespace="##local" processContents="skip"/>

```

```

377     </xsd:choice>
378     <xsd:attributeGroup ref="AG_Id"/>
379     <xsd:attributeGroup ref="AG_Style"/>
380     <xsd:attribute name="inset" type="xsd:string" use="optional"/>
381     <xsd:attribute ref="o:singleclick"/>
382     <xsd:attribute ref="o:insetmode"/>
383 </xsd:complexType>
384 <xsd:complexType name="CT_TextPath">
385     <xsd:attributeGroup ref="AG_Id"/>
386     <xsd:attributeGroup ref="AG_Style"/>
387     <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
388     <xsd:attribute name="fitshape" type="s:ST TrueFalse" use="optional"/>
389     <xsd:attribute name="fitpath" type="s:ST TrueFalse" use="optional"/>
390     <xsd:attribute name="trim" type="s:ST TrueFalse" use="optional"/>
391     <xsd:attribute name="xscale" type="s:ST TrueFalse" use="optional"/>
392     <xsd:attribute name="string" type="xsd:string" use="optional"/>
393 </xsd:complexType>
394 <xsd:element name="arc" type="CT_Arc"/>
395 <xsd:element name="curve" type="CT_Curve"/>
396 <xsd:element name="image" type="CT_Image"/>
397 <xsd:element name="line" type="CT_Line"/>
398 <xsd:element name="oval" type="CT_Oval"/>
399 <xsd:element name="polyline" type="CT_PolyLine"/>
400 <xsd:element name="rect" type="CT_Rect"/>
401 <xsd:element name="roundrect" type="CT_RoundRect"/>
402 <xsd:complexType name="CT_Arc">
403     <xsd:sequence>
404         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
405     </xsd:sequence>
406     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
407     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
408     <xsd:attribute name="startAngle" type="xsd:decimal" use="optional"/>
409     <xsd:attribute name="endAngle" type="xsd:decimal" use="optional"/>
410 </xsd:complexType>
411 <xsd:complexType name="CT_Curve">
412     <xsd:sequence>
413         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
414     </xsd:sequence>
415     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
416     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
417     <xsd:attribute name="from" type="xsd:string" use="optional"/>
418     <xsd:attribute name="control1" type="xsd:string" use="optional"/>
419     <xsd:attribute name="control2" type="xsd:string" use="optional"/>
420     <xsd:attribute name="to" type="xsd:string" use="optional"/>
421 </xsd:complexType>
422 <xsd:complexType name="CT_Image">
423     <xsd:sequence>
424         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
425     </xsd:sequence>
426     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
427     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
428     <xsd:attributeGroup ref="AG_ImageAttributes"/>
429 </xsd:complexType>

```



```

430 <xsd:complexType name="CT_Line">
431   <xsd:sequence>
432     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
433   </xsd:sequence>
434   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
435   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
436   <xsd:attribute name="from" type="xsd:string" use="optional"/>
437   <xsd:attribute name="to" type="xsd:string" use="optional"/>
438 </xsd:complexType>
439 <xsd:complexType name="CT_Oval">
440   <xsd:choice maxOccurs="unbounded">
441     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
442   </xsd:choice>
443   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
444   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
445 </xsd:complexType>
446 <xsd:complexType name="CT_PolyLine">
447   <xsd:choice minOccurs="0" maxOccurs="unbounded">
448     <xsd:group ref="EG_ShapeElements"/>
449     <xsd:element ref="o:ink"/>
450   </xsd:choice>
451   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
452   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
453   <xsd:attribute name="points" type="xsd:string" use="optional"/>
454 </xsd:complexType>
455 <xsd:complexType name="CT_Rect">
456   <xsd:choice maxOccurs="unbounded">
457     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
458   </xsd:choice>
459   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
460   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
461 </xsd:complexType>
462 <xsd:complexType name="CT_RoundRect">
463   <xsd:choice maxOccurs="unbounded">
464     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
465   </xsd:choice>
466   <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
467   <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
468   <xsd:attribute name="arcsz" type="xsd:string" use="optional"/>
469 </xsd:complexType>
470 <xsd:simpleType name="ST_Ext">
471   <xsd:restriction base="xsd:string">
472     <xsd:enumeration value="view"/>
473     <xsd:enumeration value="edit"/>
474     <xsd:enumeration value="backwardCompatible"/>
475   </xsd:restriction>
476 </xsd:simpleType>
477 <xsd:simpleType name="ST_FillType">
478   <xsd:restriction base="xsd:string">
479     <xsd:enumeration value="solid"/>
480     <xsd:enumeration value="gradient"/>
481     <xsd:enumeration value="gradientRadial"/>
482     <xsd:enumeration value="tile"/>

```

```

483         <xsd:enumeration value="pattern"/>
484         <xsd:enumeration value="frame"/>
485     </xsd:restriction>
486 </xsd:simpleType>
487 <xsd:simpleType name="ST_FillMethod">
488     <xsd:restriction base="xsd:string">
489         <xsd:enumeration value="none"/>
490         <xsd:enumeration value="linear"/>
491         <xsd:enumeration value="sigma"/>
492         <xsd:enumeration value="any"/>
493         <xsd:enumeration value="linear sigma"/>
494     </xsd:restriction>
495 </xsd:simpleType>
496 <xsd:simpleType name="ST_ShadowType">
497     <xsd:restriction base="xsd:string">
498         <xsd:enumeration value="single"/>
499         <xsd:enumeration value="double"/>
500         <xsd:enumeration value="emboss"/>
501         <xsd:enumeration value="perspective"/>
502     </xsd:restriction>
503 </xsd:simpleType>
504 <xsd:simpleType name="ST_StrokelineStyle">
505     <xsd:restriction base="xsd:string">
506         <xsd:enumeration value="single"/>
507         <xsd:enumeration value="thinThin"/>
508         <xsd:enumeration value="thinThick"/>
509         <xsd:enumeration value="thickThin"/>
510         <xsd:enumeration value="thickBetweenThin"/>
511     </xsd:restriction>
512 </xsd:simpleType>
513 <xsd:simpleType name="ST_StrokeJoinStyle">
514     <xsd:restriction base="xsd:string">
515         <xsd:enumeration value="round"/>
516         <xsd:enumeration value="bevel"/>
517         <xsd:enumeration value="miter"/>
518     </xsd:restriction>
519 </xsd:simpleType>
520 <xsd:simpleType name="ST_StrokeEndCap">
521     <xsd:restriction base="xsd:string">
522         <xsd:enumeration value="flat"/>
523         <xsd:enumeration value="square"/>
524         <xsd:enumeration value="round"/>
525     </xsd:restriction>
526 </xsd:simpleType>
527 <xsd:simpleType name="ST_StrokeArrowLength">
528     <xsd:restriction base="xsd:string">
529         <xsd:enumeration value="short"/>
530         <xsd:enumeration value="medium"/>
531         <xsd:enumeration value="long"/>
532     </xsd:restriction>
533 </xsd:simpleType>
534 <xsd:simpleType name="ST_StrokeArrowWidth">
535     <xsd:restriction base="xsd:string">

```

```

536         <xsd:enumeration value="narrow"/>
537         <xsd:enumeration value="medium"/>
538         <xsd:enumeration value="wide"/>
539     </xsd:restriction>
540 </xsd:simpleType>
541 <xsd:simpleType name="ST_StrokeArrowType">
542     <xsd:restriction base="xsd:string">
543         <xsd:enumeration value="none"/>
544         <xsd:enumeration value="block"/>
545         <xsd:enumeration value="classic"/>
546         <xsd:enumeration value="oval"/>
547         <xsd:enumeration value="diamond"/>
548         <xsd:enumeration value="open"/>
549     </xsd:restriction>
550 </xsd:simpleType>
551 <xsd:simpleType name="ST_ImageAspect">
552     <xsd:restriction base="xsd:string">
553         <xsd:enumeration value="ignore"/>
554         <xsd:enumeration value="atMost"/>
555         <xsd:enumeration value="atLeast"/>
556     </xsd:restriction>
557 </xsd:simpleType>
558 <xsd:simpleType name="ST_EditAs">
559     <xsd:restriction base="xsd:string">
560         <xsd:enumeration value="canvas"/>
561         <xsd:enumeration value="orgchart"/>
562         <xsd:enumeration value="radial"/>
563         <xsd:enumeration value="cycle"/>
564         <xsd:enumeration value="stacked"/>
565         <xsd:enumeration value="venn"/>
566         <xsd:enumeration value="bullseye"/>
567     </xsd:restriction>
568 </xsd:simpleType>
569 </xsd:schema>

```

A.6.2 VML - Office Drawing

This schema is available in the file vml-officeDrawing.xsd.

```

1 <xsd:schema xmlns="urn:schemas-microsoft-com:office:office" xmlns:v="urn:schemas-microsoft-com:vml"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
5   targetNamespace="urn:schemas-microsoft-com:office:office" elementFormDefault="qualified"
6   attributeFormDefault="unqualified">
7     <xsd:import namespace="urn:schemas-microsoft-com:vml" schemaLocation="vml-main.xsd"/>
8     <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9       schemaLocation="shared-relationshipReference.xsd"/>
10    <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11      schemaLocation="shared-commonSimpleTypes.xsd"/>
12    <xsd:attribute name="bwmode" type="ST_BWMode"/>
13    <xsd:attribute name="bwpure" type="ST_BWMode"/>
14    <xsd:attribute name="bwnormal" type="ST_BWMode"/>

```

```

15 <xsd:attribute name="targetscreenSize" type="ST_ScreenSize"/>
16 <xsd:attribute name="insetmode" type="ST_InsetMode" default="custom"/>
17 <xsd:attribute name="spt" type="xsd:float"/>
18 <xsd:attribute name="wrapcoords" type="xsd:string"/>
19 <xsd:attribute name="oned" type="s:ST_TrueFalse"/>
20 <xsd:attribute name="regroupid" type="xsd:integer"/>
21 <xsd:attribute name="doubleclicknotify" type="s:ST_TrueFalse"/>
22 <xsd:attribute name="connectortype" type="ST_ConnectorType" default="straight"/>
23 <xsd:attribute name="button" type="s:ST_TrueFalse"/>
24 <xsd:attribute name="userhidden" type="s:ST_TrueFalse"/>
25 <xsd:attribute name="forcedash" type="s:ST_TrueFalse"/>
26 <xsd:attribute name="oleicon" type="s:ST_TrueFalse"/>
27 <xsd:attribute name="ole" type="s:ST_TrueFalseBlank"/>
28 <xsd:attribute name="preferrelative" type="s:ST_TrueFalse"/>
29 <xsd:attribute name="cliptowrap" type="s:ST_TrueFalse"/>
30 <xsd:attribute name="clip" type="s:ST_TrueFalse"/>
31 <xsd:attribute name="bullet" type="s:ST_TrueFalse"/>
32 <xsd:attribute name="hr" type="s:ST_TrueFalse"/>
33 <xsd:attribute name="hrstd" type="s:ST_TrueFalse"/>
34 <xsd:attribute name="hrnoshade" type="s:ST_TrueFalse"/>
35 <xsd:attribute name="hrpct" type="xsd:float"/>
36 <xsd:attribute name="hralign" type="ST_HrAlign" default="left"/>
37 <xsd:attribute name="allowincell" type="s:ST_TrueFalse"/>
38 <xsd:attribute name="allowoverlap" type="s:ST_TrueFalse"/>
39 <xsd:attribute name="userdrawn" type="s:ST_TrueFalse"/>
40 <xsd:attribute name="bordertopcolor" type="xsd:string"/>
41 <xsd:attribute name="borderleftcolor" type="xsd:string"/>
42 <xsd:attribute name="borderbottomcolor" type="xsd:string"/>
43 <xsd:attribute name="borderrightcolor" type="xsd:string"/>
44 <xsd:attribute name="connecttype" type="ST_ConnectType"/>
45 <xsd:attribute name="connectlocs" type="xsd:string"/>
46 <xsd:attribute name="connectangles" type="xsd:string"/>
47 <xsd:attribute name="master" type="xsd:string"/>
48 <xsd:attribute name="extrusionok" type="s:ST_TrueFalse"/>
49 <xsd:attribute name="href" type="xsd:string"/>
50 <xsd:attribute name="althref" type="xsd:string"/>
51 <xsd:attribute name="title" type="xsd:string"/>
52 <xsd:attribute name="singleclick" type="s:ST_TrueFalse"/>
53 <xsd:attribute name="oleid" type="xsd:float"/>
54 <xsd:attribute name="detectmouseclick" type="s:ST_TrueFalse"/>
55 <xsd:attribute name="movie" type="xsd:float"/>
56 <xsd:attribute name="spid" type="xsd:string"/>
57 <xsd:attribute name="opacity2" type="xsd:string"/>
58 <xsd:attribute name="relid" type="r:ST_RelationshipId"/>
59 <xsd:attribute name="dgmlayout" type="ST_DiagramLayout"/>
60 <xsd:attribute name="dgmnodekind" type="xsd:integer"/>
61 <xsd:attribute name="dgmlayoutmru" type="ST_DiagramLayout"/>
62 <xsd:attribute name="gfxdata" type="xsd:base64Binary"/>
63 <xsd:attribute name="tableproperties" type="xsd:string"/>
64 <xsd:attribute name="tablelimits" type="xsd:string"/>
65 <xsd:element name="shapedefaults" type="CT_ShapeDefaults"/>
66 <xsd:element name="shapelayout" type="CT_ShapeLayout"/>
67 <xsd:element name="signatureline" type="CT_SignatureLine"/>

```

```

68 <xsd:element name="ink" type="CT_Ink"/>
69 <xsd:element name="diagram" type="CT_Diagram"/>
70 <xsd:element name="equationxml" type="CT_EquationXml"/>
71 <xsd:complexType name="CT_ShapeDefaults">
72   <xsd:all minOccurs="0">
73     <xsd:element ref="v:fill" minOccurs="0"/>
74     <xsd:element ref="v:stroke" minOccurs="0"/>
75     <xsd:element ref="v:textbox" minOccurs="0"/>
76     <xsd:element ref="v:shadow" minOccurs="0"/>
77     <xsd:element ref="skew" minOccurs="0"/>
78     <xsd:element ref="extrusion" minOccurs="0"/>
79     <xsd:element ref="callout" minOccurs="0"/>
80     <xsd:element ref="lock" minOccurs="0"/>
81     <xsd:element name="colormru" minOccurs="0" type="CT_ColorMru"/>
82     <xsd:element name="colormenu" minOccurs="0" type="CT_ColorMenu"/>
83   </xsd:all>
84   <xsd:attributeGroup ref="v:AG_Ext"/>
85   <xsd:attribute name="spidmax" type="xsd:integer" use="optional"/>
86   <xsd:attribute name="style" type="xsd:string" use="optional"/>
87   <xsd:attribute name="fill" type="s:ST_TrueFalse" use="optional"/>
88   <xsd:attribute name="fillcolor" type="s:ST_ColorType" use="optional"/>
89   <xsd:attribute name="stroke" type="s:ST_TrueFalse" use="optional"/>
90   <xsd:attribute name="strokecolor" type="s:ST_ColorType"/>
91   <xsd:attribute name="allowincell" form="qualified" type="s:ST_TrueFalse"/>
92 </xsd:complexType>
93 <xsd:complexType name="CT_Ink">
94   <xsd:sequence/>
95   <xsd:attribute name="i" type="xsd:string"/>
96   <xsd:attribute name="annotation" type="s:ST_TrueFalse"/>
97   <xsd:attribute name="contentType" type="ST_ContentType" use="optional"/>
98 </xsd:complexType>
99 <xsd:complexType name="CT_SignatureLine">
100   <xsd:attributeGroup ref="v:AG_Ext"/>
101   <xsd:attribute name="issignatureline" type="s:ST_TrueFalse"/>
102   <xsd:attribute name="id" type="s:ST_Guid"/>
103   <xsd:attribute name="provid" type="s:ST_Guid"/>
104   <xsd:attribute name="signinginstructionsset" type="s:ST_TrueFalse"/>
105   <xsd:attribute name="allowcomments" type="s:ST_TrueFalse"/>
106   <xsd:attribute name="showsigndate" type="s:ST_TrueFalse"/>
107   <xsd:attribute name="suggestedsigner" type="xsd:string" form="qualified"/>
108   <xsd:attribute name="suggestedsigner2" type="xsd:string" form="qualified"/>
109   <xsd:attribute name="suggestedsigneremail" type="xsd:string" form="qualified"/>
110   <xsd:attribute name="signinginstructions" type="xsd:string"/>
111   <xsd:attribute name="addlxml" type="xsd:string"/>
112   <xsd:attribute name="sigprovurl" type="xsd:string"/>
113 </xsd:complexType>
114 <xsd:complexType name="CT_ShapeLayout">
115   <xsd:all>
116     <xsd:element name="idmap" type="CT_IdMap" minOccurs="0"/>
117     <xsd:element name="regrouptable" type="CT_RegroupTable" minOccurs="0"/>
118     <xsd:element name="rules" type="CT_Rules" minOccurs="0"/>
119   </xsd:all>
120   <xsd:attributeGroup ref="v:AG_Ext"/>

```

```

121 </xsd:complexType>
122 <xsd:complexType name="CT_IdMap">
123   <xsd:attributeGroup ref="v:AG Ext"/>
124   <xsd:attribute name="data" type="xsd:string" use="optional"/>
125 </xsd:complexType>
126 <xsd:complexType name="CT_RegroupTable">
127   <xsd:sequence>
128     <xsd:element name="entry" type="CT_Entry" minOccurs="0" maxOccurs="unbounded"/>
129   </xsd:sequence>
130   <xsd:attributeGroup ref="v:AG Ext"/>
131 </xsd:complexType>
132 <xsd:complexType name="CT_Entry">
133   <xsd:attribute name="new" type="xsd:int" use="optional"/>
134   <xsd:attribute name="old" type="xsd:int" use="optional"/>
135 </xsd:complexType>
136 <xsd:complexType name="CT_Rules">
137   <xsd:sequence>
138     <xsd:element name="r" type="CT_R" minOccurs="0" maxOccurs="unbounded"/>
139   </xsd:sequence>
140   <xsd:attributeGroup ref="v:AG Ext"/>
141 </xsd:complexType>
142 <xsd:complexType name="CT_R">
143   <xsd:sequence>
144     <xsd:element name="proxy" type="CT_Proxy" minOccurs="0" maxOccurs="unbounded"/>
145   </xsd:sequence>
146   <xsd:attribute name="id" type="xsd:string" use="required"/>
147   <xsd:attribute name="type" type="ST_RType" use="optional"/>
148   <xsd:attribute name="how" type="ST_How" use="optional"/>
149   <xsd:attribute name="idref" type="xsd:string" use="optional"/>
150 </xsd:complexType>
151 <xsd:complexType name="CT_Proxy">
152   <xsd:attribute name="start" type="s:ST_TrueFalseBlank" use="optional" default="false"/>
153   <xsd:attribute name="end" type="s:ST_TrueFalseBlank" use="optional" default="false"/>
154   <xsd:attribute name="idref" type="xsd:string" use="optional"/>
155   <xsd:attribute name="connectloc" type="xsd:int" use="optional"/>
156 </xsd:complexType>
157 <xsd:complexType name="CT_Diagram">
158   <xsd:sequence>
159     <xsd:element name="relationtable" type="CT_RelationTable" minOccurs="0"/>
160   </xsd:sequence>
161   <xsd:attributeGroup ref="v:AG Ext"/>
162   <xsd:attribute name="dgmstyle" type="xsd:integer" use="optional"/>
163   <xsd:attribute name="autoformat" type="s:ST_TrueFalse" use="optional"/>
164   <xsd:attribute name="reverse" type="s:ST_TrueFalse" use="optional"/>
165   <xsd:attribute name="autolayout" type="s:ST_TrueFalse" use="optional"/>
166   <xsd:attribute name="dgmscalex" type="xsd:integer" use="optional"/>
167   <xsd:attribute name="dgmscaley" type="xsd:integer" use="optional"/>
168   <xsd:attribute name="dgmfontsize" type="xsd:integer" use="optional"/>
169   <xsd:attribute name="constrainbounds" type="xsd:string" use="optional"/>
170   <xsd:attribute name="dgmbasetextscale" type="xsd:integer" use="optional"/>
171 </xsd:complexType>
172 <xsd:complexType name="CT_EquationXml">
173   <xsd:sequence>

```

```

174     <xsd:any namespace="##any"/>
175 </xsd:sequence>
176 <xsd:attribute name="contentType" type="ST_AlternateMathContentType" use="optional"/>
177 </xsd:complexType>
178 <xsd:simpleType name="ST_AlternateMathContentType">
179   <xsd:restriction base="xsd:string"/>
180 </xsd:simpleType>
181 <xsd:complexType name="CT_RelationTable">
182   <xsd:sequence>
183     <xsd:element name="rel" type="CT_Relation" minOccurs="0" maxOccurs="unbounded"/>
184   </xsd:sequence>
185   <xsd:attributeGroup ref="v:AG_Ext"/>
186 </xsd:complexType>
187 <xsd:complexType name="CT_Relation">
188   <xsd:attributeGroup ref="v:AG_Ext"/>
189   <xsd:attribute name="idsrc" type="xsd:string" use="optional"/>
190   <xsd:attribute name="iddest" type="xsd:string" use="optional"/>
191   <xsd:attribute name="idcntr" type="xsd:string" use="optional"/>
192 </xsd:complexType>
193 <xsd:complexType name="CT_ColorMru">
194   <xsd:attributeGroup ref="v:AG_Ext"/>
195   <xsd:attribute name="colors" type="xsd:string"/>
196 </xsd:complexType>
197 <xsd:complexType name="CT_ColorMenu">
198   <xsd:attributeGroup ref="v:AG_Ext"/>
199   <xsd:attribute name="strokecolor" type="s:ST_ColorType"/>
200   <xsd:attribute name="fillcolor" type="s:ST_ColorType"/>
201   <xsd:attribute name="shadowcolor" type="s:ST_ColorType"/>
202   <xsd:attribute name="extrusioncolor" type="s:ST_ColorType"/>
203 </xsd:complexType>
204 <xsd:element name="skew" type="CT_Skew"/>
205 <xsd:element name="extrusion" type="CT_Extrusion"/>
206 <xsd:element name="callout" type="CT_Callout"/>
207 <xsd:element name="lock" type="CT_Lock"/>
208 <xsd:element name="OLEObject" type="CT_OLEObject"/>
209 <xsd:element name="complex" type="CT_Complex"/>
210 <xsd:element name="left" type="CT_StrokeChild"/>
211 <xsd:element name="top" type="CT_StrokeChild"/>
212 <xsd:element name="right" type="CT_StrokeChild"/>
213 <xsd:element name="bottom" type="CT_StrokeChild"/>
214 <xsd:element name="column" type="CT_StrokeChild"/>
215 <xsd:element name="clippath" type="CT_ClipPath"/>
216 <xsd:element name="fill" type="CT_Fill"/>
217 <xsd:complexType name="CT_Skew">
218   <xsd:attributeGroup ref="v:AG_Ext"/>
219   <xsd:attribute name="id" type="xsd:string" use="optional"/>
220   <xsd:attribute name="on" type="s:ST_TrueFalse" use="optional"/>
221   <xsd:attribute name="offset" type="xsd:string" use="optional"/>
222   <xsd:attribute name="origin" type="xsd:string" use="optional"/>
223   <xsd:attribute name="matrix" type="xsd:string" use="optional"/>
224 </xsd:complexType>
225 <xsd:complexType name="CT_Extrusion">
226   <xsd:attributeGroup ref="v:AG_Ext"/>

```

```

227 <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
228 <xsd:attribute name="type" type="ST ExtrusionType" default="parallel" use="optional"/>
229 <xsd:attribute name="render" type="ST ExtrusionRender" default="solid" use="optional"/>
230 <xsd:attribute name="viewpointorigin" type="xsd:string" use="optional"/>
231 <xsd:attribute name="viewpoint" type="xsd:string" use="optional"/>
232 <xsd:attribute name="plane" type="ST ExtrusionPlane" default="XY" use="optional"/>
233 <xsd:attribute name="skewangle" type="xsd:float" use="optional"/>
234 <xsd:attribute name="skewamt" type="xsd:string" use="optional"/>
235 <xsd:attribute name="foredepth" type="xsd:string" use="optional"/>
236 <xsd:attribute name="backdepth" type="xsd:string" use="optional"/>
237 <xsd:attribute name="orientation" type="xsd:string" use="optional"/>
238 <xsd:attribute name="orientationangle" type="xsd:float" use="optional"/>
239 <xsd:attribute name="lockrotationcenter" type="s:ST TrueFalse" use="optional"/>
240 <xsd:attribute name="autorotationcenter" type="s:ST TrueFalse" use="optional"/>
241 <xsd:attribute name="rotationcenter" type="xsd:string" use="optional"/>
242 <xsd:attribute name="rotationangle" type="xsd:string" use="optional"/>
243 <xsd:attribute name="colormode" type="ST ColorMode" use="optional"/>
244 <xsd:attribute name="color" type="s:ST ColorType" use="optional"/>
245 <xsd:attribute name="shininess" type="xsd:float" use="optional"/>
246 <xsd:attribute name="specularity" type="xsd:string" use="optional"/>
247 <xsd:attribute name="diffusivity" type="xsd:string" use="optional"/>
248 <xsd:attribute name="metal" type="s:ST TrueFalse" use="optional"/>
249 <xsd:attribute name="edge" type="xsd:string" use="optional"/>
250 <xsd:attribute name="facet" type="xsd:string" use="optional"/>
251 <xsd:attribute name="lightface" type="s:ST TrueFalse" use="optional"/>
252 <xsd:attribute name="brightness" type="xsd:string" use="optional"/>
253 <xsd:attribute name="lightposition" type="xsd:string" use="optional"/>
254 <xsd:attribute name="lightlevel" type="xsd:string" use="optional"/>
255 <xsd:attribute name="lightharsh" type="s:ST TrueFalse" use="optional"/>
256 <xsd:attribute name="lightposition2" type="xsd:string" use="optional"/>
257 <xsd:attribute name="lightlevel2" type="xsd:string" use="optional"/>
258 <xsd:attribute name="lightharsh2" type="s:ST TrueFalse" use="optional"/>
259 </xsd:complexType>
260 <xsd:complexType name="CT_Callout">
261 <xsd:attributeGroup ref="v:AG Ext"/>
262 <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
263 <xsd:attribute name="type" type="xsd:string" use="optional"/>
264 <xsd:attribute name="gap" type="xsd:string" use="optional"/>
265 <xsd:attribute name="angle" type="ST Angle" use="optional"/>
266 <xsd:attribute name="dropauto" type="s:ST TrueFalse" use="optional"/>
267 <xsd:attribute name="drop" type="ST CalloutDrop" use="optional"/>
268 <xsd:attribute name="distance" type="xsd:string" use="optional"/>
269 <xsd:attribute name="lengthspecified" type="s:ST TrueFalse" default="f" use="optional"/>
270 <xsd:attribute name="length" type="xsd:string" use="optional"/>
271 <xsd:attribute name="accentbar" type="s:ST TrueFalse" use="optional"/>
272 <xsd:attribute name="textborder" type="s:ST TrueFalse" use="optional"/>
273 <xsd:attribute name="minusx" type="s:ST TrueFalse" use="optional"/>
274 <xsd:attribute name="minusy" type="s:ST TrueFalse" use="optional"/>
275 </xsd:complexType>
276 <xsd:complexType name="CT_Lock">
277 <xsd:attributeGroup ref="v:AG Ext"/>
278 <xsd:attribute name="position" type="s:ST TrueFalse" use="optional"/>
279 <xsd:attribute name="selection" type="s:ST TrueFalse" use="optional"/>

```



```

280 <xsd:attribute name="grouping" type="s:ST TrueFalse" use="optional"/>
281 <xsd:attribute name="ungrouping" type="s:ST TrueFalse" use="optional"/>
282 <xsd:attribute name="rotation" type="s:ST TrueFalse" use="optional"/>
283 <xsd:attribute name="cropping" type="s:ST TrueFalse" use="optional"/>
284 <xsd:attribute name="verticies" type="s:ST TrueFalse" use="optional"/>
285 <xsd:attribute name="adjusthandles" type="s:ST TrueFalse" use="optional"/>
286 <xsd:attribute name="text" type="s:ST TrueFalse" use="optional"/>
287 <xsd:attribute name="aspectratio" type="s:ST TrueFalse" use="optional"/>
288 <xsd:attribute name="shapetype" type="s:ST TrueFalse" use="optional"/>
289 </xsd:complexType>
290 <xsd:complexType name="CT_OLEObject">
291   <xsd:sequence>
292     <xsd:element name="LinkType" type="ST_OLELinkType" minOccurs="0"/>
293     <xsd:element name="LockedField" type="s:ST TrueFalseBlank" minOccurs="0"/>
294     <xsd:element name="FieldCodes" type="xsd:string" minOccurs="0"/>
295   </xsd:sequence>
296   <xsd:attribute name="Type" type="ST_OLEType" use="optional"/>
297   <xsd:attribute name="ProgID" type="xsd:string" use="optional"/>
298   <xsd:attribute name="ShapeID" type="xsd:string" use="optional"/>
299   <xsd:attribute name="DrawAspect" type="ST_OLEDrawAspect" use="optional"/>
300   <xsd:attribute name="ObjectID" type="xsd:string" use="optional"/>
301   <xsd:attribute ref="r:id" use="optional"/>
302   <xsd:attribute name="UpdateMode" type="ST_OLEUpdateMode" use="optional"/>
303 </xsd:complexType>
304 <xsd:complexType name="CT_Complex">
305   <xsd:attributeGroup ref="v:AG_Ext"/>
306 </xsd:complexType>
307 <xsd:complexType name="CT_StrokeChild">
308   <xsd:attributeGroup ref="v:AG_Ext"/>
309   <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
310   <xsd:attribute name="weight" type="xsd:string" use="optional"/>
311   <xsd:attribute name="color" type="s:ST ColorType" use="optional"/>
312   <xsd:attribute name="color2" type="s:ST ColorType" use="optional"/>
313   <xsd:attribute name="opacity" type="xsd:string" use="optional"/>
314   <xsd:attribute name="linestyle" type="v:ST StrokeLineStyle" use="optional"/>
315   <xsd:attribute name="miterlimit" type="xsd:decimal" use="optional"/>
316   <xsd:attribute name="joinstyle" type="v:ST StrokeJoinStyle" use="optional"/>
317   <xsd:attribute name="endcap" type="v:ST StrokeEndCap" use="optional"/>
318   <xsd:attribute name="dashstyle" type="xsd:string" use="optional"/>
319   <xsd:attribute name="insetpen" type="s:ST TrueFalse" use="optional"/>
320   <xsd:attribute name="filltype" type="v:ST FillType" use="optional"/>
321   <xsd:attribute name="src" type="xsd:string" use="optional"/>
322   <xsd:attribute name="imageaspect" type="v:ST ImageAspect" use="optional"/>
323   <xsd:attribute name="imagesize" type="xsd:string" use="optional"/>
324   <xsd:attribute name="imagealignshape" type="s:ST TrueFalse" use="optional"/>
325   <xsd:attribute name="startarrow" type="v:ST StrokeArrowType" use="optional"/>
326   <xsd:attribute name="startarrowwidth" type="v:ST StrokeArrowWidth" use="optional"/>
327   <xsd:attribute name="startarrowlength" type="v:ST StrokeArrowLength" use="optional"/>
328   <xsd:attribute name="endarrow" type="v:ST StrokeArrowType" use="optional"/>
329   <xsd:attribute name="endarrowwidth" type="v:ST StrokeArrowWidth" use="optional"/>
330   <xsd:attribute name="endarrowlength" type="v:ST StrokeArrowLength" use="optional"/>
331   <xsd:attribute ref="href"/>
332   <xsd:attribute ref="althref"/>

```

```

333     <xsd:attribute ref="title"/>
334     <xsd:attribute ref="forcedash"/>
335 </xsd:complexType>
336 <xsd:complexType name="CT_ClipPath">
337     <xsd:attribute name="v" type="xsd:string" use="required" form="qualified"/>
338 </xsd:complexType>
339 <xsd:complexType name="CT_Fill">
340     <xsd:attributeGroup ref="v:AG_Ext"/>
341     <xsd:attribute name="type" type="ST_FillType"/>
342 </xsd:complexType>
343 <xsd:simpleType name="ST_RType">
344     <xsd:restriction base="xsd:string">
345         <xsd:enumeration value="arc"/>
346         <xsd:enumeration value="callout"/>
347         <xsd:enumeration value="connector"/>
348         <xsd:enumeration value="align"/>
349     </xsd:restriction>
350 </xsd:simpleType>
351 <xsd:simpleType name="ST_How">
352     <xsd:restriction base="xsd:string">
353         <xsd:enumeration value="top"/>
354         <xsd:enumeration value="middle"/>
355         <xsd:enumeration value="bottom"/>
356         <xsd:enumeration value="left"/>
357         <xsd:enumeration value="center"/>
358         <xsd:enumeration value="right"/>
359     </xsd:restriction>
360 </xsd:simpleType>
361 <xsd:simpleType name="ST_BWMode">
362     <xsd:restriction base="xsd:string">
363         <xsd:enumeration value="color"/>
364         <xsd:enumeration value="auto"/>
365         <xsd:enumeration value="grayScale"/>
366         <xsd:enumeration value="lightGrayscale"/>
367         <xsd:enumeration value="inverseGray"/>
368         <xsd:enumeration value="grayOutline"/>
369         <xsd:enumeration value="highContrast"/>
370         <xsd:enumeration value="black"/>
371         <xsd:enumeration value="white"/>
372         <xsd:enumeration value="hide"/>
373         <xsd:enumeration value="undrawn"/>
374         <xsd:enumeration value="blackTextAndLines"/>
375     </xsd:restriction>
376 </xsd:simpleType>
377 <xsd:simpleType name="ST_ScreenSize">
378     <xsd:restriction base="xsd:string">
379         <xsd:enumeration value="544,376"/>
380         <xsd:enumeration value="640,480"/>
381         <xsd:enumeration value="720,512"/>
382         <xsd:enumeration value="800,600"/>
383         <xsd:enumeration value="1024,768"/>
384         <xsd:enumeration value="1152,862"/>
385     </xsd:restriction>

```

```

386 </xsd:simpleType>
387 <xsd:simpleType name="ST_InsetMode">
388   <xsd:restriction base="xsd:string">
389     <xsd:enumeration value="auto"/>
390     <xsd:enumeration value="custom"/>
391   </xsd:restriction>
392 </xsd:simpleType>
393 <xsd:simpleType name="ST_ColorMode">
394   <xsd:restriction base="xsd:string">
395     <xsd:enumeration value="auto"/>
396     <xsd:enumeration value="custom"/>
397   </xsd:restriction>
398 </xsd:simpleType>
399 <xsd:simpleType name="ST_ContentType">
400   <xsd:restriction base="xsd:string"/>
401 </xsd:simpleType>
402 <xsd:simpleType name="ST_DiagramLayout">
403   <xsd:restriction base="xsd:integer">
404     <xsd:enumeration value="0"/>
405     <xsd:enumeration value="1"/>
406     <xsd:enumeration value="2"/>
407     <xsd:enumeration value="3"/>
408   </xsd:restriction>
409 </xsd:simpleType>
410 <xsd:simpleType name="ST_ExtrusionType">
411   <xsd:restriction base="xsd:string">
412     <xsd:enumeration value="perspective"/>
413     <xsd:enumeration value="parallel"/>
414   </xsd:restriction>
415 </xsd:simpleType>
416 <xsd:simpleType name="ST_ExtrusionRender">
417   <xsd:restriction base="xsd:string">
418     <xsd:enumeration value="solid"/>
419     <xsd:enumeration value="wireFrame"/>
420     <xsd:enumeration value="boundingCube"/>
421   </xsd:restriction>
422 </xsd:simpleType>
423 <xsd:simpleType name="ST_ExtrusionPlane">
424   <xsd:restriction base="xsd:string">
425     <xsd:enumeration value="XY"/>
426     <xsd:enumeration value="ZX"/>
427     <xsd:enumeration value="YZ"/>
428   </xsd:restriction>
429 </xsd:simpleType>
430 <xsd:simpleType name="ST_Angle">
431   <xsd:restriction base="xsd:string">
432     <xsd:enumeration value="any"/>
433     <xsd:enumeration value="30"/>
434     <xsd:enumeration value="45"/>
435     <xsd:enumeration value="60"/>
436     <xsd:enumeration value="90"/>
437     <xsd:enumeration value="auto"/>
438   </xsd:restriction>

```

```

439 </xsd:simpleType>
440 <xsd:simpleType name="ST_CalloutDrop">
441   <xsd:restriction base="xsd:string"/>
442 </xsd:simpleType>
443 <xsd:simpleType name="ST_CalloutPlacement">
444   <xsd:restriction base="xsd:string">
445     <xsd:enumeration value="top"/>
446     <xsd:enumeration value="center"/>
447     <xsd:enumeration value="bottom"/>
448     <xsd:enumeration value="user"/>
449   </xsd:restriction>
450 </xsd:simpleType>
451 <xsd:simpleType name="ST_ConnectorType">
452   <xsd:restriction base="xsd:string">
453     <xsd:enumeration value="none"/>
454     <xsd:enumeration value="straight"/>
455     <xsd:enumeration value="elbow"/>
456     <xsd:enumeration value="curved"/>
457   </xsd:restriction>
458 </xsd:simpleType>
459 <xsd:simpleType name="ST_HrAlign">
460   <xsd:restriction base="xsd:string">
461     <xsd:enumeration value="left"/>
462     <xsd:enumeration value="right"/>
463     <xsd:enumeration value="center"/>
464   </xsd:restriction>
465 </xsd:simpleType>
466 <xsd:simpleType name="ST_ConnectType">
467   <xsd:restriction base="xsd:string">
468     <xsd:enumeration value="none"/>
469     <xsd:enumeration value="rect"/>
470     <xsd:enumeration value="segments"/>
471     <xsd:enumeration value="custom"/>
472   </xsd:restriction>
473 </xsd:simpleType>
474 <xsd:simpleType name="ST_OLELinkType">
475   <xsd:restriction base="xsd:string"/>
476 </xsd:simpleType>
477 <xsd:simpleType name="ST_OLEType">
478   <xsd:restriction base="xsd:string">
479     <xsd:enumeration value="Embed"/>
480     <xsd:enumeration value="Link"/>
481   </xsd:restriction>
482 </xsd:simpleType>
483 <xsd:simpleType name="ST_OLEDrawAspect">
484   <xsd:restriction base="xsd:string">
485     <xsd:enumeration value="Content"/>
486     <xsd:enumeration value="Icon"/>
487   </xsd:restriction>
488 </xsd:simpleType>
489 <xsd:simpleType name="ST_OLEUpdateMode">
490   <xsd:restriction base="xsd:string">
491     <xsd:enumeration value="Always"/>

```

```

492         <xsd:enumeration value="OnCall"/>
493     </xsd:restriction>
494 </xsd:simpleType>
495 <xsd:simpleType name="ST_FillType">
496     <xsd:restriction base="xsd:string">
497         <xsd:enumeration value="gradientCenter"/>
498         <xsd:enumeration value="solid"/>
499         <xsd:enumeration value="pattern"/>
500         <xsd:enumeration value="tile"/>
501         <xsd:enumeration value="frame"/>
502         <xsd:enumeration value="gradientUnscaled"/>
503         <xsd:enumeration value="gradientRadial"/>
504         <xsd:enumeration value="gradient"/>
505         <xsd:enumeration value="background"/>
506     </xsd:restriction>
507 </xsd:simpleType>
508 </xsd:schema>

```

A.6.3 VML - WordprocessingML Drawing

This schema is available in the file vml-wordprocessingDrawing.xsd.

```

1  <xsd:schema xmlns="urn:schemas-microsoft-com:office:word"
2  xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:schemas-microsoft-com:office:word"
3  elementFormDefault="qualified" attributeFormDefault="unqualified">
4      <xsd:element name="borderTop" type="CT_Border"/>
5      <xsd:element name="borderLeft" type="CT_Border"/>
6      <xsd:element name="borderRight" type="CT_Border"/>
7      <xsd:element name="borderBottom" type="CT_Border"/>
8      <xsd:complexType name="CT_Border">
9          <xsd:attribute name="type" type="ST_BorderType" use="optional"/>
10         <xsd:attribute name="width" type="xsd:positiveInteger" use="optional"/>
11         <xsd:attribute name="shadow" type="ST_BorderShadow" use="optional"/>
12     </xsd:complexType>
13     <xsd:element name="wrap" type="CT_Wrap"/>
14     <xsd:complexType name="CT_Wrap">
15         <xsd:attribute name="type" type="ST_WrapType" use="optional"/>
16         <xsd:attribute name="side" type="ST_WrapSide" use="optional"/>
17         <xsd:attribute name="anchorx" type="ST_HorizontalAnchor" use="optional"/>
18         <xsd:attribute name="anchory" type="ST_VerticalAnchor" use="optional"/>
19     </xsd:complexType>
20     <xsd:element name="anchorlock" type="CT_AnchorLock"/>
21     <xsd:complexType name="CT_AnchorLock"/>
22     <xsd:simpleType name="ST_BorderType">
23         <xsd:restriction base="xsd:string">
24             <xsd:enumeration value="none"/>
25             <xsd:enumeration value="single"/>
26             <xsd:enumeration value="thick"/>
27             <xsd:enumeration value="double"/>
28             <xsd:enumeration value="hairline"/>
29             <xsd:enumeration value="dot"/>
30             <xsd:enumeration value="dash"/>
31             <xsd:enumeration value="dotDash"/>

```

```

32      <xsd:enumeration value="dashDotDot"/>
33      <xsd:enumeration value="triple"/>
34      <xsd:enumeration value="thinThickSmall"/>
35      <xsd:enumeration value="thickThinSmall"/>
36      <xsd:enumeration value="thickBetweenThinSmall"/>
37      <xsd:enumeration value="thinThick"/>
38      <xsd:enumeration value="thickThin"/>
39      <xsd:enumeration value="thickBetweenThin"/>
40      <xsd:enumeration value="thinThickLarge"/>
41      <xsd:enumeration value="thickThinLarge"/>
42      <xsd:enumeration value="thickBetweenThinLarge"/>
43      <xsd:enumeration value="wave"/>
44      <xsd:enumeration value="doubleWave"/>
45      <xsd:enumeration value="dashedSmall"/>
46      <xsd:enumeration value="dashDotStroked"/>
47      <xsd:enumeration value="threeDEmboss"/>
48      <xsd:enumeration value="threeDEngrave"/>
49      <xsd:enumeration value="HTMLOutset"/>
50      <xsd:enumeration value="HTMLInset"/>
51    </xsd:restriction>
52  </xsd:simpleType>
53  <xsd:simpleType name="ST_BorderShadow">
54    <xsd:restriction base="xsd:string">
55      <xsd:enumeration value="t"/>
56      <xsd:enumeration value="true"/>
57      <xsd:enumeration value="f"/>
58      <xsd:enumeration value="false"/>
59    </xsd:restriction>
60  </xsd:simpleType>
61  <xsd:simpleType name="ST_WrapType">
62    <xsd:restriction base="xsd:string">
63      <xsd:enumeration value="topAndBottom"/>
64      <xsd:enumeration value="square"/>
65      <xsd:enumeration value="none"/>
66      <xsd:enumeration value="tight"/>
67      <xsd:enumeration value="through"/>
68    </xsd:restriction>
69  </xsd:simpleType>
70  <xsd:simpleType name="ST_WrapSide">
71    <xsd:restriction base="xsd:string">
72      <xsd:enumeration value="both"/>
73      <xsd:enumeration value="left"/>
74      <xsd:enumeration value="right"/>
75      <xsd:enumeration value="largest"/>
76    </xsd:restriction>
77  </xsd:simpleType>
78  <xsd:simpleType name="ST_HorizontalAnchor">
79    <xsd:restriction base="xsd:string">
80      <xsd:enumeration value="margin"/>
81      <xsd:enumeration value="page"/>
82      <xsd:enumeration value="text"/>
83      <xsd:enumeration value="char"/>
84    </xsd:restriction>

```

```

85 </xsd:simpleType>
86 <xsd:simpleType name="ST_VerticalAnchor">
87   <xsd:restriction base="xsd:string">
88     <xsd:enumeration value="margin"/>
89     <xsd:enumeration value="page"/>
90     <xsd:enumeration value="text"/>
91     <xsd:enumeration value="line"/>
92   </xsd:restriction>
93 </xsd:simpleType>
94 </xsd:schema>

```

A.6.4 VML - SpreadsheetML Drawing

This schema is available in the file vml-spreadsheetDrawing.xsd.

```

1 <xsd:schema xmlns="urn:schemas-microsoft-com:office:excel"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4   targetNamespace="urn:schemas-microsoft-com:office:excel" elementFormDefault="qualified"
5   attributeFormDefault="unqualified">
6   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7     schemaLocation="shared-commonSimpleTypes.xsd"/>
8   <xsd:element name="ClientData" type="CT_ClientData"/>
9   <xsd:complexType name="CT_ClientData">
10    <xsd:choice minOccurs="0" maxOccurs="unbounded">
11      <xsd:element name="MoveWithCells" type="s:ST_TrueFalseBlank"/>
12      <xsd:element name="SizeWithCells" type="s:ST_TrueFalseBlank"/>
13      <xsd:element name="Anchor" type="xsd:string"/>
14      <xsd:element name="Locked" type="s:ST_TrueFalseBlank"/>
15      <xsd:element name="DefaultSize" type="s:ST_TrueFalseBlank"/>
16      <xsd:element name="PrintObject" type="s:ST_TrueFalseBlank"/>
17      <xsd:element name="Disabled" type="s:ST_TrueFalseBlank"/>
18      <xsd:element name="AutoFill" type="s:ST_TrueFalseBlank"/>
19      <xsd:element name="AutoLine" type="s:ST_TrueFalseBlank"/>
20      <xsd:element name="AutoPict" type="s:ST_TrueFalseBlank"/>
21      <xsd:element name="FmlaMacro" type="xsd:string"/>
22      <xsd:element name="TextHAlign" type="xsd:string"/>
23      <xsd:element name="TextVAlign" type="xsd:string"/>
24      <xsd:element name="LockText" type="s:ST_TrueFalseBlank"/>
25      <xsd:element name="JustLastX" type="s:ST_TrueFalseBlank"/>
26      <xsd:element name="SecretEdit" type="s:ST_TrueFalseBlank"/>
27      <xsd:element name="Default" type="s:ST_TrueFalseBlank"/>
28      <xsd:element name="Help" type="s:ST_TrueFalseBlank"/>
29      <xsd:element name="Cancel" type="s:ST_TrueFalseBlank"/>
30      <xsd:element name="Dismiss" type="s:ST_TrueFalseBlank"/>
31      <xsd:element name="Accel" type="xsd:integer"/>
32      <xsd:element name="Accel2" type="xsd:integer"/>
33      <xsd:element name="Row" type="xsd:integer"/>
34      <xsd:element name="Column" type="xsd:integer"/>
35      <xsd:element name="Visible" type="s:ST_TrueFalseBlank"/>
36      <xsd:element name="RowHidden" type="s:ST_TrueFalseBlank"/>
37      <xsd:element name="ColHidden" type="s:ST_TrueFalseBlank"/>
38      <xsd:element name="VTEdit" type="xsd:integer"/>

```

```

39      <xsd:element name="Multiline" type="s:ST TrueFalseBlank"/>
40      <xsd:element name="VScroll" type="s:ST TrueFalseBlank"/>
41      <xsd:element name="ValidIds" type="s:ST TrueFalseBlank"/>
42      <xsd:element name="FmlaRange" type="xsd:string"/>
43      <xsd:element name="WidthMin" type="xsd:integer"/>
44      <xsd:element name="Sel" type="xsd:integer"/>
45      <xsd:element name="NoThreeD2" type="s:ST TrueFalseBlank"/>
46      <xsd:element name="SelType" type="xsd:string"/>
47      <xsd:element name="MultiSel" type="xsd:string"/>
48      <xsd:element name="LCT" type="xsd:string"/>
49      <xsd:element name="ListItem" type="xsd:string"/>
50      <xsd:element name="DropStyle" type="xsd:string"/>
51      <xsd:element name="Colored" type="s:ST TrueFalseBlank"/>
52      <xsd:element name="DropLines" type="xsd:integer"/>
53      <xsd:element name="Checked" type="xsd:integer"/>
54      <xsd:element name="FmlaLink" type="xsd:string"/>
55      <xsd:element name="FmlaPict" type="xsd:string"/>
56      <xsd:element name="NoThreeD" type="s:ST TrueFalseBlank"/>
57      <xsd:element name="FirstButton" type="s:ST TrueFalseBlank"/>
58      <xsd:element name="FmlaGroup" type="xsd:string"/>
59      <xsd:element name="Val" type="xsd:integer"/>
60      <xsd:element name="Min" type="xsd:integer"/>
61      <xsd:element name="Max" type="xsd:integer"/>
62      <xsd:element name="Inc" type="xsd:integer"/>
63      <xsd:element name="Page" type="xsd:integer"/>
64      <xsd:element name="Horiz" type="s:ST TrueFalseBlank"/>
65      <xsd:element name="Dx" type="xsd:integer"/>
66      <xsd:element name="MapOCX" type="s:ST TrueFalseBlank"/>
67      <xsd:element name="CF" type="ST CF"/>
68      <xsd:element name="Camera" type="s:ST TrueFalseBlank"/>
69      <xsd:element name="RecalcAlways" type="s:ST TrueFalseBlank"/>
70      <xsd:element name="AutoScale" type="s:ST TrueFalseBlank"/>
71      <xsd:element name="DDE" type="s:ST TrueFalseBlank"/>
72      <xsd:element name="UIObj" type="s:ST TrueFalseBlank"/>
73      <xsd:element name="ScriptText" type="xsd:string"/>
74      <xsd:element name="ScriptExtended" type="xsd:string"/>
75      <xsd:element name="ScriptLanguage" type="xsd:nonNegativeInteger"/>
76      <xsd:element name="ScriptLocation" type="xsd:nonNegativeInteger"/>
77      <xsd:element name="FmlaTxbx" type="xsd:string"/>
78    </xsd:choice>
79    <xsd:attribute name="ObjectType" type="ST ObjectType" use="required"/>
80  </xsd:complexType>
81  <xsd:simpleType name="ST_CF">
82    <xsd:restriction base="xsd:string"/>
83  </xsd:simpleType>
84  <xsd:simpleType name="ST_ObjectType">
85    <xsd:restriction base="xsd:string">
86      <xsd:enumeration value="Button"/>
87      <xsd:enumeration value="Checkbox"/>
88      <xsd:enumeration value="Dialog"/>
89      <xsd:enumeration value="Drop"/>
90      <xsd:enumeration value="Edit"/>
91      <xsd:enumeration value="GBox"/>

```



```

92         <xsd:enumeration value="Label"/>
93         <xsd:enumeration value="LineA"/>
94         <xsd:enumeration value="List"/>
95         <xsd:enumeration value="Movie"/>
96         <xsd:enumeration value="Note"/>
97         <xsd:enumeration value="Pict"/>
98         <xsd:enumeration value="Radio"/>
99         <xsd:enumeration value="RectA"/>
100        <xsd:enumeration value="Scroll"/>
101        <xsd:enumeration value="Spin"/>
102        <xsd:enumeration value="Shape"/>
103        <xsd:enumeration value="Group"/>
104        <xsd:enumeration value="Rect"/>
105    </xsd:restriction>
106 </xsd:simpleType>
107 </xsd:schema>

```

A.6.5 VML - PresentationML Drawing

This schema is available in the file vml-presentationDrawing.xsd.

```

1 <xsd:schema xmlns="urn:schemas-microsoft-com:office:powerpoint"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:schemas-microsoft-
3   com:office:powerpoint" elementFormDefault="qualified" attributeFormDefault="unqualified">
4   <xsd:element name="iscomment" type="CT_Empty"/>
5   <xsd:element name="textdata" type="CT_Rel"/>
6   <xsd:complexType name="CT_Empty"/>
7   <xsd:complexType name="CT_Rel">
8       <xsd:attribute name="id" type="xsd:string"/>
9   </xsd:complexType>
10 </xsd:schema>

```

A.7 Shared MLs

A.7.1 Math

This schema is available in the file shared-math.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/officeDocument/2006/math"
3   xmlns:m="http://schemas.openxmlformats.org/officeDocument/2006/math"
4   xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
5   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6   elementFormDefault="qualified" attributeFormDefault="qualified" blockDefault="#all"
7   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/math">
8   <xsd:import namespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
9       schemaLocation="wml.xsd"/>
10  <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11      schemaLocation="shared-commonSimpleTypes.xsd"/>
12  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"/>
13  <xsd:simpleType name="ST_Integer255">
14      <xsd:restriction base="xsd:integer">
15          <xsd:minInclusive value="1"/>

```

```

16         <xsd:maxInclusive value="255"/>
17     </xsd:restriction>
18 </xsd:simpleType>
19 <xsd:complexType name="CT_Integer255">
20     <xsd:attribute name="val" type="ST_Integer255" use="required"/>
21 </xsd:complexType>
22 <xsd:simpleType name="ST_Integer2">
23     <xsd:restriction base="xsd:integer">
24         <xsd:minInclusive value="-2"/>
25         <xsd:maxInclusive value="2"/>
26     </xsd:restriction>
27 </xsd:simpleType>
28 <xsd:complexType name="CT_Integer2">
29     <xsd:attribute name="val" type="ST_Integer2" use="required"/>
30 </xsd:complexType>
31 <xsd:simpleType name="ST_SpacingRule">
32     <xsd:restriction base="xsd:integer">
33         <xsd:minInclusive value="0"/>
34         <xsd:maxInclusive value="4"/>
35     </xsd:restriction>
36 </xsd:simpleType>
37 <xsd:complexType name="CT_SpacingRule">
38     <xsd:attribute name="val" type="ST_SpacingRule" use="required"/>
39 </xsd:complexType>
40 <xsd:simpleType name="ST_UnSignedInteger">
41     <xsd:restriction base="xsd:unsignedInt"/>
42 </xsd:simpleType>
43 <xsd:complexType name="CT_UnSignedInteger">
44     <xsd:attribute name="val" type="ST_UnSignedInteger" use="required"/>
45 </xsd:complexType>
46 <xsd:simpleType name="ST_Char">
47     <xsd:restriction base="xsd:string">
48         <xsd:maxLength value="1"/>
49     </xsd:restriction>
50 </xsd:simpleType>
51 <xsd:complexType name="CT_Char">
52     <xsd:attribute name="val" type="ST_Char" use="required"/>
53 </xsd:complexType>
54 <xsd:complexType name="CT_OnOff">
55     <xsd:attribute name="val" type="s:ST_OnOff"/>
56 </xsd:complexType>
57 <xsd:complexType name="CT_String">
58     <xsd:attribute name="val" type="s:ST_String"/>
59 </xsd:complexType>
60 <xsd:complexType name="CT_XAlign">
61     <xsd:attribute name="val" type="s:ST_XAlign" use="required"/>
62 </xsd:complexType>
63 <xsd:complexType name="CT_YAlign">
64     <xsd:attribute name="val" type="s:ST_YAlign" use="required"/>
65 </xsd:complexType>
66 <xsd:simpleType name="ST_Shp">
67     <xsd:restriction base="xsd:string">
68         <xsd:enumeration value="centered"/>

```

```

69         <xsd:enumeration value="match"/>
70     </xsd:restriction>
71 </xsd:simpleType>
72 <xsd:complexType name="CT_Shp">
73     <xsd:attribute name="val" type="ST_Shp" use="required"/>
74 </xsd:complexType>
75 <xsd:simpleType name="ST_FType">
76     <xsd:restriction base="xsd:string">
77         <xsd:enumeration value="bar"/>
78         <xsd:enumeration value="skw"/>
79         <xsd:enumeration value="lin"/>
80         <xsd:enumeration value="noBar"/>
81     </xsd:restriction>
82 </xsd:simpleType>
83 <xsd:complexType name="CT_FType">
84     <xsd:attribute name="val" type="ST_FType" use="required"/>
85 </xsd:complexType>
86 <xsd:simpleType name="ST_LimLoc">
87     <xsd:restriction base="xsd:string">
88         <xsd:enumeration value="undOvr"/>
89         <xsd:enumeration value="subSup"/>
90     </xsd:restriction>
91 </xsd:simpleType>
92 <xsd:complexType name="CT_LimLoc">
93     <xsd:attribute name="val" type="ST_LimLoc" use="required"/>
94 </xsd:complexType>
95 <xsd:simpleType name="ST_TopBot">
96     <xsd:restriction base="xsd:string">
97         <xsd:enumeration value="top"/>
98         <xsd:enumeration value="bot"/>
99     </xsd:restriction>
100 </xsd:simpleType>
101 <xsd:complexType name="CT_TopBot">
102     <xsd:attribute name="val" type="ST_TopBot" use="required"/>
103 </xsd:complexType>
104 <xsd:simpleType name="ST_Script">
105     <xsd:restriction base="xsd:string">
106         <xsd:enumeration value="roman"/>
107         <xsd:enumeration value="script"/>
108         <xsd:enumeration value="fraktur"/>
109         <xsd:enumeration value="double-struck"/>
110         <xsd:enumeration value="sans-serif"/>
111         <xsd:enumeration value="monospace"/>
112     </xsd:restriction>
113 </xsd:simpleType>
114 <xsd:complexType name="CT_Script">
115     <xsd:attribute name="val" type="ST_Script"/>
116 </xsd:complexType>
117 <xsd:simpleType name="ST_Style">
118     <xsd:restriction base="xsd:string">
119         <xsd:enumeration value="p"/>
120         <xsd:enumeration value="b"/>
121         <xsd:enumeration value="i"/>

```

```

122     <xsd:enumeration value="bi"/>
123   </xsd:restriction>
124 </xsd:simpleType>
125 <xsd:complexType name="CT_Style">
126   <xsd:attribute name="val" type="ST_Style"/>
127 </xsd:complexType>
128 <xsd:complexType name="CT_ManualBreak">
129   <xsd:attribute name="alnAt" type="ST_Integer255"/>
130 </xsd:complexType>
131 <xsd:group name="EG_ScriptStyle">
132   <xsd:sequence>
133     <xsd:element name="scr" minOccurs="0" type="CT_Script"/>
134     <xsd:element name="sty" minOccurs="0" type="CT_Style"/>
135   </xsd:sequence>
136 </xsd:group>
137 <xsd:complexType name="CT_RPR">
138   <xsd:sequence>
139     <xsd:element name="lit" minOccurs="0" type="CT_OnOff"/>
140     <xsd:choice>
141       <xsd:element name="nor" minOccurs="0" type="CT_OnOff"/>
142       <xsd:sequence>
143         <xsd:group ref="EG_ScriptStyle"/>
144       </xsd:sequence>
145     </xsd:choice>
146     <xsd:element name="brk" minOccurs="0" type="CT_ManualBreak"/>
147     <xsd:element name="aln" minOccurs="0" type="CT_OnOff"/>
148   </xsd:sequence>
149 </xsd:complexType>
150 <xsd:complexType name="CT_Text">
151   <xsd:simpleContent>
152     <xsd:extension base="s:ST_String">
153       <xsd:attribute ref="xml:space" use="optional"/>
154     </xsd:extension>
155   </xsd:simpleContent>
156 </xsd:complexType>
157 <xsd:complexType name="CT_R">
158   <xsd:sequence>
159     <xsd:element name="rPr" type="CT_RPR" minOccurs="0"/>
160     <xsd:group ref="w:EG_RPr" minOccurs="0"/>
161     <xsd:choice minOccurs="0" maxOccurs="unbounded">
162       <xsd:group ref="w:EG_RunInnerContent"/>
163       <xsd:element name="t" type="CT_Text" minOccurs="0"/>
164     </xsd:choice>
165   </xsd:sequence>
166 </xsd:complexType>
167 <xsd:complexType name="CT_CtrlPr">
168   <xsd:sequence>
169     <xsd:group ref="w:EG_RPrMath" minOccurs="0"/>
170   </xsd:sequence>
171 </xsd:complexType>
172 <xsd:complexType name="CT_AccPr">
173   <xsd:sequence>
174     <xsd:element name="chr" type="CT_Char" minOccurs="0"/>

```

```

175     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
176   </xsd:sequence>
177 </xsd:complexType>
178 <xsd:complexType name="CT_Acc">
179   <xsd:sequence>
180     <xsd:element name="accPr" type="CT_AccPr" minOccurs="0"/>
181     <xsd:element name="e" type="CT_OMathArg"/>
182   </xsd:sequence>
183 </xsd:complexType>
184 <xsd:complexType name="CT_BarPr">
185   <xsd:sequence>
186     <xsd:element name="pos" type="CT_TopBot" minOccurs="0"/>
187     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
188   </xsd:sequence>
189 </xsd:complexType>
190 <xsd:complexType name="CT_Bar">
191   <xsd:sequence>
192     <xsd:element name="barPr" type="CT_BarPr" minOccurs="0"/>
193     <xsd:element name="e" type="CT_OMathArg"/>
194   </xsd:sequence>
195 </xsd:complexType>
196 <xsd:complexType name="CT_BoxPr">
197   <xsd:sequence>
198     <xsd:element name="opEmu" type="CT_OnOff" minOccurs="0"/>
199     <xsd:element name="noBreak" type="CT_OnOff" minOccurs="0"/>
200     <xsd:element name="diff" type="CT_OnOff" minOccurs="0"/>
201     <xsd:element name="brk" type="CT_ManualBreak" minOccurs="0"/>
202     <xsd:element name="aln" type="CT_OnOff" minOccurs="0"/>
203     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
204   </xsd:sequence>
205 </xsd:complexType>
206 <xsd:complexType name="CT_Box">
207   <xsd:sequence>
208     <xsd:element name="boxPr" type="CT_BoxPr" minOccurs="0"/>
209     <xsd:element name="e" type="CT_OMathArg"/>
210   </xsd:sequence>
211 </xsd:complexType>
212 <xsd:complexType name="CT_BorderBoxPr">
213   <xsd:sequence>
214     <xsd:element name="hideTop" type="CT_OnOff" minOccurs="0"/>
215     <xsd:element name="hideBot" type="CT_OnOff" minOccurs="0"/>
216     <xsd:element name="hideLeft" type="CT_OnOff" minOccurs="0"/>
217     <xsd:element name="hideRight" type="CT_OnOff" minOccurs="0"/>
218     <xsd:element name="strikeH" type="CT_OnOff" minOccurs="0"/>
219     <xsd:element name="strikeV" type="CT_OnOff" minOccurs="0"/>
220     <xsd:element name="strikeBLTR" type="CT_OnOff" minOccurs="0"/>
221     <xsd:element name="strikeTLBR" type="CT_OnOff" minOccurs="0"/>
222     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
223   </xsd:sequence>
224 </xsd:complexType>
225 <xsd:complexType name="CT_BorderBox">
226   <xsd:sequence>
227     <xsd:element name="borderBoxPr" type="CT_BorderBoxPr" minOccurs="0"/>

```

```

228     <xsd:element name="e" type="CT_OMathArg"/>
229   </xsd:sequence>
230 </xsd:complexType>
231 <xsd:complexType name="CT_DPr">
232   <xsd:sequence>
233     <xsd:element name="begChr" type="CT_Char" minOccurs="0"/>
234     <xsd:element name="sepChr" type="CT_Char" minOccurs="0"/>
235     <xsd:element name="endChr" type="CT_Char" minOccurs="0"/>
236     <xsd:element name="grow" type="CT_OnOff" minOccurs="0"/>
237     <xsd:element name="shp" type="CT_Shp" minOccurs="0"/>
238     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
239   </xsd:sequence>
240 </xsd:complexType>
241 <xsd:complexType name="CT_D">
242   <xsd:sequence>
243     <xsd:element name="dPr" type="CT_DPr" minOccurs="0"/>
244     <xsd:element name="e" type="CT_OMathArg" maxOccurs="unbounded"/>
245   </xsd:sequence>
246 </xsd:complexType>
247 <xsd:complexType name="CT_EqArrPr">
248   <xsd:sequence>
249     <xsd:element name="baseJc" type="CT_VAlign" minOccurs="0"/>
250     <xsd:element name="maxDist" type="CT_OnOff" minOccurs="0"/>
251     <xsd:element name="objDist" type="CT_OnOff" minOccurs="0"/>
252     <xsd:element name="rSpRule" type="CT_SpacingRule" minOccurs="0"/>
253     <xsd:element name="rSp" type="CT_UnSignedInteger" minOccurs="0"/>
254     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
255   </xsd:sequence>
256 </xsd:complexType>
257 <xsd:complexType name="CT_EqArr">
258   <xsd:sequence>
259     <xsd:element name="eqArrPr" type="CT_EqArrPr" minOccurs="0"/>
260     <xsd:element name="e" type="CT_OMathArg" maxOccurs="unbounded"/>
261   </xsd:sequence>
262 </xsd:complexType>
263 <xsd:complexType name="CT_FPr">
264   <xsd:sequence>
265     <xsd:element name="type" type="CT_FType" minOccurs="0"/>
266     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
267   </xsd:sequence>
268 </xsd:complexType>
269 <xsd:complexType name="CT_F">
270   <xsd:sequence>
271     <xsd:element name="fPr" type="CT_FPr" minOccurs="0"/>
272     <xsd:element name="num" type="CT_OMathArg"/>
273     <xsd:element name="den" type="CT_OMathArg"/>
274   </xsd:sequence>
275 </xsd:complexType>
276 <xsd:complexType name="CT_FuncPr">
277   <xsd:sequence>
278     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
279   </xsd:sequence>
280 </xsd:complexType>

```

```

281 <xsd:complexType name="CT_Func">
282   <xsd:sequence>
283     <xsd:element name="funcPr" type="CT_FuncPr" minOccurs="0"/>
284     <xsd:element name="fName" type="CT_OMathArg"/>
285     <xsd:element name="e" type="CT_OMathArg"/>
286   </xsd:sequence>
287 </xsd:complexType>
288 <xsd:complexType name="CT_GroupChrPr">
289   <xsd:sequence>
290     <xsd:element name="chr" type="CT_Char" minOccurs="0"/>
291     <xsd:element name="pos" type="CT_TopBot" minOccurs="0"/>
292     <xsd:element name="vertJc" type="CT_TopBot" minOccurs="0"/>
293     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
294   </xsd:sequence>
295 </xsd:complexType>
296 <xsd:complexType name="CT_GroupChr">
297   <xsd:sequence>
298     <xsd:element name="groupChrPr" type="CT_GroupChrPr" minOccurs="0"/>
299     <xsd:element name="e" type="CT_OMathArg"/>
300   </xsd:sequence>
301 </xsd:complexType>
302 <xsd:complexType name="CT_LimLowPr">
303   <xsd:sequence>
304     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
305   </xsd:sequence>
306 </xsd:complexType>
307 <xsd:complexType name="CT_LimLow">
308   <xsd:sequence>
309     <xsd:element name="limLowPr" type="CT_LimLowPr" minOccurs="0"/>
310     <xsd:element name="e" type="CT_OMathArg"/>
311     <xsd:element name="lim" type="CT_OMathArg"/>
312   </xsd:sequence>
313 </xsd:complexType>
314 <xsd:complexType name="CT_LimUppPr">
315   <xsd:sequence>
316     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
317   </xsd:sequence>
318 </xsd:complexType>
319 <xsd:complexType name="CT_LimUpp">
320   <xsd:sequence>
321     <xsd:element name="limUppPr" type="CT_LimUppPr" minOccurs="0"/>
322     <xsd:element name="e" type="CT_OMathArg"/>
323     <xsd:element name="lim" type="CT_OMathArg"/>
324   </xsd:sequence>
325 </xsd:complexType>
326 <xsd:complexType name="CT_MCPr">
327   <xsd:sequence>
328     <xsd:element name="count" type="CT_Integer255" minOccurs="0"/>
329     <xsd:element name="mcJc" type="CT_XAlign" minOccurs="0"/>
330   </xsd:sequence>
331 </xsd:complexType>
332 <xsd:complexType name="CT_MC">
333   <xsd:sequence>

```

```

334     <xsd:element name="mcPr" type="CT_MCPPr" minOccurs="0"/>
335   </xsd:sequence>
336 </xsd:complexType>
337 <xsd:complexType name="CT_MCS">
338   <xsd:sequence>
339     <xsd:element name="mc" type="CT_MC" maxOccurs="unbounded"/>
340   </xsd:sequence>
341 </xsd:complexType>
342 <xsd:complexType name="CT_MPr">
343   <xsd:sequence>
344     <xsd:element name="baseJc" type="CT_YAlign" minOccurs="0"/>
345     <xsd:element name="plcHide" type="CT_OnOff" minOccurs="0"/>
346     <xsd:element name="rSpRule" type="CT_SpacingRule" minOccurs="0"/>
347     <xsd:element name="cGpRule" type="CT_SpacingRule" minOccurs="0"/>
348     <xsd:element name="rSp" type="CT_UnSignedInteger" minOccurs="0"/>
349     <xsd:element name="cSp" type="CT_UnSignedInteger" minOccurs="0"/>
350     <xsd:element name="cGp" type="CT_UnSignedInteger" minOccurs="0"/>
351     <xsd:element name="mcs" type="CT_MCS" minOccurs="0"/>
352     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
353   </xsd:sequence>
354 </xsd:complexType>
355 <xsd:complexType name="CT_MR">
356   <xsd:sequence>
357     <xsd:element name="e" type="CT_OMathArg" maxOccurs="unbounded"/>
358   </xsd:sequence>
359 </xsd:complexType>
360 <xsd:complexType name="CT_M">
361   <xsd:sequence>
362     <xsd:element name="mPr" type="CT_MPr" minOccurs="0"/>
363     <xsd:element name="mr" type="CT_MR" maxOccurs="unbounded"/>
364   </xsd:sequence>
365 </xsd:complexType>
366 <xsd:complexType name="CT_NaryPr">
367   <xsd:sequence>
368     <xsd:element name="chr" type="CT_Char" minOccurs="0"/>
369     <xsd:element name="limLoc" type="CT_LimLoc" minOccurs="0"/>
370     <xsd:element name="grow" type="CT_OnOff" minOccurs="0"/>
371     <xsd:element name="subHide" type="CT_OnOff" minOccurs="0"/>
372     <xsd:element name="supHide" type="CT_OnOff" minOccurs="0"/>
373     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
374   </xsd:sequence>
375 </xsd:complexType>
376 <xsd:complexType name="CT_Nary">
377   <xsd:sequence>
378     <xsd:element name="naryPr" type="CT_NaryPr" minOccurs="0"/>
379     <xsd:element name="sub" type="CT_OMathArg"/>
380     <xsd:element name="sup" type="CT_OMathArg"/>
381     <xsd:element name="e" type="CT_OMathArg"/>
382   </xsd:sequence>
383 </xsd:complexType>
384 <xsd:complexType name="CT_PhantPr">
385   <xsd:sequence>
386     <xsd:element name="show" type="CT_OnOff" minOccurs="0"/>

```



```

387     <xsd:element name="zeroWid" type="CT_OnOff" minOccurs="0"/>
388     <xsd:element name="zeroAsc" type="CT_OnOff" minOccurs="0"/>
389     <xsd:element name="zeroDesc" type="CT_OnOff" minOccurs="0"/>
390     <xsd:element name="transp" type="CT_OnOff" minOccurs="0"/>
391     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
392   </xsd:sequence>
393 </xsd:complexType>
394 <xsd:complexType name="CT_Phant">
395   <xsd:sequence>
396     <xsd:element name="phantPr" type="CT_PhantPr" minOccurs="0"/>
397     <xsd:element name="e" type="CT_OMathArg"/>
398   </xsd:sequence>
399 </xsd:complexType>
400 <xsd:complexType name="CT_RadPr">
401   <xsd:sequence>
402     <xsd:element name="degHide" type="CT_OnOff" minOccurs="0"/>
403     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
404   </xsd:sequence>
405 </xsd:complexType>
406 <xsd:complexType name="CT_Rad">
407   <xsd:sequence>
408     <xsd:element name="radPr" type="CT_RadPr" minOccurs="0"/>
409     <xsd:element name="deg" type="CT_OMathArg"/>
410     <xsd:element name="e" type="CT_OMathArg"/>
411   </xsd:sequence>
412 </xsd:complexType>
413 <xsd:complexType name="CT_SPrePr">
414   <xsd:sequence>
415     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
416   </xsd:sequence>
417 </xsd:complexType>
418 <xsd:complexType name="CT_SPre">
419   <xsd:sequence>
420     <xsd:element name="sPrePr" type="CT_SPrePr" minOccurs="0"/>
421     <xsd:element name="sub" type="CT_OMathArg"/>
422     <xsd:element name="sup" type="CT_OMathArg"/>
423     <xsd:element name="e" type="CT_OMathArg"/>
424   </xsd:sequence>
425 </xsd:complexType>
426 <xsd:complexType name="CT_SSubPr">
427   <xsd:sequence>
428     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
429   </xsd:sequence>
430 </xsd:complexType>
431 <xsd:complexType name="CT_SSub">
432   <xsd:sequence>
433     <xsd:element name="sSubPr" type="CT_SSubPr" minOccurs="0"/>
434     <xsd:element name="e" type="CT_OMathArg"/>
435     <xsd:element name="sub" type="CT_OMathArg"/>
436   </xsd:sequence>
437 </xsd:complexType>
438 <xsd:complexType name="CT_SSubSupPr">
439   <xsd:sequence>

```

```

440     <xsd:element name="alnScr" type="CT_OnOff" minOccurs="0"/>
441     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
442   </xsd:sequence>
443 </xsd:complexType>
444 <xsd:complexType name="CT_SSubSup">
445   <xsd:sequence>
446     <xsd:element name="sSubSupPr" type="CT_SSubSupPr" minOccurs="0"/>
447     <xsd:element name="e" type="CT_OMathArg"/>
448     <xsd:element name="sub" type="CT_OMathArg"/>
449     <xsd:element name="sup" type="CT_OMathArg"/>
450   </xsd:sequence>
451 </xsd:complexType>
452 <xsd:complexType name="CT_SSupPr">
453   <xsd:sequence>
454     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
455   </xsd:sequence>
456 </xsd:complexType>
457 <xsd:complexType name="CT_SSup">
458   <xsd:sequence>
459     <xsd:element name="sSupPr" type="CT_SSupPr" minOccurs="0"/>
460     <xsd:element name="e" type="CT_OMathArg"/>
461     <xsd:element name="sup" type="CT_OMathArg"/>
462   </xsd:sequence>
463 </xsd:complexType>
464 <xsd:group name="EG_OMathMathElements">
465   <xsd:choice>
466     <xsd:element name="acc" type="CT_Acc"/>
467     <xsd:element name="bar" type="CT_Bar"/>
468     <xsd:element name="box" type="CT_Box"/>
469     <xsd:element name="borderBox" type="CT_BorderBox"/>
470     <xsd:element name="d" type="CT_D"/>
471     <xsd:element name="eqArr" type="CT_EqArr"/>
472     <xsd:element name="f" type="CT_F"/>
473     <xsd:element name="func" type="CT_Func"/>
474     <xsd:element name="groupChr" type="CT_GroupChr"/>
475     <xsd:element name="limLow" type="CT_LimLow"/>
476     <xsd:element name="limUpp" type="CT_LimUpp"/>
477     <xsd:element name="m" type="CT_M"/>
478     <xsd:element name="nary" type="CT_Nary"/>
479     <xsd:element name="phant" type="CT_Phant"/>
480     <xsd:element name="rad" type="CT_Rad"/>
481     <xsd:element name="sPre" type="CT_SPre"/>
482     <xsd:element name="sSub" type="CT_SSub"/>
483     <xsd:element name="sSubSup" type="CT_SSubSup"/>
484     <xsd:element name="sSup" type="CT_SSup"/>
485     <xsd:element name="r" type="CT_R"/>
486   </xsd:choice>
487 </xsd:group>
488 <xsd:group name="EG_OMathElements">
489   <xsd:choice>
490     <xsd:group ref="EG_OMathMathElements"/>
491     <xsd:group ref="w:EG_PContentMath"/>
492   </xsd:choice>

```

```

493 </xsd:group>
494 <xsd:complexType name="CT_OMathArgPr">
495   <xsd:sequence>
496     <xsd:element name="argSz" type="CT_Integer2" minOccurs="0"/>
497   </xsd:sequence>
498 </xsd:complexType>
499 <xsd:complexType name="CT_OMathArg">
500   <xsd:sequence>
501     <xsd:element name="argPr" type="CT_OMathArgPr" minOccurs="0"/>
502     <xsd:group ref="EG_OMathElements" minOccurs="0" maxOccurs="unbounded"/>
503     <xsd:element name="ctrlPr" type="CT_CtrlPr" minOccurs="0"/>
504   </xsd:sequence>
505 </xsd:complexType>
506 <xsd:simpleType name="ST_Jc">
507   <xsd:restriction base="xsd:string">
508     <xsd:enumeration value="left"/>
509     <xsd:enumeration value="right"/>
510     <xsd:enumeration value="center"/>
511     <xsd:enumeration value="centerGroup"/>
512   </xsd:restriction>
513 </xsd:simpleType>
514 <xsd:complexType name="CT_OMathJc">
515   <xsd:attribute name="val" type="ST_Jc"/>
516 </xsd:complexType>
517 <xsd:complexType name="CT_OMathParaPr">
518   <xsd:sequence>
519     <xsd:element name="jc" type="CT_OMathJc" minOccurs="0"/>
520   </xsd:sequence>
521 </xsd:complexType>
522 <xsd:complexType name="CT_TwipsMeasure">
523   <xsd:attribute name="val" type="s:ST_TwipsMeasure" use="required"/>
524 </xsd:complexType>
525 <xsd:simpleType name="ST_BreakBin">
526   <xsd:restriction base="xsd:string">
527     <xsd:enumeration value="before"/>
528     <xsd:enumeration value="after"/>
529     <xsd:enumeration value="repeat"/>
530   </xsd:restriction>
531 </xsd:simpleType>
532 <xsd:complexType name="CT_BreakBin">
533   <xsd:attribute name="val" type="ST_BreakBin"/>
534 </xsd:complexType>
535 <xsd:simpleType name="ST_BreakBinSub">
536   <xsd:restriction base="xsd:string">
537     <xsd:enumeration value="--"/>
538     <xsd:enumeration value="-+"/>
539     <xsd:enumeration value="+-"/>
540   </xsd:restriction>
541 </xsd:simpleType>
542 <xsd:complexType name="CT_BreakBinSub">
543   <xsd:attribute name="val" type="ST_BreakBinSub"/>
544 </xsd:complexType>
545 <xsd:complexType name="CT_MathPr">

```

```

546     <xsd:sequence>
547         <xsd:element name="mathFont" type="CT_String" minOccurs="0"/>
548         <xsd:element name="brkBin" type="CT_BreakBin" minOccurs="0"/>
549         <xsd:element name="brkBinSub" type="CT_BreakBinSub" minOccurs="0"/>
550         <xsd:element name="smallFrac" type="CT_OnOff" minOccurs="0"/>
551         <xsd:element name="dispDef" type="CT_OnOff" minOccurs="0"/>
552         <xsd:element name="lMargin" type="CT_TwipsMeasure" minOccurs="0"/>
553         <xsd:element name="rMargin" type="CT_TwipsMeasure" minOccurs="0"/>
554         <xsd:element name="defJc" type="CT_OMathJc" minOccurs="0"/>
555         <xsd:element name="preSp" type="CT_TwipsMeasure" minOccurs="0"/>
556         <xsd:element name="postSp" type="CT_TwipsMeasure" minOccurs="0"/>
557         <xsd:element name="interSp" type="CT_TwipsMeasure" minOccurs="0"/>
558         <xsd:element name="intraSp" type="CT_TwipsMeasure" minOccurs="0"/>
559         <xsd:choice minOccurs="0">
560             <xsd:element name="wrapIndent" type="CT_TwipsMeasure"/>
561             <xsd:element name="wrapRight" type="CT_OnOff"/>
562         </xsd:choice>
563         <xsd:element name="intLim" type="CT_LimLoc" minOccurs="0"/>
564         <xsd:element name="naryLim" type="CT_LimLoc" minOccurs="0"/>
565     </xsd:sequence>
566 </xsd:complexType>
567 <xsd:element name="mathPr" type="CT_MathPr"/>
568 <xsd:complexType name="CT_OMathPara">
569     <xsd:sequence>
570         <xsd:element name="oMathParaPr" type="CT_OMathParaPr" minOccurs="0"/>
571         <xsd:element name="oMath" type="CT_OMath" maxOccurs="unbounded"/>
572     </xsd:sequence>
573 </xsd:complexType>
574 <xsd:complexType name="CT_OMath">
575     <xsd:sequence>
576         <xsd:group ref="EG_OMathElements" minOccurs="0" maxOccurs="unbounded"/>
577     </xsd:sequence>
578 </xsd:complexType>
579 <xsd:element name="oMathPara" type="CT_OMathPara"/>
580 <xsd:element name="oMath" type="CT_OMath"/>
581 </xsd:schema>

```

A.7.2 Extended Properties

This schema is available in the file `shared-documentPropertiesExtended.xsd`.

```

1  <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
2  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3  xmlns:vt="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
4  targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
5  elementFormDefault="qualified" blockDefault="#all">
6      <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
7      schemaLocation="shared-documentPropertiesVariantTypes.xsd"/>
8      <xsd:element name="Properties" type="CT_Properties"/>
9      <xsd:complexType name="CT_Properties">
10         <xsd:all>
11             <xsd:element name="Template" minOccurs="0" maxOccurs="1" type="xsd:string"/>
12             <xsd:element name="Manager" minOccurs="0" maxOccurs="1" type="xsd:string"/>

```

```

13      <xsd:element name="Company" minOccurs="0" maxOccurs="1" type="xsd:string"/>
14      <xsd:element name="Pages" minOccurs="0" maxOccurs="1" type="xsd:int"/>
15      <xsd:element name="Words" minOccurs="0" maxOccurs="1" type="xsd:int"/>
16      <xsd:element name="Characters" minOccurs="0" maxOccurs="1" type="xsd:int"/>
17      <xsd:element name="PresentationFormat" minOccurs="0" maxOccurs="1" type="xsd:string"/>
18      <xsd:element name="Lines" minOccurs="0" maxOccurs="1" type="xsd:int"/>
19      <xsd:element name="Paragraphs" minOccurs="0" maxOccurs="1" type="xsd:int"/>
20      <xsd:element name="Slides" minOccurs="0" maxOccurs="1" type="xsd:int"/>
21      <xsd:element name="Notes" minOccurs="0" maxOccurs="1" type="xsd:int"/>
22      <xsd:element name="TotalTime" minOccurs="0" maxOccurs="1" type="xsd:int"/>
23      <xsd:element name="HiddenSlides" minOccurs="0" maxOccurs="1" type="xsd:int"/>
24      <xsd:element name="MMClips" minOccurs="0" maxOccurs="1" type="xsd:int"/>
25      <xsd:element name="ScaleCrop" minOccurs="0" maxOccurs="1" type="xsd:boolean"/>
26      <xsd:element name="HeadingPairs" minOccurs="0" maxOccurs="1" type="CT_VectorVariant"/>
27      <xsd:element name="TitlesOfParts" minOccurs="0" maxOccurs="1" type="CT_VectorLpstr"/>
28      <xsd:element name="LinksUpToDate" minOccurs="0" maxOccurs="1" type="xsd:boolean"/>
29      <xsd:element name="CharactersWithSpaces" minOccurs="0" maxOccurs="1" type="xsd:int"/>
30      <xsd:element name="SharedDoc" minOccurs="0" maxOccurs="1" type="xsd:boolean"/>
31      <xsd:element name="HyperlinkBase" minOccurs="0" maxOccurs="1" type="xsd:string"/>
32      <xsd:element name="HLinks" minOccurs="0" maxOccurs="1" type="CT_VectorVariant"/>
33      <xsd:element name="HyperlinksChanged" minOccurs="0" maxOccurs="1" type="xsd:boolean"/>
34      <xsd:element name="DigSig" minOccurs="0" maxOccurs="1" type="CT_DigSigBlob"/>
35      <xsd:element name="Application" minOccurs="0" maxOccurs="1" type="xsd:string"/>
36      <xsd:element name="AppVersion" minOccurs="0" maxOccurs="1" type="xsd:string"/>
37      <xsd:element name="DocSecurity" minOccurs="0" maxOccurs="1" type="xsd:int"/>
38  </xsd:all>
39  </xsd:complexType>
40  <xsd:complexType name="CT_VectorVariant">
41    <xsd:sequence minOccurs="1" maxOccurs="1">
42      <xsd:element ref="vt:vector"/>
43    </xsd:sequence>
44  </xsd:complexType>
45  <xsd:complexType name="CT_VectorLpstr">
46    <xsd:sequence minOccurs="1" maxOccurs="1">
47      <xsd:element ref="vt:vector"/>
48    </xsd:sequence>
49  </xsd:complexType>
50  <xsd:complexType name="CT_DigSigBlob">
51    <xsd:sequence minOccurs="1" maxOccurs="1">
52      <xsd:element ref="vt:blob"/>
53    </xsd:sequence>
54  </xsd:complexType>
55 </xsd:schema>

```

A.7.3 Custom Properties

This schema is available in the file shared-documentPropertiesCustom.xsd.

```

1 <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/custom-properties"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   xmlns:vt="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
4   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"

```

```

5 targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/custom-properties"
6 blockDefault="#all" elementFormDefault="qualified">
7   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
8     schemaLocation="shared-documentPropertiesVariantTypes.xsd"/>
9   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
10     schemaLocation="shared-commonSimpleTypes.xsd"/>
11   <xsd:element name="Properties" type="CT_Properties"/>
12   <xsd:complexType name="CT_Properties">
13     <xsd:sequence>
14       <xsd:element name="property" minOccurs="0" maxOccurs="unbounded" type="CT_Property"/>
15     </xsd:sequence>
16   </xsd:complexType>
17   <xsd:complexType name="CT_Property">
18     <xsd:choice minOccurs="1" maxOccurs="1">
19       <xsd:element ref="vt:vector"/>
20       <xsd:element ref="vt:array"/>
21       <xsd:element ref="vt:blob"/>
22       <xsd:element ref="vt:oblob"/>
23       <xsd:element ref="vt:empty"/>
24       <xsd:element ref="vt:null"/>
25       <xsd:element ref="vt:i1"/>
26       <xsd:element ref="vt:i2"/>
27       <xsd:element ref="vt:i4"/>
28       <xsd:element ref="vt:i8"/>
29       <xsd:element ref="vt:int"/>
30       <xsd:element ref="vt:ui1"/>
31       <xsd:element ref="vt:ui2"/>
32       <xsd:element ref="vt:ui4"/>
33       <xsd:element ref="vt:ui8"/>
34       <xsd:element ref="vt:uint"/>
35       <xsd:element ref="vt:r4"/>
36       <xsd:element ref="vt:r8"/>
37       <xsd:element ref="vt:decimal"/>
38       <xsd:element ref="vt:lpstr"/>
39       <xsd:element ref="vt:lpwstr"/>
40       <xsd:element ref="vt:bstr"/>
41       <xsd:element ref="vt:date"/>
42       <xsd:element ref="vt:filetime"/>
43       <xsd:element ref="vt:bool"/>
44       <xsd:element ref="vt:cy"/>
45       <xsd:element ref="vt:error"/>
46       <xsd:element ref="vt:stream"/>
47       <xsd:element ref="vt:ostream"/>
48       <xsd:element ref="vt:storage"/>
49       <xsd:element ref="vt:ostorage"/>
50       <xsd:element ref="vt:vstream"/>
51       <xsd:element ref="vt:clsid"/>
52     </xsd:choice>
53     <xsd:attribute name="fmtid" use="required" type="s:ST_Guid"/>
54     <xsd:attribute name="pid" use="required" type="xsd:int"/>
55     <xsd:attribute name="name" use="optional" type="xsd:string"/>
56     <xsd:attribute name="linkTarget" use="optional" type="xsd:string"/>
57   </xsd:complexType>

```

58 </xsd:schema>

A.7.4 Variant Types

This schema is available in the file shared-documentPropertiesVariantTypes.xsd.

```

1 <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
5   blockDefault="#all" elementFormDefault="qualified">
6   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7     schemaLocation="shared-commonSimpleTypes.xsd"/>
8   <xsd:simpleType name="ST_VectorBaseType">
9     <xsd:restriction base="xsd:string">
10       <xsd:enumeration value="variant"/>
11       <xsd:enumeration value="i1"/>
12       <xsd:enumeration value="i2"/>
13       <xsd:enumeration value="i4"/>
14       <xsd:enumeration value="i8"/>
15       <xsd:enumeration value="ui1"/>
16       <xsd:enumeration value="ui2"/>
17       <xsd:enumeration value="ui4"/>
18       <xsd:enumeration value="ui8"/>
19       <xsd:enumeration value="r4"/>
20       <xsd:enumeration value="r8"/>
21       <xsd:enumeration value="lpstr"/>
22       <xsd:enumeration value="lpwstr"/>
23       <xsd:enumeration value="bstr"/>
24       <xsd:enumeration value="date"/>
25       <xsd:enumeration value="filetime"/>
26       <xsd:enumeration value="bool"/>
27       <xsd:enumeration value="cy"/>
28       <xsd:enumeration value="error"/>
29       <xsd:enumeration value="clsid"/>
30     </xsd:restriction>
31   </xsd:simpleType>
32   <xsd:simpleType name="ST_ArrayBaseType">
33     <xsd:restriction base="xsd:string">
34       <xsd:enumeration value="variant"/>
35       <xsd:enumeration value="i1"/>
36       <xsd:enumeration value="i2"/>
37       <xsd:enumeration value="i4"/>
38       <xsd:enumeration value="int"/>
39       <xsd:enumeration value="ui1"/>
40       <xsd:enumeration value="ui2"/>
41       <xsd:enumeration value="ui4"/>
42       <xsd:enumeration value="uint"/>
43       <xsd:enumeration value="r4"/>
44       <xsd:enumeration value="r8"/>
45       <xsd:enumeration value="decimal"/>
46       <xsd:enumeration value="bstr"/>
47       <xsd:enumeration value="date"/>

```

```

48         <xsd:enumeration value="bool"/>
49         <xsd:enumeration value="cy"/>
50         <xsd:enumeration value="error"/>
51     </xsd:restriction>
52 </xsd:simpleType>
53 <xsd:simpleType name="ST_Cy">
54     <xsd:restriction base="xsd:string">
55         <xsd:pattern value="\s*[0-9]*\.[0-9]{4}\s*" />
56     </xsd:restriction>
57 </xsd:simpleType>
58 <xsd:simpleType name="ST_Error">
59     <xsd:restriction base="xsd:string">
60         <xsd:pattern value="\s*0x[0-9A-Za-z]{8}\s*" />
61     </xsd:restriction>
62 </xsd:simpleType>
63 <xsd:complexType name="CT_Empty"/>
64 <xsd:complexType name="CT_Null"/>
65 <xsd:complexType name="CT_Vector">
66     <xsd:choice minOccurs="1" maxOccurs="unbounded">
67         <xsd:element ref="variant"/>
68         <xsd:element ref="i1"/>
69         <xsd:element ref="i2"/>
70         <xsd:element ref="i4"/>
71         <xsd:element ref="i8"/>
72         <xsd:element ref="ui1"/>
73         <xsd:element ref="ui2"/>
74         <xsd:element ref="ui4"/>
75         <xsd:element ref="ui8"/>
76         <xsd:element ref="r4"/>
77         <xsd:element ref="r8"/>
78         <xsd:element ref="lpstr"/>
79         <xsd:element ref="lpwstr"/>
80         <xsd:element ref="bstr"/>
81         <xsd:element ref="date"/>
82         <xsd:element ref="filetime"/>
83         <xsd:element ref="bool"/>
84         <xsd:element ref="cy"/>
85         <xsd:element ref="error"/>
86         <xsd:element ref="clsid"/>
87     </xsd:choice>
88     <xsd:attribute name="baseType" type="ST_VectorBaseType" use="required"/>
89     <xsd:attribute name="size" type="xsd:unsignedInt" use="required"/>
90 </xsd:complexType>
91 <xsd:complexType name="CT_Array">
92     <xsd:choice minOccurs="1" maxOccurs="unbounded">
93         <xsd:element ref="variant"/>
94         <xsd:element ref="i1"/>
95         <xsd:element ref="i2"/>
96         <xsd:element ref="i4"/>
97         <xsd:element ref="int"/>
98         <xsd:element ref="ui1"/>
99         <xsd:element ref="ui2"/>
100        <xsd:element ref="ui4"/>

```



```

101         <xsd:element ref="uint"/>
102         <xsd:element ref="r4"/>
103         <xsd:element ref="r8"/>
104         <xsd:element ref="decimal"/>
105         <xsd:element ref="bstr"/>
106         <xsd:element ref="date"/>
107         <xsd:element ref="bool"/>
108         <xsd:element ref="error"/>
109         <xsd:element ref="cy"/>
110     </xsd:choice>
111     <xsd:attribute name="lBounds" type="xsd:int" use="required"/>
112     <xsd:attribute name="uBounds" type="xsd:int" use="required"/>
113     <xsd:attribute name="baseType" type="ST_ArrayBaseType" use="required"/>
114 </xsd:complexType>
115 <xsd:complexType name="CT_Variant">
116     <xsd:choice minOccurs="1" maxOccurs="1">
117         <xsd:element ref="variant"/>
118         <xsd:element ref="vector"/>
119         <xsd:element ref="array"/>
120         <xsd:element ref="blob"/>
121         <xsd:element ref="oblob"/>
122         <xsd:element ref="empty"/>
123         <xsd:element ref="null"/>
124         <xsd:element ref="i1"/>
125         <xsd:element ref="i2"/>
126         <xsd:element ref="i4"/>
127         <xsd:element ref="i8"/>
128         <xsd:element ref="int"/>
129         <xsd:element ref="ui1"/>
130         <xsd:element ref="ui2"/>
131         <xsd:element ref="ui4"/>
132         <xsd:element ref="ui8"/>
133         <xsd:element ref="uint"/>
134         <xsd:element ref="r4"/>
135         <xsd:element ref="r8"/>
136         <xsd:element ref="decimal"/>
137         <xsd:element ref="lpstr"/>
138         <xsd:element ref="lpwstr"/>
139         <xsd:element ref="bstr"/>
140         <xsd:element ref="date"/>
141         <xsd:element ref="filetime"/>
142         <xsd:element ref="bool"/>
143         <xsd:element ref="cy"/>
144         <xsd:element ref="error"/>
145         <xsd:element ref="stream"/>
146         <xsd:element ref="ostream"/>
147         <xsd:element ref="storage"/>
148         <xsd:element ref="ostorage"/>
149         <xsd:element ref="vstream"/>
150         <xsd:element ref="clsid"/>
151     </xsd:choice>
152 </xsd:complexType>
153 <xsd:complexType name="CT_Vstream">

```

```

154     <xsd:simpleContent>
155         <xsd:extension base="xsd:base64Binary">
156             <xsd:attribute name="version" type="s:ST_Guid"/>
157         </xsd:extension>
158     </xsd:simpleContent>
159 </xsd:complexType>
160 <xsd:element name="variant" type="CT_Variant"/>
161 <xsd:element name="vector" type="CT_Vector"/>
162 <xsd:element name="array" type="CT_Array"/>
163 <xsd:element name="blob" type="xsd:base64Binary"/>
164 <xsd:element name="oblob" type="xsd:base64Binary"/>
165 <xsd:element name="empty" type="CT_Empty"/>
166 <xsd:element name="null" type="CT_Null"/>
167 <xsd:element name="i1" type="xsd:byte"/>
168 <xsd:element name="i2" type="xsd:short"/>
169 <xsd:element name="i4" type="xsd:int"/>
170 <xsd:element name="i8" type="xsd:long"/>
171 <xsd:element name="int" type="xsd:int"/>
172 <xsd:element name="ui1" type="xsd:unsignedByte"/>
173 <xsd:element name="ui2" type="xsd:unsignedShort"/>
174 <xsd:element name="ui4" type="xsd:unsignedInt"/>
175 <xsd:element name="ui8" type="xsd:unsignedLong"/>
176 <xsd:element name="uint" type="xsd:unsignedInt"/>
177 <xsd:element name="r4" type="xsd:float"/>
178 <xsd:element name="r8" type="xsd:double"/>
179 <xsd:element name="decimal" type="xsd:decimal"/>
180 <xsd:element name="lpstr" type="xsd:string"/>
181 <xsd:element name="lpwstr" type="xsd:string"/>
182 <xsd:element name="bstr" type="xsd:string"/>
183 <xsd:element name="date" type="xsd:dateTime"/>
184 <xsd:element name="filetime" type="xsd:dateTime"/>
185 <xsd:element name="bool" type="xsd:boolean"/>
186 <xsd:element name="cy" type="ST_Cy"/>
187 <xsd:element name="error" type="ST_Error"/>
188 <xsd:element name="stream" type="xsd:base64Binary"/>
189 <xsd:element name="ostream" type="xsd:base64Binary"/>
190 <xsd:element name="storage" type="xsd:base64Binary"/>
191 <xsd:element name="ostorage" type="xsd:base64Binary"/>
192 <xsd:element name="vstream" type="CT_Vstream"/>
193 <xsd:element name="clsid" type="s:ST_Guid"/>
194 </xsd:schema>

```

A.7.5 Custom XML Data Properties

This schema is available in the file shared-customXmlDataProperties.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/officeDocument/2006/customXml"
3   xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/customXml"
5   elementFormDefault="qualified" attributeFormDefault="qualified" blockDefault="#all">
6   <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7     schemaLocation="shared-commonSimpleTypes.xsd"/>

```

```

8      <xsd:complexType name="CT_DatastoreSchemaRef">
9          <xsd:attribute name="uri" type="xsd:string" use="required"/>
10     </xsd:complexType>
11     <xsd:complexType name="CT_DatastoreSchemaRefs">
12         <xsd:sequence>
13             <xsd:element name="schemaRef" type="CT_DatastoreSchemaRef" minOccurs="0"
14                 maxOccurs="unbounded"/>
15         </xsd:sequence>
16     </xsd:complexType>
17     <xsd:complexType name="CT_DatastoreItem">
18         <xsd:sequence>
19             <xsd:element name="schemaRefs" type="CT_DatastoreSchemaRefs" minOccurs="0"/>
20         </xsd:sequence>
21         <xsd:attribute name="itemID" type="s:ST_Guid" use="required"/>
22     </xsd:complexType>
23     <xsd:element name="datastoreItem" type="CT_DatastoreItem"/>
24 </xsd:schema>

```

A.7.6 Bibliography

This schema is available in the file shared-bibliography.xsd.

```

1  <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/bibliography"
2      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3      xmlns:s="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4      targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/bibliography"
5      elementFormDefault="qualified">
6      <xsd:import namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7          schemaLocation="shared-commonSimpleTypes.xsd"/>
8      <xsd:simpleType name="ST_SourceType">
9          <xsd:restriction base="s:ST_String">
10             <xsd:enumeration value="ArticleInAPeriodical"/>
11             <xsd:enumeration value="Book"/>
12             <xsd:enumeration value="BookSection"/>
13             <xsd:enumeration value="JournalArticle"/>
14             <xsd:enumeration value="ConferenceProceedings"/>
15             <xsd:enumeration value="Report"/>
16             <xsd:enumeration value="SoundRecording"/>
17             <xsd:enumeration value="Performance"/>
18             <xsd:enumeration value="Art"/>
19             <xsd:enumeration value="DocumentFromInternetSite"/>
20             <xsd:enumeration value="InternetSite"/>
21             <xsd:enumeration value="Film"/>
22             <xsd:enumeration value="Interview"/>
23             <xsd:enumeration value="Patent"/>
24             <xsd:enumeration value="ElectronicSource"/>
25             <xsd:enumeration value="Case"/>
26             <xsd:enumeration value="Misc"/>
27         </xsd:restriction>
28     </xsd:simpleType>
29     <xsd:complexType name="CT_NameListType">
30         <xsd:sequence>
31             <xsd:element name="Person" type="CT_PersonType" minOccurs="1" maxOccurs="unbounded"/>

```

```

32     </xsd:sequence>
33 </xsd:complexType>
34 <xsd:complexType name="CT_PersonType">
35     <xsd:sequence>
36         <xsd:element name="Last" type="s:ST_String" minOccurs="0" maxOccurs="unbounded"/>
37         <xsd:element name="First" type="s:ST_String" minOccurs="0" maxOccurs="unbounded"/>
38         <xsd:element name="Middle" type="s:ST_String" minOccurs="0" maxOccurs="unbounded"/>
39     </xsd:sequence>
40 </xsd:complexType>
41 <xsd:complexType name="CT_NameType">
42     <xsd:sequence>
43         <xsd:element name="NameList" type="CT_NamelistType" minOccurs="1" maxOccurs="1"/>
44     </xsd:sequence>
45 </xsd:complexType>
46 <xsd:complexType name="CT_NameOrCorporateType">
47     <xsd:sequence>
48         <xsd:choice minOccurs="0" maxOccurs="1">
49             <xsd:element name="NameList" type="CT_NamelistType" minOccurs="1" maxOccurs="1"/>
50             <xsd:element name="Corporate" minOccurs="1" maxOccurs="1" type="s:ST_String"/>
51         </xsd:choice>
52     </xsd:sequence>
53 </xsd:complexType>
54 <xsd:complexType name="CT_AuthorType">
55     <xsd:sequence>
56         <xsd:choice minOccurs="0" maxOccurs="unbounded">
57             <xsd:element name="Artist" type="CT_NameType"/>
58             <xsd:element name="Author" type="CT_NameOrCorporateType"/>
59             <xsd:element name="BookAuthor" type="CT_NameType"/>
60             <xsd:element name="Compiler" type="CT_NameType"/>
61             <xsd:element name="Composer" type="CT_NameType"/>
62             <xsd:element name="Conductor" type="CT_NameType"/>
63             <xsd:element name="Counsel" type="CT_NameType"/>
64             <xsd:element name="Director" type="CT_NameType"/>
65             <xsd:element name="Editor" type="CT_NameType"/>
66             <xsd:element name="Interviewee" type="CT_NameType"/>
67             <xsd:element name="Interviewer" type="CT_NameType"/>
68             <xsd:element name="Inventor" type="CT_NameType"/>
69             <xsd:element name="Performer" type="CT_NameOrCorporateType"/>
70             <xsd:element name="ProducerName" type="CT_NameType"/>
71             <xsd:element name="Translator" type="CT_NameType"/>
72             <xsd:element name="Writer" type="CT_NameType"/>
73         </xsd:choice>
74     </xsd:sequence>
75 </xsd:complexType>
76 <xsd:complexType name="CT_SourceType">
77     <xsd:sequence>
78         <xsd:choice minOccurs="0" maxOccurs="unbounded">
79             <xsd:element name="AbbreviatedCaseNumber" type="s:ST_String"/>
80             <xsd:element name="AlbumTitle" type="s:ST_String"/>
81             <xsd:element name="Author" type="CT_AuthorType"/>
82             <xsd:element name="BookTitle" type="s:ST_String"/>
83             <xsd:element name="Broadcaster" type="s:ST_String"/>
84             <xsd:element name="BroadcastTitle" type="s:ST_String"/>

```

```

85      <xsd:element name="CaseNumber" type="s:ST String"/>
86      <xsd:element name="ChapterNumber" type="s:ST String"/>
87      <xsd:element name="City" type="s:ST String"/>
88      <xsd:element name="Comments" type="s:ST String"/>
89      <xsd:element name="ConferenceName" type="s:ST String"/>
90      <xsd:element name="CountryRegion" type="s:ST String"/>
91      <xsd:element name="Court" type="s:ST String"/>
92      <xsd:element name="Day" type="s:ST String"/>
93      <xsd:element name="DayAccessed" type="s:ST String"/>
94      <xsd:element name="Department" type="s:ST String"/>
95      <xsd:element name="Distributor" type="s:ST String"/>
96      <xsd:element name="Edition" type="s:ST String"/>
97      <xsd:element name="Guid" type="s:ST String"/>
98      <xsd:element name="Institution" type="s:ST String"/>
99      <xsd:element name="InternetSiteTitle" type="s:ST String"/>
100     <xsd:element name="Issue" type="s:ST String"/>
101     <xsd:element name="JournalName" type="s:ST String"/>
102     <xsd:element name="LCID" type="s:ST Lang"/>
103     <xsd:element name="Medium" type="s:ST String"/>
104     <xsd:element name="Month" type="s:ST String"/>
105     <xsd:element name="MonthAccessed" type="s:ST String"/>
106     <xsd:element name="NumberVolumes" type="s:ST String"/>
107     <xsd:element name="Pages" type="s:ST String"/>
108     <xsd:element name="PatentNumber" type="s:ST String"/>
109     <xsd:element name="PeriodicalTitle" type="s:ST String"/>
110     <xsd:element name="ProductionCompany" type="s:ST String"/>
111     <xsd:element name="PublicationTitle" type="s:ST String"/>
112     <xsd:element name="Publisher" type="s:ST String"/>
113     <xsd:element name="RecordingNumber" type="s:ST String"/>
114     <xsd:element name="RefOrder" type="s:ST String"/>
115     <xsd:element name="Reporter" type="s:ST String"/>
116     <xsd:element name="SourceType" type="ST SourceType"/>
117     <xsd:element name="ShortTitle" type="s:ST String"/>
118     <xsd:element name="StandardNumber" type="s:ST String"/>
119     <xsd:element name="StateProvince" type="s:ST String"/>
120     <xsd:element name="Station" type="s:ST String"/>
121     <xsd:element name="Tag" type="s:ST String"/>
122     <xsd:element name="Theater" type="s:ST String"/>
123     <xsd:element name="ThesisType" type="s:ST String"/>
124     <xsd:element name="Title" type="s:ST String"/>
125     <xsd:element name="Type" type="s:ST String"/>
126     <xsd:element name="URL" type="s:ST String"/>
127     <xsd:element name="Version" type="s:ST String"/>
128     <xsd:element name="Volume" type="s:ST String"/>
129     <xsd:element name="Year" type="s:ST String"/>
130     <xsd:element name="YearAccessed" type="s:ST String"/>
131   </xsd:choice>
132 </xsd:sequence>
133 </xsd:complexType>
134 <xsd:element name="Sources" type="CT Sources"/>
135 <xsd:complexType name="CT_Sources">
136   <xsd:sequence>
137     <xsd:element name="Source" type="CT SourceType" minOccurs="0" maxOccurs="unbounded"/>

```

```

138     </xsd:sequence>
139     <xsd:attribute name="SelectedStyle" type="s:ST_String"/>
140     <xsd:attribute name="StyleName" type="s:ST_String"/>
141     <xsd:attribute name="URI" type="s:ST_String"/>
142 </xsd:complexType>
143 </xsd:schema>

```

A.7.7 Additional Characteristics

This schema is available in the file shared-additionalCharacteristics.xsd.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/officeDocument/2006/characteristics"
3   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/characteristics"
4   elementFormDefault="qualified">
5   <xsd:complexType name="CT_AdditionalCharacteristics">
6     <xsd:sequence>
7       <xsd:element name="characteristic" type="CT_Characteristic" minOccurs="0"
8         maxOccurs="unbounded"/>
9     </xsd:sequence>
10  </xsd:complexType>
11  <xsd:complexType name="CT_Characteristic">
12    <xsd:attribute name="name" type="xsd:string" use="required"/>
13    <xsd:attribute name="relation" type="ST_Relation" use="required"/>
14    <xsd:attribute name="val" type="xsd:string" use="required"/>
15    <xsd:attribute name="vocabulary" type="xsd:anyURI" use="optional"/>
16  </xsd:complexType>
17  <xsd:simpleType name="ST_Relation">
18    <xsd:restriction base="xsd:string">
19      <xsd:enumeration value="ge"/>
20      <xsd:enumeration value="le"/>
21      <xsd:enumeration value="gt"/>
22      <xsd:enumeration value="lt"/>
23      <xsd:enumeration value="eq"/>
24    </xsd:restriction>
25  </xsd:simpleType>
26  <xsd:element name="additionalCharacteristics" type="CT_AdditionalCharacteristics"/>
27 </xsd:schema>

```

A.7.8 Office Document Relationships

This schema is available in the file shared-relationshipReference.xsd.

```

1 <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
2   xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
3   xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
4   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/relationships"
5   blockDefault="#all">
6   <xsd:simpleType name="ST_RelationshipId">
7     <xsd:restriction base="xsd:string"/>
8   </xsd:simpleType>
9   <xsd:attribute name="id" type="ST_RelationshipId"/>
10  <xsd:attribute name="embed" type="ST_RelationshipId"/>

```

```

11 <xsd:attribute name="link" type="ST_RelationshipId"/>
12 <xsd:attribute name="dm" type="ST_RelationshipId" default=""/>
13 <xsd:attribute name="lo" type="ST_RelationshipId" default=""/>
14 <xsd:attribute name="qs" type="ST_RelationshipId" default=""/>
15 <xsd:attribute name="cs" type="ST_RelationshipId" default=""/>
16 <xsd:attribute name="blip" type="ST_RelationshipId" default=""/>
17 <xsd:attribute name="pict" type="ST_RelationshipId"/>
18 <xsd:attribute name="href" type="ST_RelationshipId"/>
19 <xsd:attribute name="topLeft" type="ST_RelationshipId"/>
20 <xsd:attribute name="topRight" type="ST_RelationshipId"/>
21 <xsd:attribute name="bottomLeft" type="ST_RelationshipId"/>
22 <xsd:attribute name="bottomRight" type="ST_RelationshipId"/>
23 </xsd:schema>

```

A.7.9 Shared Simple Types

This schema is available in the file shared-commonSimpleTypes.xsd.

```

1 <xsd:schema xmlns="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
2   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4   elementFormDefault="qualified">
5   <xsd:simpleType name="ST_Lang">
6     <xsd:restriction base="xsd:string"/>
7   </xsd:simpleType>
8   <xsd:simpleType name="ST_HexColorRGB">
9     <xsd:restriction base="xsd:hexBinary">
10       <xsd:length value="3" fixed="true"/>
11     </xsd:restriction>
12   </xsd:simpleType>
13   <xsd:simpleType name="ST_Panose">
14     <xsd:restriction base="xsd:hexBinary">
15       <xsd:length value="10"/>
16     </xsd:restriction>
17   </xsd:simpleType>
18   <xsd:simpleType name="ST_CalendarType">
19     <xsd:restriction base="xsd:string">
20       <xsd:enumeration value="gregorian"/>
21       <xsd:enumeration value="gregorianUs"/>
22       <xsd:enumeration value="gregorianMeFrench"/>
23       <xsd:enumeration value="gregorianArabic"/>
24       <xsd:enumeration value="hijri"/>
25       <xsd:enumeration value="hebrew"/>
26       <xsd:enumeration value="taiwan"/>
27       <xsd:enumeration value="japan"/>
28       <xsd:enumeration value="thai"/>
29       <xsd:enumeration value="korea"/>
30       <xsd:enumeration value="saka"/>
31       <xsd:enumeration value="gregorianXlitEnglish"/>
32       <xsd:enumeration value="gregorianXlitFrench"/>
33       <xsd:enumeration value="none"/>
34     </xsd:restriction>
35   </xsd:simpleType>

```

```

36 <xsd:simpleType name="ST_Algorithm">
37   <xsd:restriction base="xsd:string">
38     <xsd:enumeration value="hash"/>
39     <xsd:enumeration value="custom"/>
40   </xsd:restriction>
41 </xsd:simpleType>
42 <xsd:simpleType name="ST_CryptProv">
43   <xsd:restriction base="xsd:string">
44     <xsd:enumeration value="rsaAES"/>
45     <xsd:enumeration value="rsaFull"/>
46     <xsd:enumeration value="custom"/>
47   </xsd:restriction>
48 </xsd:simpleType>
49 <xsd:simpleType name="ST_AlgorithmType">
50   <xsd:restriction base="xsd:string">
51     <xsd:enumeration value="typeAny"/>
52     <xsd:enumeration value="custom"/>
53   </xsd:restriction>
54 </xsd:simpleType>
55 <xsd:simpleType name="ST_ColorType">
56   <xsd:restriction base="xsd:string"/>
57 </xsd:simpleType>
58 <xsd:simpleType name="ST_Guid">
59   <xsd:restriction base="xsd:token">
60     <xsd:pattern value="\{[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}\}"/>
61   </xsd:restriction>
62 </xsd:simpleType>
63 <xsd:simpleType name="ST_OnOff">
64   <xsd:union memberTypes="xsd:boolean ST_OnOff1"/>
65 </xsd:simpleType>
66 <xsd:simpleType name="ST_OnOff1">
67   <xsd:restriction base="xsd:string">
68     <xsd:enumeration value="on"/>
69     <xsd:enumeration value="off"/>
70   </xsd:restriction>
71 </xsd:simpleType>
72 <xsd:simpleType name="ST_String">
73   <xsd:restriction base="xsd:string"/>
74 </xsd:simpleType>
75 <xsd:simpleType name="ST_XmlName">
76   <xsd:restriction base="xsd:NCName">
77     <xsd:minLength value="1"/>
78     <xsd:maxLength value="255"/>
79   </xsd:restriction>
80 </xsd:simpleType>
81 <xsd:simpleType name="ST_TrueFalse">
82   <xsd:restriction base="xsd:string">
83     <xsd:enumeration value="t"/>
84     <xsd:enumeration value="f"/>
85     <xsd:enumeration value="true"/>
86     <xsd:enumeration value="false"/>
87   </xsd:restriction>
88 </xsd:simpleType>

```



```

89 <xsd:simpleType name="ST_TrueFalseBlank">
90   <xsd:restriction base="xsd:string">
91     <xsd:enumeration value="t"/>
92     <xsd:enumeration value="f"/>
93     <xsd:enumeration value="true"/>
94     <xsd:enumeration value="false"/>
95     <xsd:enumeration value=""/>
96     <xsd:enumeration value="True"/>
97     <xsd:enumeration value="False"/>
98   </xsd:restriction>
99 </xsd:simpleType>
100 <xsd:simpleType name="ST_UnsignedDecimalNumber">
101   <xsd:restriction base="xsd:unsignedLong"/>
102 </xsd:simpleType>
103 <xsd:simpleType name="ST_TwipsMeasure">
104   <xsd:union memberTypes="ST_UnsignedDecimalNumber ST_PositiveUniversalMeasure"/>
105 </xsd:simpleType>
106 <xsd:simpleType name="ST_VerticalAlignRun">
107   <xsd:restriction base="xsd:string">
108     <xsd:enumeration value="baseline"/>
109     <xsd:enumeration value="superscript"/>
110     <xsd:enumeration value="subscript"/>
111   </xsd:restriction>
112 </xsd:simpleType>
113 <xsd:simpleType name="ST_Xstring">
114   <xsd:restriction base="xsd:string"/>
115 </xsd:simpleType>
116 <xsd:simpleType name="ST_XAlign">
117   <xsd:restriction base="xsd:string">
118     <xsd:enumeration value="left"/>
119     <xsd:enumeration value="center"/>
120     <xsd:enumeration value="right"/>
121     <xsd:enumeration value="inside"/>
122     <xsd:enumeration value="outside"/>
123   </xsd:restriction>
124 </xsd:simpleType>
125 <xsd:simpleType name="ST_YAlign">
126   <xsd:restriction base="xsd:string">
127     <xsd:enumeration value="inline"/>
128     <xsd:enumeration value="top"/>
129     <xsd:enumeration value="center"/>
130     <xsd:enumeration value="bottom"/>
131     <xsd:enumeration value="inside"/>
132     <xsd:enumeration value="outside"/>
133   </xsd:restriction>
134 </xsd:simpleType>
135 <xsd:simpleType name="ST_ConformanceClass">
136   <xsd:restriction base="xsd:string">
137     <xsd:enumeration value="strict"/>
138     <xsd:enumeration value="transitional"/>
139   </xsd:restriction>
140 </xsd:simpleType>
141 <xsd:simpleType name="ST_UniversalMeasure">

```

```

142     <xsd:restriction base="xsd:string">
143       <xsd:pattern value="-?[0-9]+(\.[0-9]+)?(mm|cm|in|pt|pc|pi)"/>
144     </xsd:restriction>
145   </xsd:simpleType>
146   <xsd:simpleType name="ST_PositiveUniversalMeasure">
147     <xsd:restriction base="ST_UniversalMeasure">
148       <xsd:pattern value="[0-9]+(\.[0-9]+)?(mm|cm|in|pt|pc|pi)"/>
149     </xsd:restriction>
150   </xsd:simpleType>
151   <xsd:simpleType name="ST_Percentage">
152     <xsd:restriction base="xsd:string">
153       <xsd:pattern value="-?[0-9]+(\.[0-9]+)?%/>
154     </xsd:restriction>
155   </xsd:simpleType>
156   <xsd:simpleType name="ST_FixedPercentage">
157     <xsd:restriction base="ST_Percentage">
158       <xsd:pattern value="-?((100)|([0-9][0-9]?))(\.[0-9][0-9]?)?%/>
159     </xsd:restriction>
160   </xsd:simpleType>
161   <xsd:simpleType name="ST_PositivePercentage">
162     <xsd:restriction base="ST_Percentage">
163       <xsd:pattern value="[0-9]+(\.[0-9]+)?%/>
164     </xsd:restriction>
165   </xsd:simpleType>
166   <xsd:simpleType name="ST_PositiveFixedPercentage">
167     <xsd:restriction base="ST_Percentage">
168       <xsd:pattern value="((100)|([0-9][0-9]?))(\.[0-9][0-9]?)?%/>
169     </xsd:restriction>
170   </xsd:simpleType>
171 </xsd:schema>

```

A.8 Custom XML Schema References

This schema is available in the file `shared-customXmlSchemaProperties.xsd`.

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/schemaLibrary/2006/main"
3   targetNamespace="http://schemas.openxmlformats.org/schemaLibrary/2006/main"
4   attributeFormDefault="qualified" elementFormDefault="qualified">
5   <xsd:complexType name="CT_Schema">
6     <xsd:attribute name="uri" type="xsd:string" default=""/>
7     <xsd:attribute name="manifestLocation" type="xsd:string"/>
8     <xsd:attribute name="schemaLocation" type="xsd:string"/>
9     <xsd:attribute name="schemaLanguage" type="xsd:token"/>
10  </xsd:complexType>
11  <xsd:complexType name="CT_SchemaLibrary">
12    <xsd:sequence>
13      <xsd:element name="schema" type="CT_Schema" minOccurs="0" maxOccurs="unbounded"/>
14    </xsd:sequence>
15  </xsd:complexType>
16  <xsd:element name="schemaLibrary" type="CT_SchemaLibrary"/>
17 </xsd:schema>

```

Annex B.

(informative)

Schemas – RELAX NG

This annex is informative.

This Office Open XML specification includes a family of schemas defined using the RELAX NG syntax. The definitions of these schemas follow below, and they also reside in an accompanying file named OfficeOpenXML-RELAXNG-Transitional.zip, which is distributed in electronic form.

As well as the differences between RELAX NG and XML Schemas described in Part 1, §B, “Schemas – RELAX NG”, here are some other differences:

- The RELAX NG schemas represent co-occurrence constraints between elements and attributes. For example, pml.rnc specifies that the pic element and the attribute spid in p_CT_OleObject are mutually exclusive. Meanwhile, pml.xsd simply allows both in CT_OleObject.
- VML drawing parts (§8.1) can be validated against RELAX NG schemas, but cannot be validated against XSD schemas. This is because there are no XSD schemas for the unqualified xml element, which is the root element of VML drawing parts.

B.1 WordprocessingML

This schema is available in the file wml.rnc.

```

1 namespace m =
2   "http://schemas.openxmlformats.org/officeDocument/2006/math"
3 namespace o = "urn:schemas-microsoft-com:office:office"
4 namespace r =
5   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
6 namespace s =
7   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
8 namespace sl =
9   "http://schemas.openxmlformats.org/schemaLibrary/2006/main"
10 namespace v = "urn:schemas-microsoft-com:vml"
11 default namespace w =
12   "http://schemas.openxmlformats.org/wordprocessingml/2006/main"
13 namespace w10 = "urn:schemas-microsoft-com:office:word"
14 namespace wp =
15   "http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
16 namespace x = "urn:schemas-microsoft-com:office:excel"
17
18 w_CT_Empty = empty
19 w_CT_OnOff = attribute w:val { s_ST_OnOff }?
```

```

20 w_ST_LongHexNumber = xsd:hexBinary { length = "4" }
21 w_CT_LongHexNumber = attribute w:val { w_ST_LongHexNumber }
22 w_ST_ShortHexNumber = xsd:hexBinary { length = "2" }
23 w_ST_UcharHexNumber = xsd:hexBinary { length = "1" }
24 w_CT_Charset =
25     attribute w:val { w_ST_UcharHexNumber }?,
26     attribute w:characterSet { s_ST_String }?
27 w_ST_DecimalNumberOrPercent =
28     w_ST_UnqualifiedPercentage | s_ST_Percentage
29 w_ST_UnqualifiedPercentage = xsd:integer
30 w_ST_DecimalNumber = xsd:integer
31 w_CT_DecimalNumber = attribute w:val { w_ST_DecimalNumber }
32 w_CT_UnsignedDecimalNumber =
33     attribute w:val { s_ST_UnsignedDecimalNumber }
34 w_CT_DecimalNumberOrPrecent =
35     attribute w:val { w_ST_DecimalNumberOrPercent }
36 w_CT_TwipsMeasure = attribute w:val { s_ST_TwipsMeasure }
37 w_ST_SignedTwipsMeasure = xsd:integer | s_ST_UniversalMeasure
38 w_CT_SignedTwipsMeasure = attribute w:val { w_ST_SignedTwipsMeasure }
39 w_ST_PixelsMeasure = s_ST_UnsignedDecimalNumber
40 w_CT_PixelsMeasure = attribute w:val { w_ST_PixelsMeasure }
41 w_ST_HpsMeasure =
42     s_ST_UnsignedDecimalNumber | s_ST_PositiveUniversalMeasure
43 w_CT_HpsMeasure = attribute w:val { w_ST_HpsMeasure }
44 w_ST_SignedHpsMeasure = xsd:integer | s_ST_UniversalMeasure
45 w_CT_SignedHpsMeasure = attribute w:val { w_ST_SignedHpsMeasure }
46 w_ST_DateTime = xsd:dateTime
47 w_ST_MacroName = xsd:string { maxLength = "33" }
48 w_CT_MacroName = attribute w:val { w_ST_MacroName }
49 w_ST_EighthPointMeasure = s_ST_UnsignedDecimalNumber
50 w_ST_PointMeasure = s_ST_UnsignedDecimalNumber
51 w_CT_String = attribute w:val { s_ST_String }
52 w_ST_TextScale = w_ST_TextScalePercent | w_ST_TextScaleDecimal
53 w_ST_TextScalePercent = xsd:string { pattern = "0*(600|([0-5]?[0-9]?[0-9]))%" }
54 w_ST_TextScaleDecimal = xsd:integer { minInclusive = "0" maxInclusive = "600" }
55 w_CT_TextScale = attribute w:val { w_ST_TextScale }?
56 w_ST_HighlightColor =
57     string "black"
58     | string "blue"
59     | string "cyan"
60     | string "green"
61     | string "magenta"
62     | string "red"
63     | string "yellow"
64     | string "white"
65     | string "darkBlue"
66     | string "darkCyan"
67     | string "darkGreen"
68     | string "darkMagenta"
69     | string "darkRed"
70     | string "darkYellow"
71     | string "darkGray"
72     | string "lightGray"

```

```

73 | string "none"
74 w_CT_Highlight = attribute w:val { w_ST_HighlightColor }
75 w_ST_HexColorAuto = string "auto"
76 w_ST_HexColor = w_ST_HexColorAuto | s_ST_HexColorRGB
77 w_CT_Color =
78   attribute w:val { w_ST_HexColor },
79   attribute w:themeColor { w_ST_ThemeColor }?,
80   attribute w:themeTint { w_ST_UcharHexNumber }?,
81   attribute w:themeShade { w_ST_UcharHexNumber }?
82 w_CT_Lang = attribute w:val { s_ST_Lang }
83 w_CT_Guid = attribute w:val { s_ST_Guid }?
84 w_ST_Underline =
85   string "single"
86   | string "words"
87   | string "double"
88   | string "thick"
89   | string "dotted"
90   | string "dottedHeavy"
91   | string "dash"
92   | string "dashedHeavy"
93   | string "dashLong"
94   | string "dashLongHeavy"
95   | string "dotDash"
96   | string "dashDotHeavy"
97   | string "dotDotDash"
98   | string "dashDotDotHeavy"
99   | string "wave"
100  | string "wavyHeavy"
101  | string "wavyDouble"
102  | string "none"
103 w_CT_Underline =
104   attribute w:val { w_ST_Underline }?,
105   attribute w:color { w_ST_HexColor }?,
106   attribute w:themeColor { w_ST_ThemeColor }?,
107   attribute w:themeTint { w_ST_UcharHexNumber }?,
108   attribute w:themeShade { w_ST_UcharHexNumber }?
109 w_ST_TextEffect =
110   string "blinkBackground"
111   | string "lights"
112   | string "antsBlack"
113   | string "antsRed"
114   | string "shimmer"
115   | string "sparkle"
116   | string "none"
117 w_CT_TextEffect = attribute w:val { w_ST_TextEffect }
118 w_ST_Border =
119   string "nil"
120   | string "none"
121   | string "single"
122   | string "thick"
123   | string "double"
124   | string "dotted"
125   | string "dashed"

```

```

126 | string "dotDash"
127 | string "dotDotDash"
128 | string "triple"
129 | string "thinThickSmallGap"
130 | string "thickThinSmallGap"
131 | string "thinThickThinSmallGap"
132 | string "thinThickMediumGap"
133 | string "thickThinMediumGap"
134 | string "thinThickThinMediumGap"
135 | string "thinThickLargeGap"
136 | string "thickThinLargeGap"
137 | string "thinThickThinLargeGap"
138 | string "wave"
139 | string "doubleWave"
140 | string "dashSmallGap"
141 | string "dashDotStroked"
142 | string "threeDEmboss"
143 | string "threeDEngrave"
144 | string "outset"
145 | string "inset"
146 | string "apples"
147 | string "archedScallops"
148 | string "babyPacifier"
149 | string "babyRattle"
150 | string "balloons3Colors"
151 | string "balloonsHotAir"
152 | string "basicBlackDashes"
153 | string "basicBlackDots"
154 | string "basicBlackSquares"
155 | string "basicThinLines"
156 | string "basicWhiteDashes"
157 | string "basicWhiteDots"
158 | string "basicWhiteSquares"
159 | string "basicWideInline"
160 | string "basicWideMidline"
161 | string "basicWideOutline"
162 | string "bats"
163 | string "birds"
164 | string "birdsFlight"
165 | string "cabins"
166 | string "cakeSlice"
167 | string "candyCorn"
168 | string "celticKnotwork"
169 | string "certificateBanner"
170 | string "chainLink"
171 | string "champagneBottle"
172 | string "checkedBarBlack"
173 | string "checkedBarColor"
174 | string "checkered"
175 | string "christmasTree"
176 | string "circlesLines"
177 | string "circlesRectangles"
178 | string "classicalWave"

```

```

179 | string "clocks"
180 | string "compass"
181 | string "confetti"
182 | string "confettiGrays"
183 | string "confettiOutline"
184 | string "confettiStreamers"
185 | string "confettiWhite"
186 | string "cornerTriangles"
187 | string "couponCutoutDashes"
188 | string "couponCutoutDots"
189 | string "crazyMaze"
190 | string "creaturesButterfly"
191 | string "creaturesFish"
192 | string "creaturesInsects"
193 | string "creaturesLadyBug"
194 | string "crossStitch"
195 | string "cup"
196 | string "decoArch"
197 | string "decoArchColor"
198 | string "decoBlocks"
199 | string "diamondsGray"
200 | string "doubleD"
201 | string "doubleDiamonds"
202 | string "earth1"
203 | string "earth2"
204 | string "earth3"
205 | string "eclipsingSquares1"
206 | string "eclipsingSquares2"
207 | string "eggsBlack"
208 | string "fans"
209 | string "film"
210 | string "firecrackers"
211 | string "flowersBlockPrint"
212 | string "flowersDaisies"
213 | string "flowersModern1"
214 | string "flowersModern2"
215 | string "flowersPansy"
216 | string "flowersRedRose"
217 | string "flowersRoses"
218 | string "flowersTeacup"
219 | string "flowersTiny"
220 | string "gems"
221 | string "gingerbreadMan"
222 | string "gradient"
223 | string "handmade1"
224 | string "handmade2"
225 | string "heartBalloon"
226 | string "heartGray"
227 | string "hearts"
228 | string "heebieJeebies"
229 | string "holly"
230 | string "houseFunky"
231 | string "hypnotic"

```

232		string	"iceCreamCones"
233		string	"lightBulb"
234		string	"lightning1"
235		string	"lightning2"
236		string	"mapPins"
237		string	"mapleLeaf"
238		string	"mapleMuffins"
239		string	"marquee"
240		string	"marqueeToothed"
241		string	"moons"
242		string	"mosaic"
243		string	"musicNotes"
244		string	"northwest"
245		string	"ovals"
246		string	"packages"
247		string	"palmsBlack"
248		string	"palmsColor"
249		string	"paperClips"
250		string	"papyrus"
251		string	"partyFavor"
252		string	"partyGlass"
253		string	"pencils"
254		string	"people"
255		string	"peopleWaving"
256		string	"peopleHats"
257		string	"poinsettias"
258		string	"postageStamp"
259		string	"pumpkin1"
260		string	"pushPinNote2"
261		string	"pushPinNote1"
262		string	"pyramids"
263		string	"pyramidsAbove"
264		string	"quadrants"
265		string	"rings"
266		string	"safari"
267		string	"sawtooth"
268		string	"sawtoothGray"
269		string	"scaredCat"
270		string	"seattle"
271		string	"shadowedSquares"
272		string	"sharksTeeth"
273		string	"shorebirdTracks"
274		string	"skyrocket"
275		string	"snowflakeFancy"
276		string	"snowflakes"
277		string	"sombrero"
278		string	"southwest"
279		string	"stars"
280		string	"starsTop"
281		string	"stars3d"
282		string	"starsBlack"
283		string	"starsShadowed"
284		string	"sun"


```

285 | string "swirligig"
286 | string "tornPaper"
287 | string "tornPaperBlack"
288 | string "trees"
289 | string "triangleParty"
290 | string "triangles"
291 | string "triangle1"
292 | string "triangle2"
293 | string "triangleCircle1"
294 | string "triangleCircle2"
295 | string "shapes1"
296 | string "shapes2"
297 | string "twistedLines1"
298 | string "twistedLines2"
299 | string "vine"
300 | string "waveline"
301 | string "weavingAngles"
302 | string "weavingBraid"
303 | string "weavingRibbon"
304 | string "weavingStrips"
305 | string "whiteFlowers"
306 | string "woodwork"
307 | string "xIllusions"
308 | string "zanyTriangles"
309 | string "zigZag"
310 | string "zigZagStitch"
311 | string "custom"
312 w_CT_Border =
313     attribute w:val { w_ST_Border },
314     attribute w:color { w_ST_HexColor }?,
315     attribute w:themeColor { w_ST_ThemeColor }?,
316     attribute w:themeTint { w_ST_UcharHexNumber }?,
317     attribute w:themeShade { w_ST_UcharHexNumber }?,
318     attribute w:sz { w_ST_EighthPointMeasure }?,
319     attribute w:space { w_ST_PointMeasure }?,
320     attribute w:shadow { s_ST_OnOff }?,
321     attribute w:frame { s_ST_OnOff }?
322 w_ST_Shd =
323     string "nil"
324     | string "clear"
325     | string "solid"
326     | string "horzStripe"
327     | string "vertStripe"
328     | string "reverseDiagStripe"
329     | string "diagStripe"
330     | string "horzCross"
331     | string "diagCross"
332     | string "thinHorzStripe"
333     | string "thinVertStripe"
334     | string "thinReverseDiagStripe"
335     | string "thinDiagStripe"
336     | string "thinHorzCross"
337     | string "thinDiagCross"

```

```

338 | string "pct5"
339 | string "pct10"
340 | string "pct12"
341 | string "pct15"
342 | string "pct20"
343 | string "pct25"
344 | string "pct30"
345 | string "pct35"
346 | string "pct37"
347 | string "pct40"
348 | string "pct45"
349 | string "pct50"
350 | string "pct55"
351 | string "pct60"
352 | string "pct62"
353 | string "pct65"
354 | string "pct70"
355 | string "pct75"
356 | string "pct80"
357 | string "pct85"
358 | string "pct87"
359 | string "pct90"
360 | string "pct95"
361 w_CT_Shd =
362   attribute w:val { w_ST_Shd },
363   attribute w:color { w_ST_HexColor }?,
364   attribute w:themeColor { w_ST_ThemeColor }?,
365   attribute w:themeTint { w_ST_UcharHexNumber }?,
366   attribute w:themeShade { w_ST_UcharHexNumber }?,
367   attribute w:fill { w_ST_HexColor }?,
368   attribute w:themeFill { w_ST_ThemeColor }?,
369   attribute w:themeFillTint { w_ST_UcharHexNumber }?,
370   attribute w:themeFillShade { w_ST_UcharHexNumber }?
371 w_CT_VerticalAlignRun = attribute w:val { s_ST_VerticalAlignRun }
372 w_CT_FitText =
373   attribute w:val { s_ST_TwipsMeasure },
374   attribute w:id { w_ST_DecimalNumber }?
375 w_ST_Em =
376   string "none"
377   | string "dot"
378   | string "comma"
379   | string "circle"
380   | string "underDot"
381 w_CT_Em = attribute w:val { w_ST_Em }
382 w_CT_Language =
383   attribute w:val { s_ST_Lang }?,
384   attribute w:eastAsia { s_ST_Lang }?,
385   attribute w:bidirectional { s_ST_Lang }?
386 w_ST_CombineBrackets =
387   string "none"
388   | string "round"
389   | string "square"
390   | string "angle"

```

```

391 | string "curly"
392 w_CT_EastAsianLayout =
393   attribute w:id { w_ST_DecimalNumber }?,
394   attribute w:combine { s_ST_OnOff }?,
395   attribute w:combineBrackets { w_ST_CombineBrackets }?,
396   attribute w:vert { s_ST_OnOff }?,
397   attribute w:vertCompress { s_ST_OnOff }?
398 w_ST_HeightRule = string "auto" | string "exact" | string "atLeast"
399 w_ST_Wrap =
400   string "auto"
401   | string "notBeside"
402   | string "around"
403   | string "tight"
404   | string "through"
405   | string "none"
406 w_ST_VAnchor = string "text" | string "margin" | string "page"
407 w_ST_HAnchor = string "text" | string "margin" | string "page"
408 w_ST_DropCap = string "none" | string "drop" | string "margin"
409 w_CT_FramePr =
410   attribute w:dropCap { w_ST_DropCap }?,
411   attribute w:lines { w_ST_DecimalNumber }?,
412   attribute w:w { s_ST_TwipsMeasure }?,
413   attribute w:h { s_ST_TwipsMeasure }?,
414   attribute w:vSpace { s_ST_TwipsMeasure }?,
415   attribute w:hSpace { s_ST_TwipsMeasure }?,
416   attribute w:wrap { w_ST_Wrap }?,
417   attribute w:hAnchor { w_ST_HAnchor }?,
418   attribute w:vAnchor { w_ST_VAnchor }?,
419   attribute w:x { w_ST_SignedTwipsMeasure }?,
420   attribute w:xAlign { s_ST_XAlign }?,
421   attribute w:y { w_ST_SignedTwipsMeasure }?,
422   attribute w:yAlign { s_ST_YAlign }?,
423   attribute w:hRule { w_ST_HeightRule }?,
424   attribute w:anchorLock { s_ST_OnOff }?
425 w_ST_TabJc =
426   string "clear"
427   | string "start"
428   | string "center"
429   | string "end"
430   | string "decimal"
431   | string "bar"
432   | string "num"
433   | string "left"
434   | string "right"
435 w_ST_TabTlc =
436   string "none"
437   | string "dot"
438   | string "hyphen"
439   | string "underscore"
440   | string "heavy"
441   | string "middleDot"
442 w_CT_TabStop =
443   attribute w:val { w_ST_TabJc },

```

```

444     attribute w:leader { w_ST_TabTlc }?,
445     attribute w:pos { w_ST_SignedTwipsMeasure }
446 w_ST_LineSpacingRule = string "auto" | string "exact" | string "atLeast"
447 w_CT_Spacing =
448     attribute w:before { s_ST_TwipsMeasure }?,
449     attribute w:beforeLines { w_ST_DecimalNumber }?,
450     attribute w:beforeAutospacing { s_ST_OnOff }?,
451     attribute w:after { s_ST_TwipsMeasure }?,
452     attribute w:afterLines { w_ST_DecimalNumber }?,
453     attribute w:afterAutospacing { s_ST_OnOff }?,
454     attribute w:line { w_ST_SignedTwipsMeasure }?,
455     attribute w:lineRule { w_ST_LineSpacingRule }?
456 w_CT_Ind =
457     attribute w:start { w_ST_SignedTwipsMeasure }?,
458     attribute w:startChars { w_ST_DecimalNumber }?,
459     attribute w:end { w_ST_SignedTwipsMeasure }?,
460     attribute w:endChars { w_ST_DecimalNumber }?,
461     attribute w:left { w_ST_SignedTwipsMeasure }?,
462     attribute w:leftChars { w_ST_DecimalNumber }?,
463     attribute w:right { w_ST_SignedTwipsMeasure }?,
464     attribute w:rightChars { w_ST_DecimalNumber }?,
465     attribute w:hanging { s_ST_TwipsMeasure }?,
466     attribute w:hangingChars { w_ST_DecimalNumber }?,
467     attribute w:firstLine { s_ST_TwipsMeasure }?,
468     attribute w:firstLineChars { w_ST_DecimalNumber }?
469 w_ST_Jc =
470     string "start"
471     | string "center"
472     | string "end"
473     | string "both"
474     | string "mediumKashida"
475     | string "distribute"
476     | string "numTab"
477     | string "highKashida"
478     | string "lowKashida"
479     | string "thaiDistribute"
480     | string "left"
481     | string "right"
482 w_ST_JcTable =
483     string "center"
484     | string "end"
485     | string "left"
486     | string "right"
487     | string "start"
488 w_CT_Jc = attribute w:val { w_ST_Jc }
489 w_CT_JcTable = attribute w:val { w_ST_JcTable }
490 w_ST_View =
491     string "none"
492     | string "print"
493     | string "outline"
494     | string "masterPages"
495     | string "normal"
496     | string "web"

```

```

497 w_CT_View = attribute w:val { w_ST_View }
498 w_ST_Zoom =
499     string "none"
500     | string "fullPage"
501     | string "bestFit"
502     | string "textFit"
503 w_CT_Zoom =
504     attribute w:val { w_ST_Zoom }?,
505     attribute w:percent { w_ST_DecimalNumberOrPercent }
506 w_CT_WritingStyle =
507     attribute w:lang { s_ST_Lang },
508     attribute w:vendorID { s_ST_String },
509     attribute w:dllVersion { s_ST_String },
510     attribute w:nlCheck { s_ST_OnOff }?,
511     attribute w:checkStyle { s_ST_OnOff },
512     attribute w:appName { s_ST_String }
513 w_ST_Proof = string "clean" | string "dirty"
514 w_CT_Proof =
515     attribute w:spelling { w_ST_Proof }?,
516     attribute w:grammar { w_ST_Proof }?
517 w_ST_DocType = xsd:string
518 w_CT_DocType = attribute w:val { w_ST_DocType }
519 w_ST_DocProtect =
520     string "none"
521     | string "readOnly"
522     | string "comments"
523     | string "trackedChanges"
524     | string "forms"
525 w_AG_Password =
526     attribute w:algorithmName { s_ST_String }?,
527     attribute w:hashValue { xsd:base64Binary }?,
528     attribute w:saltValue { xsd:base64Binary }?,
529     attribute w:spinCount { w_ST_DecimalNumber }?
530 w_AG_TransitionalPassword =
531     attribute w:cryptProviderType { s_ST_CryptProv }?,
532     attribute w:cryptAlgorithmClass { s_ST_AlgClass }?,
533     attribute w:cryptAlgorithmType { s_ST_AlgType }?,
534     attribute w:cryptAlgorithmSid { w_ST_DecimalNumber }?,
535     attribute w:cryptSpinCount { w_ST_DecimalNumber }?,
536     attribute w:cryptProvider { s_ST_String }?,
537     attribute w:algIdExt { w_ST_LongHexNumber }?,
538     attribute w:algIdExtSource { s_ST_String }?,
539     attribute w:cryptProviderTypeExt { w_ST_LongHexNumber }?,
540     attribute w:cryptProviderTypeExtSource { s_ST_String }?,
541     attribute w:hash { xsd:base64Binary }?,
542     attribute w:salt { xsd:base64Binary }?
543 w_CT_DocProtect =
544     attribute w:edit { w_ST_DocProtect }?,
545     attribute w:formatting { s_ST_OnOff }?,
546     attribute w:enforcement { s_ST_OnOff }?,
547     w_AG_Password,
548     w_AG_TransitionalPassword
549 w_ST_MailMergeDocType =

```

```

550     string "catalog"
551     | string "envelopes"
552     | string "mailingLabels"
553     | string "formLetters"
554     | string "email"
555     | string "fax"
556 w_CT_MailMergeDocType = attribute w:val { w_ST_MailMergeDocType }
557 w_ST_MailMergeDataType = xsd:string
558 w_CT_MailMergeDataType = attribute w:val { w_ST_MailMergeDataType }
559 w_ST_MailMergeDest =
560     string "newDocument"
561     | string "printer"
562     | string "email"
563     | string "fax"
564 w_CT_MailMergeDest = attribute w:val { w_ST_MailMergeDest }
565 w_ST_MailMergeOdsoFMDFieldType = string "null" | string "dbColumn"
566 w_CT_MailMergeOdsoFMDFieldType =
567     attribute w:val { w_ST_MailMergeOdsoFMDFieldType }
568 w_CT_TrackChangesView =
569     attribute w:markup { s_ST_OnOff }?,
570     attribute w:comments { s_ST_OnOff }?,
571     attribute w:insDel { s_ST_OnOff }?,
572     attribute w:formatting { s_ST_OnOff }?,
573     attribute w:inkAnnotations { s_ST_OnOff }?
574 w_CT_Kinsoku =
575     attribute w:lang { s_ST_Lang },
576     attribute w:val { s_ST_String }
577 w_ST_TextDirection =
578     string "tb"
579     | string "rl"
580     | string "lr"
581     | string "tbV"
582     | string "rlV"
583     | string "lrV"
584     | string "btLr"
585     | string "lrTb"
586     | string "lrTbV"
587     | string "tbLrV"
588     | string "tbRl"
589     | string "tbRlV"
590 w_CT_TextDirection = attribute w:val { w_ST_TextDirection }
591 w_ST_TextAlignment =
592     string "top"
593     | string "center"
594     | string "baseline"
595     | string "bottom"
596     | string "auto"
597 w_CT_TextAlignment = attribute w:val { w_ST_TextAlignment }
598 w_ST_DisplacedByCustomXml = string "next" | string "prev"
599 w_ST_AnnotationVMerge = string "cont" | string "rest"
600 w_CT_Markup = attribute w:id { w_ST_DecimalNumber }
601 w_CT_TrackChange =
602     w_CT_Markup,

```

```

603     attribute w:author { s_ST_String },
604     attribute w:date { w_ST_DateTime }?
605 w_CT_CellMergeTrackChange =
606     w_CT_TrackChange,
607     attribute w:vMerge { w_ST_AnnotationVMerge }?,
608     attribute w:vMergeOrig { w_ST_AnnotationVMerge }?
609 w_CT_TrackChangeRange =
610     w_CT_TrackChange,
611     attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
612 w_CT_MarkupRange =
613     w_CT_Markup,
614     attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
615 w_CT_BookmarkRange =
616     w_CT_MarkupRange,
617     attribute w:colFirst { w_ST_DecimalNumber }?,
618     attribute w:colLast { w_ST_DecimalNumber }?
619 w_CT_Bookmark =
620     w_CT_BookmarkRange,
621     attribute w:name { s_ST_String }
622 w_CT_MoveBookmark =
623     w_CT_Bookmark,
624     attribute w:author { s_ST_String },
625     attribute w:date { w_ST_DateTime }
626 w_CT_Comment =
627     w_CT_TrackChange,
628     w_EG_BlockLevelElts*,
629     attribute w:initials { s_ST_String }?
630 w_CT_TrackChangeNumbering =
631     w_CT_TrackChange,
632     attribute w:original { s_ST_String }?
633 w_CT_TblPrExChange =
634     w_CT_TrackChange,
635     element tblPrEx { w_CT_TblPrExBase }
636 w_CT_TcPrChange =
637     w_CT_TrackChange,
638     element tcPr { w_CT_TcPrInner }
639 w_CT_TrPrChange =
640     w_CT_TrackChange,
641     element trPr { w_CT_TrPrBase }
642 w_CT_TblGridChange =
643     w_CT_Markup,
644     element tblGrid { w_CT_TblGridBase }
645 w_CT_TblPrChange =
646     w_CT_TrackChange,
647     element tblPr { w_CT_TblPrBase }
648 w_CT_SectPrChange =
649     w_CT_TrackChange,
650     element sectPr { w_CT_SectPrBase }?
651 w_CT_PPrChange =
652     w_CT_TrackChange,
653     element pPr { w_CT_PPrBase }
654 w_CT_RPrChange =
655     w_CT_TrackChange,

```

```

656     element rPr { w_CT_RPrOriginal }
657 w_CT_ParaRPrChange =
658     w_CT_TrackChange,
659     element rPr { w_CT_ParaRPrOriginal }
660 w_CT_RunTrackChange =
661     w_CT_TrackChange, (w_EG_ContentRunContent | m_EG_OMathMathElements)*
662 w_EG_PContentMath = w_EG_PContentBase* | w_EG_ContentRunContentBase*
663 w_EG_PContentBase =
664     element customXml { w_CT_CustomXmlRun }
665     | element fldSimple { w_CT_SimpleField }*
666     | element hyperlink { w_CT_Hyperlink }
667 w_EG_ContentRunContentBase =
668     element smartTag { w_CT_SmartTagRun }
669     | element sdt { w_CT_SdtRun }
670     | w_EG_RunLevelElts*
671 w_EG_CellMarkupElements =
672     element cellIns { w_CT_TrackChange }?
673     | element cellDel { w_CT_TrackChange }?
674     | element cellMerge { w_CT_CellMergeTrackChange }?
675 w_EG_RangeMarkupElements =
676     element bookmarkStart { w_CT_Bookmark }
677     | element bookmarkEnd { w_CT_MarkupRange }
678     | element moveFromRangeStart { w_CT_MoveBookmark }
679     | element moveFromRangeEnd { w_CT_MarkupRange }
680     | element moveToRangeStart { w_CT_MoveBookmark }
681     | element moveToRangeEnd { w_CT_MarkupRange }
682     | element commentRangeStart { w_CT_MarkupRange }
683     | element commentRangeEnd { w_CT_MarkupRange }
684     | element customXmlInsRangeStart { w_CT_TrackChange }
685     | element customXmlInsRangeEnd { w_CT_Markup }
686     | element customXmlDelRangeStart { w_CT_TrackChange }
687     | element customXmlDelRangeEnd { w_CT_Markup }
688     | element customXmlMoveFromRangeStart { w_CT_TrackChange }
689     | element customXmlMoveFromRangeEnd { w_CT_Markup }
690     | element customXmlMoveToRangeStart { w_CT_TrackChange }
691     | element customXmlMoveToRangeEnd { w_CT_Markup }
692 w_CT_NumPr =
693     element ilvl { w_CT_DecimalNumber }?,
694     element numId { w_CT_DecimalNumber }?,
695     element numberingChange { w_CT_TrackChangeNumbering }?,
696     element ins { w_CT_TrackChange }?
697 w_CT_PBdr =
698     element top { w_CT_Border }?,
699     element left { w_CT_Border }?,
700     element bottom { w_CT_Border }?,
701     element right { w_CT_Border }?,
702     element between { w_CT_Border }?,
703     element bar { w_CT_Border }?
704 w_CT_Tabs = element tab { w_CT_TabStop }+
705 w_ST_TextboxTightWrap =
706     string "none"
707     | string "allLines"
708     | string "firstAndLastLine"

```



```

709 | string "firstLineOnly"
710 | string "lastLineOnly"
711 w_CT_TextboxTightWrap = attribute w:val { w_ST_TextboxTightWrap }
712 w_CT_PPr =
713   w_CT_PPrBase,
714   element rPr { w_CT_ParaRPr }?,
715   element sectPr { w_CT_SectPr }?,
716   element pPrChange { w_CT_PPrChange }?
717 w_CT_PPrBase =
718   element pStyle { w_CT_String }?,
719   element keepNext { w_CT_OnOff }?,
720   element keepLines { w_CT_OnOff }?,
721   element pageBreakBefore { w_CT_OnOff }?,
722   element framePr { w_CT_FramePr }?,
723   element widowControl { w_CT_OnOff }?,
724   element numPr { w_CT_NumPr }?,
725   element suppressLineNumbers { w_CT_OnOff }?,
726   element pBdr { w_CT_PBdr }?,
727   element shd { w_CT_Shds }?,
728   element tabs { w_CT_Tabs }?,
729   element suppressAutoHyphens { w_CT_OnOff }?,
730   element kinsoku { w_CT_OnOff }?,
731   element wordWrap { w_CT_OnOff }?,
732   element overflowPunct { w_CT_OnOff }?,
733   element topLinePunct { w_CT_OnOff }?,
734   element autoSpaceDE { w_CT_OnOff }?,
735   element autoSpaceDN { w_CT_OnOff }?,
736   element bidi { w_CT_OnOff }?,
737   element adjustRightInd { w_CT_OnOff }?,
738   element snapToGrid { w_CT_OnOff }?,
739   element spacing { w_CT_Spacing }?,
740   element ind { w_CT_Ind }?,
741   element contextualSpacing { w_CT_OnOff }?,
742   element mirrorIndents { w_CT_OnOff }?,
743   element suppressOverlap { w_CT_OnOff }?,
744   element jc { w_CT_Jc }?,
745   element textDirection { w_CT_TextDirection }?,
746   element textAlignment { w_CT_TextAlignment }?,
747   element textboxTightWrap { w_CT_TextboxTightWrap }?,
748   element outlineLvl { w_CT_DecimalNumber }?,
749   element divId { w_CT_DecimalNumber }?,
750   element cnfStyle { w_CT_Cnf }?
751 w_CT_PPrGeneral =
752   w_CT_PPrBase,
753   element pPrChange { w_CT_PPrChange }?
754 w_CT_Control =
755   attribute w:name { s_ST_String }?,
756   attribute w:shapeid { s_ST_String }?,
757   r_id?
758 w_CT_Background =
759   attribute w:color { w_ST_HexColor }?,
760   attribute w:themeColor { w_ST_ThemeColor }?,
761   attribute w:themeTint { w_ST_UcharHexNumber }?,

```

```

762     attribute w:themeShade { w_ST_UcharHexNumber }?,
763     (w_any_vml_vml*, w_any_vml_office*)+,
764     element drawing { w_CT_Drawing }?
765 w_CT_Rel = r_id
766 w_CT_Object =
767     attribute w:dxaOrig { s_ST_TwipsMeasure }?,
768     attribute w:dyaOrig { s_ST_TwipsMeasure }?,
769     (w_any_vml_vml*, w_any_vml_office*)+,
770     element drawing { w_CT_Drawing }?,
771     (element control { w_CT_Control }
772     | element objectLink { w_CT_ObjectLink }
773     | element objectEmbed { w_CT_ObjectEmbed }
774     | element movie { w_CT_Rel })?
775 w_CT_Picture =
776     (w_any_vml_vml*, w_any_vml_office*)+,
777     element movie { w_CT_Rel }?,
778     element control { w_CT_Control }?
779 w_CT_ObjectEmbed =
780     attribute w:drawAspect { w_ST_ObjectDrawAspect }?,
781     r_id,
782     attribute w:progId { s_ST_String }?,
783     attribute w:shapeId { s_ST_String }?,
784     attribute w:fieldCodes { s_ST_String }?
785 w_ST_ObjectDrawAspect = string "content" | string "icon"
786 w_CT_ObjectLink =
787     w_CT_ObjectEmbed,
788     attribute w:updateMode { w_ST_ObjectUpdateMode },
789     attribute w:lockedField { s_ST_OnOff }?
790 w_ST_ObjectUpdateMode = string "always" | string "onCall"
791 w_CT_Drawing = (wp_anchor? | wp_inline?)+
792 w_CT_SimpleField =
793     attribute w:instr { s_ST_String },
794     attribute w:fldLock { s_ST_OnOff }?,
795     attribute w:dirty { s_ST_OnOff }?,
796     element fldData { w_CT_Text }?,
797     w_EG_PContent*
798 w_ST_FldCharType = string "begin" | string "separate" | string "end"
799 w_ST_InfoTextType = string "text" | string "autoText"
800 w_ST_FFHelpTextVal = xsd:string { maxLength = "256" }
801 w_ST_FFStatusTextVal = xsd:string { maxLength = "140" }
802 w_ST_FFName = xsd:string { maxLength = "65" }
803 w_ST_FFTextType =
804     string "regular"
805     | string "number"
806     | string "date"
807     | string "currentTime"
808     | string "currentDate"
809     | string "calculated"
810 w_CT_FFTextType = attribute w:val { w_ST_FFTextType }
811 w_CT_FFName = attribute w:val { w_ST_FFName }?
812 w_CT_FldChar =
813     attribute w:fldCharType { w_ST_FldCharType },
814     attribute w:fldLock { s_ST_OnOff }?,

```

```

815 attribute w:dirty { s_ST_OnOff }?,
816 (element fldData { w_CT_Text }?
817 | element ffData { w_CT_FFData }?
818 | element numberingChange { w_CT_TrackChangeNumbering }?)
819 w_CT_Hyperlink =
820 attribute w:tgtFrame { s_ST_String }?,
821 attribute w:tooltip { s_ST_String }?,
822 attribute w:docLocation { s_ST_String }?,
823 attribute w:history { s_ST_OnOff }?,
824 attribute w:anchor { s_ST_String }?,
825 r_id?,
826 w_EG_PContent*
827 w_CT_FFData =
828 (element name { w_CT_FFName }
829 | element label { w_CT_DecimalNumber }?
830 | element tabIndex { w_CT_UnsignedDecimalNumber }?
831 | element enabled { w_CT_OnOff }
832 | element calcOnExit { w_CT_OnOff }
833 | element entryMacro { w_CT_MacroName }?
834 | element exitMacro { w_CT_MacroName }?
835 | element helpText { w_CT_FFHelpText }?
836 | element statusText { w_CT_FFStatusText }?
837 | (element checkBox { w_CT_FFCheckBox }
838 | element ddList { w_CT_FFDDLList }
839 | element textInput { w_CT_FFTextInput }))) +
840 w_CT_FFHelpText =
841 attribute w:type { w_ST_InfoTextType }?,
842 attribute w:val { w_ST_FFHelpTextVal }?
843 w_CT_FFStatusText =
844 attribute w:type { w_ST_InfoTextType }?,
845 attribute w:val { w_ST_FFStatusTextVal }?
846 w_CT_FFCheckBox =
847 (element size { w_CT_HpsMeasure }
848 | element sizeAuto { w_CT_OnOff }?),
849 element default { w_CT_OnOff }?,
850 element checked { w_CT_OnOff }?
851 w_CT_FFDDLList =
852 element result { w_CT_DecimalNumber }?,
853 element default { w_CT_DecimalNumber }?,
854 element listEntry { w_CT_String }*
855 w_CT_FFTextInput =
856 element type { w_CT_FFTextType }?,
857 element default { w_CT_String }?,
858 element maxLength { w_CT_DecimalNumber }?,
859 element format { w_CT_String }?
860 w_ST_SectionMark =
861 string "nextPage"
862 | string "nextColumn"
863 | string "continuous"
864 | string "evenPage"
865 | string "oddPage"
866 w_CT_SectType = attribute w:val { w_ST_SectionMark }?
867 w_CT_PaperSource =

```

```

868     attribute w:first { w_ST_DecimalNumber }?,
869     attribute w:other { w_ST_DecimalNumber }?
870 w_ST_NumberFormat =
871     string "decimal"
872     | string "upperRoman"
873     | string "lowerRoman"
874     | string "upperLetter"
875     | string "lowerLetter"
876     | string "ordinal"
877     | string "cardinalText"
878     | string "ordinalText"
879     | string "hex"
880     | string "chicago"
881     | string "ideographDigital"
882     | string "japaneseCounting"
883     | string "aiueo"
884     | string "iroha"
885     | string "decimalFullWidth"
886     | string "decimalHalfWidth"
887     | string "japaneseLegal"
888     | string "japaneseDigitalTenThousand"
889     | string "decimalEnclosedCircle"
890     | string "decimalFullWidth2"
891     | string "aiueoFullWidth"
892     | string "irohaFullWidth"
893     | string "decimalZero"
894     | string "bullet"
895     | string "ganada"
896     | string "chosung"
897     | string "decimalEnclosedFullstop"
898     | string "decimalEnclosedParen"
899     | string "decimalEnclosedCircleChinese"
900     | string "ideographEnclosedCircle"
901     | string "ideographTraditional"
902     | string "ideographZodiac"
903     | string "ideographZodiacTraditional"
904     | string "taiwaneseCounting"
905     | string "ideographLegalTraditional"
906     | string "taiwaneseCountingThousand"
907     | string "taiwaneseDigital"
908     | string "chineseCounting"
909     | string "chineseLegalSimplified"
910     | string "chineseCountingThousand"
911     | string "koreanDigital"
912     | string "koreanCounting"
913     | string "koreanLegal"
914     | string "koreanDigital2"
915     | string "vietnameseCounting"
916     | string "russianLower"
917     | string "russianUpper"
918     | string "none"
919     | string "numberInDash"
920     | string "hebrew1"

```

```

921 | string "hebrew2"
922 | string "arabicAlpha"
923 | string "arabicAbjad"
924 | string "hindiVowels"
925 | string "hindiConsonants"
926 | string "hindiNumbers"
927 | string "hindiCounting"
928 | string "thaiLetters"
929 | string "thaiNumbers"
930 | string "thaiCounting"
931 | string "bahtText"
932 | string "dollarText"
933 | string "custom"
934 w_ST_PageOrientation = string "portrait" | string "landscape"
935 w_CT_PageSz =
936     attribute w:w { s_ST_TwipsMeasure }?,
937     attribute w:h { s_ST_TwipsMeasure }?,
938     attribute w:orient { w_ST_PageOrientation }?,
939     attribute w:code { w_ST_DecimalNumber }?
940 w_CT_PageMar =
941     attribute w:top { w_ST_SignedTwipsMeasure },
942     attribute w:right { s_ST_TwipsMeasure },
943     attribute w:bottom { w_ST_SignedTwipsMeasure },
944     attribute w:left { s_ST_TwipsMeasure },
945     attribute w:header { s_ST_TwipsMeasure },
946     attribute w:footer { s_ST_TwipsMeasure },
947     attribute w:gutter { s_ST_TwipsMeasure }
948 w_ST_PageBorderZOrder = string "front" | string "back"
949 w_ST_PageBorderDisplay =
950     string "allPages" | string "firstPage" | string "notFirstPage"
951 w_ST_PageBorderOffset = string "page" | string "text"
952 w_CT_PageBorders =
953     attribute w:zOrder { w_ST_PageBorderZOrder }?,
954     attribute w:display { w_ST_PageBorderDisplay }?,
955     attribute w:offsetFrom { w_ST_PageBorderOffset }?,
956     element top { w_CT_TopPageBorder }?,
957     element left { w_CT_PageBorder }?,
958     element bottom { w_CT_BottomPageBorder }?,
959     element right { w_CT_PageBorder }?
960 w_CT_PageBorder = w_CT_Border, r_id?
961 w_CT_BottomPageBorder = w_CT_PageBorder, r_bottomLeft?, r_bottomRight?
962 w_CT_TopPageBorder = w_CT_PageBorder, r_topLeft?, r_topRight?
963 w_ST_ChapterSep =
964     string "hyphen"
965     | string "period"
966     | string "colon"
967     | string "emDash"
968     | string "enDash"
969 w_ST_LineNumberRestart =
970     string "newPage" | string "newSection" | string "continuous"
971 w_CT_LineNumber =
972     attribute w:countBy { w_ST_DecimalNumber }?,
973     attribute w:start { w_ST_DecimalNumber }?,

```

```

974     attribute w:distance { s_ST_TwipsMeasure }?,
975     attribute w:restart { w_ST_LineNumberRestart }?
976 w_CT_PageNumber =
977     attribute w:fmt { w_ST_NumberFormat }?,
978     attribute w:start { w_ST_DecimalNumber }?,
979     attribute w:chapStyle { w_ST_DecimalNumber }?,
980     attribute w:chapSep { w_ST_ChapterSep }?
981 w_CT_Column =
982     attribute w:w { s_ST_TwipsMeasure }?,
983     attribute w:space { s_ST_TwipsMeasure }?
984 w_CT_Columns =
985     attribute w:equalWidth { s_ST_OnOff }?,
986     attribute w:space { s_ST_TwipsMeasure }?,
987     attribute w:num { w_ST_DecimalNumber }?,
988     attribute w:sep { s_ST_OnOff }?,
989     element col { w_CT_Column }*
990 w_ST_VerticalJc =
991     string "top" | string "center" | string "both" | string "bottom"
992 w_CT_VerticalJc = attribute w:val { w_ST_VerticalJc }
993 w_ST_DocGrid =
994     string "default"
995     | string "lines"
996     | string "linesAndChars"
997     | string "snapToChars"
998 w_CT_DocGrid =
999     attribute w:type { w_ST_DocGrid }?,
1000     attribute w:linePitch { w_ST_DecimalNumber }?,
1001     attribute w:charSpace { w_ST_DecimalNumber }?
1002 w_ST_HdrFtr = string "even" | string "default" | string "first"
1003 w_ST_FtnEdn =
1004     string "normal"
1005     | string "separator"
1006     | string "continuationSeparator"
1007     | string "continuationNotice"
1008 w_CT_HdrFtrRef =
1009     w_CT_Rel,
1010     attribute w:type { w_ST_HdrFtr }
1011 w_EG_HdrFtrReferences =
1012     element headerReference { w_CT_HdrFtrRef }?
1013     | element footerReference { w_CT_HdrFtrRef }?
1014 w_CT_HdrFtr = w_EG_BlockLevelElts+
1015 w_EG_SectPrContents =
1016     element footnotePr { w_CT_FtnProps }?,
1017     element endnotePr { w_CT_EdnProps }?,
1018     element type { w_CT_SectType }?,
1019     element pgSz { w_CT_PageSz }?,
1020     element pgMar { w_CT_PageMar }?,
1021     element paperSrc { w_CT_PaperSource }?,
1022     element pgBorders { w_CT_PageBorders }?,
1023     element lnNumType { w_CT_LineNumber }?,
1024     element pgNumType { w_CT_PageNumber }?,
1025     element cols { w_CT_Columns }?,
1026     element formProt { w_CT_OnOff }?,

```

```

1027     element vAlign { w_CT_VerticalJc }?,
1028     element noEndnote { w_CT_OnOff }?,
1029     element titlePg { w_CT_OnOff }?,
1030     element textDirection { w_CT_TextDirection }?,
1031     element bidi { w_CT_OnOff }?,
1032     element rtlGutter { w_CT_OnOff }?,
1033     element docGrid { w_CT_DocGrid }?,
1034     element printerSettings { w_CT_Rel }?
1035 w_AG_SectPrAttributes =
1036     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1037     attribute w:rsidDel { w_ST_LongHexNumber }?,
1038     attribute w:rsidR { w_ST_LongHexNumber }?,
1039     attribute w:rsidSect { w_ST_LongHexNumber }?
1040 w_CT_SectPrBase = w_AG_SectPrAttributes, w_EG_SectPrContents?
1041 w_CT_SectPr =
1042     w_AG_SectPrAttributes,
1043     w_EG_HdrFtrReferences*,
1044     w_EG_SectPrContents?,
1045     element sectPrChange { w_CT_SectPrChange }?
1046 w_ST_BrType = string "page" | string "column" | string "textWrapping"
1047 w_ST_BrClear =
1048     string "none" | string "left" | string "right" | string "all"
1049 w_CT_Br =
1050     attribute w:type { w_ST_BrType }?,
1051     attribute w:clear { w_ST_BrClear }?
1052 w_ST_PTabAlignment = string "left" | string "center" | string "right"
1053 w_ST_PTabRelativeTo = string "margin" | string "indent"
1054 w_ST_PTabLeader =
1055     string "none"
1056     | string "dot"
1057     | string "hyphen"
1058     | string "underscore"
1059     | string "middleDot"
1060 w_CT_PTab =
1061     attribute w:alignment { w_ST_PTabAlignment },
1062     attribute w:relativeTo { w_ST_PTabRelativeTo },
1063     attribute w:leader { w_ST_PTabLeader }
1064 w_CT_Sym =
1065     attribute w:font { s_ST_String }?,
1066     attribute w:char { w_ST_ShortHexNumber }?
1067 w_ST_ProofErr =
1068     string "spellStart"
1069     | string "spellEnd"
1070     | string "gramStart"
1071     | string "gramEnd"
1072 w_CT_ProofErr = attribute w:type { w_ST_ProofErr }
1073 w_ST_EdGrp =
1074     string "none"
1075     | string "everyone"
1076     | string "administrators"
1077     | string "contributors"
1078     | string "editors"
1079     | string "owners"

```

```

1080 | string "current"
1081 w_CT_Perm =
1082   attribute w:id { s_ST_String },
1083   attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
1084 w_CT_PermStart =
1085   w_CT_Perm,
1086   attribute w:edGrp { w_ST_EdGrp }?,
1087   attribute w:ed { s_ST_String }?,
1088   attribute w:colFirst { w_ST_DecimalNumber }?,
1089   attribute w:colLast { w_ST_DecimalNumber }?
1090 w_CT_Text = s_ST_String, xml_space?
1091 w_EG_RunInnerContent =
1092   element br { w_CT_Br }
1093   | element t { w_CT_Text }
1094   | element contentPart { w_CT_Rel }
1095   | element delText { w_CT_Text }
1096   | element instrText { w_CT_Text }
1097   | element delInstrText { w_CT_Text }
1098   | element noBreakHyphen { w_CT_Empty }
1099   | element softHyphen { w_CT_Empty }?
1100   | element dayShort { w_CT_Empty }?
1101   | element monthShort { w_CT_Empty }?
1102   | element yearShort { w_CT_Empty }?
1103   | element dayLong { w_CT_Empty }?
1104   | element monthLong { w_CT_Empty }?
1105   | element yearLong { w_CT_Empty }?
1106   | element annotationRef { w_CT_Empty }?
1107   | element footnoteRef { w_CT_Empty }?
1108   | element endnoteRef { w_CT_Empty }?
1109   | element separator { w_CT_Empty }?
1110   | element continuationSeparator { w_CT_Empty }?
1111   | element sym { w_CT_Sym }?
1112   | element pgNum { w_CT_Empty }?
1113   | element cr { w_CT_Empty }?
1114   | element tab { w_CT_Empty }?
1115   | element object { w_CT_Object }
1116   | element pict { w_CT_Picture }
1117   | element fldChar { w_CT_FldChar }
1118   | element ruby { w_CT_Ruby }
1119   | element footnoteReference { w_CT_FtnEdnRef }
1120   | element endnoteReference { w_CT_FtnEdnRef }
1121   | element commentReference { w_CT_Markup }
1122   | element drawing { w_CT_Drawing }
1123   | element ptab { w_CT_PTab }?
1124   | element lastRenderedPageBreak { w_CT_Empty }?
1125 w_CT_R =
1126   attribute w:rsidRPr { w_ST_LongHexNumber }?,
1127   attribute w:rsidDel { w_ST_LongHexNumber }?,
1128   attribute w:rsidR { w_ST_LongHexNumber }?,
1129   w_EG_RPr?,
1130   w_EG_RunInnerContent*
1131 w_ST_Hint = string "default" | string "eastAsia" | string "cs"
1132 w_ST_Theme =

```



```

1133 string "majorEastAsia"
1134 | string "majorBidi"
1135 | string "majorAscii"
1136 | string "majorHAnsi"
1137 | string "minorEastAsia"
1138 | string "minorBidi"
1139 | string "minorAscii"
1140 | string "minorHAnsi"
1141 w_CT_Fonts =
1142   attribute w:hint { w_ST_Hint }?,
1143   attribute w:ascii { s_ST_String }?,
1144   attribute w:hAnsi { s_ST_String }?,
1145   attribute w:eastAsia { s_ST_String }?,
1146   attribute w:cs { s_ST_String }?,
1147   attribute w:asciiTheme { w_ST_Theme }?,
1148   attribute w:hAnsiTheme { w_ST_Theme }?,
1149   attribute w:eastAsiaTheme { w_ST_Theme }?,
1150   attribute w:cstheme { w_ST_Theme }?
1151 w_EG_RPrBase =
1152   element rStyle { w_CT_String }?&
1153   element rFonts { w_CT_Fonts }?&
1154   element b { w_CT_OnOff }?&
1155   element bCs { w_CT_OnOff }?&
1156   element i { w_CT_OnOff }?&
1157   element iCs { w_CT_OnOff }?&
1158   element caps { w_CT_OnOff }?&
1159   element smallCaps { w_CT_OnOff }?&
1160   element strike { w_CT_OnOff }?&
1161   element dstrike { w_CT_OnOff }?&
1162   element outline { w_CT_OnOff }?&
1163   element shadow { w_CT_OnOff }?&
1164   element emboss { w_CT_OnOff }?&
1165   element imprint { w_CT_OnOff }?&
1166   element noProof { w_CT_OnOff }?&
1167   element snapToGrid { w_CT_OnOff }?&
1168   element vanish { w_CT_OnOff }?&
1169   element webHidden { w_CT_OnOff }?&
1170   element color { w_CT_Color }?&
1171   element spacing { w_CT_SignedTwipsMeasure }?&
1172   element w { w_CT_TextScale }?&
1173   element kern { w_CT_HpsMeasure }?&
1174   element position { w_CT_SignedHpsMeasure }?&
1175   element sz { w_CT_HpsMeasure }?&
1176   element szCs { w_CT_HpsMeasure }?&
1177   element highlight { w_CT_Highlight }?&
1178   element u { w_CT_Underline }?&
1179   element effect { w_CT_TextEffect }?&
1180   element bdr { w_CT_Border }?&
1181   element shd { w_CT_Shadow }?&
1182   element fitText { w_CT_FitText }?&
1183   element vertAlign { w_CT_VerticalAlignRun }?&
1184   element rtl { w_CT_OnOff }?&
1185   element cs { w_CT_OnOff }?&

```

```

1186     element em { w_CT_Em }?&
1187     element lang { w_CT_Language }?&
1188     element eastAsianLayout { w_CT_EastAsianLayout }?&
1189     element specVanish { w_CT_OnOff }?&
1190     element oMath { w_CT_OnOff }?
1191 w_EG_RPrContent =
1192     w_EG_RPrBase?,
1193     element rPrChange { w_CT_RPrChange }?
1194 w_CT_RPr = w_EG_RPrContent?
1195 w_EG_RPr = element rPr { w_CT_RPr }?
1196 w_EG_RPrMath =
1197     w_EG_RPr
1198     | element ins { w_CT_MathCtrlIns }
1199     | element del { w_CT_MathCtrlDel }
1200 w_CT_MathCtrlIns =
1201     w_CT_TrackChange,
1202     (element del { w_CT_RPrChange }
1203     | element rPr { w_CT_RPr })?
1204 w_CT_MathCtrlDel =
1205     w_CT_TrackChange,
1206     (element rPr { w_CT_RPr })?
1207 w_CT_RPrOriginal = w_EG_RPrBase*
1208 w_CT_ParaRPrOriginal = w_EG_ParaRPrTrackChanges?, w_EG_RPrBase*
1209 w_CT_ParaRPr =
1210     w_EG_ParaRPrTrackChanges?,
1211     w_EG_RPrBase?,
1212     element rPrChange { w_CT_ParaRPrChange }?
1213 w_EG_ParaRPrTrackChanges =
1214     element ins { w_CT_TrackChange }?,
1215     element del { w_CT_TrackChange }?,
1216     element moveFrom { w_CT_TrackChange }?,
1217     element moveTo { w_CT_TrackChange }?
1218 w_CT_AltChunk =
1219     r_id?,
1220     element altChunkPr { w_CT_AltChunkPr }?
1221 w_CT_AltChunkPr = element matchSrc { w_CT_OnOff }?
1222 w_ST_RubyAlign =
1223     string "center"
1224     | string "distributeLetter"
1225     | string "distributeSpace"
1226     | string "left"
1227     | string "right"
1228     | string "rightVertical"
1229 w_CT_RubyAlign = attribute w:val { w_ST_RubyAlign }
1230 w_CT_RubyPr =
1231     element rubyAlign { w_CT_RubyAlign },
1232     element hps { w_CT_HpsMeasure },
1233     element hpsRaise { w_CT_HpsMeasure },
1234     element hpsBaseText { w_CT_HpsMeasure },
1235     element lid { w_CT_Lang },
1236     element dirty { w_CT_OnOff }?
1237 w_EG_RubyContent =
1238     element r { w_CT_R }

```

```

1239 | w_EG_RunLevelElts*
1240 w_CT_RubyContent = w_EG_RubyContent*
1241 w_CT_Ruby =
1242     element rubyPr { w_CT_RubyPr },
1243     element rt { w_CT_RubyContent },
1244     element rubyBase { w_CT_RubyContent }
1245 w_ST_Lock =
1246     string "sdtLocked"
1247     | string "contentLocked"
1248     | string "unlocked"
1249     | string "sdtContentLocked"
1250 w_CT_Lock = attribute w:val { w_ST_Lock }?
1251 w_CT_SdtListItem =
1252     attribute w:displayText { s_ST_String }?,
1253     attribute w:value { s_ST_String }?
1254 w_ST_SdtDateMappingType =
1255     string "text" | string "date" | string "dateTime"
1256 w_CT_SdtDateMappingType = attribute w:val { w_ST_SdtDateMappingType }?
1257 w_CT_CalendarType = attribute w:val { s_ST_CalendarType }?
1258 w_CT_SdtDate =
1259     attribute w:fullDate { w_ST_DateTime }?,
1260     element dateFormat { w_CT_String }?,
1261     element lid { w_CT_Lang }?,
1262     element storeMappedDataAs { w_CT_SdtDateMappingType }?,
1263     element calendar { w_CT_CalendarType }?
1264 w_CT_SdtComboBox =
1265     attribute w:lastValue { s_ST_String }?,
1266     element listItem { w_CT_SdtListItem }*
1267 w_CT_SdtDocPart =
1268     element docPartGallery { w_CT_String }?,
1269     element docPartCategory { w_CT_String }?,
1270     element docPartUnique { w_CT_OnOff }?
1271 w_CT_SdtDropDownList =
1272     attribute w:lastValue { s_ST_String }?,
1273     element listItem { w_CT_SdtListItem }*
1274 w_CT_Placeholder = element docPart { w_CT_String }
1275 w_CT_SdtText = attribute w:multiline { s_ST_OnOff }?
1276 w_CT_DataBinding =
1277     attribute w:prefixMappings { s_ST_String }?,
1278     attribute w:xpath { s_ST_String },
1279     attribute w:storeItemID { s_ST_String }
1280 w_CT_SdtPr =
1281     element rPr { w_CT_RPr }?,
1282     element alias { w_CT_String }?,
1283     element tag { w_CT_String }?,
1284     element id { w_CT_DecimalNumber }?,
1285     element lock { w_CT_Lock }?,
1286     element placeholder { w_CT_Placeholder }?,
1287     element temporary { w_CT_OnOff }?,
1288     element showingPlcHdr { w_CT_OnOff }?,
1289     element dataBinding { w_CT_DataBinding }?,
1290     element label { w_CT_DecimalNumber }?,
1291     element tabIndex { w_CT_UnsignedDecimalNumber }?,

```

```

1292 (element equation { w_CT_Empty }
1293   | element comboBox { w_CT_SdtComboBox }
1294   | element date { w_CT_SdtDate }
1295   | element docPartObj { w_CT_SdtDocPart }
1296   | element docPartList { w_CT_SdtDocPart }
1297   | element dropDownList { w_CT_SdtDropDownList }
1298   | element picture { w_CT_Empty }
1299   | element richText { w_CT_Empty }
1300   | element text { w_CT_SdtText }
1301   | element citation { w_CT_Empty }
1302   | element group { w_CT_Empty }
1303   | element bibliography { w_CT_Empty }))?
1304 w_CT_SdtEndPr = (element rPr { w_CT_RPr }?)+
1305 w_EG_ContentRunContent =
1306   element customXml { w_CT_CustomXmlRun }
1307   | element smartTag { w_CT_SmartTagRun }
1308   | element sdt { w_CT_SdtRun }
1309   | element dir { w_CT_DirContentRun }
1310   | element bdo { w_CT_BdoContentRun }
1311   | element r { w_CT_R }
1312   | w_EG_RunLevelElts*
1313 w_CT_DirContentRun =
1314   attribute w:val { w_ST_Direction }?,
1315   w_EG_PContent*
1316 w_CT_BdoContentRun =
1317   attribute w:val { w_ST_Direction }?,
1318   w_EG_PContent*
1319 w_ST_Direction = string "ltr" | string "rtl"
1320 w_CT_SdtContentRun = w_EG_PContent*
1321 w_EG_ContentBlockContent =
1322   element customXml { w_CT_CustomXmlBlock }
1323   | element sdt { w_CT_SdtBlock }
1324   | element p { w_CT_P }*
1325   | element tbl { w_CT_Tbl }*
1326   | w_EG_RunLevelElts*
1327 w_CT_SdtContentBlock = w_EG_ContentBlockContent*
1328 w_EG_ContentRowContent =
1329   element tr { w_CT_Row }*
1330   | element customXml { w_CT_CustomXmlRow }
1331   | element sdt { w_CT_SdtRow }
1332   | w_EG_RunLevelElts*
1333 w_CT_SdtContentRow = w_EG_ContentRowContent*
1334 w_EG_ContentCellContent =
1335   element tc { w_CT_Tc }*
1336   | element customXml { w_CT_CustomXmlCell }
1337   | element sdt { w_CT_SdtCell }
1338   | w_EG_RunLevelElts*
1339 w_CT_SdtContentCell = w_EG_ContentCellContent*
1340 w_CT_SdtBlock =
1341   element sdtPr { w_CT_SdtPr }?,
1342   element sdtEndPr { w_CT_SdtEndPr }?,
1343   element sdtContent { w_CT_SdtContentBlock }?
1344 w_CT_SdtRun =

```

```

1345     element sdtPr { w_CT_SdtPr }?,
1346     element sdtEndPr { w_CT_SdtEndPr }?,
1347     element sdtContent { w_CT_SdtContentRun }?
1348 w_CT_SdtCell =
1349     element sdtPr { w_CT_SdtPr }?,
1350     element sdtEndPr { w_CT_SdtEndPr }?,
1351     element sdtContent { w_CT_SdtContentCell }?
1352 w_CT_SdtRow =
1353     element sdtPr { w_CT_SdtPr }?,
1354     element sdtEndPr { w_CT_SdtEndPr }?,
1355     element sdtContent { w_CT_SdtContentRow }?
1356 w_CT_Attr =
1357     attribute w:uri { s_ST_String }?,
1358     attribute w:name { s_ST_String },
1359     attribute w:val { s_ST_String }
1360 w_CT_CustomXmlRun =
1361     attribute w:uri { s_ST_String }?,
1362     attribute w:element { s_ST_XmlName },
1363     element customXmlPr { w_CT_CustomXmlPr }?,
1364     w_EG_PContent*
1365 w_CT_SmartTagRun =
1366     attribute w:uri { s_ST_String }?,
1367     attribute w:element { s_ST_XmlName },
1368     element smartTagPr { w_CT_SmartTagPr }?,
1369     w_EG_PContent*
1370 w_CT_CustomXmlBlock =
1371     attribute w:uri { s_ST_String }?,
1372     attribute w:element { s_ST_XmlName },
1373     element customXmlPr { w_CT_CustomXmlPr }?,
1374     w_EG_ContentBlockContent*
1375 w_CT_CustomXmlPr =
1376     element placeholder { w_CT_String }?,
1377     element attr { w_CT_Attr }*
1378 w_CT_CustomXmlRow =
1379     attribute w:uri { s_ST_String }?,
1380     attribute w:element { s_ST_XmlName },
1381     element customXmlPr { w_CT_CustomXmlPr }?,
1382     w_EG_ContentRowContent*
1383 w_CT_CustomXmlCell =
1384     attribute w:uri { s_ST_String }?,
1385     attribute w:element { s_ST_XmlName },
1386     element customXmlPr { w_CT_CustomXmlPr }?,
1387     w_EG_ContentCellContent*
1388 w_CT_SmartTagPr = element attr { w_CT_Attr }*
1389 w_EG_PContent =
1390     w_EG_ContentRunContent*
1391     | element fldSimple { w_CT_SimpleField }*
1392     | element hyperlink { w_CT_Hyperlink }
1393     | element subDoc { w_CT_Rel }
1394 w_CT_P =
1395     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1396     attribute w:rsidR { w_ST_LongHexNumber }?,
1397     attribute w:rsidDel { w_ST_LongHexNumber }?,

```

```

1398     attribute w:rsidP { w_ST_LongHexNumber }?,
1399     attribute w:rsidRDefault { w_ST_LongHexNumber }?,
1400     element pPr { w_CT_PPr }?,
1401     w_EG_PContent*
1402 w_ST_TblWidth =
1403     string "nil" | string "pct" | string "dxa" | string "auto"
1404 w_CT_Height =
1405     attribute w:val { s_ST_TwipsMeasure }?,
1406     attribute w:hRule { w_ST_HeightRule }?
1407 w_ST_MeasurementOrPercent = w_ST_DecimalNumberOrPercent | s_ST_UniversalMeasure
1408 w_CT_TblWidth =
1409     attribute w:w { w_ST_MeasurementOrPercent}?,
1410     attribute w:type { w_ST_TblWidth }?
1411 w_CT_TblGridCol = attribute w:w { s_ST_TwipsMeasure }?
1412 w_CT_TblGridBase = element gridCol { w_CT_TblGridCol }*
1413 w_CT_TblGrid =
1414     w_CT_TblGridBase,
1415     element tblGridChange { w_CT_TblGridChange }?
1416 w_CT_TcBorders =
1417     element top { w_CT_Border }?,
1418     element start { w_CT_Border }?,
1419     element left { w_CT_Border }?,
1420     element bottom { w_CT_Border }?,
1421     element end { w_CT_Border }?,
1422     element right { w_CT_Border }?,
1423     element insideH { w_CT_Border }?,
1424     element insideV { w_CT_Border }?,
1425     element tl2br { w_CT_Border }?,
1426     element tr2bl { w_CT_Border }?
1427 w_CT_TcMar =
1428     element top { w_CT_TblWidth }?,
1429     element start { w_CT_TblWidth }?,
1430     element left { w_CT_TblWidth }?,
1431     element bottom { w_CT_TblWidth }?,
1432     element end { w_CT_TblWidth }?,
1433     element right { w_CT_TblWidth }?
1434 w_ST_Merge = string "continue" | string "restart"
1435 w_CT_VMerge = attribute w:val { w_ST_Merge }?
1436 w_CT_HMerge = attribute w:val { w_ST_Merge }?
1437 w_CT_TcPrBase =
1438     element cnfStyle { w_CT_Cnf }?,
1439     element tcW { w_CT_TblWidth }?,
1440     element gridSpan { w_CT_DecimalNumber }?,
1441     element hMerge { w_CT_HMerge }?,
1442     element vMerge { w_CT_VMerge }?,
1443     element tcBorders { w_CT_TcBorders }?,
1444     element shd { w_CT_Shd }?,
1445     element nowrap { w_CT_OnOff }?,
1446     element tcMar { w_CT_TcMar }?,
1447     element textDirection { w_CT_TextDirection }?,
1448     element tcFitText { w_CT_OnOff }?,
1449     element vAlign { w_CT_VerticalJc }?,
1450     element hideMark { w_CT_OnOff }?,

```

```

1451     element headers { w_CT_Headers }?
1452 w_CT_TcPr =
1453     w_CT_TcPrInner,
1454     element tcPrChange { w_CT_TcPrChange }?
1455 w_CT_TcPrInner = w_CT_TcPrBase, w_EG_CellMarkupElements?
1456 w_CT_Tc =
1457     attribute w:id { s_ST_String }?,
1458     element tcPr { w_CT_TcPr }?,
1459     w_EG_BlockLevelElts+
1460 w_ST_Cnf = xsd:string { length = "12" pattern = "[01]*" }
1461 w_CT_Cnf =
1462     attribute w:val { w_ST_Cnf }?,
1463     attribute w:firstRow { s_ST_OnOff }?,
1464     attribute w:lastRow { s_ST_OnOff }?,
1465     attribute w:firstColumn { s_ST_OnOff }?,
1466     attribute w:lastColumn { s_ST_OnOff }?,
1467     attribute w:oddVBand { s_ST_OnOff }?,
1468     attribute w:evenVBand { s_ST_OnOff }?,
1469     attribute w:oddHBand { s_ST_OnOff }?,
1470     attribute w:evenHBand { s_ST_OnOff }?,
1471     attribute w:firstRowFirstColumn { s_ST_OnOff }?,
1472     attribute w:firstRowLastColumn { s_ST_OnOff }?,
1473     attribute w:lastRowFirstColumn { s_ST_OnOff }?,
1474     attribute w:lastRowLastColumn { s_ST_OnOff }?
1475 w_CT_Headers = element header { w_CT_String }*
1476 w_CT_TrPrBase =
1477     (element cnfStyle { w_CT_Cnf }?
1478     | element divId { w_CT_DecimalNumber }?
1479     | element gridBefore { w_CT_DecimalNumber }?
1480     | element gridAfter { w_CT_DecimalNumber }?
1481     | element wBefore { w_CT_TblWidth }?
1482     | element wAfter { w_CT_TblWidth }?
1483     | element cantSplit { w_CT_OnOff }?
1484     | element trHeight { w_CT_Height }?
1485     | element tblHeader { w_CT_OnOff }?
1486     | element tblCellSpacing { w_CT_TblWidth }?
1487     | element jc { w_CT_JcTable }?
1488     | element hidden { w_CT_OnOff }?)+
1489 w_CT_TrPr =
1490     w_CT_TrPrBase,
1491     element ins { w_CT_TrackChange }?,
1492     element del { w_CT_TrackChange }?,
1493     element trPrChange { w_CT_TrPrChange }?
1494 w_CT_Row =
1495     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1496     attribute w:rsidR { w_ST_LongHexNumber }?,
1497     attribute w:rsidDel { w_ST_LongHexNumber }?,
1498     attribute w:rsidTr { w_ST_LongHexNumber }?,
1499     element tblPrEx { w_CT_TblPrEx }?,
1500     element trPr { w_CT_TrPr }?,
1501     w_EG_ContentCellContent*
1502 w_ST_TblLayoutType = string "fixed" | string "autofit"
1503 w_CT_TblLayoutType = attribute w:type { w_ST_TblLayoutType }?

```

```

1504 w_ST_TblOverlap = string "never" | string "overlap"
1505 w_CT_TblOverlap = attribute w:val { w_ST_TblOverlap }
1506 w_CT_TblPPr =
1507     attribute w:leftFromText { s_ST_TwipsMeasure }?,
1508     attribute w:rightFromText { s_ST_TwipsMeasure }?,
1509     attribute w:topFromText { s_ST_TwipsMeasure }?,
1510     attribute w:bottomFromText { s_ST_TwipsMeasure }?,
1511     attribute w:vertAnchor { w_ST_VAnchor }?,
1512     attribute w:horzAnchor { w_ST_HAnchor }?,
1513     attribute w:tblpXSpec { s_ST_XAlign }?,
1514     attribute w:tblpX { w_ST_SignedTwipsMeasure }?,
1515     attribute w:tblpYSpec { s_ST_YAlign }?,
1516     attribute w:tblpY { w_ST_SignedTwipsMeasure }?
1517 w_CT_TblCellMar =
1518     element top { w_CT_TblWidth }?,
1519     element start { w_CT_TblWidth }?,
1520     element left { w_CT_TblWidth }?,
1521     element bottom { w_CT_TblWidth }?,
1522     element end { w_CT_TblWidth }?,
1523     element right { w_CT_TblWidth }?
1524 w_CT_TblBorders =
1525     element top { w_CT_Border }?,
1526     element start { w_CT_Border }?,
1527     element left { w_CT_Border }?,
1528     element bottom { w_CT_Border }?,
1529     element end { w_CT_Border }?,
1530     element right { w_CT_Border }?,
1531     element insideH { w_CT_Border }?,
1532     element insideV { w_CT_Border }?
1533 w_CT_TblPrBase =
1534     element tblStyle { w_CT_String }?,
1535     element tblpPr { w_CT_TblPPr }?,
1536     element tblOverlap { w_CT_TblOverlap }?,
1537     element bidiVisual { w_CT_OnOff }?,
1538     element tblStyleRowBandSize { w_CT_DecimalNumber }?,
1539     element tblStyleColBandSize { w_CT_DecimalNumber }?,
1540     element tblW { w_CT_TblWidth }?,
1541     element jc { w_CT_JcTable }?,
1542     element tblCellSpacing { w_CT_TblWidth }?,
1543     element tblInd { w_CT_TblWidth }?,
1544     element tblBorders { w_CT_TblBorders }?,
1545     element shd { w_CT_Shd }?,
1546     element tblLayout { w_CT_TblLayoutType }?,
1547     element tblCellMar { w_CT_TblCellMar }?,
1548     element tblLook { w_CT_TblLook }?,
1549     element tblCaption { w_CT_String }?,
1550     element tblDescription { w_CT_String }?
1551 w_CT_TblPr =
1552     w_CT_TblPrBase,
1553     element tblPrChange { w_CT_TblPrChange }?
1554 w_CT_TblPrExBase =
1555     element tblW { w_CT_TblWidth }?,
1556     element jc { w_CT_JcTable }?,

```



```

1557     element tblCellSpacing { w_CT_TblWidth }?,
1558     element tblInd { w_CT_TblWidth }?,
1559     element tblBorders { w_CT_TblBorders }?,
1560     element shd { w_CT_Shd }?,
1561     element tblLayout { w_CT_TblLayoutType }?,
1562     element tblCellMar { w_CT_TblCellMar }?,
1563     element tblLook { w_CT_TblLook }?
1564 w_CT_TblPrEx =
1565     w_CT_TblPrExBase,
1566     element tblPrExChange { w_CT_TblPrExChange }?
1567 w_CT_Tbl =
1568     w_EG_RangeMarkupElements*,
1569     element tblPr { w_CT_TblPr },
1570     element tblGrid { w_CT_TblGrid },
1571     w_EG_ContentRowContent*
1572 w_CT_TblLook =
1573     attribute w:firstRow { s_ST_OnOff }?,
1574     attribute w:lastRow { s_ST_OnOff }?,
1575     attribute w:firstColumn { s_ST_OnOff }?,
1576     attribute w:lastColumn { s_ST_OnOff }?,
1577     attribute w:noHBand { s_ST_OnOff }?,
1578     attribute w:noVBand { s_ST_OnOff }?,
1579     attribute w:val { w_ST_ShortHexNumber }?
1580 w_ST_FtnPos =
1581     string "pageBottom"
1582     | string "beneathText"
1583     | string "sectEnd"
1584     | string "docEnd"
1585 w_CT_FtnPos = attribute w:val { w_ST_FtnPos }
1586 w_ST_EdnPos = string "sectEnd" | string "docEnd"
1587 w_CT_EdnPos = attribute w:val { w_ST_EdnPos }
1588 w_CT_NumFmt =
1589     attribute w:val { w_ST_NumberFormat },
1590     attribute w:format { s_ST_String }?
1591 w_ST_RestartNumber =
1592     string "continuous" | string "eachSect" | string "eachPage"
1593 w_CT_NumRestart = attribute w:val { w_ST_RestartNumber }
1594 w_CT_FtnEdnRef =
1595     attribute w:customMarkFollows { s_ST_OnOff }?,
1596     attribute w:id { w_ST_DecimalNumber }
1597 w_CT_FtnEdnSepRef = attribute w:id { w_ST_DecimalNumber }
1598 w_CT_FtnEdn =
1599     attribute w:type { w_ST_FtnEdn }?,
1600     attribute w:id { w_ST_DecimalNumber },
1601     w_EG_BlockLevelElts+
1602 w_EG_FtnEdnNumProps =
1603     element numStart { w_CT_DecimalNumber }?,
1604     element numRestart { w_CT_NumRestart }?
1605 w_CT_FtnProps =
1606     element pos { w_CT_FtnPos }?,
1607     element numFmt { w_CT_NumFmt }?,
1608     w_EG_FtnEdnNumProps?
1609 w_CT_EdnProps =

```

```

1610     element pos { w_CT_EdnPos }?,
1611     element numFmt { w_CT_NumFmt }?,
1612     w_EG_FtnEdnNumProps?
1613 w_CT_FtnDocProps =
1614     w_CT_FtnProps,
1615     element footnote { w_CT_FtnEdnSepRef }*
1616 w_CT_EdnDocProps =
1617     w_CT_EdnProps,
1618     element endnote { w_CT_FtnEdnSepRef }*
1619 w_CT_RecipientData =
1620     element active { w_CT_OnOff }?,
1621     element column { w_CT_DecimalNumber },
1622     element uniqueTag { w_CT_Base64Binary}
1623 w_CT_Base64Binary = attribute w:val { xsd:base64Binary }
1624 w_CT_Recipients = element recipientData { w_CT_RecipientData }+
1625 w_recipients = element recipients { w_CT_Recipients }
1626 w_CT_OdsoFieldMapData =
1627     element type { w_CT_MailMergeOdsoFMDFieldType }?,
1628     element name { w_CT_String }?,
1629     element mappedName { w_CT_String }?,
1630     element column { w_CT_DecimalNumber }?,
1631     element lid { w_CT_Lang }?,
1632     element dynamicAddress { w_CT_OnOff }?
1633 w_ST_MailMergeSourceType =
1634     string "database"
1635     | string "addressBook"
1636     | string "document1"
1637     | string "document2"
1638     | string "text"
1639     | string "email"
1640     | string "native"
1641     | string "legacy"
1642     | string "master"
1643 w_CT_MailMergeSourceType = attribute w:val { w_ST_MailMergeSourceType }
1644 w_CT_Odso =
1645     element udl { w_CT_String }?,
1646     element table { w_CT_String }?,
1647     element src { w_CT_Rel }?,
1648     element colDelim { w_CT_DecimalNumber }?,
1649     element type { w_CT_MailMergeSourceType }?,
1650     element fHdr { w_CT_OnOff }?,
1651     element fieldMapData { w_CT_OdsoFieldMapData }*,
1652     element recipientData { w_CT_Rel }*
1653 w_CT_MailMerge =
1654     element mainDocumentType { w_CT_MailMergeDocType },
1655     element linkToQuery { w_CT_OnOff }?,
1656     element dataType { w_CT_MailMergeDataType },
1657     element connectString { w_CT_String }?,
1658     element query { w_CT_String }?,
1659     element dataSource { w_CT_Rel }?,
1660     element headerSource { w_CT_Rel }?,
1661     element doNotSuppressBlankLines { w_CT_OnOff }?,
1662     element destination { w_CT_MailMergeDest }?,

```

```

1663     element addressFieldName { w_CT_String }?,
1664     element mailSubject { w_CT_String }?,
1665     element mailAsAttachment { w_CT_OnOff }?,
1666     element viewMergedData { w_CT_OnOff }?,
1667     element activeRecord { w_CT_DecimalNumber }?,
1668     element checkErrors { w_CT_DecimalNumber }?,
1669     element odso { w_CT_Odso }?
1670 w_ST_TargetScreenSz =
1671     string "544x376"
1672     | string "640x480"
1673     | string "720x512"
1674     | string "800x600"
1675     | string "1024x768"
1676     | string "1152x882"
1677     | string "1152x900"
1678     | string "1280x1024"
1679     | string "1600x1200"
1680     | string "1800x1440"
1681     | string "1920x1200"
1682 w_CT_TargetScreenSz = attribute w:val { w_ST_TargetScreenSz }
1683 w_CT_Compat =
1684     element useSingleBorderforContiguousCells { w_CT_OnOff }?,
1685     element wpJustification { w_CT_OnOff }?,
1686     element noTabHangInd { w_CT_OnOff }?,
1687     element noLeading { w_CT_OnOff }?,
1688     element spaceForUL { w_CT_OnOff }?,
1689     element noColumnBalance { w_CT_OnOff }?,
1690     element balanceSingleByteDoubleByteWidth { w_CT_OnOff }?,
1691     element noExtraLineSpacing { w_CT_OnOff }?,
1692     element doNotLeaveBackslashAlone { w_CT_OnOff }?,
1693     element ulTrailSpace { w_CT_OnOff }?,
1694     element doNotExpandShiftReturn { w_CT_OnOff }?,
1695     element spacingInWholePoints { w_CT_OnOff }?,
1696     element lineWrapLikeWord6 { w_CT_OnOff }?,
1697     element printBodyTextBeforeHeader { w_CT_OnOff }?,
1698     element printColBlack { w_CT_OnOff }?,
1699     element wpSpaceWidth { w_CT_OnOff }?,
1700     element showBreaksInFrames { w_CT_OnOff }?,
1701     element subFontBySize { w_CT_OnOff }?,
1702     element suppressBottomSpacing { w_CT_OnOff }?,
1703     element suppressTopSpacing { w_CT_OnOff }?,
1704     element suppressSpacingAtTopOfPage { w_CT_OnOff }?,
1705     element suppressTopSpacingWP { w_CT_OnOff }?,
1706     element suppressSpBfAfterPgBrk { w_CT_OnOff }?,
1707     element swapBordersFacingPages { w_CT_OnOff }?,
1708     element convMailMergeEsc { w_CT_OnOff }?,
1709     element truncateFontHeightsLikeWP6 { w_CT_OnOff }?,
1710     element mwSmallCaps { w_CT_OnOff }?,
1711     element usePrinterMetrics { w_CT_OnOff }?,
1712     element doNotSuppressParagraphBorders { w_CT_OnOff }?,
1713     element wrapTrailSpaces { w_CT_OnOff }?,
1714     element footnoteLayoutLikeWW8 { w_CT_OnOff }?,
1715     element shapeLayoutLikeWW8 { w_CT_OnOff }?,

```

```

1716 element alignTablesRowByRow { w_CT_OnOff }?,
1717 element forgetLastTabAlignment { w_CT_OnOff }?,
1718 element adjustLineHeightInTable { w_CT_OnOff }?,
1719 element autoSpaceLikeWord95 { w_CT_OnOff }?,
1720 element noSpaceRaiseLower { w_CT_OnOff }?,
1721 element doNotUseHTMLParagraphAutoSpacing { w_CT_OnOff }?,
1722 element layoutRawTableWidth { w_CT_OnOff }?,
1723 element layoutTableRowsApart { w_CT_OnOff }?,
1724 element useWord97LineBreakRules { w_CT_OnOff }?,
1725 element doNotBreakWrappedTables { w_CT_OnOff }?,
1726 element doNotSnapToGridInCell { w_CT_OnOff }?,
1727 element selectFldWithFirstOrLastChar { w_CT_OnOff }?,
1728 element applyBreakingRules { w_CT_OnOff }?,
1729 element doNotWrapTextWithPunct { w_CT_OnOff }?,
1730 element doNotUseEastAsianBreakRules { w_CT_OnOff }?,
1731 element useWord2002TableStyleRules { w_CT_OnOff }?,
1732 element growAutofit { w_CT_OnOff }?,
1733 element useFELayout { w_CT_OnOff }?,
1734 element useNormalStyleForList { w_CT_OnOff }?,
1735 element doNotUseIndentAsNumberingTabStop { w_CT_OnOff }?,
1736 element useAltKinsokuLineBreakRules { w_CT_OnOff }?,
1737 element allowSpaceOfSameStyleInTable { w_CT_OnOff }?,
1738 element doNotSuppressIndentation { w_CT_OnOff }?,
1739 element doNotAutofitConstrainedTables { w_CT_OnOff }?,
1740 element autofitToFirstFixedWidthCell { w_CT_OnOff }?,
1741 element underlineTabInNumList { w_CT_OnOff }?,
1742 element displayHangulFixedWidth { w_CT_OnOff }?,
1743 element splitPgBreakAndParaMark { w_CT_OnOff }?,
1744 element doNotVertAlignCellWithSp { w_CT_OnOff }?,
1745 element doNotBreakConstrainedForcedTable { w_CT_OnOff }?,
1746 element doNotVertAlignInTxbx { w_CT_OnOff }?,
1747 element useAnsiKerningPairs { w_CT_OnOff }?,
1748 element cachedColBalance { w_CT_OnOff }?,
1749 element compatSetting { w_CT_CompatSetting }*
1750 w_CT_CompatSetting =
1751   attribute w:name { s_ST_String }?,
1752   attribute w:uri { s_ST_String }?,
1753   attribute w:val { s_ST_String }?
1754 w_CT_DocVar =
1755   attribute w:name { s_ST_String },
1756   attribute w:val { s_ST_String }
1757 w_CT_DocVars = element docVar { w_CT_DocVar }*
1758 w_CT_DocRsids =
1759   element rsidRoot { w_CT_LongHexNumber }?,
1760   element rsid { w_CT_LongHexNumber }*
1761 w_ST_CharacterSpacing =
1762   string "doNotCompress"
1763   | string "compressPunctuation"
1764   | string "compressPunctuationAndJapaneseKana"
1765 w_CT_CharacterSpacing = attribute w:val { w_ST_CharacterSpacing }
1766 w_CT_SaveThroughXslt =
1767   r_id?,
1768   attribute w:solutionID { s_ST_String }?

```

```

1769 w_CT_RPrDefault = element rPr { w_CT_RPr }?
1770 w_CT_PPrDefault = element pPr { w_CT_PPrGeneral }?
1771 w_CT_DocDefaults =
1772     element rPrDefault { w_CT_RPrDefault }?,
1773     element pPrDefault { w_CT_PPrDefault }?
1774 w_ST_WmlColorSchemeIndex =
1775     string "dark1"
1776     | string "light1"
1777     | string "dark2"
1778     | string "light2"
1779     | string "accent1"
1780     | string "accent2"
1781     | string "accent3"
1782     | string "accent4"
1783     | string "accent5"
1784     | string "accent6"
1785     | string "hyperlink"
1786     | string "followedHyperlink"
1787 w_CT_ColorSchemeMapping =
1788     attribute w:bg1 { w_ST_WmlColorSchemeIndex }?,
1789     attribute w:t1 { w_ST_WmlColorSchemeIndex }?,
1790     attribute w:bg2 { w_ST_WmlColorSchemeIndex }?,
1791     attribute w:t2 { w_ST_WmlColorSchemeIndex }?,
1792     attribute w:accent1 { w_ST_WmlColorSchemeIndex }?,
1793     attribute w:accent2 { w_ST_WmlColorSchemeIndex }?,
1794     attribute w:accent3 { w_ST_WmlColorSchemeIndex }?,
1795     attribute w:accent4 { w_ST_WmlColorSchemeIndex }?,
1796     attribute w:accent5 { w_ST_WmlColorSchemeIndex }?,
1797     attribute w:accent6 { w_ST_WmlColorSchemeIndex }?,
1798     attribute w:hyperlink { w_ST_WmlColorSchemeIndex }?,
1799     attribute w:followedHyperlink { w_ST_WmlColorSchemeIndex }?
1800 w_CT_ReadingModeInkLockDown =
1801     attribute w:actualPg { s_ST_OnOff },
1802     attribute w:w { w_ST_PixelsMeasure },
1803     attribute w:h { w_ST_PixelsMeasure },
1804     attribute w:fontSz { w_ST_DecimalNumberOrPercent }
1805 w_CT_WriteProtection =
1806     attribute w:recommended { s_ST_OnOff }?,
1807     w_AG_Password,
1808     w_AG_TransitionalPassword
1809 w_CT_Settings =
1810     element writeProtection { w_CT_WriteProtection }?,
1811     element view { w_CT_View }?,
1812     element zoom { w_CT_Zoom }?,
1813     element removePersonalInformation { w_CT_OnOff }?,
1814     element removeDateAndTime { w_CT_OnOff }?,
1815     element doNotDisplayPageBoundaries { w_CT_OnOff }?,
1816     element displayBackgroundShape { w_CT_OnOff }?,
1817     element printPostScriptOverText { w_CT_OnOff }?,
1818     element printFractionalCharacterWidth { w_CT_OnOff }?,
1819     element printFormsData { w_CT_OnOff }?,
1820     element embedTrueTypeFonts { w_CT_OnOff }?,
1821     element embedSystemFonts { w_CT_OnOff }?,

```

```

1822 element saveSubsetFonts { w_CT_OnOff }?,
1823 element saveFormsData { w_CT_OnOff }?,
1824 element mirrorMargins { w_CT_OnOff }?,
1825 element alignBordersAndEdges { w_CT_OnOff }?,
1826 element bordersDoNotSurroundHeader { w_CT_OnOff }?,
1827 element bordersDoNotSurroundFooter { w_CT_OnOff }?,
1828 element gutterAtTop { w_CT_OnOff }?,
1829 element hideSpellingErrors { w_CT_OnOff }?,
1830 element hideGrammaticalErrors { w_CT_OnOff }?,
1831 element activeWritingStyle { w_CT_WritingStyle }*,
1832 element proofState { w_CT_Proof }?,
1833 element formsDesign { w_CT_OnOff }?,
1834 element attachedTemplate { w_CT_Rel }?,
1835 element linkStyles { w_CT_OnOff }?,
1836 element stylePaneFormatFilter { w_CT_StylePaneFilter }?,
1837 element stylePaneSortMethod { w_CT_StyleSort }?,
1838 element documentType { w_CT_DocType }?,
1839 element mailMerge { w_CT_MailMerge }?,
1840 element revisionView { w_CT_TrackChangesView }?,
1841 element trackRevisions { w_CT_OnOff }?,
1842 element doNotTrackMoves { w_CT_OnOff }?,
1843 element doNotTrackFormatting { w_CT_OnOff }?,
1844 element documentProtection { w_CT_DocProtect }?,
1845 element autoFormatOverride { w_CT_OnOff }?,
1846 element styleLockTheme { w_CT_OnOff }?,
1847 element styleLockQFSet { w_CT_OnOff }?,
1848 element defaultTabStop { w_CT_TwipsMeasure }?,
1849 element autoHyphenation { w_CT_OnOff }?,
1850 element consecutiveHyphenLimit { w_CT_DecimalNumber }?,
1851 element hyphenationZone { w_CT_TwipsMeasure }?,
1852 element doNotHyphenateCaps { w_CT_OnOff }?,
1853 element showEnvelope { w_CT_OnOff }?,
1854 element summaryLength { w_CT_DecimalNumberOrPrecent }?,
1855 element clickAndTypeStyle { w_CT_String }?,
1856 element defaultTableStyle { w_CT_String }?,
1857 element evenAndOddHeaders { w_CT_OnOff }?,
1858 element bookFoldRevPrinting { w_CT_OnOff }?,
1859 element bookFoldPrinting { w_CT_OnOff }?,
1860 element bookFoldPrintingSheets { w_CT_DecimalNumber }?,
1861 element drawingGridHorizontalSpacing { w_CT_TwipsMeasure }?,
1862 element drawingGridVerticalSpacing { w_CT_TwipsMeasure }?,
1863 element displayHorizontalDrawingGridEvery { w_CT_DecimalNumber }?,
1864 element displayVerticalDrawingGridEvery { w_CT_DecimalNumber }?,
1865 element doNotUseMarginsForDrawingGridOrigin { w_CT_OnOff }?,
1866 element drawingGridHorizontalOrigin { w_CT_TwipsMeasure }?,
1867 element drawingGridVerticalOrigin { w_CT_TwipsMeasure }?,
1868 element doNotShadeFormData { w_CT_OnOff }?,
1869 element noPunctuationKerning { w_CT_OnOff }?,
1870 element characterSpacingControl { w_CT_CharacterSpacing }?,
1871 element printTwoOnOne { w_CT_OnOff }?,
1872 element strictFirstAndLastChars { w_CT_OnOff }?,
1873 element noLineBreaksAfter { w_CT_Kinsoku }?,
1874 element noLineBreaksBefore { w_CT_Kinsoku }?,

```

```

1875 element savePreviewPicture { w_CT_OnOff }?,
1876 element doNotValidateAgainstSchema { w_CT_OnOff }?,
1877 element saveInvalidXml { w_CT_OnOff }?,
1878 element ignoreMixedContent { w_CT_OnOff }?,
1879 element alwaysShowPlaceholderText { w_CT_OnOff }?,
1880 element doNotDemarcateInvalidXml { w_CT_OnOff }?,
1881 element saveXmlDataOnly { w_CT_OnOff }?,
1882 element useXSLTWhenSaving { w_CT_OnOff }?,
1883 element saveThroughXslt { w_CT_SaveThroughXslt }?,
1884 element showXMLTags { w_CT_OnOff }?,
1885 element alwaysMergeEmptyNamespace { w_CT_OnOff }?,
1886 element updateFields { w_CT_OnOff }?,
1887 element hdrShapeDefaults { w_CT_ShapeDefaults }?,
1888 element footnotePr { w_CT_FtnDocProps }?,
1889 element endnotePr { w_CT_EdnDocProps }?,
1890 element compat { w_CT_Compat }?,
1891 element docVars { w_CT_DocVars }?,
1892 element rsids { w_CT_DocRsids }?,
1893 m_mathPr?,
1894 element attachedSchema { w_CT_String }*,
1895 element themeFontLang { w_CT_Language }?,
1896 element clrSchemeMapping { w_CT_ColorSchemeMapping }?,
1897 element doNotIncludeSubdocsInStats { w_CT_OnOff }?,
1898 element doNotAutoCompressPictures { w_CT_OnOff }?,
1899 element forceUpgrade { w_CT_Empty }?,
1900 element captions { w_CT_Captions }?,
1901 element readModeInkLockDown { w_CT_ReadingModeInkLockDown }?,
1902 element smartTagType { w_CT_SmartTagType }*,
1903 sl_schemaLibrary?,
1904 element shapeDefaults { w_CT_ShapeDefaults }?,
1905 element doNotEmbedSmartTags { w_CT_OnOff }?,
1906 element decimalSymbol { w_CT_String }?,
1907 element listSeparator { w_CT_String }?
1908 w_CT_StyleSort = attribute w:val { w_ST_StyleSort }
1909 w_CT_StylePaneFilter =
1910   attribute w:allStyles { s_ST_OnOff }?,
1911   attribute w:customStyles { s_ST_OnOff }?,
1912   attribute w:latentStyles { s_ST_OnOff }?,
1913   attribute w:stylesInUse { s_ST_OnOff }?,
1914   attribute w:headingStyles { s_ST_OnOff }?,
1915   attribute w:numberingStyles { s_ST_OnOff }?,
1916   attribute w:tableStyles { s_ST_OnOff }?,
1917   attribute w:directFormattingOnRuns { s_ST_OnOff }?,
1918   attribute w:directFormattingOnParagraphs { s_ST_OnOff }?,
1919   attribute w:directFormattingOnNumbering { s_ST_OnOff }?,
1920   attribute w:directFormattingOnTables { s_ST_OnOff }?,
1921   attribute w:clearFormatting { s_ST_OnOff }?,
1922   attribute w:top3HeadingStyles { s_ST_OnOff }?,
1923   attribute w:visibleStyles { s_ST_OnOff }?,
1924   attribute w:alternateStyleNames { s_ST_OnOff }?,
1925   attribute w:val { w_ST_ShortHexNumber }?
1926 w_ST_StyleSort =
1927   string "name"

```

```

1928 | string "priority"
1929 | string "default"
1930 | string "font"
1931 | string "basedOn"
1932 | string "type"
1933 | string "0000"
1934 | string "0001"
1935 | string "0002"
1936 | string "0003"
1937 | string "0004"
1938 | string "0005"
1939 w_CT_WebSettings =
1940   element frameset { w_CT_Frameset }?,
1941   element divs { w_CT_Divs }?,
1942   element encoding { w_CT_String }?,
1943   element optimizeForBrowser { w_CT_OptimizeForBrowser }?,
1944   element relyOnVML { w_CT_OnOff }?,
1945   element allowPNG { w_CT_OnOff }?,
1946   element doNotRelyOnCSS { w_CT_OnOff }?,
1947   element doNotSaveAsSingleFile { w_CT_OnOff }?,
1948   element doNotOrganizeInFolder { w_CT_OnOff }?,
1949   element doNotUseLongFileNames { w_CT_OnOff }?,
1950   element pixelsPerInch { w_CT_DecimalNumber }?,
1951   element targetScreenSz { w_CT_TargetScreenSz }?,
1952   element saveSmartTagsAsXml { w_CT_OnOff }?
1953 w_ST_FrameScrollbar = string "on" | string "off" | string "auto"
1954 w_CT_FrameScrollbar = attribute w:val { w_ST_FrameScrollbar }
1955 w_CT_OptimizeForBrowser =
1956   w_CT_OnOff,
1957   attribute w:target { s_ST_String }?
1958 w_CT_Frame =
1959   element sz { w_CT_String }?,
1960   element name { w_CT_String }?,
1961   element title { w_CT_String }?,
1962   element longDesc { w_CT_Rel }?,
1963   element sourceFileName { w_CT_Rel }?,
1964   element marW { w_CT_PixelsMeasure }?,
1965   element marH { w_CT_PixelsMeasure }?,
1966   element scrollbar { w_CT_FrameScrollbar }?,
1967   element noResizeAllowed { w_CT_OnOff }?,
1968   element linkedToFile { w_CT_OnOff }?
1969 w_ST_FrameLayout = string "rows" | string "cols" | string "none"
1970 w_CT_FrameLayout = attribute w:val { w_ST_FrameLayout }
1971 w_CT_FramesetSplitbar =
1972   element w { w_CT_TwipsMeasure }?,
1973   element color { w_CT_Color }?,
1974   element noBorder { w_CT_OnOff }?,
1975   element flatBorders { w_CT_OnOff }?
1976 w_CT_Frameset =
1977   element sz { w_CT_String }?,
1978   element framesetSplitbar { w_CT_FramesetSplitbar }?,
1979   element frameLayout { w_CT_FrameLayout }?,
1980   element title { w_CT_String }?,

```



```

1981 (element frameset { w_CT_Frameset }*)
1982 | element frame { w_CT_Frame }*)*
1983 w_CT_NumPicBullet =
1984   attribute w:numPicBulletId { w_ST_DecimalNumber },
1985   (element pict { w_CT_Picture }
1986   | element drawing { w_CT_Drawing })
1987 w_ST_LevelSuffix = string "tab" | string "space" | string "nothing"
1988 w_CT_LevelSuffix = attribute w:val { w_ST_LevelSuffix }
1989 w_CT_LevelText =
1990   attribute w:val { s_ST_String }?,
1991   attribute w:null { s_ST_OnOff }?
1992 w_CT_LvlLegacy =
1993   attribute w:legacy { s_ST_OnOff }?,
1994   attribute w:legacySpace { s_ST_TwipsMeasure }?,
1995   attribute w:legacyIndent { w_ST_SignedTwipsMeasure }?
1996 w_CT_Lvl =
1997   attribute w:ilvl { w_ST_DecimalNumber },
1998   attribute w:tplc { w_ST_LongHexNumber }?,
1999   attribute w:tentative { s_ST_OnOff }?,
2000   element start { w_CT_DecimalNumber }?,
2001   element numFmt { w_CT_NumFmt }?,
2002   element lvlRestart { w_CT_DecimalNumber }?,
2003   element pStyle { w_CT_String }?,
2004   element isLgl { w_CT_OnOff }?,
2005   element suff { w_CT_LevelSuffix }?,
2006   element lvlText { w_CT_LevelText }?,
2007   element lvlPicBulletId { w_CT_DecimalNumber }?,
2008   element legacy { w_CT_LvlLegacy }?,
2009   element lvlJc { w_CT_Jc }?,
2010   element pPr { w_CT_PPrGeneral }?,
2011   element rPr { w_CT_RPr }?
2012 w_ST_MultiLevelType =
2013   string "singleLevel" | string "multilevel" | string "hybridMultilevel"
2014 w_CT_MultiLevelType = attribute w:val { w_ST_MultiLevelType }
2015 w_CT_AbstractNum =
2016   attribute w:abstractNumId { w_ST_DecimalNumber },
2017   element nsid { w_CT_LongHexNumber }?,
2018   element multiLevelType { w_CT_MultiLevelType }?,
2019   element tpl { w_CT_LongHexNumber }?,
2020   element name { w_CT_String }?,
2021   element styleLink { w_CT_String }?,
2022   element numStyleLink { w_CT_String }?,
2023   element lvl { w_CT_Lvl }*
2024 w_CT_NumLvl =
2025   attribute w:ilvl { w_ST_DecimalNumber },
2026   element startOverride { w_CT_DecimalNumber }?,
2027   element lvl { w_CT_Lvl }?
2028 w_CT_Num =
2029   attribute w:numId { w_ST_DecimalNumber },
2030   element abstractNumId { w_CT_DecimalNumber },
2031   element lvlOverride { w_CT_NumLvl }*
2032 w_CT_Numbering =
2033   element numPicBullet { w_CT_NumPicBullet }*,

```

```

2034     element abstractNum { w_CT_AbstractNum }*,
2035     element num { w_CT_Num }*,
2036     element numIdMacAtCleanup { w_CT_DecimalNumber }?
2037 w_ST_TblStyleOverrideType =
2038     string "wholeTable"
2039     | string "firstRow"
2040     | string "lastRow"
2041     | string "firstCol"
2042     | string "lastCol"
2043     | string "band1Vert"
2044     | string "band2Vert"
2045     | string "band1Horz"
2046     | string "band2Horz"
2047     | string "neCell"
2048     | string "nwCell"
2049     | string "seCell"
2050     | string "swCell"
2051 w_CT_TblStylePr =
2052     attribute w:type { w_ST_TblStyleOverrideType },
2053     element pPr { w_CT_PPrGeneral }?,
2054     element rPr { w_CT_RPr }?,
2055     element tblPr { w_CT_TblPrBase }?,
2056     element trPr { w_CT_TrPr }?,
2057     element tcPr { w_CT_TcPr }?
2058 w_ST_StyleType =
2059     string "paragraph"
2060     | string "character"
2061     | string "table"
2062     | string "numbering"
2063 w_CT_Style =
2064     attribute w:type { w_ST_StyleType }?,
2065     attribute w:styleId { s_ST_String }?,
2066     attribute w:default { s_ST_OnOff }?,
2067     attribute w:customStyle { s_ST_OnOff }?,
2068     element name { w_CT_String }?,
2069     element aliases { w_CT_String }?,
2070     element basedOn { w_CT_String }?,
2071     element next { w_CT_String }?,
2072     element link { w_CT_String }?,
2073     element autoRedefine { w_CT_OnOff }?,
2074     element hidden { w_CT_OnOff }?,
2075     element uiPriority { w_CT_DecimalNumber }?,
2076     element semiHidden { w_CT_OnOff }?,
2077     element unhideWhenUsed { w_CT_OnOff }?,
2078     element qFormat { w_CT_OnOff }?,
2079     element locked { w_CT_OnOff }?,
2080     element personal { w_CT_OnOff }?,
2081     element personalCompose { w_CT_OnOff }?,
2082     element personalReply { w_CT_OnOff }?,
2083     element rsid { w_CT_LongHexNumber }?,
2084     element pPr { w_CT_PPrGeneral }?,
2085     element rPr { w_CT_RPr }?,
2086     element tblPr { w_CT_TblPrBase }?,

```

```

2087     element trPr { w_CT_TrPr }?,
2088     element tcPr { w_CT_TcPr }?,
2089     element tblStylePr { w_CT_TblStylePr }*
2090 w_CT_LsdException =
2091     attribute w:name { s_ST_String },
2092     attribute w:locked { s_ST_OnOff }?,
2093     attribute w:uiPriority { w_ST_DecimalNumber }?,
2094     attribute w:semiHidden { s_ST_OnOff }?,
2095     attribute w:unhideWhenUsed { s_ST_OnOff }?,
2096     attribute w:qFormat { s_ST_OnOff }?
2097 w_CT_LatentStyles =
2098     attribute w:defLockedState { s_ST_OnOff }?,
2099     attribute w:defUIPriority { w_ST_DecimalNumber }?,
2100     attribute w:defSemiHidden { s_ST_OnOff }?,
2101     attribute w:defUnhideWhenUsed { s_ST_OnOff }?,
2102     attribute w:defQFormat { s_ST_OnOff }?,
2103     attribute w:count { w_ST_DecimalNumber }?,
2104     element lsdException { w_CT_LsdException }*
2105 w_CT_Styles =
2106     element docDefaults { w_CT_DocDefaults }?,
2107     element latentStyles { w_CT_LatentStyles }?,
2108     element style { w_CT_Style }*
2109 w_CT_Panose = attribute w:val { s_ST_Panose }
2110 w_ST_FontFamily =
2111     string "decorative"
2112     | string "modern"
2113     | string "roman"
2114     | string "script"
2115     | string "swiss"
2116     | string "auto"
2117 w_CT_FontFamily = attribute w:val { w_ST_FontFamily }
2118 w_ST_Pitch = string "fixed" | string "variable" | string "default"
2119 w_CT_Pitch = attribute w:val { w_ST_Pitch }
2120 w_CT_FontSig =
2121     attribute w:usb0 { w_ST_LongHexNumber },
2122     attribute w:usb1 { w_ST_LongHexNumber },
2123     attribute w:usb2 { w_ST_LongHexNumber },
2124     attribute w:usb3 { w_ST_LongHexNumber },
2125     attribute w:csb0 { w_ST_LongHexNumber },
2126     attribute w:csb1 { w_ST_LongHexNumber }
2127 w_CT_FontRel =
2128     w_CT_Rel,
2129     attribute w:fontKey { s_ST_Guid }?,
2130     attribute w:subsetting { s_ST_OnOff }?
2131 w_CT_Font =
2132     attribute w:name { s_ST_String },
2133     element altName { w_CT_String }?,
2134     element panose1 { w_CT_Panose }?,
2135     element charset { w_CT_Charset }?,
2136     element family { w_CT_FontFamily }?,
2137     element notTrueType { w_CT_OnOff }?,
2138     element pitch { w_CT_Pitch }?,
2139     element sig { w_CT_FontSig }?,

```

```

2140     element embedRegular { w_CT_FontRel }?,
2141     element embedBold { w_CT_FontRel }?,
2142     element embedItalic { w_CT_FontRel }?,
2143     element embedBoldItalic { w_CT_FontRel }?
2144 w_CT_FontsList = element font { w_CT_Font }*
2145 w_CT_DivBdr =
2146     element top { w_CT_Border }?,
2147     element left { w_CT_Border }?,
2148     element bottom { w_CT_Border }?,
2149     element right { w_CT_Border }?
2150 w_CT_Div =
2151     attribute w:id { w_ST_DecimalNumber },
2152     element blockQuote { w_CT_OnOff }?,
2153     element bodyDiv { w_CT_OnOff }?,
2154     element marLeft { w_CT_SignedTwipsMeasure },
2155     element marRight { w_CT_SignedTwipsMeasure },
2156     element marTop { w_CT_SignedTwipsMeasure },
2157     element marBottom { w_CT_SignedTwipsMeasure },
2158     element divBdr { w_CT_DivBdr }?,
2159     element divsChild { w_CT_Divs }*
2160 w_CT_Divs = element div { w_CT_Div }+
2161 w_CT_TxbxContent = w_EG_BlockLevelElts+
2162 w_txbxContent = element txbxContent { w_CT_TxbxContent }
2163 w_EG_MathContent = m_oMathPara | m_oMath
2164 w_EG_BlockLevelChunkElts = w_EG_ContentBlockContent*
2165 w_EG_BlockLevelElts =
2166     w_EG_BlockLevelChunkElts*
2167     | element altChunk { w_CT_AltChunk }*
2168 w_EG_RunLevelElts =
2169     element proofErr { w_CT_ProofErr }?
2170     | element permStart { w_CT_PermStart }?
2171     | element permEnd { w_CT_Perm }?
2172     | w_EG_RangeMarkupElements*
2173     | element ins { w_CT_RunTrackChange }?
2174     | element del { w_CT_RunTrackChange }?
2175     | element moveFrom { w_CT_RunTrackChange }
2176     | element moveTo { w_CT_RunTrackChange }
2177     | w_EG_MathContent*
2178 w_CT_Body =
2179     w_EG_BlockLevelElts*,
2180     element sectPr { w_CT_SectPr }?
2181 w_CT_ShapeDefaults = (w_any_vml_office*)+
2182 w_CT_Comments = element comment { w_CT_Comment }*
2183 w_comments = element comments { w_CT_Comments }
2184 w_CT_Footnotes = element footnote { w_CT_FtnEdn }*
2185 w_footnotes = element footnotes { w_CT_Footnotes }
2186 w_CT_Endnotes = element endnote { w_CT_FtnEdn }*
2187 w_endnotes = element endnotes { w_CT_Endnotes }
2188 w_hdr = element hdr { w_CT_HdrFtr }
2189 w_ftr = element ftr { w_CT_HdrFtr }
2190 w_CT_SmartTagType =
2191     attribute w:namespaceuri { s_ST_String }?,
2192     attribute w:name { s_ST_String }?,

```

```

2193     attribute w:url { s_ST_String }?
2194 w_ST_ThemeColor =
2195     string "dark1"
2196     | string "light1"
2197     | string "dark2"
2198     | string "light2"
2199     | string "accent1"
2200     | string "accent2"
2201     | string "accent3"
2202     | string "accent4"
2203     | string "accent5"
2204     | string "accent6"
2205     | string "hyperlink"
2206     | string "followedHyperlink"
2207     | string "none"
2208     | string "background1"
2209     | string "text1"
2210     | string "background2"
2211     | string "text2"
2212 w_ST_DocPartBehavior = string "content" | string "p" | string "pg"
2213 w_CT_DocPartBehavior = attribute w:val { w_ST_DocPartBehavior }
2214 w_CT_DocPartBehaviors = element behavior { w_CT_DocPartBehavior }+
2215 w_ST_DocPartType =
2216     string "none"
2217     | string "normal"
2218     | string "autoExp"
2219     | string "toolbar"
2220     | string "speller"
2221     | string "formFld"
2222     | string "bbPlcHdr"
2223 w_CT_DocPartType = attribute w:val { w_ST_DocPartType }
2224 w_CT_DocPartTypes =
2225     attribute w:all { s_ST_OnOff }?,
2226     (element type { w_CT_DocPartType }+)
2227 w_ST_DocPartGallery =
2228     string "placeholder"
2229     | string "any"
2230     | string "default"
2231     | string "docParts"
2232     | string "coverPg"
2233     | string "eq"
2234     | string "ftrs"
2235     | string "hdrs"
2236     | string "pgNum"
2237     | string "tbls"
2238     | string "watermarks"
2239     | string "autoTxt"
2240     | string "txtBox"
2241     | string "pgNumT"
2242     | string "pgNumB"
2243     | string "pgNumMargins"
2244     | string "tblOfContents"
2245     | string "bib"

```

```

2246 | string "custQuickParts"
2247 | string "custCoverPg"
2248 | string "custEq"
2249 | string "custFtrs"
2250 | string "custHdrs"
2251 | string "custPgNum"
2252 | string "custTbls"
2253 | string "custWatermarks"
2254 | string "custAutoTxt"
2255 | string "custTxtBox"
2256 | string "custPgNumT"
2257 | string "custPgNumB"
2258 | string "custPgNumMargins"
2259 | string "custTblOfContents"
2260 | string "custBib"
2261 | string "custom1"
2262 | string "custom2"
2263 | string "custom3"
2264 | string "custom4"
2265 | string "custom5"
2266 w_CT_DocPartGallery = attribute w:val { w_ST_DocPartGallery }
2267 w_CT_DocPartCategory =
2268   element name { w_CT_String },
2269   element gallery { w_CT_DocPartGallery }
2270 w_CT_DocPartName =
2271   attribute w:val { s_ST_String },
2272   attribute w:decorated { s_ST_OnOff }?
2273 w_CT_DocPartPr =
2274   element name { w_CT_DocPartName }&
2275   element style { w_CT_String }?&
2276   element category { w_CT_DocPartCategory }?&
2277   element types { w_CT_DocPartTypes }?&
2278   element behaviors { w_CT_DocPartBehaviors }?&
2279   element description { w_CT_String }?&
2280   element guid { w_CT_Guid }?
2281 w_CT_DocPart =
2282   element docPartPr { w_CT_DocPartPr }?,
2283   element docPartBody { w_CT_Body }?
2284 w_CT_DocParts = element docPart { w_CT_DocPart }+
2285 w_settings = element settings { w_CT_Settings }
2286 w_webSettings = element webSettings { w_CT_WebSettings }
2287 w_fonts = element fonts { w_CT_FontsList }
2288 w_numbering = element numbering { w_CT_Numbering }
2289 w_styles = element styles { w_CT_Styles }
2290 w_ST_CaptionPos =
2291   string "above" | string "below" | string "left" | string "right"
2292 w_CT_Caption =
2293   attribute w:name { s_ST_String },
2294   attribute w:pos { w_ST_CaptionPos }?,
2295   attribute w:chapNum { s_ST_OnOff }?,
2296   attribute w:heading { w_ST_DecimalNumber }?,
2297   attribute w:noLabel { s_ST_OnOff }?,
2298   attribute w:numFmt { w_ST_NumberFormat }?,

```

```

2299     attribute w:sep { w_ST_ChapterSep }?
2300 w_CT_AutoCaption =
2301     attribute w:name { s_ST_String },
2302     attribute w:caption { s_ST_String }
2303 w_CT_AutoCaptions = element autoCaption { w_CT_AutoCaption }+
2304 w_CT_Captions =
2305     element caption { w_CT_Caption }+,
2306     element autoCaptions { w_CT_AutoCaptions }?
2307 w_CT_DocumentBase = element background { w_CT_Background }?
2308 w_CT_Document =
2309     w_CT_DocumentBase,
2310     element body { w_CT_Body }?,
2311     attribute w:conformance { s_ST_ConformanceClass }?
2312 w_CT_GlossaryDocument =
2313     w_CT_DocumentBase,
2314     element docParts { w_CT_DocParts }?
2315 w_document = element document { w_CT_Document }
2316 w_glossaryDocument = element glossaryDocument { w_CT_GlossaryDocument }
2317 w_any_vml_office =
2318     o_shapedefaults
2319     | o_shapelayout
2320     | o_signatureline
2321     | o_ink
2322     | o_diagram
2323     | o_skew
2324     | o_extrusion
2325     | o_callout
2326     | o_lock
2327     | o_OLEObject
2328     | o_complex
2329     | o_left
2330     | o_top
2331     | o_right
2332     | o_bottom
2333     | o_column
2334     | o_clippath
2335     | o_fill
2336 w_any_vml_vml =
2337     v_shape
2338     | v_shapetype
2339     | v_group
2340     | v_background
2341     | v_fill
2342     | v_formulas
2343     | v_handles
2344     | v_imagedata
2345     | v_path
2346     | v_textbox
2347     | v_shadow
2348     | v_stroke
2349     | v_textpath
2350     | v_arc
2351     | v_curve

```

```

2352 | v_image
2353 | v_line
2354 | v_oval
2355 | v_polyline
2356 | v_rect
2357 | v_roundrect

```

B.1.1 Part Schemas

B.1.1.1 Comments Part

This schema is available in the file WordprocessingML_Comments.rnc.

```

1  include "wml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-wordprocessingDrawing.rnc"
4  include "dml-main.rnc"
5  include "dml-diagram.rnc"
6  include "shared-commonSimpleTypes.rnc"
7  include "dml-lockedCanvas.rnc"
8  include "any.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_comments

```

B.1.1.2 Document Settings Part

This schema is available in the file WordprocessingML_Document_Settings.rnc.

```

1  include "wml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-wordprocessingDrawing.rnc"
4  include "dml-main.rnc"
5  include "dml-diagram.rnc"
6  include "shared-commonSimpleTypes.rnc"
7  include "dml-lockedCanvas.rnc"
8  include "any.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"

```



```

17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_settings

```

B.1.1.3 Endnotes Part

This schema is available in the file WordprocessingML_Endnotes.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_endnotes

```

B.1.1.4 Font Table Part

This schema is available in the file WordprocessingML_Font_Table.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_fonts

```

B.1.1.5 Footer Part

This schema is available in the file WordprocessingML_Footer.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_ftr

```

B.1.1.6 Footnotes Part

This schema is available in the file WordprocessingML_Footnotes.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_footnotes

```

B.1.1.7 Glossary Document Part

This schema is available in the file WordprocessingML_Glossary_Document.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_glossaryDocument

```

B.1.1.8 Header Part

This schema is available in the file WordprocessingML_Header.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_hdr

```

B.1.1.9 Mail Merge Recipient Data Part

This schema is available in the file WordprocessingML_Mail_Merge_Recipient_Data.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"

```

```

5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_recipients

```

B.1.1.10 Main Document Part

This schema is available in the file WordprocessingML_Main_Document.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_document

```

B.1.1.11 Numbering Definitions Part

This schema is available in the file WordprocessingML_Numbering_Definitions.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"

```

```

9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_numbering

```

B.1.1.12 Style Definitions Part

This schema is available in the file WordprocessingML_Style_Definitions.rnc.

```

1  include "wml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-wordprocessingDrawing.rnc"
4  include "dml-main.rnc"
5  include "dml-diagram.rnc"
6  include "shared-commonSimpleTypes.rnc"
7  include "dml-lockedCanvas.rnc"
8  include "any.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_styles

```

B.1.1.13 Web Settings Part

This schema is available in the file WordprocessingML_Web_Settings.rnc.

```

1  include "wml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-wordprocessingDrawing.rnc"
4  include "dml-main.rnc"
5  include "dml-diagram.rnc"
6  include "shared-commonSimpleTypes.rnc"
7  include "dml-lockedCanvas.rnc"
8  include "any.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"

```

```

13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = w_webSettings

```

B.2 SpreadsheetML

This schema is available in the file sml.rnc.

```

1 namespace o = "urn:schemas-microsoft-com:office:office"
2 namespace r =
3   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 default namespace sml =
7   "http://schemas.openxmlformats.org/spreadsheetml/2006/main"
8 namespace v = "urn:schemas-microsoft-com:vml"
9 namespace w10 = "urn:schemas-microsoft-com:office:word"
10 namespace x = "urn:schemas-microsoft-com:office:excel"
11 namespace xdr =
12   "http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
13
14 sml_CT_AutoFilter =
15   attribute ref { sml_ST_Ref }?,
16   element filterColumn { sml_CT_FilterColumn }*,
17   element sortState { sml_CT_SortState }?,
18   element extLst { sml_CT_ExtensionList }?
19 sml_CT_FilterColumn =
20   attribute colId { xsd:unsignedInt },
21
22   ## default value: false
23   attribute hiddenButton { xsd:boolean }?,
24
25   ## default value: true
26   attribute showButton { xsd:boolean }?,
27   (element filters { sml_CT_Filters }?
28     | element top10 { sml_CT_Top10 }?
29     | element customFilters { sml_CT_CustomFilters }?
30     | element dynamicFilter { sml_CT_DynamicFilter }?
31     | element colorFilter { sml_CT_ColorFilter }?
32     | element iconFilter { sml_CT_IconFilter }?
33     | element extLst { sml_CT_ExtensionList }?)?
34 sml_CT_Filters =
35
36   ## default value: false
37   attribute blank { xsd:boolean }?,
38
39   ## default value: none
40   attribute calendarType { s_ST_CalendarType }?,

```

```

41     element filter { sml_CT_Filter }*,
42     element dateGroupItem { sml_CT_DateGroupItem }*
43 sml_CT_Filter = attribute val { s_ST_Xstring }?
44 sml_CT_CustomFilters =
45
46     ## default value: false
47     attribute and { xsd:boolean }?,
48     element customFilter { sml_CT_CustomFilter }+
49 sml_CT_CustomFilter =
50
51     ## default value: equal
52     attribute operator { sml_ST_FilterOperator }?,
53     attribute val { s_ST_Xstring }?
54 sml_CT_Top10 =
55
56     ## default value: true
57     attribute top { xsd:boolean }?,
58
59     ## default value: false
60     attribute percent { xsd:boolean }?,
61     attribute val { xsd:double },
62     attribute filterVal { xsd:double }?
63 sml_CT_ColorFilter =
64     attribute dxfId { sml_ST_DxfId }?,
65
66     ## default value: true
67     attribute cellColor { xsd:boolean }?
68 sml_CT_IconFilter =
69     attribute iconSet { sml_ST_IconSetType },
70     attribute iconId { xsd:unsignedInt }?
71 sml_ST_FilterOperator =
72     string "equal"
73     | string "lessThan"
74     | string "lessThanOrEqual"
75     | string "notEqual"
76     | string "greaterThanOrEqual"
77     | string "greaterThan"
78 sml_CT_DynamicFilter =
79     attribute type { sml_ST_DynamicFilterType },
80     attribute val { xsd:double }?,
81     attribute valIso { xsd:dateTime }?,
82     attribute maxVal { xsd:double }?,
83     attribute maxValIso { xsd:dateTime }?
84 sml_ST_DynamicFilterType =
85     string "null"
86     | string "aboveAverage"
87     | string "belowAverage"
88     | string "tomorrow"
89     | string "today"
90     | string "yesterday"
91     | string "nextWeek"
92     | string "thisWeek"
93     | string "lastWeek"

```

```

94 | string "nextMonth"
95 | string "thisMonth"
96 | string "lastMonth"
97 | string "nextQuarter"
98 | string "thisQuarter"
99 | string "lastQuarter"
100 | string "nextYear"
101 | string "thisYear"
102 | string "lastYear"
103 | string "yearToDate"
104 | string "Q1"
105 | string "Q2"
106 | string "Q3"
107 | string "Q4"
108 | string "M1"
109 | string "M2"
110 | string "M3"
111 | string "M4"
112 | string "M5"
113 | string "M6"
114 | string "M7"
115 | string "M8"
116 | string "M9"
117 | string "M10"
118 | string "M11"
119 | string "M12"
120 sml_ST_IconSetType =
121   string "3Arrows"
122   | string "3ArrowsGray"
123   | string "3Flags"
124   | string "3TrafficLights1"
125   | string "3TrafficLights2"
126   | string "3Signs"
127   | string "3Symbols"
128   | string "3Symbols2"
129   | string "4Arrows"
130   | string "4ArrowsGray"
131   | string "4RedToBlack"
132   | string "4Rating"
133   | string "4TrafficLights"
134   | string "5Arrows"
135   | string "5ArrowsGray"
136   | string "5Rating"
137   | string "5Quarters"
138 sml_CT_SortState =
139
140   ## default value: false
141   attribute columnSort { xsd:boolean }?,
142
143   ## default value: false
144   attribute caseSensitive { xsd:boolean }?,
145
146   ## default value: none

```



```

147     attribute sortMethod { sml_ST_SortMethod }?,
148     attribute ref { sml_ST_Ref },
149     element sortCondition { sml_CT_SortCondition }*,
150     element extLst { sml_CT_ExtensionList }?
151 sml_CT_SortCondition =
152
153     ## default value: false
154     attribute descending { xsd:boolean }?,
155
156     ## default value: value
157     attribute sortBy { sml_ST_SortBy }?,
158     attribute ref { sml_ST_Ref },
159     attribute customList { s_ST_Xstring }?,
160     attribute dxId { sml_ST_DxfId }?,
161
162     ## default value: 3Arrows
163     attribute iconSet { sml_ST_IconSetType }?,
164     attribute iconId { xsd:unsignedInt }?
165 sml_ST_SortBy =
166     string "value"
167     | string "cellColor"
168     | string "fontColor"
169     | string "icon"
170 sml_ST_SortMethod = string "stroke" | string "pinYin" | string "none"
171 sml_CT_DateGroupItem =
172     attribute year { xsd:unsignedShort },
173     attribute month { xsd:unsignedShort }?,
174     attribute day { xsd:unsignedShort }?,
175     attribute hour { xsd:unsignedShort }?,
176     attribute minute { xsd:unsignedShort }?,
177     attribute second { xsd:unsignedShort }?,
178     attribute dateTimeGrouping { sml_ST_DateTimeGrouping }
179 sml_ST_DateTimeGrouping =
180     string "year"
181     | string "month"
182     | string "day"
183     | string "hour"
184     | string "minute"
185     | string "second"
186 sml_ST_CellRef = xsd:string
187 sml_ST_Ref = xsd:string
188 sml_ST_RefA = xsd:string
189 sml_ST_Sqref = list { sml_ST_Ref* }
190 sml_ST_Formula = s_ST_Xstring
191 sml_ST_UnsignedIntHex = xsd:hexBinary { length = "4" }
192 sml_ST_UnsignedShortHex = xsd:hexBinary { length = "2" }
193 sml_CT_XStringElement = attribute v { s_ST_Xstring }
194 sml_CT_Extension =
195     attribute uri { xsd:token }?,
196     sml_CT_Extension_any
197 sml_CT_Extension_any =
198     element * - (o:* | v:* | w10:* | x:*) {
199         anyAttribute*,

```

```

200     mixed { anyElement* }
201   }
202   sml_CT_ObjectAnchor =
203
204     ## default value: false
205     attribute moveWithCells { xsd:boolean }?,
206
207     ## default value: false
208     attribute sizeWithCells { xsd:boolean }?,
209     xdr_from,
210     xdr_to
211   sml_EG_ExtensionList = element ext { sml_CT_Extension }*
212   sml_CT_ExtensionList = sml_EG_ExtensionList?
213   sml_calcChain = element calcChain { sml_CT_CalcChain }
214   sml_CT_CalcChain =
215     element c { sml_CT_CalcCell }+,
216     element extLst { sml_CT_ExtensionList }?
217   sml_CT_CalcCell =
218     attribute ( r | ref ) { sml_ST_CellRef },
219
220     ## default value: 0
221     attribute i { xsd:int }?,
222
223     ## default value: false
224     attribute s { xsd:boolean }?,
225
226     ## default value: false
227     attribute l { xsd:boolean }?,
228
229     ## default value: false
230     attribute t { xsd:boolean }?,
231
232     ## default value: false
233     attribute a { xsd:boolean }?
234   sml_comments = element comments { sml_CT_Comments }
235   sml_CT_Comments =
236     element authors { sml_CT_Authors },
237     element commentList { sml_CT_CommentList },
238     element extLst { sml_CT_ExtensionList }?
239   sml_CT_Authors = element author { s_ST_Xstring }*
240   sml_CT_CommentList = element comment { sml_CT_Comment }*
241   sml_CT_Comment =
242     attribute ref { sml_ST_Ref },
243     attribute authorId { xsd:unsignedInt },
244     attribute guid { s_ST_Guid }?,
245     attribute shapeId { xsd:unsignedInt }?,
246     element text { sml_CT_Rst },
247     element commentPr { sml_CT_CommentPr }?
248   sml_CT_CommentPr =
249
250     ## default value: true
251     attribute locked { xsd:boolean }?,
252

```

```

253 ## default value: true
254 attribute defaultSize { xsd:boolean }?,
255
256 ## default value: true
257 attribute print { xsd:boolean }?,
258
259 ## default value: false
260 attribute disabled { xsd:boolean }?,
261
262 ## default value: true
263 attribute autoFill { xsd:boolean }?,
264
265 ## default value: true
266 attribute autoLine { xsd:boolean }?,
267 attribute altText { s_ST_Xstring }?,
268
269 ## default value: left
270 attribute textHAlign { sml_ST_TextHAlign }?,
271
272 ## default value: top
273 attribute textVAlign { sml_ST_TextVAlign }?,
274
275 ## default value: true
276 attribute lockText { xsd:boolean }?,
277
278 ## default value: false
279 attribute justLastX { xsd:boolean }?,
280
281 ## default value: false
282 attribute autoScale { xsd:boolean }?,
283 element anchor { sml_CT_ObjectAnchor }
284 sml_ST_TextHAlign =
285     string "left"
286     | string "center"
287     | string "right"
288     | string "justify"
289     | string "distributed"
290 sml_ST_TextVAlign =
291     string "top"
292     | string "center"
293     | string "bottom"
294     | string "justify"
295     | string "distributed"
296 sml_MapInfo = element MapInfo { sml_CT_MapInfo }
297 sml_CT_MapInfo =
298     attribute SelectionNamespaces { xsd:string },
299     element Schema { sml_CT_Schema }+,
300     element Map { sml_CT_Map }+
301 sml_CT_Schema =
302     mixed {
303         attribute ID { xsd:string },
304         attribute SchemaRef { xsd:string }?,
305         attribute Namespace { xsd:string }?,

```

```

306     attribute SchemaLanguage { xsd:token }?,
307     sml_CT_Schema_any
308 }
309 sml_CT_Schema_any =
310     element * - (o:* | v:* | w10:* | x:*) {
311         anyAttribute*,
312         mixed { anyElement* }
313     }
314 sml_CT_Map =
315     attribute ID { xsd:unsignedInt },
316     attribute Name { xsd:string },
317     attribute RootElement { xsd:string },
318     attribute SchemaID { xsd:string },
319     attribute ShowImportExportValidationErrors { xsd:boolean },
320     attribute AutoFit { xsd:boolean },
321     attribute Append { xsd:boolean },
322     attribute PreserveSortAFLayout { xsd:boolean },
323     attribute PreserveFormat { xsd:boolean },
324     element DataBinding { sml_CT_DataBinding }?
325 sml_CT_DataBinding =
326     attribute DataBindingName { xsd:string }?,
327     attribute FileBinding { xsd:boolean }?,
328     attribute ConnectionID { xsd:unsignedInt }?,
329     attribute FileBindingName { xsd:string }?,
330     attribute DataBindingLoadMode { xsd:unsignedInt },
331     sml_CT_DataBinding_any
332 sml_CT_DataBinding_any =
333     element * - (o:* | v:* | w10:* | x:*) {
334         anyAttribute*,
335         mixed { anyElement* }
336     }
337 sml_connections = element connections { sml_CT_Connections }
338 sml_CT_Connections = element connection { sml_CT_Connection }+
339 sml_CT_Connection =
340     attribute id { xsd:unsignedInt },
341     attribute sourceFile { s_ST_Xstring }?,
342     attribute odcFile { s_ST_Xstring }?,
343
344     ## default value: false
345     attribute keepAlive { xsd:boolean }?,
346
347     ## default value: 0
348     attribute interval { xsd:unsignedInt }?,
349     attribute name { s_ST_Xstring }?,
350     attribute description { s_ST_Xstring }?,
351     attribute type { xsd:unsignedInt }?,
352
353     ## default value: 1
354     attribute reconnectionMethod { xsd:unsignedInt }?,
355     attribute refreshedVersion { xsd:unsignedByte },
356
357     ## default value: 0
358     attribute minRefreshableVersion { xsd:unsignedByte }?,

```

```

359
360 ## default value: false
361 attribute savePassword { xsd:boolean }?,
362
363 ## default value: false
364 attribute new { xsd:boolean }?,
365
366 ## default value: false
367 attribute deleted { xsd:boolean }?,
368
369 ## default value: false
370 attribute onlyUseConnectionFile { xsd:boolean }?,
371
372 ## default value: false
373 attribute background { xsd:boolean }?,
374
375 ## default value: false
376 attribute refreshOnLoad { xsd:boolean }?,
377
378 ## default value: false
379 attribute saveData { xsd:boolean }?,
380
381 ## default value: integrated
382 attribute credentials { sml_ST_CredMethod }?,
383 attribute singleSignOnId { s_ST_Xstring }?,
384 element dbPr { sml_CT_DbPr }?,
385 element olapPr { sml_CT_OlapPr }?,
386 element webPr { sml_CT_WebPr }?,
387 element textPr { sml_CT_TextPr }?,
388 element parameters { sml_CT_Parameters }?,
389 element extLst { sml_CT_ExtensionList }?
390 sml_ST_CredMethod =
391     string "integrated"
392     | string "none"
393     | string "stored"
394     | string "prompt"
395 sml_CT_DbPr =
396     attribute connection { s_ST_Xstring },
397     attribute command { s_ST_Xstring }?,
398     attribute serverCommand { s_ST_Xstring }?,
399
400 ## default value: 2
401 attribute commandType { xsd:unsignedInt }?
402 sml_CT_OlapPr =
403
404 ## default value: false
405 attribute local { xsd:boolean }?,
406 attribute localConnection { s_ST_Xstring }?,
407
408 ## default value: true
409 attribute localRefresh { xsd:boolean }?,
410
411 ## default value: false

```

```

412 attribute sendLocale { xsd:boolean }?,
413 attribute rowDrillCount { xsd:unsignedInt }?,
414
415 ## default value: true
416 attribute serverFill { xsd:boolean }?,
417
418 ## default value: true
419 attribute serverNumberFormat { xsd:boolean }?,
420
421 ## default value: true
422 attribute serverFont { xsd:boolean }?,
423
424 ## default value: true
425 attribute serverFontColor { xsd:boolean }?
426 sml_CT_WebPr =
427
428 ## default value: false
429 attribute xml { xsd:boolean }?,
430
431 ## default value: false
432 attribute sourceData { xsd:boolean }?,
433
434 ## default value: false
435 attribute parsePre { xsd:boolean }?,
436
437 ## default value: false
438 attribute consecutive { xsd:boolean }?,
439
440 ## default value: false
441 attribute firstRow { xsd:boolean }?,
442
443 ## default value: false
444 attribute xl97 { xsd:boolean }?,
445
446 ## default value: false
447 attribute textDates { xsd:boolean }?,
448
449 ## default value: false
450 attribute xl2000 { xsd:boolean }?,
451 attribute url { s_ST_Xstring }?,
452 attribute post { s_ST_Xstring }?,
453
454 ## default value: false
455 attribute htmlTables { xsd:boolean }?,
456
457 ## default value: none
458 attribute htmlFormat { sml_ST_HtmlFmt }?,
459 attribute editPage { s_ST_Xstring }?,
460 element tables { sml_CT_Tables }?
461 sml_ST_HtmlFmt = string "none" | string "rtf" | string "all"
462 sml_CT_Parameters =
463 attribute count { xsd:unsignedInt }?,
464 element parameter { sml_CT_Parameter }+

```

```

465 sml_CT_Parameter =
466     attribute name { s_ST_Xstring }?,
467
468     ## default value: 0
469     attribute sqlType { xsd:int }?,
470
471     ## default value: prompt
472     attribute parameterType { sml_ST_ParameterType }?,
473
474     ## default value: false
475     attribute refreshOnChange { xsd:boolean }?,
476     attribute prompt { s_ST_Xstring }?,
477     attribute boolean { xsd:boolean }?,
478     attribute double { xsd:double }?,
479     attribute integer { xsd:int }?,
480     attribute string { s_ST_Xstring }?,
481     attribute cell { s_ST_Xstring }?
482 sml_ST_ParameterType = string "prompt" | string "value" | string "cell"
483 sml_CT_Tables =
484     attribute count { xsd:unsignedInt }?,
485     (element m { sml_CT_TableMissing }
486       | element s { sml_CT_XStringElement }
487       | element x { sml_CT_Index })+
488 sml_CT_TableMissing = empty
489 sml_CT_TextPr =
490
491     ## default value: true
492     attribute prompt { xsd:boolean }?,
493
494     ## default value: win
495     attribute fileType { sml_ST_FileType }?,
496
497     ## default value: 1252
498     attribute codePage { xsd:unsignedInt }?,
499     attribute characterSet { xsd:string }?,
500
501     ## default value: 1
502     attribute firstRow { xsd:unsignedInt }?,
503     attribute sourceFile { s_ST_Xstring }?,
504
505     ## default value: true
506     attribute delimited { xsd:boolean }?,
507
508     ## default value: .
509     attribute decimal { s_ST_Xstring }?,
510
511     ## default value: ,
512     attribute thousands { s_ST_Xstring }?,
513
514     ## default value: true
515     attribute tab { xsd:boolean }?,
516
517     ## default value: false

```

```

518     attribute space { xsd:boolean }?,
519
520     ## default value: false
521     attribute comma { xsd:boolean }?,
522
523     ## default value: false
524     attribute semicolon { xsd:boolean }?,
525
526     ## default value: false
527     attribute consecutive { xsd:boolean }?,
528
529     ## default value: doubleQuote
530     attribute qualifier { sml_ST_Qualifier }?,
531     attribute delimiter { s_ST_Xstring }?,
532     element textFields { sml_CT_TextFields }?
533 sml_ST_FileType =
534     string "mac"
535     | string "win"
536     | string "dos"
537     | string "lin"
538     | string "other"
539 sml_ST_Qualifier =
540     string "doubleQuote" | string "singleQuote" | string "none"
541 sml_CT_TextFields =
542
543     ## default value: 1
544     attribute count { xsd:unsignedInt }?,
545     element textField { sml_CT_TextField }+
546 sml_CT_TextField =
547
548     ## default value: general
549     attribute type { sml_ST_ExternalConnectionType }?,
550
551     ## default value: 0
552     attribute position { xsd:unsignedInt }?
553 sml_ST_ExternalConnectionType =
554     string "general"
555     | string "text"
556     | string "MDY"
557     | string "DMY"
558     | string "YMD"
559     | string "MYD"
560     | string "DYM"
561     | string "YDM"
562     | string "skip"
563     | string "EMD"
564 sml_pivotCacheDefinition =
565     element pivotCacheDefinition { sml_CT_PivotCacheDefinition }
566 sml_pivotCacheRecords =
567     element pivotCacheRecords { sml_CT_PivotCacheRecords }
568 sml_pivotTableDefinition =
569     element pivotTableDefinition { sml_CT_pivotTableDefinition }
570 sml_CT_PivotCacheDefinition =

```



```

571     r_id?,
572
573     ## default value: false
574     attribute invalid { xsd:boolean }?,
575
576     ## default value: true
577     attribute saveData { xsd:boolean }?,
578
579     ## default value: false
580     attribute refreshOnLoad { xsd:boolean }?,
581
582     ## default value: false
583     attribute optimizeMemory { xsd:boolean }?,
584
585     ## default value: true
586     attribute enableRefresh { xsd:boolean }?,
587     attribute refreshedBy { s_ST_Xstring }?,
588     attribute refreshedDate { xsd:double }?,
589     attribute refreshedDateIso { xsd:dateTime }?,
590
591     ## default value: false
592     attribute backgroundQuery { xsd:boolean }?,
593     attribute missingItemsLimit { xsd:unsignedInt }?,
594
595     ## default value: 0
596     attribute createdVersion { xsd:unsignedByte }?,
597
598     ## default value: 0
599     attribute refreshedVersion { xsd:unsignedByte }?,
600
601     ## default value: 0
602     attribute minRefreshableVersion { xsd:unsignedByte }?,
603     attribute recordCount { xsd:unsignedInt }?,
604
605     ## default value: false
606     attribute upgradeOnRefresh { xsd:boolean }?,
607
608     ## default value: false
609     attribute tupleCache { xsd:boolean }?,
610
611     ## default value: false
612     attribute supportSubquery { xsd:boolean }?,
613
614     ## default value: false
615     attribute supportAdvancedDrill { xsd:boolean }?,
616     element cacheSource { sml_CT_CacheSource },
617     element cacheFields { sml_CT_CacheFields },
618     element cacheHierarchies { sml_CT_CacheHierarchies }?,
619     element kpis { sml_CT_PCDKPIs }?,
620     element tupleCache { sml_CT_TupleCache }?,
621     element calculatedItems { sml_CT_CalculatedItems }?,
622     element calculatedMembers { sml_CT_CalculatedMembers }?,
623     element dimensions { sml_CT_Dimensions }?,

```

```

624     element measureGroups { sml_CT_MeasureGroups }?,
625     element maps { sml_CT_MeasureDimensionMaps }?,
626     element extLst { sml_CT_ExtensionList }?
627 sml_CT_CacheFields =
628     attribute count { xsd:unsignedInt }?,
629     element cacheField { sml_CT_CacheField }*
630 sml_CT_CacheField =
631     attribute name { s_ST_Xstring },
632     attribute caption { s_ST_Xstring }?,
633     attribute propertyName { s_ST_Xstring }?,
634
635     ## default value: false
636     attribute serverField { xsd:boolean }?,
637
638     ## default value: true
639     attribute uniqueList { xsd:boolean }?,
640     attribute numFmtId { sml_ST_NumFmtId }?,
641     attribute formula { s_ST_Xstring }?,
642
643     ## default value: 0
644     attribute sqlType { xsd:int }?,
645
646     ## default value: 0
647     attribute hierarchy { xsd:int }?,
648
649     ## default value: 0
650     attribute level { xsd:unsignedInt }?,
651
652     ## default value: true
653     attribute databaseField { xsd:boolean }?,
654     attribute mappingCount { xsd:unsignedInt }?,
655
656     ## default value: false
657     attribute memberPropertyField { xsd:boolean }?,
658     element sharedItems { sml_CT_SharedItems }?,
659     element fieldGroup { sml_CT_FieldGroup }?,
660     element mpMap { sml_CT_X }*,
661     element extLst { sml_CT_ExtensionList }?
662 sml_CT_CacheSource =
663     attribute type { sml_ST_SourceType },
664
665     ## default value: 0
666     attribute connectionId { xsd:unsignedInt }?,
667     (element worksheetSource { sml_CT_WorksheetSource }
668      | element consolidation { sml_CT_Consolidation }
669      | element extLst { sml_CT_ExtensionList }?)?
670 sml_ST_SourceType =
671     string "worksheet"
672     | string "external"
673     | string "consolidation"
674     | string "scenario"
675 sml_CT_WorksheetSource =
676     attribute ref { sml_ST_Ref }?,

```

```

677     attribute name { s_ST_Xstring }?,
678     attribute sheet { s_ST_Xstring }?,
679     r_id?
680 sml_CT_Consolidation =
681
682     ## default value: true
683     attribute autoPage { xsd:boolean }?,
684     element pages { sml_CT_Pages }?,
685     element rangeSets { sml_CT_RangeSets }
686 sml_CT_Pages =
687     attribute count { xsd:unsignedInt }?,
688     element page { sml_CT_PCDSCPage }+
689 sml_CT_PCDSCPage =
690     attribute count { xsd:unsignedInt }?,
691     element pageItem { sml_CT_PageItem }*
692 sml_CT_PageItem = attribute name { s_ST_Xstring }
693 sml_CT_RangeSets =
694     attribute count { xsd:unsignedInt }?,
695     element rangeSet { sml_CT_RangeSet }+
696 sml_CT_RangeSet =
697     attribute i1 { xsd:unsignedInt }?,
698     attribute i2 { xsd:unsignedInt }?,
699     attribute i3 { xsd:unsignedInt }?,
700     attribute i4 { xsd:unsignedInt }?,
701     attribute ref { sml_ST_Ref }?,
702     attribute name { s_ST_Xstring }?,
703     attribute sheet { s_ST_Xstring }?,
704     r_id?
705 sml_CT_SharedItems =
706
707     ## default value: true
708     attribute containsSemiMixedTypes { xsd:boolean }?,
709
710     ## default value: true
711     attribute containsNonDate { xsd:boolean }?,
712
713     ## default value: false
714     attribute containsDate { xsd:boolean }?,
715
716     ## default value: true
717     attribute containsString { xsd:boolean }?,
718
719     ## default value: false
720     attribute containsBlank { xsd:boolean }?,
721
722     ## default value: false
723     attribute containsMixedTypes { xsd:boolean }?,
724
725     ## default value: false
726     attribute containsNumber { xsd:boolean }?,
727
728     ## default value: false
729     attribute containsInteger { xsd:boolean }?,

```

```

730 attribute minValue { xsd:double }?,
731 attribute maxValue { xsd:double }?,
732 attribute minDate { xsd:dateTime }?,
733 attribute maxDate { xsd:dateTime }?,
734 attribute count { xsd:unsignedInt }?,
735
736 ## default value: false
737 attribute longText { xsd:boolean }?,
738 (element m { sml_CT_Missing }
739   | element n { sml_CT_Number }
740   | element b { sml_CT_Boolean }
741   | element e { sml_CT_Error }
742   | element s { sml_CT_String }
743   | element d { sml_CT_DateTime })*
744 sml_CT_Missing =
745   attribute u { xsd:boolean }?,
746   attribute f { xsd:boolean }?,
747   attribute c { s_ST_Xstring }?,
748   attribute cp { xsd:unsignedInt }?,
749   attribute in { xsd:unsignedInt }?,
750   attribute bc { sml_ST_UnsignedIntHex }?,
751   attribute fc { sml_ST_UnsignedIntHex }?,
752
753 ## default value: false
754 attribute i { xsd:boolean }?,
755
756 ## default value: false
757 attribute un { xsd:boolean }?,
758
759 ## default value: false
760 attribute st { xsd:boolean }?,
761
762 ## default value: false
763 attribute b { xsd:boolean }?,
764 element tpls { sml_CT_Tuples }*,
765 element x { sml_CT_X }*
766 sml_CT_Number =
767   attribute v { xsd:double },
768   attribute u { xsd:boolean }?,
769   attribute f { xsd:boolean }?,
770   attribute c { s_ST_Xstring }?,
771   attribute cp { xsd:unsignedInt }?,
772   attribute in { xsd:unsignedInt }?,
773   attribute bc { sml_ST_UnsignedIntHex }?,
774   attribute fc { sml_ST_UnsignedIntHex }?,
775
776 ## default value: false
777 attribute i { xsd:boolean }?,
778
779 ## default value: false
780 attribute un { xsd:boolean }?,
781
782 ## default value: false

```

```

783 attribute st { xsd:boolean }?,
784
785 ## default value: false
786 attribute b { xsd:boolean }?,
787 element tpls { sml_CT_Tuples }*,
788 element x { sml_CT_X }*
789 sml_CT_Boolean =
790 attribute v { xsd:boolean },
791 attribute u { xsd:boolean }?,
792 attribute f { xsd:boolean }?,
793 attribute c { s_ST_Xstring }?,
794 attribute cp { xsd:unsignedInt }?,
795 element x { sml_CT_X }*
796 sml_CT_Error =
797 attribute v { s_ST_Xstring },
798 attribute u { xsd:boolean }?,
799 attribute f { xsd:boolean }?,
800 attribute c { s_ST_Xstring }?,
801 attribute cp { xsd:unsignedInt }?,
802 attribute in { xsd:unsignedInt }?,
803 attribute bc { sml_ST_UnsignedIntHex }?,
804 attribute fc { sml_ST_UnsignedIntHex }?,
805
806 ## default value: false
807 attribute i { xsd:boolean }?,
808
809 ## default value: false
810 attribute un { xsd:boolean }?,
811
812 ## default value: false
813 attribute st { xsd:boolean }?,
814
815 ## default value: false
816 attribute b { xsd:boolean }?,
817 element tpls { sml_CT_Tuples }?,
818 element x { sml_CT_X }*
819 sml_CT_String =
820 attribute v { s_ST_Xstring },
821 attribute u { xsd:boolean }?,
822 attribute f { xsd:boolean }?,
823 attribute c { s_ST_Xstring }?,
824 attribute cp { xsd:unsignedInt }?,
825 attribute in { xsd:unsignedInt }?,
826 attribute bc { sml_ST_UnsignedIntHex }?,
827 attribute fc { sml_ST_UnsignedIntHex }?,
828
829 ## default value: false
830 attribute i { xsd:boolean }?,
831
832 ## default value: false
833 attribute un { xsd:boolean }?,
834
835 ## default value: false

```

```

836 attribute st { xsd:boolean }?,
837
838 ## default value: false
839 attribute b { xsd:boolean }?,
840 element tpls { sml_CT_Tuples }*,
841 element x { sml_CT_X }*
842 sml_CT_DateTime =
843 attribute v { xsd:dateTime },
844 attribute u { xsd:boolean }?,
845 attribute f { xsd:boolean }?,
846 attribute c { s_ST_Xstring }?,
847 attribute cp { xsd:unsignedInt }?,
848 element x { sml_CT_X }*
849 sml_CT_FieldGroup =
850 attribute par { xsd:unsignedInt }?,
851 attribute base { xsd:unsignedInt }?,
852 element rangePr { sml_CT_RangePr }?,
853 element discretePr { sml_CT_DiscretePr }?,
854 element groupItems { sml_CT_GroupItems }?
855 sml_CT_RangePr =
856
857 ## default value: true
858 attribute autoStart { xsd:boolean }?,
859
860 ## default value: true
861 attribute autoEnd { xsd:boolean }?,
862
863 ## default value: range
864 attribute groupBy { sml_ST_GroupBy }?,
865 attribute startNum { xsd:double }?,
866 attribute endNum { xsd:double }?,
867 attribute startDate { xsd:dateTime }?,
868 attribute endDate { xsd:dateTime }?,
869
870 ## default value: 1
871 attribute groupInterval { xsd:double }?
872 sml_ST_GroupBy =
873 string "range"
874 | string "seconds"
875 | string "minutes"
876 | string "hours"
877 | string "days"
878 | string "months"
879 | string "quarters"
880 | string "years"
881 sml_CT_DiscretePr =
882 attribute count { xsd:unsignedInt }?,
883 element x { sml_CT_Index }+
884 sml_CT_GroupItems =
885 attribute count { xsd:unsignedInt }?,
886 (element m { sml_CT_Missing }
887 | element n { sml_CT_Number }
888 | element b { sml_CT_Boolean }

```

```

889 | element e { sml_CT_Error }
890 | element s { sml_CT_String }
891 | element d { sml_CT_DateTime })+
892 sml_CT_PivotCacheRecords =
893   attribute count { xsd:unsignedInt }?,
894   element r { sml_CT_Record }*,
895   element extList { sml_CT_ExtensionList }?
896 sml_CT_Record =
897   (element m { sml_CT_Missing }
898     | element n { sml_CT_Number }
899     | element b { sml_CT_Boolean }
900     | element e { sml_CT_Error }
901     | element s { sml_CT_String }
902     | element d { sml_CT_DateTime }
903     | element x { sml_CT_Index })+
904 sml_CT_PCDKPIs =
905   attribute count { xsd:unsignedInt }?,
906   element kpi { sml_CT_PCDKPI }*
907 sml_CT_PCDKPI =
908   attribute uniqueName { s_ST_Xstring },
909   attribute caption { s_ST_Xstring }?,
910   attribute displayFolder { s_ST_Xstring }?,
911   attribute measureGroup { s_ST_Xstring }?,
912   attribute parent { s_ST_Xstring }?,
913   attribute value { s_ST_Xstring },
914   attribute goal { s_ST_Xstring }?,
915   attribute status { s_ST_Xstring }?,
916   attribute trend { s_ST_Xstring }?,
917   attribute weight { s_ST_Xstring }?,
918   attribute time { s_ST_Xstring }?
919 sml_CT_CacheHierarchies =
920   attribute count { xsd:unsignedInt }?,
921   element cacheHierarchy { sml_CT_CacheHierarchy }*
922 sml_CT_CacheHierarchy =
923   attribute uniqueName { s_ST_Xstring },
924   attribute caption { s_ST_Xstring }?,
925
926   ## default value: false
927   attribute measure { xsd:boolean }?,
928
929   ## default value: false
930   attribute set { xsd:boolean }?,
931   attribute parentSet { xsd:unsignedInt }?,
932
933   ## default value: 0
934   attribute iconSet { xsd:int }?,
935
936   ## default value: false
937   attribute attribute { xsd:boolean }?,
938
939   ## default value: false
940   attribute time { xsd:boolean }?,
941

```

```

942  ## default value: false
943  attribute keyAttribute { xsd:boolean }?,
944  attribute defaultMemberUniqueName { s_ST_Xstring }?,
945  attribute allUniqueName { s_ST_Xstring }?,
946  attribute allCaption { s_ST_Xstring }?,
947  attribute dimensionUniqueName { s_ST_Xstring }?,
948  attribute displayFolder { s_ST_Xstring }?,
949  attribute measureGroup { s_ST_Xstring }?,
950
951  ## default value: false
952  attribute measures { xsd:boolean }?,
953  attribute count { xsd:unsignedInt },
954
955  ## default value: false
956  attribute oneField { xsd:boolean }?,
957  attribute memberValueDatatype { xsd:unsignedShort }?,
958  attribute unbalanced { xsd:boolean }?,
959  attribute unbalancedGroup { xsd:boolean }?,
960
961  ## default value: false
962  attribute hidden { xsd:boolean }?,
963  element fieldsUsage { sml_CT_FieldsUsage }?,
964  element groupLevels { sml_CT_GroupLevels }?,
965  element extLst { sml_CT_ExtensionList }?
966  sml_CT_FieldsUsage =
967    attribute count { xsd:unsignedInt }?,
968    element fieldUsage { sml_CT_FieldUsage }*
969  sml_CT_FieldUsage = attribute x { xsd:int }
970  sml_CT_GroupLevels =
971    attribute count { xsd:unsignedInt }?,
972    element groupLevel { sml_CT_GroupLevel }+
973  sml_CT_GroupLevel =
974    attribute uniqueName { s_ST_Xstring },
975    attribute caption { s_ST_Xstring },
976
977  ## default value: false
978  attribute user { xsd:boolean }?,
979
980  ## default value: false
981  attribute customRollUp { xsd:boolean }?,
982  element groups { sml_CT_Groups }?,
983  element extLst { sml_CT_ExtensionList }?
984  sml_CT_Groups =
985    attribute count { xsd:unsignedInt }?,
986    element group { sml_CT_LevelGroup }+
987  sml_CT_LevelGroup =
988    attribute name { s_ST_Xstring },
989    attribute uniqueName { s_ST_Xstring },
990    attribute caption { s_ST_Xstring },
991    attribute uniqueParent { s_ST_Xstring }?,
992    attribute id { xsd:int }?,
993    element groupMembers { sml_CT_GroupMembers }
994  sml_CT_GroupMembers =

```



```

995     attribute count { xsd:unsignedInt }?,
996     element groupMember { sml_CT_GroupMember }+
997 sml_CT_GroupMember =
998     attribute uniqueName { s_ST_Xstring },
999
1000     ## default value: false
1001     attribute group { xsd:boolean }?
1002 sml_CT_TupleCache =
1003     element entries { sml_CT_PCSDTCEntries }?,
1004     element sets { sml_CT_Sets }?,
1005     element queryCache { sml_CT_QueryCache }?,
1006     element serverFormats { sml_CT_ServerFormats }?,
1007     element extLst { sml_CT_ExtensionList }?
1008 sml_CT_ServerFormat =
1009     attribute culture { s_ST_Xstring }?,
1010     attribute format { s_ST_Xstring }?
1011 sml_CT_ServerFormats =
1012     attribute count { xsd:unsignedInt }?,
1013     element serverFormat { sml_CT_ServerFormat }*
1014 sml_CT_PCSDTCEntries =
1015     attribute count { xsd:unsignedInt }?,
1016     (element m { sml_CT_Missing }
1017      | element n { sml_CT_Number }
1018      | element e { sml_CT_Error }
1019      | element s { sml_CT_String })+
1020 sml_CT_Tuples =
1021     attribute c { xsd:unsignedInt }?,
1022     element tpl { sml_CT_Tuple }+
1023 sml_CT_Tuple =
1024     attribute fld { xsd:unsignedInt }?,
1025     attribute hier { xsd:unsignedInt }?,
1026     attribute item { xsd:unsignedInt }
1027 sml_CT_Sets =
1028     attribute count { xsd:unsignedInt }?,
1029     element set { sml_CT_Set }+
1030 sml_CT_Set =
1031     attribute count { xsd:unsignedInt }?,
1032     attribute maxRank { xsd:int },
1033     attribute setDefinition { s_ST_Xstring },
1034
1035     ## default value: none
1036     attribute sortType { sml_ST_SortType }?,
1037
1038     ## default value: false
1039     attribute queryFailed { xsd:boolean }?,
1040     element tpls { sml_CT_Tuples }*,
1041     element sortByTuple { sml_CT_Tuples }?
1042 sml_ST_SortType =
1043     string "none"
1044     | string "ascending"
1045     | string "descending"
1046     | string "ascendingAlpha"
1047     | string "descendingAlpha"

```

```

1048 | string "ascendingNatural"
1049 | string "descendingNatural"
1050 sml_CT_QueryCache =
1051   attribute count { xsd:unsignedInt }?,
1052   element query { sml_CT_Query }+
1053 sml_CT_Query =
1054   attribute mdx { s_ST_Xstring },
1055   element tpls { sml_CT_Tuples }?
1056 sml_CT_CalculatedItems =
1057   attribute count { xsd:unsignedInt }?,
1058   element calculatedItem { sml_CT_CalculatedItem }+
1059 sml_CT_CalculatedItem =
1060   attribute field { xsd:unsignedInt }?,
1061   attribute formula { s_ST_Xstring }?,
1062   element pivotArea { sml_CT_PivotArea },
1063   element extLst { sml_CT_ExtensionList }?
1064 sml_CT_CalculatedMembers =
1065   attribute count { xsd:unsignedInt }?,
1066   element calculatedMember { sml_CT_CalculatedMember }+
1067 sml_CT_CalculatedMember =
1068   attribute name { s_ST_Xstring },
1069   attribute mdx { s_ST_Xstring },
1070   attribute memberName { s_ST_Xstring }?,
1071   attribute hierarchy { s_ST_Xstring }?,
1072   attribute parent { s_ST_Xstring }?,
1073
1074   ## default value: 0
1075   attribute solveOrder { xsd:int }?,
1076
1077   ## default value: false
1078   attribute set { xsd:boolean }?,
1079   element extLst { sml_CT_ExtensionList }?
1080 sml_CT_pivotTableDefinition =
1081   attribute name { s_ST_Xstring },
1082   attribute cacheId { xsd:unsignedInt },
1083
1084   ## default value: false
1085   attribute dataOnRows { xsd:boolean }?,
1086   attribute dataPosition { xsd:unsignedInt }?,
1087   sml_AG_AutoFormat,
1088   attribute dataCaption { s_ST_Xstring },
1089   attribute grandTotalCaption { s_ST_Xstring }?,
1090   attribute errorCaption { s_ST_Xstring }?,
1091
1092   ## default value: false
1093   attribute showError { xsd:boolean }?,
1094   attribute missingCaption { s_ST_Xstring }?,
1095
1096   ## default value: true
1097   attribute showMissing { xsd:boolean }?,
1098   attribute pageStyle { s_ST_Xstring }?,
1099   attribute pivotTableStyle { s_ST_Xstring }?,
1100   attribute vacatedStyle { s_ST_Xstring }?,

```

```

1101 attribute tag { s_ST_Xstring }?,
1102
1103 ## default value: 0
1104 attribute updatedVersion { xsd:unsignedByte }?,
1105
1106 ## default value: 0
1107 attribute minRefreshableVersion { xsd:unsignedByte }?,
1108
1109 ## default value: false
1110 attribute asteriskTotals { xsd:boolean }?,
1111
1112 ## default value: true
1113 attribute showItems { xsd:boolean }?,
1114
1115 ## default value: false
1116 attribute editData { xsd:boolean }?,
1117
1118 ## default value: false
1119 attribute disableFieldList { xsd:boolean }?,
1120
1121 ## default value: true
1122 attribute showCalcMbrs { xsd:boolean }?,
1123
1124 ## default value: true
1125 attribute visualTotals { xsd:boolean }?,
1126
1127 ## default value: true
1128 attribute showMultipleLabel { xsd:boolean }?,
1129
1130 ## default value: true
1131 attribute showDataDropDown { xsd:boolean }?,
1132
1133 ## default value: true
1134 attribute showDrill { xsd:boolean }?,
1135
1136 ## default value: false
1137 attribute printDrill { xsd:boolean }?,
1138
1139 ## default value: true
1140 attribute showMemberPropertyTips { xsd:boolean }?,
1141
1142 ## default value: true
1143 attribute showDataTips { xsd:boolean }?,
1144
1145 ## default value: true
1146 attribute enableWizard { xsd:boolean }?,
1147
1148 ## default value: true
1149 attribute enableDrill { xsd:boolean }?,
1150
1151 ## default value: true
1152 attribute enableFieldProperties { xsd:boolean }?,
1153

```

```
1154 ## default value: true
1155 attribute preserveFormatting { xsd:boolean }?,
1156
1157 ## default value: false
1158 attribute useAutoFormatting { xsd:boolean }?,
1159
1160 ## default value: 0
1161 attribute pageWrap { xsd:unsignedInt }?,
1162
1163 ## default value: false
1164 attribute pageOverThenDown { xsd:boolean }?,
1165
1166 ## default value: false
1167 attribute subtotalHiddenItems { xsd:boolean }?,
1168
1169 ## default value: true
1170 attribute rowGrandTotals { xsd:boolean }?,
1171
1172 ## default value: true
1173 attribute colGrandTotals { xsd:boolean }?,
1174
1175 ## default value: false
1176 attribute fieldPrintTitles { xsd:boolean }?,
1177
1178 ## default value: false
1179 attribute itemPrintTitles { xsd:boolean }?,
1180
1181 ## default value: false
1182 attribute mergeItem { xsd:boolean }?,
1183
1184 ## default value: true
1185 attribute showDropZones { xsd:boolean }?,
1186
1187 ## default value: 0
1188 attribute createdVersion { xsd:unsignedByte }?,
1189
1190 ## default value: 1
1191 attribute indent { xsd:unsignedInt }?,
1192
1193 ## default value: false
1194 attribute showEmptyRow { xsd:boolean }?,
1195
1196 ## default value: false
1197 attribute showEmptyCol { xsd:boolean }?,
1198
1199 ## default value: true
1200 attribute showHeaders { xsd:boolean }?,
1201
1202 ## default value: true
1203 attribute compact { xsd:boolean }?,
1204
1205 ## default value: false
1206 attribute outline { xsd:boolean }?,
```

```

1207
1208 ## default value: false
1209 attribute outlineData { xsd:boolean }?,
1210
1211 ## default value: true
1212 attribute compactData { xsd:boolean }?,
1213
1214 ## default value: false
1215 attribute published { xsd:boolean }?,
1216
1217 ## default value: false
1218 attribute gridDropZones { xsd:boolean }?,
1219
1220 ## default value: true
1221 attribute immersive { xsd:boolean }?,
1222
1223 ## default value: true
1224 attribute multipleFieldFilters { xsd:boolean }?,
1225
1226 ## default value: 0
1227 attribute chartFormat { xsd:unsignedInt }?,
1228 attribute rowHeaderCaption { s_ST_Xstring }?,
1229 attribute colHeaderCaption { s_ST_Xstring }?,
1230
1231 ## default value: false
1232 attribute fieldListSortAscending { xsd:boolean }?,
1233
1234 ## default value: false
1235 attribute mdxSubqueries { xsd:boolean }?,
1236
1237 ## default value: true
1238 attribute customListSort { xsd:boolean }?,
1239 element location { sml_CT_Location },
1240 element pivotFields { sml_CT_PivotFields }?,
1241 element rowFields { sml_CT_RowFields }?,
1242 element rowItems { sml_CT_rowItems }?,
1243 element colFields { sml_CT_ColFields }?,
1244 element colItems { sml_CT_colItems }?,
1245 element pageFields { sml_CT_PageFields }?,
1246 element dataFields { sml_CT_DataFields }?,
1247 element formats { sml_CT_Formats }?,
1248 element conditionalFormats { sml_CT_ConditionalFormats }?,
1249 element chartFormats { sml_CT_ChartFormats }?,
1250 element pivotHierarchies { sml_CT_PivotHierarchies }?,
1251 element pivotTableStyleInfo { sml_CT_PivotTableStyle }?,
1252 element filters { sml_CT_PivotFilters }?,
1253 element rowHierarchiesUsage { sml_CT_RowHierarchiesUsage }?,
1254 element colHierarchiesUsage { sml_CT_ColHierarchiesUsage }?,
1255 element extLst { sml_CT_ExtensionList }?
1256 sml_CT_Location =
1257 attribute ref { sml_ST_Ref },
1258 attribute firstHeaderRow { xsd:unsignedInt },
1259 attribute firstDataRow { xsd:unsignedInt },

```

```

1260     attribute firstDataCol { xsd:unsignedInt },
1261
1262     ## default value: 0
1263     attribute rowPageCount { xsd:unsignedInt }?,
1264
1265     ## default value: 0
1266     attribute colPageCount { xsd:unsignedInt }?
1267 sml_CT_PivotFields =
1268     attribute count { xsd:unsignedInt }?,
1269     element pivotField { sml_CT_PivotField }+
1270 sml_CT_PivotField =
1271     attribute name { s_ST_Xstring }?,
1272     attribute axis { sml_ST_Axis }?,
1273
1274     ## default value: false
1275     attribute dataField { xsd:boolean }?,
1276     attribute subtotalCaption { s_ST_Xstring }?,
1277
1278     ## default value: true
1279     attribute showDropDowns { xsd:boolean }?,
1280
1281     ## default value: false
1282     attribute hiddenLevel { xsd:boolean }?,
1283     attribute uniqueMemberProperty { s_ST_Xstring }?,
1284
1285     ## default value: true
1286     attribute compact { xsd:boolean }?,
1287
1288     ## default value: false
1289     attribute allDrilled { xsd:boolean }?,
1290     attribute numFmtId { sml_ST_NumFmtId }?,
1291
1292     ## default value: true
1293     attribute outline { xsd:boolean }?,
1294
1295     ## default value: true
1296     attribute subtotalTop { xsd:boolean }?,
1297
1298     ## default value: true
1299     attribute dragToRow { xsd:boolean }?,
1300
1301     ## default value: true
1302     attribute dragToCol { xsd:boolean }?,
1303
1304     ## default value: false
1305     attribute multipleItemSelectionAllowed { xsd:boolean }?,
1306
1307     ## default value: true
1308     attribute dragToPage { xsd:boolean }?,
1309
1310     ## default value: true
1311     attribute dragToData { xsd:boolean }?,
1312

```

```

1313 ## default value: true
1314 attribute dragOff { xsd:boolean }?,
1315
1316 ## default value: true
1317 attribute showAll { xsd:boolean }?,
1318
1319 ## default value: false
1320 attribute insertBlankRow { xsd:boolean }?,
1321
1322 ## default value: false
1323 attribute serverField { xsd:boolean }?,
1324
1325 ## default value: false
1326 attribute insertPageBreak { xsd:boolean }?,
1327
1328 ## default value: false
1329 attribute autoShow { xsd:boolean }?,
1330
1331 ## default value: true
1332 attribute topAutoShow { xsd:boolean }?,
1333
1334 ## default value: false
1335 attribute hideNewItems { xsd:boolean }?,
1336
1337 ## default value: false
1338 attribute measureFilter { xsd:boolean }?,
1339
1340 ## default value: false
1341 attribute includeNewItemsInFilter { xsd:boolean }?,
1342
1343 ## default value: 10
1344 attribute itemPageCount { xsd:unsignedInt }?,
1345
1346 ## default value: manual
1347 attribute sortType { sml_ST_FieldSortType }?,
1348 attribute dataSourceSort { xsd:boolean }?,
1349
1350 ## default value: false
1351 attribute nonAutoSortDefault { xsd:boolean }?,
1352 attribute rankBy { xsd:unsignedInt }?,
1353
1354 ## default value: true
1355 attribute defaultSubtotal { xsd:boolean }?,
1356
1357 ## default value: false
1358 attribute sumSubtotal { xsd:boolean }?,
1359
1360 ## default value: false
1361 attribute countASubtotal { xsd:boolean }?,
1362
1363 ## default value: false
1364 attribute avgSubtotal { xsd:boolean }?,
1365

```

```

1366   ## default value: false
1367   attribute maxSubtotal { xsd:boolean }?,
1368
1369   ## default value: false
1370   attribute minSubtotal { xsd:boolean }?,
1371
1372   ## default value: false
1373   attribute productSubtotal { xsd:boolean }?,
1374
1375   ## default value: false
1376   attribute countSubtotal { xsd:boolean }?,
1377
1378   ## default value: false
1379   attribute stdDevSubtotal { xsd:boolean }?,
1380
1381   ## default value: false
1382   attribute stdDevPSubtotal { xsd:boolean }?,
1383
1384   ## default value: false
1385   attribute varSubtotal { xsd:boolean }?,
1386
1387   ## default value: false
1388   attribute varPSubtotal { xsd:boolean }?,
1389
1390   ## default value: false
1391   attribute showPropCell { xsd:boolean }?,
1392
1393   ## default value: false
1394   attribute showPropTip { xsd:boolean }?,
1395
1396   ## default value: false
1397   attribute showPropAsCaption { xsd:boolean }?,
1398
1399   ## default value: false
1400   attribute defaultAttributeDrillState { xsd:boolean }?,
1401   element items { sml_CT_Items }?,
1402   element autoSortScope { sml_CT_AutoSortScope }?,
1403   element extLst { sml_CT_ExtensionList }?
1404 sml_CT_AutoSortScope = element pivotArea { sml_CT_PivotArea }
1405 sml_CT_Items =
1406   attribute count { xsd:unsignedInt }?,
1407   element item { sml_CT_Item }+
1408 sml_CT_Item =
1409   attribute n { s_ST_Xstring }?,
1410
1411   ## default value: data
1412   attribute t { sml_ST_ItemType }?,
1413
1414   ## default value: false
1415   attribute h { xsd:boolean }?,
1416
1417   ## default value: false
1418   attribute s { xsd:boolean }?,

```



```

1419
1420   ## default value: true
1421   attribute sd { xsd:boolean }?,
1422
1423   ## default value: false
1424   attribute f { xsd:boolean }?,
1425
1426   ## default value: false
1427   attribute m { xsd:boolean }?,
1428
1429   ## default value: false
1430   attribute c { xsd:boolean }?,
1431   attribute x { xsd:unsignedInt }?,
1432
1433   ## default value: false
1434   attribute d { xsd:boolean }?,
1435
1436   ## default value: true
1437   attribute e { xsd:boolean }?
1438 sml_CT_PageFields =
1439   attribute count { xsd:unsignedInt }?,
1440   element pageField { sml_CT_PageField }+
1441 sml_CT_PageField =
1442   attribute fld { xsd:int },
1443   attribute item { xsd:unsignedInt }?,
1444   attribute hier { xsd:int }?,
1445   attribute name { s_ST_Xstring }?,
1446   attribute cap { s_ST_Xstring }?,
1447   element extLst { sml_CT_ExtensionList }?
1448 sml_CT_DataFields =
1449   attribute count { xsd:unsignedInt }?,
1450   element dataField { sml_CT_DataField }+
1451 sml_CT_DataField =
1452   attribute name { s_ST_Xstring }?,
1453   attribute fld { xsd:unsignedInt },
1454
1455   ## default value: sum
1456   attribute subtotal { sml_ST_DataConsolidateFunction }?,
1457
1458   ## default value: normal
1459   attribute showDataAs { sml_ST_ShowDataAs }?,
1460
1461   ## default value: -1
1462   attribute baseField { xsd:int }?,
1463
1464   ## default value: 1048832
1465   attribute baseItem { xsd:unsignedInt }?,
1466   attribute numFmtId { sml_ST_NumFmtId }?,
1467   element extLst { sml_CT_ExtensionList }?
1468 sml_CT_rowItems =
1469   attribute count { xsd:unsignedInt }?,
1470   element i { sml_CT_I }+
1471 sml_CT_colItems =

```

```

1472     attribute count { xsd:unsignedInt }?,
1473     element i { sml_CT_I }+
1474 sml_CT_I =
1475
1476     ## default value: data
1477     attribute t { sml_ST_ItemType }?,
1478
1479     ## default value: 0
1480     attribute r { xsd:unsignedInt }?,
1481
1482     ## default value: 0
1483     attribute i { xsd:unsignedInt }?,
1484     element x { sml_CT_X }*
1485 sml_CT_X =
1486
1487     ## default value: 0
1488     attribute v { xsd:int }?
1489 sml_CT_RowFields =
1490
1491     ## default value: 0
1492     attribute count { xsd:unsignedInt }?,
1493     element field { sml_CT_Field }+
1494 sml_CT_ColFields =
1495
1496     ## default value: 0
1497     attribute count { xsd:unsignedInt }?,
1498     element field { sml_CT_Field }+
1499 sml_CT_Field = attribute x { xsd:int }
1500 sml_CT_Formats =
1501
1502     ## default value: 0
1503     attribute count { xsd:unsignedInt }?,
1504     element format { sml_CT_Format }+
1505 sml_CT_Format =
1506
1507     ## default value: formatting
1508     attribute action { sml_ST_FormatAction }?,
1509     attribute dxId { sml_ST_DxId }?,
1510     element pivotArea { sml_CT_PivotArea },
1511     element extLst { sml_CT_ExtensionList }?
1512 sml_CT_ConditionalFormats =
1513
1514     ## default value: 0
1515     attribute count { xsd:unsignedInt }?,
1516     element conditionalFormat { sml_CT_ConditionalFormat }+
1517 sml_CT_ConditionalFormat =
1518
1519     ## default value: selection
1520     attribute scope { sml_ST_Scope }?,
1521
1522     ## default value: none
1523     attribute type { sml_ST_Type }?,
1524     attribute priority { xsd:unsignedInt },

```

```

1525     element pivotAreas { sml_CT_PivotAreas },
1526     element extLst { sml_CT_ExtensionList }?
1527 sml_CT_PivotAreas =
1528     attribute count { xsd:unsignedInt }?,
1529     element pivotArea { sml_CT_PivotArea }*
1530 sml_ST_Scope = string "selection" | string "data" | string "field"
1531 sml_ST_Type =
1532     string "none" | string "all" | string "row" | string "column"
1533 sml_CT_ChartFormats =
1534
1535     ## default value: 0
1536     attribute count { xsd:unsignedInt }?,
1537     element chartFormat { sml_CT_ChartFormat }+
1538 sml_CT_ChartFormat =
1539     attribute chart { xsd:unsignedInt },
1540     attribute format { xsd:unsignedInt },
1541
1542     ## default value: false
1543     attribute series { xsd:boolean }?,
1544     element pivotArea { sml_CT_PivotArea }
1545 sml_CT_PivotHierarchies =
1546     attribute count { xsd:unsignedInt }?,
1547     element pivotHierarchy { sml_CT_PivotHierarchy }+
1548 sml_CT_PivotHierarchy =
1549
1550     ## default value: false
1551     attribute outline { xsd:boolean }?,
1552
1553     ## default value: false
1554     attribute multipleItemSelectionAllowed { xsd:boolean }?,
1555
1556     ## default value: false
1557     attribute subtotalTop { xsd:boolean }?,
1558
1559     ## default value: true
1560     attribute showInFieldList { xsd:boolean }?,
1561
1562     ## default value: true
1563     attribute dragToRow { xsd:boolean }?,
1564
1565     ## default value: true
1566     attribute dragToCol { xsd:boolean }?,
1567
1568     ## default value: true
1569     attribute dragToPage { xsd:boolean }?,
1570
1571     ## default value: false
1572     attribute dragToData { xsd:boolean }?,
1573
1574     ## default value: true
1575     attribute dragOff { xsd:boolean }?,
1576
1577     ## default value: false

```

```

1578     attribute includeNewItemInFilter { xsd:boolean }?,
1579     attribute caption { s_ST_Xstring }?,
1580     element mps { sml_CT_MemberProperties }?,
1581     element members { sml_CT_Members }*,
1582     element extLst { sml_CT_ExtensionList }?
1583 sml_CT_RowHierarchiesUsage =
1584     attribute count { xsd:unsignedInt }?,
1585     element rowHierarchyUsage { sml_CT_HierarchyUsage }+
1586 sml_CT_ColHierarchiesUsage =
1587     attribute count { xsd:unsignedInt }?,
1588     element colHierarchyUsage { sml_CT_HierarchyUsage }+
1589 sml_CT_HierarchyUsage = attribute hierarchyUsage { xsd:int }
1590 sml_CT_MemberProperties =
1591     attribute count { xsd:unsignedInt }?,
1592     element mp { sml_CT_MemberProperty }+
1593 sml_CT_MemberProperty =
1594     attribute name { s_ST_Xstring }?,
1595
1596     ## default value: false
1597     attribute showCell { xsd:boolean }?,
1598
1599     ## default value: false
1600     attribute showTip { xsd:boolean }?,
1601
1602     ## default value: false
1603     attribute showAsCaption { xsd:boolean }?,
1604     attribute nameLen { xsd:unsignedInt }?,
1605     attribute pPos { xsd:unsignedInt }?,
1606     attribute plen { xsd:unsignedInt }?,
1607     attribute level { xsd:unsignedInt }?,
1608     attribute field { xsd:unsignedInt }
1609 sml_CT_Members =
1610     attribute count { xsd:unsignedInt }?,
1611     attribute level { xsd:unsignedInt }?,
1612     element member { sml_CT_Member }+
1613 sml_CT_Member = attribute name { s_ST_Xstring }
1614 sml_CT_Dimensions =
1615     attribute count { xsd:unsignedInt }?,
1616     element dimension { sml_CT_PivotDimension }*
1617 sml_CT_PivotDimension =
1618
1619     ## default value: false
1620     attribute measure { xsd:boolean }?,
1621     attribute name { s_ST_Xstring },
1622     attribute uniqueName { s_ST_Xstring },
1623     attribute caption { s_ST_Xstring }
1624 sml_CT_MeasureGroups =
1625     attribute count { xsd:unsignedInt }?,
1626     element measureGroup { sml_CT_MeasureGroup }*
1627 sml_CT_MeasureDimensionMaps =
1628     attribute count { xsd:unsignedInt }?,
1629     element map { sml_CT_MeasureDimensionMap }*
1630 sml_CT_MeasureGroup =

```

```

1631     attribute name { s_ST_Xstring },
1632     attribute caption { s_ST_Xstring }
1633 sml_CT_MeasureDimensionMap =
1634     attribute measureGroup { xsd:unsignedInt }?,
1635     attribute dimension { xsd:unsignedInt }?
1636 sml_CT_PivotTableStyle =
1637     attribute name { xsd:string }?,
1638     attribute showRowHeaders { xsd:boolean }?,
1639     attribute showColHeaders { xsd:boolean }?,
1640     attribute showRowStripes { xsd:boolean }?,
1641     attribute showColStripes { xsd:boolean }?,
1642     attribute showLastColumn { xsd:boolean }?
1643 sml_CT_PivotFilters =
1644
1645     ## default value: 0
1646     attribute count { xsd:unsignedInt }?,
1647     element filter { sml_CT_PivotFilter }*
1648 sml_CT_PivotFilter =
1649     attribute fld { xsd:unsignedInt },
1650     attribute mpFld { xsd:unsignedInt }?,
1651     attribute type { sml_ST_PivotFilterType },
1652
1653     ## default value: 0
1654     attribute evalOrder { xsd:int }?,
1655     attribute id { xsd:unsignedInt },
1656     attribute iMeasureHier { xsd:unsignedInt }?,
1657     attribute iMeasureFld { xsd:unsignedInt }?,
1658     attribute name { s_ST_Xstring }?,
1659     attribute description { s_ST_Xstring }?,
1660     attribute stringValue1 { s_ST_Xstring }?,
1661     attribute stringValue2 { s_ST_Xstring }?,
1662     element autoFilter { sml_CT_AutoFilter },
1663     element extLst { sml_CT_ExtensionList }?
1664 sml_ST_ShowDataAs =
1665     string "normal"
1666     | string "difference"
1667     | string "percent"
1668     | string "percentDiff"
1669     | string "runTotal"
1670     | string "percentOfRow"
1671     | string "percentOfCol"
1672     | string "percentOfTotal"
1673     | string "index"
1674 sml_ST_ItemType =
1675     string "data"
1676     | string "default"
1677     | string "sum"
1678     | string "countA"
1679     | string "avg"
1680     | string "max"
1681     | string "min"
1682     | string "product"
1683     | string "count"

```

```

1684 | string "stdDev"
1685 | string "stdDevP"
1686 | string "var"
1687 | string "varP"
1688 | string "grand"
1689 | string "blank"
1690 sml_ST_FormatAction =
1691     string "blank"
1692     | string "formatting"
1693     | string "drill"
1694     | string "formula"
1695 sml_ST_FieldSortType =
1696     string "manual" | string "ascending" | string "descending"
1697 sml_ST_PivotFilterType =
1698     string "unknown"
1699     | string "count"
1700     | string "percent"
1701     | string "sum"
1702     | string "captionEqual"
1703     | string "captionNotEqual"
1704     | string "captionBeginsWith"
1705     | string "captionNotBeginsWith"
1706     | string "captionEndsWith"
1707     | string "captionNotEndsWith"
1708     | string "captionContains"
1709     | string "captionNotContains"
1710     | string "captionGreaterThan"
1711     | string "captionGreaterThanOrEqual"
1712     | string "captionLessThan"
1713     | string "captionLessThanOrEqual"
1714     | string "captionBetween"
1715     | string "captionNotBetween"
1716     | string "valueEqual"
1717     | string "valueNotEqual"
1718     | string "valueGreaterThan"
1719     | string "valueGreaterThanOrEqual"
1720     | string "valueLessThan"
1721     | string "valueLessThanOrEqual"
1722     | string "valueBetween"
1723     | string "valueNotBetween"
1724     | string "dateEqual"
1725     | string "dateNotEqual"
1726     | string "dateOlderThan"
1727     | string "dateOlderThanOrEqual"
1728     | string "dateNewerThan"
1729     | string "dateNewerThanOrEqual"
1730     | string "dateBetween"
1731     | string "dateNotBetween"
1732     | string "tomorrow"
1733     | string "today"
1734     | string "yesterday"
1735     | string "nextWeek"
1736     | string "thisWeek"

```

```

1737 | string "lastWeek"
1738 | string "nextMonth"
1739 | string "thisMonth"
1740 | string "lastMonth"
1741 | string "nextQuarter"
1742 | string "thisQuarter"
1743 | string "lastQuarter"
1744 | string "nextYear"
1745 | string "thisYear"
1746 | string "lastYear"
1747 | string "yearToDate"
1748 | string "Q1"
1749 | string "Q2"
1750 | string "Q3"
1751 | string "Q4"
1752 | string "M1"
1753 | string "M2"
1754 | string "M3"
1755 | string "M4"
1756 | string "M5"
1757 | string "M6"
1758 | string "M7"
1759 | string "M8"
1760 | string "M9"
1761 | string "M10"
1762 | string "M11"
1763 | string "M12"
1764 sml_CT_PivotArea =
1765     attribute field { xsd:int }?,
1766
1767     ## default value: normal
1768     attribute type { sml_ST_PivotAreaType }?,
1769
1770     ## default value: true
1771     attribute dataOnly { xsd:boolean }?,
1772
1773     ## default value: false
1774     attribute labelOnly { xsd:boolean }?,
1775
1776     ## default value: false
1777     attribute grandRow { xsd:boolean }?,
1778
1779     ## default value: false
1780     attribute grandCol { xsd:boolean }?,
1781
1782     ## default value: false
1783     attribute cacheIndex { xsd:boolean }?,
1784
1785     ## default value: true
1786     attribute outline { xsd:boolean }?,
1787     attribute offset { sml_ST_Ref }?,
1788
1789     ## default value: false

```

```

1790     attribute collapsedLevelsAreSubtotals { xsd:boolean }?,
1791     attribute axis { sml_ST_Axis }?,
1792     attribute fieldPosition { xsd:unsignedInt }?,
1793     element references { sml_CT_PivotAreaReferences }?,
1794     element extLst { sml_CT_ExtensionList }?
1795 sml_ST_PivotAreaType =
1796     string "none"
1797     | string "normal"
1798     | string "data"
1799     | string "all"
1800     | string "origin"
1801     | string "button"
1802     | string "topEnd"
1803     | string "topRight"
1804 sml_CT_PivotAreaReferences =
1805     attribute count { xsd:unsignedInt }?,
1806     element reference { sml_CT_PivotAreaReference }+
1807 sml_CT_PivotAreaReference =
1808     attribute field { xsd:unsignedInt }?,
1809     attribute count { xsd:unsignedInt }?,
1810
1811     ## default value: true
1812     attribute selected { xsd:boolean }?,
1813
1814     ## default value: false
1815     attribute byPosition { xsd:boolean }?,
1816
1817     ## default value: false
1818     attribute relative { xsd:boolean }?,
1819
1820     ## default value: false
1821     attribute defaultSubtotal { xsd:boolean }?,
1822
1823     ## default value: false
1824     attribute sumSubtotal { xsd:boolean }?,
1825
1826     ## default value: false
1827     attribute countASubtotal { xsd:boolean }?,
1828
1829     ## default value: false
1830     attribute avgSubtotal { xsd:boolean }?,
1831
1832     ## default value: false
1833     attribute maxSubtotal { xsd:boolean }?,
1834
1835     ## default value: false
1836     attribute minSubtotal { xsd:boolean }?,
1837
1838     ## default value: false
1839     attribute productSubtotal { xsd:boolean }?,
1840
1841     ## default value: false
1842     attribute countSubtotal { xsd:boolean }?,

```



```

1843
1844   ## default value: false
1845   attribute stdDevSubtotal { xsd:boolean }?,
1846
1847   ## default value: false
1848   attribute stdDevPSubtotal { xsd:boolean }?,
1849
1850   ## default value: false
1851   attribute varSubtotal { xsd:boolean }?,
1852
1853   ## default value: false
1854   attribute varPSubtotal { xsd:boolean }?,
1855   element x { sml_CT_Index }*,
1856   element extLst { sml_CT_ExtensionList }?
1857 sml_CT_Index = attribute v { xsd:unsignedInt }
1858 sml_ST_Axis =
1859     string "axisRow"
1860     | string "axisCol"
1861     | string "axisPage"
1862     | string "axisValues"
1863 sml_queryTable = element queryTable { sml_CT_QueryTable }
1864 sml_CT_QueryTable =
1865     attribute name { s_ST_Xstring },
1866
1867     ## default value: true
1868     attribute headers { xsd:boolean }?,
1869
1870     ## default value: false
1871     attribute rowNumbers { xsd:boolean }?,
1872
1873     ## default value: false
1874     attribute disableRefresh { xsd:boolean }?,
1875
1876     ## default value: true
1877     attribute backgroundRefresh { xsd:boolean }?,
1878
1879     ## default value: false
1880     attribute firstBackgroundRefresh { xsd:boolean }?,
1881
1882     ## default value: false
1883     attribute refreshOnLoad { xsd:boolean }?,
1884
1885     ## default value: insertDelete
1886     attribute growShrinkType { sml_ST_GrowShrinkType }?,
1887
1888     ## default value: false
1889     attribute fillFormulas { xsd:boolean }?,
1890
1891     ## default value: false
1892     attribute removeDataOnSave { xsd:boolean }?,
1893
1894     ## default value: false
1895     attribute disableEdit { xsd:boolean }?,

```

```

1896
1897   ## default value: true
1898   attribute preserveFormatting { xsd:boolean }?,
1899
1900   ## default value: true
1901   attribute adjustColumnWidth { xsd:boolean }?,
1902
1903   ## default value: false
1904   attribute intermediate { xsd:boolean }?,
1905   attribute connectionId { xsd:unsignedInt },
1906   sml_AG_AutoFormat,
1907   element queryTableRefresh { sml_CT_QueryTableRefresh }?,
1908   element extLst { sml_CT_ExtensionList }?
1909 sml_CT_QueryTableRefresh =
1910
1911   ## default value: true
1912   attribute preserveSortFilterLayout { xsd:boolean }?,
1913
1914   ## default value: false
1915   attribute fieldIdWrapped { xsd:boolean }?,
1916
1917   ## default value: true
1918   attribute headersInLastRefresh { xsd:boolean }?,
1919
1920   ## default value: 0
1921   attribute minimumVersion { xsd:unsignedByte }?,
1922
1923   ## default value: 1
1924   attribute nextId { xsd:unsignedInt }?,
1925
1926   ## default value: 0
1927   attribute unboundColumnsLeft { xsd:unsignedInt }?,
1928
1929   ## default value: 0
1930   attribute unboundColumnsRight { xsd:unsignedInt }?,
1931   element queryTableFields { sml_CT_QueryTableFields },
1932   element queryTableDeletedFields { sml_CT_QueryTableDeletedFields }?,
1933   element sortState { sml_CT_SortState }?,
1934   element extLst { sml_CT_ExtensionList }?
1935 sml_CT_QueryTableDeletedFields =
1936   attribute count { xsd:unsignedInt }?,
1937   element deletedField { sml_CT_DeletedField }+
1938 sml_CT_DeletedField = attribute name { s_ST_Xstring }
1939 sml_CT_QueryTableFields =
1940
1941   ## default value: 0
1942   attribute count { xsd:unsignedInt }?,
1943   element queryTableField { sml_CT_QueryTableField }*
1944 sml_CT_QueryTableField =
1945   attribute id { xsd:unsignedInt },
1946   attribute name { s_ST_Xstring }?,
1947
1948   ## default value: true

```

```

1949 attribute dataBound { xsd:boolean }?,
1950
1951 ## default value: false
1952 attribute rowNumbers { xsd:boolean }?,
1953
1954 ## default value: false
1955 attribute fillFormulas { xsd:boolean }?,
1956
1957 ## default value: false
1958 attribute clipped { xsd:boolean }?,
1959
1960 ## default value: 0
1961 attribute tableColumnId { xsd:unsignedInt }?,
1962 element extLst { sml_CT_ExtensionList }?
1963 sml_ST_GrowShrinkType =
1964     string "insertDelete" | string "insertClear" | string "overwriteClear"
1965 sml_sst = element sst { sml_CT_Sst }
1966 sml_CT_Sst =
1967     attribute count { xsd:unsignedInt }?,
1968     attribute uniqueCount { xsd:unsignedInt }?,
1969     element si { sml_CT_Rst }*,
1970     element extLst { sml_CT_ExtensionList }?
1971 sml_ST_PhoneticType =
1972     string "halfwidthKatakana"
1973     | string "fullwidthKatakana"
1974     | string "Hiragana"
1975     | string "noConversion"
1976 sml_ST_PhoneticAlignment =
1977     string "noControl"
1978     | string "left"
1979     | string "center"
1980     | string "distributed"
1981 sml_CT_PhoneticRun =
1982     attribute sb { xsd:unsignedInt },
1983     attribute eb { xsd:unsignedInt },
1984     element t { s_ST_Xstring }
1985 sml_CT_RElt =
1986     element rPr { sml_CT_RPrElt }?,
1987     element t { s_ST_Xstring }
1988 sml_CT_RPrElt =
1989     (element rFont { sml_CT_FontName }?
1990     | element charset { sml_CT_IntProperty }?
1991     | element family { sml_CT_IntProperty }?
1992     | element b { sml_CT_BooleanProperty }?
1993     | element i { sml_CT_BooleanProperty }?
1994     | element strike { sml_CT_BooleanProperty }?
1995     | element outline { sml_CT_BooleanProperty }?
1996     | element shadow { sml_CT_BooleanProperty }?
1997     | element condense { sml_CT_BooleanProperty }?
1998     | element extend { sml_CT_BooleanProperty }?
1999     | element color { sml_CT_Color }?
2000     | element sz { sml_CT_FontSize }?
2001     | element u { sml_CT_UnderlineProperty }?)

```

```

2002 | element vertAlign { sml_CT_VerticalAlignFontProperty }?
2003 | element scheme { sml_CT_FontScheme }?)+
2004 sml_CT_Rst =
2005     element t { s_ST_Xstring }?,
2006     element r { sml_CT_RElt }*,
2007     element rPh { sml_CT_PhoneticRun }*,
2008     element phoneticPr { sml_CT_PhoneticPr }?
2009 sml_CT_PhoneticPr =
2010     attribute fontId { sml_ST_FontId },
2011
2012     ## default value: fullwidthKatakana
2013     attribute type { sml_ST_PhoneticType }?,
2014
2015     ## default value: left
2016     attribute alignment { sml_ST_PhoneticAlignment }?
2017 sml_headers = element headers { sml_CT_RevisionHeaders }
2018 sml_revisions = element revisions { sml_CT_Revisions }
2019 sml_CT_RevisionHeaders =
2020     attribute guid { s_ST_Guid },
2021     attribute lastGuid { s_ST_Guid }?,
2022
2023     ## default value: true
2024     attribute shared { xsd:boolean }?,
2025
2026     ## default value: false
2027     attribute diskRevisions { xsd:boolean }?,
2028
2029     ## default value: true
2030     attribute history { xsd:boolean }?,
2031
2032     ## default value: true
2033     attribute trackRevisions { xsd:boolean }?,
2034
2035     ## default value: false
2036     attribute exclusive { xsd:boolean }?,
2037
2038     ## default value: 0
2039     attribute revisionId { xsd:unsignedInt }?,
2040
2041     ## default value: 1
2042     attribute version { xsd:int }?,
2043
2044     ## default value: true
2045     attribute keepChangeHistory { xsd:boolean }?,
2046
2047     ## default value: false
2048     attribute protected { xsd:boolean }?,
2049
2050     ## default value: 30
2051     attribute preserveHistory { xsd:unsignedInt }?,
2052     element header { sml_CT_RevisionHeader }+
2053 sml_CT_Revisions =
2054     (element rrc { sml_CT_RevisionRowColumn }*)

```

```

2055 | element rm { sml_CT_RevisionMove }*
2056 | element rcv { sml_CT_RevisionCustomView }*
2057 | element rsnm { sml_CT_RevisionSheetRename }*
2058 | element ris { sml_CT_RevisionInsertSheet }*
2059 | element rcc { sml_CT_RevisionCellChange }*
2060 | element rfmt { sml_CT_RevisionFormatting }*
2061 | element raf { sml_CT_RevisionAutoFormatting }*
2062 | element rdn { sml_CT_RevisionDefinedName }*
2063 | element rcmt { sml_CT_RevisionComment }*
2064 | element rqt { sml_CT_RevisionQueryTableField }*
2065 | element rcft { sml_CT_RevisionConflict }*)+
2066 sml_AG_RevData =
2067     attribute rId { xsd:unsignedInt },
2068
2069     ## default value: false
2070     attribute ua { xsd:boolean }?,
2071
2072     ## default value: false
2073     attribute ra { xsd:boolean }?
2074 sml_CT_RevisionHeader =
2075     attribute guid { s_ST_Guid },
2076     attribute dateTime { xsd:dateTime },
2077     attribute maxSheetId { xsd:unsignedInt },
2078     attribute userName { s_ST_Xstring },
2079     r_id,
2080     attribute minRId { xsd:unsignedInt }?,
2081     attribute maxRId { xsd:unsignedInt }?,
2082     element sheetIdMap { sml_CT_SheetIdMap },
2083     element reviewedList { sml_CT_ReviewedRevisions }?,
2084     element extLst { sml_CT_ExtensionList }?
2085 sml_CT_SheetIdMap =
2086     attribute count { xsd:unsignedInt }?,
2087     element sheetId { sml_CT_SheetId }+
2088 sml_CT_SheetId = attribute val { xsd:unsignedInt }
2089 sml_CT_ReviewedRevisions =
2090     attribute count { xsd:unsignedInt }?,
2091     element reviewed { sml_CT_Reviewed }+
2092 sml_CT_Reviewed = attribute rId { xsd:unsignedInt }
2093 sml_CT_UndoInfo =
2094     attribute index { xsd:unsignedInt },
2095     attribute exp { sml_ST_FormulaExpression },
2096
2097     ## default value: false
2098     attribute ref3D { xsd:boolean }?,
2099
2100     ## default value: false
2101     attribute array { xsd:boolean }?,
2102
2103     ## default value: false
2104     attribute v { xsd:boolean }?,
2105
2106     ## default value: false
2107     attribute nf { xsd:boolean }?,

```

```

2108
2109     ## default value: false
2110     attribute cs { xsd:boolean }?,
2111     attribute dr { sml_ST_RefA },
2112     attribute dn { s_ST_Xstring }?,
2113     attribute r { sml_ST_CellRef }?,
2114     attribute sId { xsd:unsignedInt }?
2115 sml_CT_RevisionRowColumn =
2116     sml_AG_RevData,
2117     attribute sId { xsd:unsignedInt },
2118
2119     ## default value: false
2120     attribute eol { xsd:boolean }?,
2121     attribute ref { sml_ST_Ref },
2122     attribute action { sml_ST_rwColActionType },
2123
2124     ## default value: false
2125     attribute edge { xsd:boolean }?,
2126     (element undo { sml_CT_UndoInfo }*
2127       | element rcc { sml_CT_RevisionCellChange }*
2128       | element rfmt { sml_CT_RevisionFormatting }*)*
2129 sml_CT_RevisionMove =
2130     sml_AG_RevData,
2131     attribute sheetId { xsd:unsignedInt },
2132     attribute source { sml_ST_Ref },
2133     attribute destination { sml_ST_Ref },
2134
2135     ## default value: 0
2136     attribute sourceSheetId { xsd:unsignedInt }?,
2137     (element undo { sml_CT_UndoInfo }*
2138       | element rcc { sml_CT_RevisionCellChange }*
2139       | element rfmt { sml_CT_RevisionFormatting }*)*
2140 sml_CT_RevisionCustomView =
2141     attribute guid { s_ST_Guid },
2142     attribute action { sml_ST_RevisionAction }
2143 sml_CT_RevisionSheetRename =
2144     sml_AG_RevData,
2145     attribute sheetId { xsd:unsignedInt },
2146     attribute oldName { s_ST_Xstring },
2147     attribute newName { s_ST_Xstring },
2148     element extLst { sml_CT_ExtensionList }?
2149 sml_CT_RevisionInsertSheet =
2150     sml_AG_RevData,
2151     attribute sheetId { xsd:unsignedInt },
2152     attribute name { s_ST_Xstring },
2153     attribute sheetPosition { xsd:unsignedInt }
2154 sml_CT_RevisionCellChange =
2155     sml_AG_RevData,
2156     attribute sId { xsd:unsignedInt },
2157
2158     ## default value: false
2159     attribute odxf { xsd:boolean }?,
2160

```

```

2161 ## default value: false
2162 attribute xfDxf { xsd:boolean }?,
2163
2164 ## default value: false
2165 attribute s { xsd:boolean }?,
2166
2167 ## default value: false
2168 attribute dxf { xsd:boolean }?,
2169 attribute numFmtId { sml_ST_NumFmtId }?,
2170
2171 ## default value: false
2172 attribute quotePrefix { xsd:boolean }?,
2173
2174 ## default value: false
2175 attribute oldQuotePrefix { xsd:boolean }?,
2176
2177 ## default value: false
2178 attribute ph { xsd:boolean }?,
2179
2180 ## default value: false
2181 attribute oldPh { xsd:boolean }?,
2182
2183 ## default value: false
2184 attribute endOfListFormulaUpdate { xsd:boolean }?,
2185 element oc { sml_CT_Cell }?,
2186 element nc { sml_CT_Cell },
2187 element odxf { sml_CT_Dxf }?,
2188 element ndxf { sml_CT_Dxf }?,
2189 element extLst { sml_CT_ExtensionList }?
2190 sml_CT_RevisionFormatting =
2191   attribute sheetId { xsd:unsignedInt },
2192
2193 ## default value: false
2194 attribute xfDxf { xsd:boolean }?,
2195
2196 ## default value: false
2197 attribute s { xsd:boolean }?,
2198 attribute sqref { sml_ST_Sqref },
2199 attribute start { xsd:unsignedInt }?,
2200 attribute length { xsd:unsignedInt }?,
2201 element dxf { sml_CT_Dxf }?,
2202 element extLst { sml_CT_ExtensionList }?
2203 sml_CT_RevisionAutoFormatting =
2204   attribute sheetId { xsd:unsignedInt },
2205   sml_AG_AutoFormat,
2206   attribute ref { sml_ST_Ref }
2207 sml_CT_RevisionComment =
2208   attribute sheetId { xsd:unsignedInt },
2209   attribute cell { sml_ST_CellRef },
2210   attribute guid { s_ST_Guid },
2211
2212 ## default value: add
2213 attribute action { sml_ST_RevisionAction }?,

```

```

2214
2215   ## default value: false
2216   attribute alwaysShow { xsd:boolean }?,
2217
2218   ## default value: false
2219   attribute old { xsd:boolean }?,
2220
2221   ## default value: false
2222   attribute hiddenRow { xsd:boolean }?,
2223
2224   ## default value: false
2225   attribute hiddenColumn { xsd:boolean }?,
2226   attribute author { s_ST_Xstring },
2227
2228   ## default value: 0
2229   attribute oldLength { xsd:unsignedInt }?,
2230
2231   ## default value: 0
2232   attribute newLength { xsd:unsignedInt }?
2233 sml_CT_RevisionDefinedName =
2234   sml_AG_RevData,
2235   attribute localSheetId { xsd:unsignedInt }?,
2236
2237   ## default value: false
2238   attribute customView { xsd:boolean }?,
2239   attribute name { s_ST_Xstring },
2240
2241   ## default value: false
2242   attribute function { xsd:boolean }?,
2243
2244   ## default value: false
2245   attribute oldFunction { xsd:boolean }?,
2246   attribute functionGroupId { xsd:unsignedByte }?,
2247   attribute oldFunctionGroupId { xsd:unsignedByte }?,
2248   attribute shortcutKey { xsd:unsignedByte }?,
2249   attribute oldShortcutKey { xsd:unsignedByte }?,
2250
2251   ## default value: false
2252   attribute hidden { xsd:boolean }?,
2253
2254   ## default value: false
2255   attribute oldHidden { xsd:boolean }?,
2256   attribute customMenu { s_ST_Xstring }?,
2257   attribute oldCustomMenu { s_ST_Xstring }?,
2258   attribute description { s_ST_Xstring }?,
2259   attribute oldDescription { s_ST_Xstring }?,
2260   attribute help { s_ST_Xstring }?,
2261   attribute oldHelp { s_ST_Xstring }?,
2262   attribute statusBar { s_ST_Xstring }?,
2263   attribute oldStatusBar { s_ST_Xstring }?,
2264   attribute comment { s_ST_Xstring }?,
2265   attribute oldComment { s_ST_Xstring }?,
2266   element formula { sml_ST_Formula }?,

```



```

2267     element oldFormula { sml_ST_Formula }?,
2268     element extLst { sml_CT_ExtensionList }?
2269 sml_CT_RevisionConflict =
2270     sml_AG_RevData,
2271     attribute sheetId { xsd:unsignedInt }?
2272 sml_CT_RevisionQueryTableField =
2273     attribute sheetId { xsd:unsignedInt },
2274     attribute ref { sml_ST_Ref },
2275     attribute fieldId { xsd:unsignedInt }
2276 sml_ST_rwColActionType =
2277     string "insertRow"
2278     | string "deleteRow"
2279     | string "insertCol"
2280     | string "deleteCol"
2281 sml_ST_RevisionAction = string "add" | string "delete"
2282 sml_ST_FormulaExpression =
2283     string "ref"
2284     | string "refError"
2285     | string "area"
2286     | string "areaError"
2287     | string "computedArea"
2288 sml_users = element users { sml_CT_Users }
2289 sml_CT_Users =
2290     attribute count { xsd:unsignedInt }?,
2291     element userInfo { sml_CT_SharedUser }*
2292 sml_CT_SharedUser =
2293     attribute guid { s_ST_Guid },
2294     attribute name { s_ST_Xstring },
2295     attribute id { xsd:int },
2296     attribute dateTime { xsd:dateTime },
2297     element extLst { sml_CT_ExtensionList }?
2298 sml_worksheet = element worksheet { sml_CT_Worksheet }
2299 sml_chartsheet = element chartsheet { sml_CT_Chartsheet }
2300 sml_dialogsheet = element dialogsheet { sml_CT_Dialogsheet }
2301 sml_CT_Macrosheet =
2302     element sheetPr { sml_CT_SheetPr }?,
2303     element dimension { sml_CT_SheetDimension }?,
2304     element sheetViews { sml_CT_SheetViews }?,
2305     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2306     element cols { sml_CT_Cols }*,
2307     element sheetData { sml_CT_SheetData },
2308     element sheetProtection { sml_CT_SheetProtection }?,
2309     element autoFilter { sml_CT_AutoFilter }?,
2310     element sortState { sml_CT_SortState }?,
2311     element dataConsolidate { sml_CT_DataConsolidate }?,
2312     element customSheetViews { sml_CT_CustomSheetViews }?,
2313     element phoneticPr { sml_CT_PhoneticPr }?,
2314     element conditionalFormatting { sml_CT_ConditionalFormatting }*,
2315     element printOptions { sml_CT_PrintOptions }?,
2316     element pageMargins { sml_CT_PageMargins }?,
2317     element pageSetup { sml_CT_PageSetup }?,
2318     element headerFooter { sml_CT_HeaderFooter }?,
2319     element rowBreaks { sml_CT_PageBreak }?,

```

```

2320     element colBreaks { sml_CT_PageBreak }?,
2321     element customProperties { sml_CT_CustomProperties }?,
2322     element drawing { sml_CT_Drawing }?,
2323     element legacyDrawing { sml_CT_LegacyDrawing }?,
2324     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2325     element drawingHF { sml_CT_DrawingHF }?,
2326     element picture { sml_CT_SheetBackgroundPicture }?,
2327     element oleObjects { sml_CT_OleObjects }?,
2328     element extLst { sml_CT_ExtensionList }?
2329 sml_CT_Dialogsheet =
2330     element sheetPr { sml_CT_SheetPr }?,
2331     element sheetViews { sml_CT_SheetViews }?,
2332     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2333     element sheetProtection { sml_CT_SheetProtection }?,
2334     element customSheetViews { sml_CT_CustomSheetViews }?,
2335     element printOptions { sml_CT_PrintOptions }?,
2336     element pageMargins { sml_CT_PageMargins }?,
2337     element pageSetup { sml_CT_PageSetup }?,
2338     element headerFooter { sml_CT_HeaderFooter }?,
2339     element drawing { sml_CT_Drawing }?,
2340     element legacyDrawing { sml_CT_LegacyDrawing }?,
2341     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2342     element drawingHF { sml_CT_DrawingHF }?,
2343     element oleObjects { sml_CT_OleObjects }?,
2344     element controls { sml_CT_Controls }?,
2345     element extLst { sml_CT_ExtensionList }?
2346 sml_CT_Worksheet =
2347     element sheetPr { sml_CT_SheetPr }?,
2348     element dimension { sml_CT_SheetDimension }?,
2349     element sheetViews { sml_CT_SheetViews }?,
2350     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2351     element cols { sml_CT_Cols }*,
2352     element sheetData { sml_CT_SheetData },
2353     element sheetCalcPr { sml_CT_SheetCalcPr }?,
2354     element sheetProtection { sml_CT_SheetProtection }?,
2355     element protectedRanges { sml_CT_ProtectedRanges }?,
2356     element scenarios { sml_CT_Scenarios }?,
2357     element autoFilter { sml_CT_AutoFilter }?,
2358     element sortState { sml_CT_SortState }?,
2359     element dataConsolidate { sml_CT_DataConsolidate }?,
2360     element customSheetViews { sml_CT_CustomSheetViews }?,
2361     element mergeCells { sml_CT_MergeCells }?,
2362     element phoneticPr { sml_CT_PhoneticPr }?,
2363     element conditionalFormatting { sml_CT_ConditionalFormatting }*,
2364     element dataValidations { sml_CT_DataValidations }?,
2365     element hyperlinks { sml_CT_Hyperlinks }?,
2366     element printOptions { sml_CT_PrintOptions }?,
2367     element pageMargins { sml_CT_PageMargins }?,
2368     element pageSetup { sml_CT_PageSetup }?,
2369     element headerFooter { sml_CT_HeaderFooter }?,
2370     element rowBreaks { sml_CT_PageBreak }?,
2371     element colBreaks { sml_CT_PageBreak }?,
2372     element customProperties { sml_CT_CustomProperties }?,

```

```

2373 element cellWatches { sml_CT_CellWatches }?,
2374 element ignoredErrors { sml_CT_IgnoredErrors }?,
2375 element smartTags { sml_CT_SmartTags }?,
2376 element drawing { sml_CT_Drawing }?,
2377 element legacyDrawing { sml_CT_LegacyDrawing }?,
2378 element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2379 element drawingHF { sml_CT_DrawingHF }?,
2380 element picture { sml_CT_SheetBackgroundPicture }?,
2381 element oleObjects { sml_CT_OleObjects }?,
2382 element controls { sml_CT_Controls }?,
2383 element webPublishItems { sml_CT_WebPublishItems }?,
2384 element tableParts { sml_CT_TableParts }?,
2385 element extLst { sml_CT_ExtensionList }?
2386 sml_CT_SheetData = element row { sml_CT_Row }*
2387 sml_CT_SheetCalcPr =
2388
2389     ## default value: false
2390     attribute fullCalcOnLoad { xsd:boolean }?
2391 sml_CT_SheetFormatPr =
2392
2393     ## default value: 8
2394     attribute baseColWidth { xsd:unsignedInt }?,
2395     attribute defaultColWidth { xsd:double }?,
2396     attribute defaultRowHeight { xsd:double },
2397
2398     ## default value: false
2399     attribute customHeight { xsd:boolean }?,
2400
2401     ## default value: false
2402     attribute zeroHeight { xsd:boolean }?,
2403
2404     ## default value: false
2405     attribute thickTop { xsd:boolean }?,
2406
2407     ## default value: false
2408     attribute thickBottom { xsd:boolean }?,
2409
2410     ## default value: 0
2411     attribute outlineLevelRow { xsd:unsignedByte }?,
2412
2413     ## default value: 0
2414     attribute outlineLevelCol { xsd:unsignedByte }?
2415 sml_CT_Cols = element col { sml_CT_Col }+
2416 sml_CT_Col =
2417     attribute min { xsd:unsignedInt },
2418     attribute max { xsd:unsignedInt },
2419     attribute width { xsd:double }?,
2420
2421     ## default value: 0
2422     attribute style { xsd:unsignedInt }?,
2423
2424     ## default value: false
2425     attribute hidden { xsd:boolean }?,

```

```

2426
2427   ## default value: false
2428   attribute bestFit { xsd:boolean }?,
2429
2430   ## default value: false
2431   attribute customWidth { xsd:boolean }?,
2432
2433   ## default value: false
2434   attribute phonetic { xsd:boolean }?,
2435
2436   ## default value: 0
2437   attribute outlineLevel { xsd:unsignedByte }?,
2438
2439   ## default value: false
2440   attribute collapsed { xsd:boolean }?
2441   sml_ST_CellSpan = xsd:string
2442   sml_ST_CellSpans = list { sml_ST_CellSpan* }
2443   sml_CT_Row =
2444     attribute r { xsd:unsignedInt }?,
2445     attribute spans { sml_ST_CellSpans }?,
2446
2447   ## default value: 0
2448   attribute s { xsd:unsignedInt }?,
2449
2450   ## default value: false
2451   attribute customFormat { xsd:boolean }?,
2452   attribute ht { xsd:double }?,
2453
2454   ## default value: false
2455   attribute hidden { xsd:boolean }?,
2456
2457   ## default value: false
2458   attribute customHeight { xsd:boolean }?,
2459
2460   ## default value: 0
2461   attribute outlineLevel { xsd:unsignedByte }?,
2462
2463   ## default value: false
2464   attribute collapsed { xsd:boolean }?,
2465
2466   ## default value: false
2467   attribute thickTop { xsd:boolean }?,
2468
2469   ## default value: false
2470   attribute thickBot { xsd:boolean }?,
2471
2472   ## default value: false
2473   attribute ph { xsd:boolean }?,
2474   element c { sml_CT_Cell }*,
2475   element extLst { sml_CT_ExtensionList }?
2476   sml_CT_Cell =
2477     attribute r { sml_ST_CellRef }?,
2478

```

```

2479 ## default value: 0
2480 attribute s { xsd:unsignedInt }?,
2481
2482 ## default value: n
2483 attribute t { sml_ST_CellType }?,
2484
2485 ## default value: 0
2486 attribute cm { xsd:unsignedInt }?,
2487
2488 ## default value: 0
2489 attribute vm { xsd:unsignedInt }?,
2490
2491 ## default value: false
2492 attribute ph { xsd:boolean }?,
2493 element f { sml_CT_CellFormula }?,
2494 element v { s_ST_Xstring }?,
2495 element is { sml_CT_Rst }?,
2496 element extLst { sml_CT_ExtensionList }?
2497 sml_ST_CellType =
2498     string "b"
2499     | string "n"
2500     | string "e"
2501     | string "s"
2502     | string "str"
2503     | string "inlineStr"
2504 sml_ST_CellFormulaType =
2505     string "normal"
2506     | string "array"
2507     | string "dataTable"
2508     | string "shared"
2509 sml_CT_SheetPr =
2510
2511 ## default value: false
2512 attribute syncHorizontal { xsd:boolean }?,
2513
2514 ## default value: false
2515 attribute syncVertical { xsd:boolean }?,
2516 attribute syncRef { sml_ST_Ref }?,
2517
2518 ## default value: false
2519 attribute transitionEvaluation { xsd:boolean }?,
2520
2521 ## default value: false
2522 attribute transitionEntry { xsd:boolean }?,
2523
2524 ## default value: true
2525 attribute published { xsd:boolean }?,
2526 attribute codeName { xsd:string }?,
2527
2528 ## default value: false
2529 attribute filterMode { xsd:boolean }?,
2530
2531 ## default value: true

```

```

2532     attribute enableFormatConditionsCalculation { xsd:boolean }?,
2533     element tabColor { sml_CT_Color }?,
2534     element outlinePr { sml_CT_OutlinePr }?,
2535     element pageSetUpPr { sml_CT_PageSetUpPr }?
2536 sml_CT_SheetDimension = attribute ref { sml_ST_Ref }
2537 sml_CT_SheetViews =
2538     element sheetView { sml_CT_SheetView }+,
2539     element extLst { sml_CT_ExtensionList }?
2540 sml_CT_SheetView =
2541
2542     ## default value: false
2543     attribute windowProtection { xsd:boolean }?,
2544
2545     ## default value: false
2546     attribute showFormulas { xsd:boolean }?,
2547
2548     ## default value: true
2549     attribute showGridLines { xsd:boolean }?,
2550
2551     ## default value: true
2552     attribute showRowColHeaders { xsd:boolean }?,
2553
2554     ## default value: true
2555     attribute showZeros { xsd:boolean }?,
2556
2557     ## default value: false
2558     attribute rightToLeft { xsd:boolean }?,
2559
2560     ## default value: false
2561     attribute tabSelected { xsd:boolean }?,
2562
2563     ## default value: true
2564     attribute showRuler { xsd:boolean }?,
2565
2566     ## default value: true
2567     attribute showOutlineSymbols { xsd:boolean }?,
2568
2569     ## default value: true
2570     attribute defaultGridColor { xsd:boolean }?,
2571
2572     ## default value: true
2573     attribute showWhiteSpace { xsd:boolean }?,
2574
2575     ## default value: normal
2576     attribute view { sml_ST_SheetViewType }?,
2577     attribute topLeftCell { sml_ST_CellRef }?,
2578
2579     ## default value: 64
2580     attribute colorId { xsd:unsignedInt }?,
2581
2582     ## default value: 100
2583     attribute zoomScale { xsd:unsignedInt }?,
2584

```

```

2585 ## default value: 0
2586 attribute zoomScaleNormal { xsd:unsignedInt }?,
2587
2588 ## default value: 0
2589 attribute zoomScaleSheetLayoutView { xsd:unsignedInt }?,
2590
2591 ## default value: 0
2592 attribute zoomScalePageLayoutView { xsd:unsignedInt }?,
2593 attribute workbookViewId { xsd:unsignedInt },
2594 element pane { sml_CT_Pane }?,
2595 element selection { sml_CT_Selection }*,
2596 element pivotSelection { sml_CT_PivotSelection }*,
2597 element extLst { sml_CT_ExtensionList }?
2598 sml_CT_Pane =
2599
2600 ## default value: 0
2601 attribute xSplit { xsd:double }?,
2602
2603 ## default value: 0
2604 attribute ySplit { xsd:double }?,
2605 attribute topLeftCell { sml_ST_CellRef }?,
2606
2607 ## default value: topLeft
2608 attribute activePane { sml_ST_Pane }?,
2609
2610 ## default value: split
2611 attribute state { sml_ST_PaneState }?
2612 sml_CT_PivotSelection =
2613
2614 ## default value: topLeft
2615 attribute pane { sml_ST_Pane }?,
2616
2617 ## default value: false
2618 attribute showHeader { xsd:boolean }?,
2619
2620 ## default value: false
2621 attribute label { xsd:boolean }?,
2622
2623 ## default value: false
2624 attribute data { xsd:boolean }?,
2625
2626 ## default value: false
2627 attribute extendable { xsd:boolean }?,
2628
2629 ## default value: 0
2630 attribute count { xsd:unsignedInt }?,
2631 attribute axis { sml_ST_Axis }?,
2632
2633 ## default value: 0
2634 attribute dimension { xsd:unsignedInt }?,
2635
2636 ## default value: 0
2637 attribute start { xsd:unsignedInt }?,

```

2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690

```

## default value: 0
attribute min { xsd:unsignedInt }?,

## default value: 0
attribute max { xsd:unsignedInt }?,

## default value: 0
attribute activeRow { xsd:unsignedInt }?,

## default value: 0
attribute activeCol { xsd:unsignedInt }?,

## default value: 0
attribute previousRow { xsd:unsignedInt }?,

## default value: 0
attribute previousCol { xsd:unsignedInt }?,

## default value: 0
attribute click { xsd:unsignedInt }?,
r_id?,
element pivotArea { sml_CT_PivotArea }
sml_CT_Selection =

## default value: topLeft
attribute pane { sml_ST_Pane }?,
attribute activeCell { sml_ST_CellRef }?,

## default value: 0
attribute activeCellId { xsd:unsignedInt }?,

## default value: A1
attribute sqref { sml_ST_Sqref }?
sml_ST_Pane =
  string "bottomRight"
  | string "topRight"
  | string "bottomLeft"
  | string "topLeft"
sml_CT_PageBreak =

## default value: 0
attribute count { xsd:unsignedInt }?,

## default value: 0
attribute manualBreakCount { xsd:unsignedInt }?,
element brk { sml_CT_Break }*
sml_CT_Break =

## default value: 0
attribute id { xsd:unsignedInt }?,

## default value: 0

```



```

2691 attribute min { xsd:unsignedInt }?,
2692
2693 ## default value: 0
2694 attribute max { xsd:unsignedInt }?,
2695
2696 ## default value: false
2697 attribute man { xsd:boolean }?,
2698
2699 ## default value: false
2700 attribute pt { xsd:boolean }?
2701 sml_ST_SheetViewType =
2702     string "normal" | string "pageBreakPreview" | string "pageLayout"
2703 sml_CT_OutlinePr =
2704
2705 ## default value: false
2706 attribute applyStyles { xsd:boolean }?,
2707
2708 ## default value: true
2709 attribute summaryBelow { xsd:boolean }?,
2710
2711 ## default value: true
2712 attribute summaryRight { xsd:boolean }?,
2713
2714 ## default value: true
2715 attribute showOutlineSymbols { xsd:boolean }?
2716 sml_CT_PageSetUpPr =
2717
2718 ## default value: true
2719 attribute autoPageBreaks { xsd:boolean }?,
2720
2721 ## default value: false
2722 attribute fitToPage { xsd:boolean }?
2723 sml_CT_DataConsolidate =
2724
2725 ## default value: sum
2726 attribute function { sml_ST_DataConsolidateFunction }?,
2727
2728 ## default value: false
2729 attribute startLabels { xsd:boolean }?,
2730
2731 ## default value: false
2732 attribute leftLabels { xsd:boolean }?,
2733
2734 ## default value: false
2735 attribute topLabels { xsd:boolean }?,
2736
2737 ## default value: false
2738 attribute link { xsd:boolean }?,
2739 element dataRefs { sml_CT_DataRefs }?
2740 sml_ST_DataConsolidateFunction =
2741     string "average"
2742     | string "count"
2743     | string "countNums"

```

```

2744 | string "max"
2745 | string "min"
2746 | string "product"
2747 | string "stdDev"
2748 | string "stdDevp"
2749 | string "sum"
2750 | string "var"
2751 | string "varp"
2752 sml_CT_DataRefs =
2753     attribute count { xsd:unsignedInt }?,
2754     element dataRef { sml_CT_DataRef }*
2755 sml_CT_DataRef =
2756     attribute ref { sml_ST_Ref }?,
2757     attribute name { s_ST_Xstring }?,
2758     attribute sheet { s_ST_Xstring }?,
2759     r_id?
2760 sml_CT_MergeCells =
2761     attribute count { xsd:unsignedInt }?,
2762     element mergeCell { sml_CT_MergeCell }+
2763 sml_CT_MergeCell = attribute ref { sml_ST_Ref }
2764 sml_CT_SmartTags = element cellSmartTags { sml_CT_CellSmartTags }+
2765 sml_CT_CellSmartTags =
2766     attribute r { sml_ST_CellRef },
2767     element cellSmartTag { sml_CT_CellSmartTag }+
2768 sml_CT_CellSmartTag =
2769     attribute type { xsd:unsignedInt },
2770
2771     ## default value: false
2772     attribute deleted { xsd:boolean }?,
2773
2774     ## default value: false
2775     attribute xmlBased { xsd:boolean }?,
2776     element cellSmartTagPr { sml_CT_CellSmartTagPr }*
2777 sml_CT_CellSmartTagPr =
2778     attribute key { s_ST_Xstring },
2779     attribute val { s_ST_Xstring }
2780 sml_CT_Drawing = r_id
2781 sml_CT_LegacyDrawing = r_id
2782 sml_CT_DrawingHF =
2783     r_id,
2784     attribute lho { xsd:unsignedInt }?,
2785     attribute lhe { xsd:unsignedInt }?,
2786     attribute lhf { xsd:unsignedInt }?,
2787     attribute cho { xsd:unsignedInt }?,
2788     attribute che { xsd:unsignedInt }?,
2789     attribute chf { xsd:unsignedInt }?,
2790     attribute rho { xsd:unsignedInt }?,
2791     attribute rhe { xsd:unsignedInt }?,
2792     attribute rhf { xsd:unsignedInt }?,
2793     attribute lfo { xsd:unsignedInt }?,
2794     attribute lfe { xsd:unsignedInt }?,
2795     attribute lff { xsd:unsignedInt }?,
2796     attribute cfo { xsd:unsignedInt }?,

```

```

2797 attribute cfe { xsd:unsignedInt }?,
2798 attribute cff { xsd:unsignedInt }?,
2799 attribute rfo { xsd:unsignedInt }?,
2800 attribute rfe { xsd:unsignedInt }?,
2801 attribute rff { xsd:unsignedInt }?
2802 sml_CT_CustomSheetViews =
2803   element customSheetView { sml_CT_CustomSheetView }+
2804 sml_CT_CustomSheetView =
2805   attribute guid { s_ST_Guid },
2806
2807   ## default value: 100
2808   attribute scale { xsd:unsignedInt }?,
2809
2810   ## default value: 64
2811   attribute colorId { xsd:unsignedInt }?,
2812
2813   ## default value: false
2814   attribute showPageBreaks { xsd:boolean }?,
2815
2816   ## default value: false
2817   attribute showFormulas { xsd:boolean }?,
2818
2819   ## default value: true
2820   attribute showGridLines { xsd:boolean }?,
2821
2822   ## default value: true
2823   attribute showRowCol { xsd:boolean }?,
2824
2825   ## default value: true
2826   attribute outlineSymbols { xsd:boolean }?,
2827
2828   ## default value: true
2829   attribute zeroValues { xsd:boolean }?,
2830
2831   ## default value: false
2832   attribute fitToPage { xsd:boolean }?,
2833
2834   ## default value: false
2835   attribute printArea { xsd:boolean }?,
2836
2837   ## default value: false
2838   attribute filter { xsd:boolean }?,
2839
2840   ## default value: false
2841   attribute showAutoFilter { xsd:boolean }?,
2842
2843   ## default value: false
2844   attribute hiddenRows { xsd:boolean }?,
2845
2846   ## default value: false
2847   attribute hiddenColumns { xsd:boolean }?,
2848
2849   ## default value: visible

```

```

2850 attribute state { sml_ST_SheetState }?,
2851
2852 ## default value: false
2853 attribute filterUnique { xsd:boolean }?,
2854
2855 ## default value: normal
2856 attribute view { sml_ST_SheetViewType }?,
2857
2858 ## default value: true
2859 attribute showRuler { xsd:boolean }?,
2860 attribute topLeftCell { sml_ST_CellRef }?,
2861 element pane { sml_CT_Pane }?,
2862 element selection { sml_CT_Selection }?,
2863 element rowBreaks { sml_CT_PageBreak }?,
2864 element colBreaks { sml_CT_PageBreak }?,
2865 element pageMargins { sml_CT_PageMargins }?,
2866 element printOptions { sml_CT_PrintOptions }?,
2867 element pageSetup { sml_CT_PageSetup }?,
2868 element headerFooter { sml_CT_HeaderFooter }?,
2869 element autoFilter { sml_CT_AutoFilter }?,
2870 element extLst { sml_CT_ExtensionList }?
2871 sml_CT_DataValidations =
2872
2873 ## default value: false
2874 attribute disablePrompts { xsd:boolean }?,
2875 attribute xWindow { xsd:unsignedInt }?,
2876 attribute yWindow { xsd:unsignedInt }?,
2877 attribute count { xsd:unsignedInt }?,
2878 element dataValidation { sml_CT_DataValidation }+
2879 sml_CT_DataValidation =
2880
2881 ## default value: none
2882 attribute type { sml_ST_DataValidationType }?,
2883
2884 ## default value: stop
2885 attribute errorStyle { sml_ST_DataValidationErrorStyle }?,
2886
2887 ## default value: noControl
2888 attribute imeMode { sml_ST_DataValidationImeMode }?,
2889
2890 ## default value: between
2891 attribute operator { sml_ST_DataValidationOperator }?,
2892
2893 ## default value: false
2894 attribute allowBlank { xsd:boolean }?,
2895
2896 ## default value: false
2897 attribute showDropDown { xsd:boolean }?,
2898
2899 ## default value: false
2900 attribute showInputMessage { xsd:boolean }?,
2901
2902 ## default value: false

```

```

2903     attribute showErrorMessage { xsd:boolean }?,
2904     attribute errorTitle { s_ST_Xstring }?,
2905     attribute error { s_ST_Xstring }?,
2906     attribute promptTitle { s_ST_Xstring }?,
2907     attribute prompt { s_ST_Xstring }?,
2908     attribute sqref { sml_ST_Sqref },
2909     element formula1 { sml_ST_Formula }?,
2910     element formula2 { sml_ST_Formula }?
2911 sml_ST_DataValidationType =
2912     string "none"
2913     | string "whole"
2914     | string "decimal"
2915     | string "list"
2916     | string "date"
2917     | string "time"
2918     | string "textLength"
2919     | string "custom"
2920 sml_ST_DataValidationOperator =
2921     string "between"
2922     | string "notBetween"
2923     | string "equal"
2924     | string "notEqual"
2925     | string "lessThan"
2926     | string "lessThanOrEqual"
2927     | string "greaterThan"
2928     | string "greaterThanOrEqual"
2929 sml_ST_DataValidationErrorStyle =
2930     string "stop" | string "warning" | string "information"
2931 sml_ST_DataValidationImeMode =
2932     string "noControl"
2933     | string "off"
2934     | string "on"
2935     | string "disabled"
2936     | string "hiragana"
2937     | string "fullKatakana"
2938     | string "halfKatakana"
2939     | string "fullAlpha"
2940     | string "halfAlpha"
2941     | string "fullHangul"
2942     | string "halfHangul"
2943 sml_ST_CfType =
2944     string "expression"
2945     | string "cellIs"
2946     | string "colorScale"
2947     | string "dataBar"
2948     | string "iconSet"
2949     | string "top10"
2950     | string "uniqueValues"
2951     | string "duplicateValues"
2952     | string "containsText"
2953     | string "notContainsText"
2954     | string "beginsWith"
2955     | string "endsWith"

```

```

2956 | string "containsBlanks"
2957 | string "notContainsBlanks"
2958 | string "containsErrors"
2959 | string "notContainsErrors"
2960 | string "timePeriod"
2961 | string "aboveAverage"
2962 sml_ST_TimePeriod =
2963     string "today"
2964     | string "yesterday"
2965     | string "tomorrow"
2966     | string "last7Days"
2967     | string "thisMonth"
2968     | string "lastMonth"
2969     | string "nextMonth"
2970     | string "thisWeek"
2971     | string "lastWeek"
2972     | string "nextWeek"
2973 sml_ST_ConditionalFormattingOperator =
2974     string "lessThan"
2975     | string "lessThanOrEqual"
2976     | string "equal"
2977     | string "notEqual"
2978     | string "greaterThanOrEqual"
2979     | string "greaterThan"
2980     | string "between"
2981     | string "notBetween"
2982     | string "containsText"
2983     | string "notContains"
2984     | string "beginsWith"
2985     | string "endsWith"
2986 sml_ST_CfvoType =
2987     string "num"
2988     | string "percent"
2989     | string "max"
2990     | string "min"
2991     | string "formula"
2992     | string "percentile"
2993 sml_CT_ConditionalFormatting =
2994
2995     ## default value: false
2996     attribute pivot { xsd:boolean }?,
2997     attribute sqref { sml_ST_Sqref }?,
2998     element cfRule { sml_CT_CfRule }+,
2999     element extLst { sml_CT_ExtensionList }?
3000 sml_CT_CfRule =
3001     attribute type { sml_ST_CfType }?,
3002     attribute dxId { sml_ST_DxfId }?,
3003     attribute priority { xsd:int },
3004
3005     ## default value: false
3006     attribute stopIfTrue { xsd:boolean }?,
3007
3008     ## default value: true

```

```

3009     attribute aboveAverage { xsd:boolean }?,
3010
3011     ## default value: false
3012     attribute percent { xsd:boolean }?,
3013
3014     ## default value: false
3015     attribute bottom { xsd:boolean }?,
3016     attribute operator { sml_ST_ConditionalFormattingOperator }?,
3017     attribute text { xsd:string }?,
3018     attribute timePeriod { sml_ST_TimePeriod }?,
3019     attribute rank { xsd:unsignedInt }?,
3020     attribute stdDev { xsd:int }?,
3021
3022     ## default value: false
3023     attribute equalAverage { xsd:boolean }?,
3024     element formula { sml_ST_Formula }*,
3025     element colorScale { sml_CT_ColorScale }?,
3026     element dataBar { sml_CT_DataBar }?,
3027     element iconSet { sml_CT_IconSet }?,
3028     element extLst { sml_CT_ExtensionList }?
3029 sml_CT_Hyperlinks = element hyperlink { sml_CT_Hyperlink }+
3030 sml_CT_Hyperlink =
3031     attribute ref { sml_ST_Ref },
3032     r_id?,
3033     attribute location { s_ST_Xstring }?,
3034     attribute tooltip { s_ST_Xstring }?,
3035     attribute display { s_ST_Xstring }?
3036 sml_CT_CellFormula =
3037     sml_ST_Formula,
3038
3039     ## default value: normal
3040     attribute t { sml_ST_CellFormulaType }?,
3041
3042     ## default value: false
3043     attribute aca { xsd:boolean }?,
3044     attribute ref { sml_ST_Ref }?,
3045
3046     ## default value: false
3047     attribute dt2D { xsd:boolean }?,
3048
3049     ## default value: false
3050     attribute dtr { xsd:boolean }?,
3051
3052     ## default value: false
3053     attribute del1 { xsd:boolean }?,
3054
3055     ## default value: false
3056     attribute del2 { xsd:boolean }?,
3057     attribute r1 { sml_ST_CellRef }?,
3058     attribute r2 { sml_ST_CellRef }?,
3059
3060     ## default value: false
3061     attribute ca { xsd:boolean }?,

```

```

3062     attribute si { xsd:unsignedInt }?,
3063
3064     ## default value: false
3065     attribute bx { xsd:boolean }?
3066 sml_CT_ColorScale =
3067     element cfvo { sml_CT_Cfvo }+,
3068     element color { sml_CT_Color }+
3069 sml_CT_DataBar =
3070
3071     ## default value: 10
3072     attribute minLength { xsd:unsignedInt }?,
3073
3074     ## default value: 90
3075     attribute maxLength { xsd:unsignedInt }?,
3076
3077     ## default value: true
3078     attribute showValue { xsd:boolean }?,
3079     element cfvo { sml_CT_Cfvo }+,
3080     element color { sml_CT_Color }
3081 sml_CT_IconSet =
3082
3083     ## default value: 3TrafficLights1
3084     attribute iconSet { sml_ST_IconSetType }?,
3085
3086     ## default value: true
3087     attribute showValue { xsd:boolean }?,
3088
3089     ## default value: true
3090     attribute percent { xsd:boolean }?,
3091
3092     ## default value: false
3093     attribute reverse { xsd:boolean }?,
3094     element cfvo { sml_CT_Cfvo }+
3095 sml_CT_Cfvo =
3096     attribute type { sml_ST_CfvoType },
3097     attribute val { s_ST_Xstring }?,
3098
3099     ## default value: true
3100     attribute gte { xsd:boolean }?,
3101     element extLst { sml_CT_ExtensionList }?
3102 sml_CT_PageMargins =
3103     attribute left { xsd:double },
3104     attribute right { xsd:double },
3105     attribute top { xsd:double },
3106     attribute bottom { xsd:double },
3107     attribute header { xsd:double },
3108     attribute footer { xsd:double }
3109 sml_CT_PrintOptions =
3110
3111     ## default value: false
3112     attribute horizontalCentered { xsd:boolean }?,
3113
3114     ## default value: false

```



```

3115 attribute verticalCentered { xsd:boolean }?,
3116
3117 ## default value: false
3118 attribute headings { xsd:boolean }?,
3119
3120 ## default value: false
3121 attribute gridLines { xsd:boolean }?,
3122
3123 ## default value: true
3124 attribute gridLinesSet { xsd:boolean }?
3125 sml_CT_PageSetup =
3126
3127 ## default value: 1
3128 attribute paperSize { xsd:unsignedInt }?,
3129 attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
3130 attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
3131
3132 ## default value: 100
3133 attribute scale { xsd:unsignedInt }?,
3134
3135 ## default value: 1
3136 attribute firstPageNumber { xsd:unsignedInt }?,
3137
3138 ## default value: 1
3139 attribute fitToWidth { xsd:unsignedInt }?,
3140
3141 ## default value: 1
3142 attribute fitToHeight { xsd:unsignedInt }?,
3143
3144 ## default value: downThenOver
3145 attribute pageOrder { sml_ST_PageOrder }?,
3146
3147 ## default value: default
3148 attribute orientation { sml_ST_Orientation }?,
3149
3150 ## default value: true
3151 attribute usePrinterDefaults { xsd:boolean }?,
3152
3153 ## default value: false
3154 attribute blackAndWhite { xsd:boolean }?,
3155
3156 ## default value: false
3157 attribute draft { xsd:boolean }?,
3158
3159 ## default value: none
3160 attribute cellComments { sml_ST_CellComments }?,
3161
3162 ## default value: false
3163 attribute useFirstPageNumber { xsd:boolean }?,
3164
3165 ## default value: displayed
3166 attribute errors { sml_ST_PrintError }?,
3167

```

```

3168 ## default value: 600
3169 attribute horizontalDpi { xsd:unsignedInt }?,
3170
3171 ## default value: 600
3172 attribute verticalDpi { xsd:unsignedInt }?,
3173
3174 ## default value: 1
3175 attribute copies { xsd:unsignedInt }?,
3176 r_id?
3177 sml_ST_PageOrder = string "downThenOver" | string "overThenDown"
3178 sml_ST_Orientation =
3179     string "default" | string "portrait" | string "landscape"
3180 sml_ST_CellComments =
3181     string "none" | string "asDisplayed" | string "atEnd"
3182 sml_CT_HeaderFooter =
3183
3184 ## default value: false
3185 attribute differentOddEven { xsd:boolean }?,
3186
3187 ## default value: false
3188 attribute differentFirst { xsd:boolean }?,
3189
3190 ## default value: true
3191 attribute scaleWithDoc { xsd:boolean }?,
3192
3193 ## default value: true
3194 attribute alignWithMargins { xsd:boolean }?,
3195 element oddHeader { s_ST_Xstring }?,
3196 element oddFooter { s_ST_Xstring }?,
3197 element evenHeader { s_ST_Xstring }?,
3198 element evenFooter { s_ST_Xstring }?,
3199 element firstHeader { s_ST_Xstring }?,
3200 element firstFooter { s_ST_Xstring }?
3201 sml_ST_PrintError =
3202     string "displayed" | string "blank" | string "dash" | string "NA"
3203 sml_CT_Scenarios =
3204     attribute current { xsd:unsignedInt }?,
3205     attribute show { xsd:unsignedInt }?,
3206     attribute sqref { sml_ST_Sqref }?,
3207     element scenario { sml_CT_Scenario }+
3208 sml_CT_SheetProtection =
3209     attribute password { sml_ST_UnsignedShortHex }?,
3210     attribute algorithmName { s_ST_Xstring }?,
3211     attribute hashValue { xsd:base64Binary }?,
3212     attribute saltValue { xsd:base64Binary }?,
3213     attribute spinCount { xsd:unsignedInt }?,
3214
3215 ## default value: false
3216 attribute sheet { xsd:boolean }?,
3217
3218 ## default value: false
3219 attribute objects { xsd:boolean }?,
3220

```

```

3221 ## default value: false
3222 attribute scenarios { xsd:boolean }?,
3223
3224 ## default value: true
3225 attribute formatCells { xsd:boolean }?,
3226
3227 ## default value: true
3228 attribute formatColumns { xsd:boolean }?,
3229
3230 ## default value: true
3231 attribute formatRows { xsd:boolean }?,
3232
3233 ## default value: true
3234 attribute insertColumns { xsd:boolean }?,
3235
3236 ## default value: true
3237 attribute insertRows { xsd:boolean }?,
3238
3239 ## default value: true
3240 attribute insertHyperlinks { xsd:boolean }?,
3241
3242 ## default value: true
3243 attribute deleteColumns { xsd:boolean }?,
3244
3245 ## default value: true
3246 attribute deleteRows { xsd:boolean }?,
3247
3248 ## default value: false
3249 attribute selectLockedCells { xsd:boolean }?,
3250
3251 ## default value: true
3252 attribute sort { xsd:boolean }?,
3253
3254 ## default value: true
3255 attribute autoFilter { xsd:boolean }?,
3256
3257 ## default value: true
3258 attribute pivotTables { xsd:boolean }?,
3259
3260 ## default value: false
3261 attribute selectUnlockedCells { xsd:boolean }?
3262 sml_CT_ProtectedRanges =
3263     element protectedRange { sml_CT_ProtectedRange }+
3264 sml_CT_ProtectedRange =
3265     attribute password { sml_ST_UnsignedShortHex }?,
3266     attribute sqref { sml_ST_Sqref },
3267     attribute name { s_ST_Xstring },
3268     attribute securityDescriptor { xsd:string }?,
3269     attribute algorithmName { s_ST_Xstring }?,
3270     attribute hashValue { xsd:base64Binary }?,
3271     attribute saltValue { xsd:base64Binary }?,
3272     attribute spinCount { xsd:unsignedInt }?,
3273     element securityDescriptor { xsd:string }*

```

```

3274 sml_CT_Scenario =
3275     attribute name { s_ST_Xstring },
3276
3277     ## default value: false
3278     attribute locked { xsd:boolean }?,
3279
3280     ## default value: false
3281     attribute hidden { xsd:boolean }?,
3282     attribute count { xsd:unsignedInt }?,
3283     attribute user { s_ST_Xstring }?,
3284     attribute comment { s_ST_Xstring }?,
3285     element inputCells { sml_CT_InputCells }+
3286 sml_CT_InputCells =
3287     attribute r { sml_ST_CellRef },
3288
3289     ## default value: false
3290     attribute deleted { xsd:boolean }?,
3291
3292     ## default value: false
3293     attribute undone { xsd:boolean }?,
3294     attribute val { s_ST_Xstring },
3295     attribute numFmtId { sml_ST_NumFmtId }?
3296 sml_CT_CellWatches = element cellWatch { sml_CT_CellWatch }+
3297 sml_CT_CellWatch = attribute r { sml_ST_CellRef }
3298 sml_CT_Chartsheet =
3299     element sheetPr { sml_CT_ChartsheetPr }?,
3300     element sheetViews { sml_CT_ChartsheetViews },
3301     element sheetProtection { sml_CT_ChartsheetProtection }?,
3302     element customSheetViews { sml_CT_CustomChartsheetViews }?,
3303     element pageMargins { sml_CT_PageMargins }?,
3304     element pageSetup { sml_CT-CsPageSetup }?,
3305     element headerFooter { sml_CT_HeaderFooter }?,
3306     element drawing { sml_CT_Drawing },
3307     element legacyDrawing { sml_CT_LegacyDrawing }?,
3308     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
3309     element drawingHF { sml_CT_DrawingHF }?,
3310     element picture { sml_CT_SheetBackgroundPicture }?,
3311     element webPublishItems { sml_CT_WebPublishItems }?,
3312     element extLst { sml_CT_ExtensionList }?
3313 sml_CT_ChartsheetPr =
3314
3315     ## default value: true
3316     attribute published { xsd:boolean }?,
3317     attribute codeName { xsd:string }?,
3318     element tabColor { sml_CT_Color }?
3319 sml_CT_ChartsheetViews =
3320     element sheetView { sml_CT_ChartsheetView }+,
3321     element extLst { sml_CT_ExtensionList }?
3322 sml_CT_ChartsheetView =
3323
3324     ## default value: false
3325     attribute tabSelected { xsd:boolean }?,
3326

```

```

3327 ## default value: 100
3328 attribute zoomScale { xsd:unsignedInt }?,
3329 attribute workbookViewId { xsd:unsignedInt },
3330
3331 ## default value: false
3332 attribute zoomToFit { xsd:boolean }?,
3333 element extLst { sml_CT_ExtensionList }?
3334 sml_CT_ChartsheetProtection =
3335   attribute password { sml_ST_UnsignedShortHex }?,
3336   attribute algorithmName { s_ST_Xstring }?,
3337   attribute hashValue { xsd:base64Binary }?,
3338   attribute saltValue { xsd:base64Binary }?,
3339   attribute spinCount { xsd:unsignedInt }?,
3340
3341 ## default value: false
3342 attribute content { xsd:boolean }?,
3343
3344 ## default value: false
3345 attribute objects { xsd:boolean }?
3346 sml_CT_CsPageSetup =
3347
3348 ## default value: 1
3349 attribute paperSize { xsd:unsignedInt }?,
3350 attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
3351 attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
3352
3353 ## default value: 1
3354 attribute firstPageNumber { xsd:unsignedInt }?,
3355
3356 ## default value: default
3357 attribute orientation { sml_ST_Orientation }?,
3358
3359 ## default value: true
3360 attribute usePrinterDefaults { xsd:boolean }?,
3361
3362 ## default value: false
3363 attribute blackAndWhite { xsd:boolean }?,
3364
3365 ## default value: false
3366 attribute draft { xsd:boolean }?,
3367
3368 ## default value: false
3369 attribute useFirstPageNumber { xsd:boolean }?,
3370
3371 ## default value: 600
3372 attribute horizontalDpi { xsd:unsignedInt }?,
3373
3374 ## default value: 600
3375 attribute verticalDpi { xsd:unsignedInt }?,
3376
3377 ## default value: 1
3378 attribute copies { xsd:unsignedInt }?,
3379 r_id?

```

```

3380 sml_CT_CustomChartsheetViews =
3381     element customSheetView { sml_CT_CustomChartsheetView }*
3382 sml_CT_CustomChartsheetView =
3383     attribute guid { s_ST_Guid },
3384
3385     ## default value: 100
3386     attribute scale { xsd:unsignedInt }?,
3387
3388     ## default value: visible
3389     attribute state { sml_ST_SheetState }?,
3390
3391     ## default value: false
3392     attribute zoomToFit { xsd:boolean }?,
3393     element pageMargins { sml_CT_PageMargins }?,
3394     element pageSetup { sml_CT_CsPageSetup }?,
3395     element headerFooter { sml_CT_HeaderFooter }?
3396 sml_CT_CustomProperties = element customPr { sml_CT_CustomProperty }+
3397 sml_CT_CustomProperty =
3398     attribute name { s_ST_Xstring },
3399     r_id
3400 sml_CT_OleObjects = element oleObject { sml_CT_OleObject }+
3401 sml_CT_OleObject =
3402     attribute progId { xsd:string }?,
3403
3404     ## default value: DVASPECT_CONTENT
3405     attribute dvAspect { sml_ST_DvAspect }?,
3406     attribute link { s_ST_Xstring }?,
3407     attribute oleUpdate { sml_ST_OleUpdate }?,
3408
3409     ## default value: false
3410     attribute autoLoad { xsd:boolean }?,
3411     attribute shapeId { xsd:unsignedInt },
3412     r_id?,
3413     element objectPr { sml_CT_ObjectPr }?
3414 sml_CT_ObjectPr =
3415
3416     ## default value: true
3417     attribute locked { xsd:boolean }?,
3418
3419     ## default value: true
3420     attribute defaultSize { xsd:boolean }?,
3421
3422     ## default value: true
3423     attribute print { xsd:boolean }?,
3424
3425     ## default value: false
3426     attribute disabled { xsd:boolean }?,
3427
3428     ## default value: false
3429     attribute uiObject { xsd:boolean }?,
3430
3431     ## default value: true
3432     attribute autoFill { xsd:boolean }?,

```

```

3433
3434   ## default value: true
3435   attribute autoLine { xsd:boolean }?,
3436
3437   ## default value: true
3438   attribute autoPict { xsd:boolean }?,
3439   attribute macro { sml_ST_Formula }?,
3440   attribute altText { s_ST_Xstring }?,
3441
3442   ## default value: false
3443   attribute dde { xsd:boolean }?,
3444   r_id?,
3445   element anchor { sml_CT_ObjectAnchor }
3446 sml_ST_DvAspect = string "DVASPECT_CONTENT" | string "DVASPECT_ICON"
3447 sml_ST_OleUpdate = string "OLEUPDATE_ALWAYS" | string "OLEUPDATE_ONCALL"
3448 sml_CT_WebPublishItems =
3449   attribute count { xsd:unsignedInt }?,
3450   element webPublishItem { sml_CT_WebPublishItem }+
3451 sml_CT_WebPublishItem =
3452   attribute id { xsd:unsignedInt },
3453   attribute divId { s_ST_Xstring },
3454   attribute sourceType { sml_ST_WebSourceType },
3455   attribute sourceRef { sml_ST_Ref }?,
3456   attribute sourceObject { s_ST_Xstring }?,
3457   attribute destinationFile { s_ST_Xstring },
3458   attribute title { s_ST_Xstring }?,
3459
3460   ## default value: false
3461   attribute autoRepublish { xsd:boolean }?
3462 sml_CT_Controls = element control { sml_CT_Control }+
3463 sml_CT_Control =
3464   attribute shapeId { xsd:unsignedInt },
3465   r_id,
3466   attribute name { xsd:string }?,
3467   element controlPr { sml_CT_ControlPr }?
3468 sml_CT_ControlPr =
3469
3470   ## default value: true
3471   attribute locked { xsd:boolean }?,
3472
3473   ## default value: true
3474   attribute defaultSize { xsd:boolean }?,
3475
3476   ## default value: true
3477   attribute print { xsd:boolean }?,
3478
3479   ## default value: false
3480   attribute disabled { xsd:boolean }?,
3481
3482   ## default value: false
3483   attribute recalcAlways { xsd:boolean }?,
3484
3485   ## default value: false

```

```

3486     attribute uiObject { xsd:boolean }?,
3487
3488     ## default value: true
3489     attribute autoFill { xsd:boolean }?,
3490
3491     ## default value: true
3492     attribute autoLine { xsd:boolean }?,
3493
3494     ## default value: true
3495     attribute autoPict { xsd:boolean }?,
3496     attribute macro { sml_ST_Formula }?,
3497     attribute altText { s_ST_Xstring }?,
3498     attribute linkedCell { sml_ST_Formula }?,
3499     attribute listFillRange { sml_ST_Formula }?,
3500
3501     ## default value: pict
3502     attribute cf { s_ST_Xstring }?,
3503     r_id?,
3504     element anchor { sml_CT_ObjectAnchor }
3505 sml_ST_WebSourceType =
3506     string "sheet"
3507     | string "printArea"
3508     | string "autoFilter"
3509     | string "range"
3510     | string "chart"
3511     | string "pivotTable"
3512     | string "query"
3513     | string "label"
3514 sml_CT_IgnoredErrors =
3515     element ignoredError { sml_CT_IgnoredError }+,
3516     element extLst { sml_CT_ExtensionList }?
3517 sml_CT_IgnoredError =
3518     attribute sqref { sml_ST_Sqref },
3519
3520     ## default value: false
3521     attribute evalError { xsd:boolean }?,
3522
3523     ## default value: false
3524     attribute twoDigitTextYear { xsd:boolean }?,
3525
3526     ## default value: false
3527     attribute numberStoredAsText { xsd:boolean }?,
3528
3529     ## default value: false
3530     attribute formula { xsd:boolean }?,
3531
3532     ## default value: false
3533     attribute formulaRange { xsd:boolean }?,
3534
3535     ## default value: false
3536     attribute unlockedFormula { xsd:boolean }?,
3537
3538     ## default value: false

```



```

3539     attribute emptyCellReference { xsd:boolean }?,
3540
3541     ## default value: false
3542     attribute listDataValidation { xsd:boolean }?,
3543
3544     ## default value: false
3545     attribute calculatedColumn { xsd:boolean }?
3546 sml_ST_PaneState =
3547     string "split" | string "frozen" | string "frozenSplit"
3548 sml_CT_TableParts =
3549     attribute count { xsd:unsignedInt }?,
3550     element tablePart { sml_CT_TablePart }*
3551 sml_CT_TablePart = r_id
3552 sml_metadata = element metadata { sml_CT_Metadata }
3553 sml_CT_Metadata =
3554     element metadataTypes { sml_CT_MetadataTypes }?,
3555     element metadataStrings { sml_CT_MetadataStrings }?,
3556     element mdxMetadata { sml_CT_MdxMetadata }?,
3557     element futureMetadata { sml_CT_FutureMetadata }*,
3558     element cellMetadata { sml_CT_MetadataBlocks }?,
3559     element valueMetadata { sml_CT_MetadataBlocks }?,
3560     element extLst { sml_CT_ExtensionList }?
3561 sml_CT_MetadataTypes =
3562
3563     ## default value: 0
3564     attribute count { xsd:unsignedInt }?,
3565     element metadataType { sml_CT_MetadataType }+
3566 sml_CT_MetadataType =
3567     attribute name { s_ST_Xstring },
3568     attribute minSupportedVersion { xsd:unsignedInt },
3569
3570     ## default value: false
3571     attribute ghostRow { xsd:boolean }?,
3572
3573     ## default value: false
3574     attribute ghostCol { xsd:boolean }?,
3575
3576     ## default value: false
3577     attribute edit { xsd:boolean }?,
3578
3579     ## default value: false
3580     attribute delete { xsd:boolean }?,
3581
3582     ## default value: false
3583     attribute copy { xsd:boolean }?,
3584
3585     ## default value: false
3586     attribute pasteAll { xsd:boolean }?,
3587
3588     ## default value: false
3589     attribute pasteFormulas { xsd:boolean }?,
3590
3591     ## default value: false

```

```
3592 attribute pasteValues { xsd:boolean }?,
3593
3594 ## default value: false
3595 attribute pasteFormats { xsd:boolean }?,
3596
3597 ## default value: false
3598 attribute pasteComments { xsd:boolean }?,
3599
3600 ## default value: false
3601 attribute pasteDataValidation { xsd:boolean }?,
3602
3603 ## default value: false
3604 attribute pasteBorders { xsd:boolean }?,
3605
3606 ## default value: false
3607 attribute pasteColWidths { xsd:boolean }?,
3608
3609 ## default value: false
3610 attribute pasteNumberFormats { xsd:boolean }?,
3611
3612 ## default value: false
3613 attribute merge { xsd:boolean }?,
3614
3615 ## default value: false
3616 attribute splitFirst { xsd:boolean }?,
3617
3618 ## default value: false
3619 attribute splitAll { xsd:boolean }?,
3620
3621 ## default value: false
3622 attribute rowColShift { xsd:boolean }?,
3623
3624 ## default value: false
3625 attribute clearAll { xsd:boolean }?,
3626
3627 ## default value: false
3628 attribute clearFormats { xsd:boolean }?,
3629
3630 ## default value: false
3631 attribute clearContents { xsd:boolean }?,
3632
3633 ## default value: false
3634 attribute clearComments { xsd:boolean }?,
3635
3636 ## default value: false
3637 attribute assign { xsd:boolean }?,
3638
3639 ## default value: false
3640 attribute coerce { xsd:boolean }?,
3641
3642 ## default value: false
3643 attribute adjust { xsd:boolean }?,
3644
```

```

3645  ## default value: false
3646  attribute cellMeta { xsd:boolean }?
3647  sml_CT_MetadataBlocks =
3648
3649  ## default value: 0
3650  attribute count { xsd:unsignedInt }?,
3651  element bk { sml_CT_MetadataBlock }+
3652  sml_CT_MetadataBlock = element rc { sml_CT_MetadataRecord }+
3653  sml_CT_MetadataRecord =
3654  attribute t { xsd:unsignedInt },
3655  attribute v { xsd:unsignedInt }
3656  sml_CT_FutureMetadata =
3657  attribute name { s_ST_Xstring },
3658
3659  ## default value: 0
3660  attribute count { xsd:unsignedInt }?,
3661  element bk { sml_CT_FutureMetadataBlock }*,
3662  element extLst { sml_CT_ExtensionList }?
3663  sml_CT_FutureMetadataBlock = element extLst { sml_CT_ExtensionList }?
3664  sml_CT_MdxMetadata =
3665
3666  ## default value: 0
3667  attribute count { xsd:unsignedInt }?,
3668  element mdx { sml_CT_Mdx }+
3669  sml_CT_Mdx =
3670  attribute n { xsd:unsignedInt },
3671  attribute f { sml_ST_MdxFunctionType },
3672  (element t { sml_CT_MdxTuple }
3673  | element ms { sml_CT_MdxSet }
3674  | element p { sml_CT_MdxMemeberProp }
3675  | element k { sml_CT_MdxKPI })
3676  sml_ST_MdxFunctionType =
3677  string "m"
3678  | string "v"
3679  | string "s"
3680  | string "c"
3681  | string "r"
3682  | string "p"
3683  | string "k"
3684  sml_CT_MdxTuple =
3685
3686  ## default value: 0
3687  attribute c { xsd:unsignedInt }?,
3688  attribute ct { s_ST_Xstring }?,
3689  attribute si { xsd:unsignedInt }?,
3690  attribute fi { xsd:unsignedInt }?,
3691  attribute bc { sml_ST_UnsignedIntHex }?,
3692  attribute fc { sml_ST_UnsignedIntHex }?,
3693
3694  ## default value: false
3695  attribute i { xsd:boolean }?,
3696
3697  ## default value: false

```

```

3698     attribute u { xsd:boolean }?,
3699
3700     ## default value: false
3701     attribute st { xsd:boolean }?,
3702
3703     ## default value: false
3704     attribute b { xsd:boolean }?,
3705     element n { sml_CT_MetadataStringIndex }*
3706 sml_CT_MdxSet =
3707     attribute ns { xsd:unsignedInt },
3708
3709     ## default value: 0
3710     attribute c { xsd:unsignedInt }?,
3711
3712     ## default value: u
3713     attribute o { sml_ST_MdxSetOrder }?,
3714     element n { sml_CT_MetadataStringIndex }*
3715 sml_ST_MdxSetOrder =
3716     string "u"
3717     | string "a"
3718     | string "d"
3719     | string "aa"
3720     | string "ad"
3721     | string "na"
3722     | string "nd"
3723 sml_CT_MdxMemeberProp =
3724     attribute n { xsd:unsignedInt },
3725     attribute np { xsd:unsignedInt }
3726 sml_CT_MdxKPI =
3727     attribute n { xsd:unsignedInt },
3728     attribute np { xsd:unsignedInt },
3729     attribute p { sml_ST_MdxKPIProperty }
3730 sml_ST_MdxKPIProperty =
3731     string "v"
3732     | string "g"
3733     | string "s"
3734     | string "t"
3735     | string "w"
3736     | string "m"
3737 sml_CT_MetadataStringIndex =
3738     attribute x { xsd:unsignedInt },
3739
3740     ## default value: false
3741     attribute s { xsd:boolean }?
3742 sml_CT_MetadataStrings =
3743
3744     ## default value: 0
3745     attribute count { xsd:unsignedInt }?,
3746     element s { sml_CT_XStringElement }+
3747 sml_singleXmlCells = element singleXmlCells { sml_CT_SingleXmlCells }
3748 sml_CT_SingleXmlCells = element singleXmlCell { sml_CT_SingleXmlCell }+
3749 sml_CT_SingleXmlCell =
3750     attribute id { xsd:unsignedInt },

```

```

3751     attribute r { sml_ST_CellRef },
3752     attribute connectionId { xsd:unsignedInt },
3753     element xmlCellPr { sml_CT_XmlCellPr },
3754     element extLst { sml_CT_ExtensionList }?
3755 sml_CT_XmlCellPr =
3756     attribute id { xsd:unsignedInt },
3757     attribute uniqueName { s_ST_Xstring }?,
3758     element xmlPr { sml_CT_XmlPr },
3759     element extLst { sml_CT_ExtensionList }?
3760 sml_CT_XmlPr =
3761     attribute mapId { xsd:unsignedInt },
3762     attribute xpath { s_ST_Xstring },
3763     attribute xmlDataType { sml_ST_XmlDataType },
3764     element extLst { sml_CT_ExtensionList }?
3765 sml_styleSheet = element styleSheet { sml_CT_Stylesheet }
3766 sml_CT_Stylesheet =
3767     element numFmts { sml_CT_NumFmts }?,
3768     element fonts { sml_CT_Fonts }?,
3769     element fills { sml_CT_Fills }?,
3770     element borders { sml_CT_Borders }?,
3771     element cellStyleXfs { sml_CT_CellStyleXfs }?,
3772     element cellXfs { sml_CT_CellXfs }?,
3773     element cellStyles { sml_CT_CellStyles }?,
3774     element dxfs { sml_CT_Dxfs }?,
3775     element tableStyles { sml_CT_TableStyles }?,
3776     element colors { sml_CT_Colors }?,
3777     element extLst { sml_CT_ExtensionList }?
3778 sml_CT_CellAlignment =
3779     attribute horizontal { sml_ST_HorizontalAlignment }?,
3780     attribute vertical { sml_ST_VerticalAlignment }?,
3781     attribute textRotation { xsd:unsignedInt }?,
3782     attribute wrapText { xsd:boolean }?,
3783     attribute indent { xsd:unsignedInt }?,
3784     attribute relativeIndent { xsd:int }?,
3785     attribute justifyLastLine { xsd:boolean }?,
3786     attribute shrinkToFit { xsd:boolean }?,
3787     attribute readingOrder { xsd:unsignedInt }?
3788 sml_ST_BorderStyle =
3789     string "none"
3790     | string "thin"
3791     | string "medium"
3792     | string "dashed"
3793     | string "dotted"
3794     | string "thick"
3795     | string "double"
3796     | string "hair"
3797     | string "mediumDashed"
3798     | string "dashDot"
3799     | string "mediumDashDot"
3800     | string "dashDotDot"
3801     | string "mediumDashDotDot"
3802     | string "slantDashDot"
3803 sml_CT_Borders =

```

```

3804     attribute count { xsd:unsignedInt }?,
3805     element border { sml_CT_Border }*
3806 sml_CT_Border =
3807     attribute diagonalUp { xsd:boolean }?,
3808     attribute diagonalDown { xsd:boolean }?,
3809
3810     ## default value: true
3811     attribute outline { xsd:boolean }?,
3812     element start { sml_CT_BorderPr }?,
3813     element end { sml_CT_BorderPr }?,
3814     element left { sml_CT_BorderPr }?,
3815     element right { sml_CT_BorderPr }?,
3816     element top { sml_CT_BorderPr }?,
3817     element bottom { sml_CT_BorderPr }?,
3818     element diagonal { sml_CT_BorderPr }?,
3819     element vertical { sml_CT_BorderPr }?,
3820     element horizontal { sml_CT_BorderPr }?
3821 sml_CT_BorderPr =
3822
3823     ## default value: none
3824     attribute style { sml_ST_BorderStyle }?,
3825     element color { sml_CT_Color }?
3826 sml_CT_CellProtection =
3827     attribute locked { xsd:boolean }?,
3828     attribute hidden { xsd:boolean }?
3829 sml_CT_Fonts =
3830     attribute count { xsd:unsignedInt }?,
3831     element font { sml_CT_Font }*
3832 sml_CT_Fills =
3833     attribute count { xsd:unsignedInt }?,
3834     element fill { sml_CT_Fill }*
3835 sml_CT_Fill =
3836     element patternFill { sml_CT_PatternFill }?
3837     | element gradientFill { sml_CT_GradientFill }?
3838 sml_CT_PatternFill =
3839     attribute patternType { sml_ST_PatternType }?,
3840     element fgColor { sml_CT_Color }?,
3841     element bgColor { sml_CT_Color }?
3842 sml_CT_Color =
3843     attribute auto { xsd:boolean }?,
3844     attribute indexed { xsd:unsignedInt }?,
3845     attribute rgb { sml_ST_UnsignedIntHex }?,
3846     attribute theme { xsd:unsignedInt }?,
3847
3848     ## default value: 0.0
3849     attribute tint { xsd:double }?
3850 sml_ST_PatternType =
3851     string "none"
3852     | string "solid"
3853     | string "mediumGray"
3854     | string "darkGray"
3855     | string "lightGray"
3856     | string "darkHorizontal"

```

```

3857 | string "darkVertical"
3858 | string "darkDown"
3859 | string "darkUp"
3860 | string "darkGrid"
3861 | string "darkTrellis"
3862 | string "lightHorizontal"
3863 | string "lightVertical"
3864 | string "lightDown"
3865 | string "lightUp"
3866 | string "lightGrid"
3867 | string "lightTrellis"
3868 | string "gray125"
3869 | string "gray0625"
3870 sml_CT_GradientFill =
3871
3872     ## default value: linear
3873     attribute type { sml_ST_GradientType }?,
3874
3875     ## default value: 0
3876     attribute degree { xsd:double }?,
3877
3878     ## default value: 0
3879     attribute left { xsd:double }?,
3880
3881     ## default value: 0
3882     attribute right { xsd:double }?,
3883
3884     ## default value: 0
3885     attribute top { xsd:double }?,
3886
3887     ## default value: 0
3888     attribute bottom { xsd:double }?,
3889     element stop { sml_CT_GradientStop }*
3890 sml_CT_GradientStop =
3891     attribute position { xsd:double },
3892     element color { sml_CT_Color }
3893 sml_ST_GradientType = string "linear" | string "path"
3894 sml_ST_HorizontalAlignment =
3895     string "general"
3896     | string "left"
3897     | string "center"
3898     | string "right"
3899     | string "fill"
3900     | string "justify"
3901     | string "centerContinuous"
3902     | string "distributed"
3903 sml_ST_VerticalAlignment =
3904     string "top"
3905     | string "center"
3906     | string "bottom"
3907     | string "justify"
3908     | string "distributed"
3909 sml_CT_NumFmts =

```

```

3910     attribute count { xsd:unsignedInt }?,
3911     element numFmt { sml_CT_NumFmt }*
3912 sml_CT_NumFmt =
3913     attribute numFmtId { sml_ST_NumFmtId },
3914     attribute formatCode { s_ST_Xstring }
3915 sml_CT_CellStyleXfs =
3916     attribute count { xsd:unsignedInt }?,
3917     element xf { sml_CT_Xf }+
3918 sml_CT_CellXfs =
3919     attribute count { xsd:unsignedInt }?,
3920     element xf { sml_CT_Xf }+
3921 sml_CT_Xf =
3922     attribute numFmtId { sml_ST_NumFmtId }?,
3923     attribute fontId { sml_ST_FontId }?,
3924     attribute fillId { sml_ST_FillId }?,
3925     attribute borderId { sml_ST_BorderId }?,
3926     attribute xfId { sml_ST_CellStyleXfId }?,
3927
3928     ## default value: false
3929     attribute quotePrefix { xsd:boolean }?,
3930
3931     ## default value: false
3932     attribute pivotButton { xsd:boolean }?,
3933     attribute applyNumberFormat { xsd:boolean }?,
3934     attribute applyFont { xsd:boolean }?,
3935     attribute applyFill { xsd:boolean }?,
3936     attribute applyBorder { xsd:boolean }?,
3937     attribute applyAlignment { xsd:boolean }?,
3938     attribute applyProtection { xsd:boolean }?,
3939     element alignment { sml_CT_CellAlignment }?,
3940     element protection { sml_CT_CellProtection }?,
3941     element extLst { sml_CT_ExtensionList }?
3942 sml_CT_CellStyles =
3943     attribute count { xsd:unsignedInt }?,
3944     element cellStyle { sml_CT_CellStyle }+
3945 sml_CT_CellStyle =
3946     attribute name { s_ST_Xstring }?,
3947     attribute xfId { sml_ST_CellStyleXfId },
3948     attribute builtinId { xsd:unsignedInt }?,
3949     attribute iLevel { xsd:unsignedInt }?,
3950     attribute hidden { xsd:boolean }?,
3951     attribute customBuiltin { xsd:boolean }?,
3952     element extLst { sml_CT_ExtensionList }?
3953 sml_CT_Dxfs =
3954     attribute count { xsd:unsignedInt }?,
3955     element dxf { sml_CT_Dxf }*
3956 sml_CT_Dxf =
3957     element font { sml_CT_Font }?,
3958     element numFmt { sml_CT_NumFmt }?,
3959     element fill { sml_CT_Fill }?,
3960     element alignment { sml_CT_CellAlignment }?,
3961     element border { sml_CT_Border }?,
3962     element protection { sml_CT_CellProtection }?,

```



```

3963     element extlst { sml_CT_ExtensionList }?
3964 sml_ST_NumFmtId = xsd:unsignedInt
3965 sml_ST_FontId = xsd:unsignedInt
3966 sml_ST_FillId = xsd:unsignedInt
3967 sml_ST_BorderId = xsd:unsignedInt
3968 sml_ST_CellStyleXfId = xsd:unsignedInt
3969 sml_ST_DxfId = xsd:unsignedInt
3970 sml_CT_Colors =
3971     element indexedColors { sml_CT_IndexedColors }?,
3972     element mruColors { sml_CT_MRUColors }?
3973 sml_CT_IndexedColors = element rgbColor { sml_CT_RgbColor }+
3974 sml_CT_MRUColors = element color { sml_CT_Color }+
3975 sml_CT_RgbColor = attribute rgb { sml_ST_UnsignedIntHex }?
3976 sml_CT_TableStyles =
3977     attribute count { xsd:unsignedInt }?,
3978     attribute defaultTableStyle { xsd:string }?,
3979     attribute defaultPivotStyle { xsd:string }?,
3980     element tableStyle { sml_CT_TableStyle }*
3981 sml_CT_TableStyle =
3982     attribute name { xsd:string },
3983
3984     ## default value: true
3985     attribute pivot { xsd:boolean }?,
3986
3987     ## default value: true
3988     attribute table { xsd:boolean }?,
3989     attribute count { xsd:unsignedInt }?,
3990     element tableStyleElement { sml_CT_TableStyleElement }*
3991 sml_CT_TableStyleElement =
3992     attribute type { sml_ST_TableStyleType },
3993
3994     ## default value: 1
3995     attribute size { xsd:unsignedInt }?,
3996     attribute dxfId { sml_ST_DxfId }?
3997 sml_ST_TableStyleType =
3998     string "wholeTable"
3999     | string "headerRow"
4000     | string "totalRow"
4001     | string "firstColumn"
4002     | string "lastColumn"
4003     | string "firstRowStripe"
4004     | string "secondRowStripe"
4005     | string "firstColumnStripe"
4006     | string "secondColumnStripe"
4007     | string "firstHeaderCell"
4008     | string "lastHeaderCell"
4009     | string "firstTotalCell"
4010     | string "lastTotalCell"
4011     | string "firstSubtotalColumn"
4012     | string "secondSubtotalColumn"
4013     | string "thirdSubtotalColumn"
4014     | string "firstSubtotalRow"
4015     | string "secondSubtotalRow"

```

```

4016 | string "thirdSubtotalRow"
4017 | string "blankRow"
4018 | string "firstColumnSubheading"
4019 | string "secondColumnSubheading"
4020 | string "thirdColumnSubheading"
4021 | string "firstRowSubheading"
4022 | string "secondRowSubheading"
4023 | string "thirdRowSubheading"
4024 | string "pageFieldLabels"
4025 | string "pageFieldValues"
4026 sml_CT_BooleanProperty =
4027
4028     ## default value: true
4029     attribute val { xsd:boolean }?
4030 sml_CT_FontSize = attribute val { xsd:double }
4031 sml_CT_IntProperty = attribute val { xsd:int }
4032 sml_CT_FontName = attribute val { s_ST_Xstring }
4033 sml_CT_VerticalAlignFontProperty =
4034     attribute val { s_ST_VerticalAlignRun }
4035 sml_CT_FontScheme = attribute val { sml_ST_FontScheme }
4036 sml_ST_FontScheme = string "none" | string "major" | string "minor"
4037 sml_CT_UnderlineProperty =
4038
4039     ## default value: single
4040     attribute val { sml_ST_UnderlineValues }?
4041 sml_ST_UnderlineValues =
4042     string "single"
4043     | string "double"
4044     | string "singleAccounting"
4045     | string "doubleAccounting"
4046     | string "none"
4047 sml_CT_Font =
4048     (element name { sml_CT_FontName }?
4049     | element charset { sml_CT_IntProperty }?
4050     | element family { sml_CT_FontFamily}?
4051     | element b { sml_CT_BooleanProperty }?
4052     | element i { sml_CT_BooleanProperty }?
4053     | element strike { sml_CT_BooleanProperty }?
4054     | element outline { sml_CT_BooleanProperty }?
4055     | element shadow { sml_CT_BooleanProperty }?
4056     | element condense { sml_CT_BooleanProperty }?
4057     | element extend { sml_CT_BooleanProperty }?
4058     | element color { sml_CT_Color }?
4059     | element sz { sml_CT_FontSize }?
4060     | element u { sml_CT_UnderlineProperty }?
4061     | element vertAlign { sml_CT_VerticalAlignFontProperty }?
4062     | element scheme { sml_CT_FontScheme }?)+
4063 sml_CT_FontFamily = attribute val { sml_ST_FontFamily }
4064 sml_ST_FontFamily = xsd:integer { minInclusive = "0" maxInclusive = "14" }
4065 sml_AG_AutoFormat =
4066     attribute autoFormatId { xsd:unsignedInt }?,
4067     attribute applyNumberFormats { xsd:boolean }?,
4068     attribute applyBorderFormats { xsd:boolean }?,

```

```

4069     attribute applyFontFormats { xsd:boolean }?,
4070     attribute applyPatternFormats { xsd:boolean }?,
4071     attribute applyAlignmentFormats { xsd:boolean }?,
4072     attribute applyWidthHeightFormats { xsd:boolean }?
4073 sml_externallink = element externalLink { sml_CT_ExternalLink }
4074 sml_CT_ExternalLink =
4075     (element externalBook { sml_CT_ExternalBook }?
4076     | element ddeLink { sml_CT_DdeLink }?
4077     | element oleLink { sml_CT_OleLink }?),
4078     element extLst { sml_CT_ExtensionList }?
4079 sml_CT_ExternalBook =
4080     r_id,
4081     element sheetNames { sml_CT_ExternalSheetNames }?,
4082     element definedNames { sml_CT_ExternalDefinedNames }?,
4083     element sheetDataSet { sml_CT_ExternalSheetDataSet }?
4084 sml_CT_ExternalSheetNames =
4085     element sheetName { sml_CT_ExternalSheetName }+
4086 sml_CT_ExternalSheetName = attribute val { s_ST_Xstring }?
4087 sml_CT_ExternalDefinedNames =
4088     element definedName { sml_CT_ExternalDefinedName }*
4089 sml_CT_ExternalDefinedName =
4090     attribute name { s_ST_Xstring },
4091     attribute refersTo { s_ST_Xstring }?,
4092     attribute sheetId { xsd:unsignedInt }?
4093 sml_CT_ExternalSheetDataSet =
4094     element sheetData { sml_CT_ExternalSheetData }+
4095 sml_CT_ExternalSheetData =
4096     attribute sheetId { xsd:unsignedInt },
4097
4098     ## default value: false
4099     attribute refreshError { xsd:boolean }?,
4100     element row { sml_CT_ExternalRow }*
4101 sml_CT_ExternalRow =
4102     attribute r { xsd:unsignedInt },
4103     element cell { sml_CT_ExternalCell }*
4104 sml_CT_ExternalCell =
4105     attribute r { sml_ST_CellRef }?,
4106
4107     ## default value: n
4108     attribute t { sml_ST_CellType }?,
4109
4110     ## default value: 0
4111     attribute vm { xsd:unsignedInt }?,
4112     element v { s_ST_Xstring }?
4113 sml_CT_DdeLink =
4114     attribute ddeService { s_ST_Xstring },
4115     attribute ddeTopic { s_ST_Xstring },
4116     element ddeItems { sml_CT_DdeItems }?
4117 sml_CT_DdeItems = element ddeItem { sml_CT_DdeItem }*
4118 sml_CT_DdeItem =
4119
4120     ## default value: 0
4121     attribute name { s_ST_Xstring }?,

```

```

4122
4123   ## default value: false
4124   attribute ole { xsd:boolean }?,
4125
4126   ## default value: false
4127   attribute advise { xsd:boolean }?,
4128
4129   ## default value: false
4130   attribute preferPic { xsd:boolean }?,
4131   element values { sml_CT_DdeValues }?
4132 sml_CT_DdeValues =
4133
4134   ## default value: 1
4135   attribute rows { xsd:unsignedInt }?,
4136
4137   ## default value: 1
4138   attribute cols { xsd:unsignedInt }?,
4139   element value { sml_CT_DdeValue }+
4140 sml_CT_DdeValue =
4141
4142   ## default value: n
4143   attribute t { sml_ST_DdeValueType }?,
4144   element val { s_ST_Xstring }
4145 sml_ST_DdeValueType =
4146   string "nil" | string "b" | string "n" | string "e" | string "str"
4147 sml_CT_OleLink =
4148   r_id,
4149   attribute progId { s_ST_Xstring },
4150   element oleItems { sml_CT_OleItems }?
4151 sml_CT_OleItems = element oleItem { sml_CT_OleItem }*
4152 sml_CT_OleItem =
4153   attribute name { s_ST_Xstring },
4154
4155   ## default value: false
4156   attribute icon { xsd:boolean }?,
4157
4158   ## default value: false
4159   attribute advise { xsd:boolean }?,
4160
4161   ## default value: false
4162   attribute preferPic { xsd:boolean }?
4163 sml_table = element table { sml_CT_Table }
4164 sml_CT_Table =
4165   attribute id { xsd:unsignedInt },
4166   attribute name { s_ST_Xstring }?,
4167   attribute displayName { s_ST_Xstring },
4168   attribute comment { s_ST_Xstring }?,
4169   attribute ref { sml_ST_Ref },
4170
4171   ## default value: worksheet
4172   attribute tableType { sml_ST_TableType }?,
4173
4174   ## default value: 1

```

```

4175 attribute headerRowCount { xsd:unsignedInt }?,
4176
4177 ## default value: false
4178 attribute insertRow { xsd:boolean }?,
4179
4180 ## default value: false
4181 attribute insertRowShift { xsd:boolean }?,
4182
4183 ## default value: 0
4184 attribute totalsRowCount { xsd:unsignedInt }?,
4185
4186 ## default value: true
4187 attribute totalsRowShown { xsd:boolean }?,
4188
4189 ## default value: false
4190 attribute published { xsd:boolean }?,
4191 attribute headerRowDxfId { sml_ST_DxfId }?,
4192 attribute dataDxfId { sml_ST_DxfId }?,
4193 attribute totalsRowDxfId { sml_ST_DxfId }?,
4194 attribute headerRowBorderDxfId { sml_ST_DxfId }?,
4195 attribute tableBorderDxfId { sml_ST_DxfId }?,
4196 attribute totalsRowBorderDxfId { sml_ST_DxfId }?,
4197 attribute headerRowCellStyle { s_ST_Xstring }?,
4198 attribute dataCellStyle { s_ST_Xstring }?,
4199 attribute totalsRowCellStyle { s_ST_Xstring }?,
4200 attribute connectionId { xsd:unsignedInt }?,
4201 element autoFilter { sml_CT_AutoFilter }?,
4202 element sortState { sml_CT_SortState }?,
4203 element tableColumns { sml_CT_TableColumns },
4204 element tableStyleInfo { sml_CT_TableStyleInfo }?,
4205 element extLst { sml_CT_ExtensionList }?
4206 sml_ST_TableType =
4207     string "worksheet" | string "xml" | string "queryTable"
4208 sml_CT_TableStyleInfo =
4209     attribute name { s_ST_Xstring }?,
4210     attribute showFirstColumn { xsd:boolean }?,
4211     attribute showLastColumn { xsd:boolean }?,
4212     attribute showRowStripes { xsd:boolean }?,
4213     attribute showColumnStripes { xsd:boolean }?
4214 sml_CT_TableColumns =
4215     attribute count { xsd:unsignedInt }?,
4216     element tableColumn { sml_CT_TableColumn }+
4217 sml_CT_TableColumn =
4218     attribute id { xsd:unsignedInt },
4219     attribute uniqueName { s_ST_Xstring }?,
4220     attribute name { s_ST_Xstring },
4221
4222 ## default value: none
4223 attribute totalsRowFunction { sml_ST_TotalsRowFunction }?,
4224 attribute totalsRowLabel { s_ST_Xstring }?,
4225 attribute queryTableFieldId { xsd:unsignedInt }?,
4226 attribute headerRowDxfId { sml_ST_DxfId }?,
4227 attribute dataDxfId { sml_ST_DxfId }?,

```

```

4228     attribute totalsRowDxfId { sml_ST_DxfId }?,
4229     attribute headerRowCellStyle { s_ST_Xstring }?,
4230     attribute dataCellStyle { s_ST_Xstring }?,
4231     attribute totalsRowCellStyle { s_ST_Xstring }?,
4232     element calculatedColumnFormula { sml_CT_TableFormula }?,
4233     element totalsRowFormula { sml_CT_TableFormula }?,
4234     element xmlColumnPr { sml_CT_XmlColumnPr }?,
4235     element extLst { sml_CT_ExtensionList }?
4236 sml_CT_TableFormula =
4237     sml_ST_Formula,
4238
4239     ## default value: false
4240     attribute array { xsd:boolean }?
4241 sml_ST_TotalsRowFunction =
4242     string "none"
4243     | string "sum"
4244     | string "min"
4245     | string "max"
4246     | string "average"
4247     | string "count"
4248     | string "countNums"
4249     | string "stdDev"
4250     | string "var"
4251     | string "custom"
4252 sml_CT_XmlColumnPr =
4253     attribute mapId { xsd:unsignedInt },
4254     attribute xpath { s_ST_Xstring },
4255
4256     ## default value: false
4257     attribute denormalized { xsd:boolean }?,
4258     attribute xmlDataType { sml_ST_XmlDataType },
4259     element extLst { sml_CT_ExtensionList }?
4260 sml_ST_XmlDataType = xsd:string
4261 sml_volTypes = element volTypes { sml_CT_VolTypes }
4262 sml_CT_VolTypes =
4263     element volType { sml_CT_VolType }+,
4264     element extLst { sml_CT_ExtensionList }?
4265 sml_CT_VolType =
4266     attribute type { sml_ST_VolDepType },
4267     element main { sml_CT_VolMain }+
4268 sml_CT_VolMain =
4269     attribute first { s_ST_Xstring },
4270     element tp { sml_CT_VolTopic }+
4271 sml_CT_VolTopic =
4272
4273     ## default value: n
4274     attribute t { sml_ST_VolValueType }?,
4275     element v { s_ST_Xstring },
4276     element stp { s_ST_Xstring }*,
4277     element tr { sml_CT_VolTopicRef }+
4278 sml_CT_VolTopicRef =
4279     attribute r { sml_ST_CellRef },
4280     attribute s { xsd:unsignedInt }

```

```

4281 sml_ST_VolDepType = string "realTimeData" | string "olapFunctions"
4282 sml_ST_VolValueType = string "b" | string "n" | string "e" | string "s"
4283 sml_workbook = element workbook { sml_CT_Workbook }
4284 sml_CT_Workbook =
4285     attribute conformance { s_ST_ConformanceClass }?,
4286     element fileVersion { sml_CT_FileVersion }?,
4287     element fileSharing { sml_CT_FileSharing }?,
4288     element workbookPr { sml_CT_WorkbookPr }?,
4289     element workbookProtection { sml_CT_WorkbookProtection }?,
4290     element bookViews { sml_CT_BookViews }?,
4291     element sheets { sml_CT_Sheets },
4292     element functionGroups { sml_CT_FunctionGroups }?,
4293     element externalReferences { sml_CT_ExternalReferences }?,
4294     element definedNames { sml_CT_DefinedNames }?,
4295     element calcPr { sml_CT_CalcPr }?,
4296     element oleSize { sml_CT_OleSize }?,
4297     element customWorkbookViews { sml_CT_CustomWorkbookViews }?,
4298     element pivotCaches { sml_CT_PivotCaches }?,
4299     element smartTagPr { sml_CT_SmartTagPr }?,
4300     element smartTagTypes { sml_CT_SmartTagTypes }?,
4301     element webPublishing { sml_CT_WebPublishing }?,
4302     element fileRecoveryPr { sml_CT_FileRecoveryPr }*,
4303     element webPublishObjects { sml_CT_WebPublishObjects }?,
4304     element extLst { sml_CT_ExtensionList }?
4305 sml_CT_FileVersion =
4306     attribute appName { xsd:string }?,
4307     attribute lastEdited { xsd:string }?,
4308     attribute lowestEdited { xsd:string }?,
4309     attribute rupBuild { xsd:string }?,
4310     attribute codeName { s_ST_Guid }?
4311 sml_CT_BookViews = element workbookView { sml_CT_BookView }+
4312 sml_CT_BookView =
4313
4314     ## default value: visible
4315     attribute visibility { sml_ST_Visibility }?,
4316
4317     ## default value: false
4318     attribute minimized { xsd:boolean }?,
4319
4320     ## default value: true
4321     attribute showHorizontalScroll { xsd:boolean }?,
4322
4323     ## default value: true
4324     attribute showVerticalScroll { xsd:boolean }?,
4325
4326     ## default value: true
4327     attribute showSheetTabs { xsd:boolean }?,
4328     attribute xWindow { xsd:int }?,
4329     attribute yWindow { xsd:int }?,
4330     attribute windowWidth { xsd:unsignedInt }?,
4331     attribute windowHeight { xsd:unsignedInt }?,
4332
4333     ## default value: 600

```

```

4334     attribute tabRatio { xsd:unsignedInt }?,
4335
4336     ## default value: 0
4337     attribute firstSheet { xsd:unsignedInt }?,
4338
4339     ## default value: 0
4340     attribute activeTab { xsd:unsignedInt }?,
4341
4342     ## default value: true
4343     attribute autoFilterDateGrouping { xsd:boolean }?,
4344     element extLst { sml_CT_ExtensionList }?
4345 sml_ST_Visibility =
4346     string "visible" | string "hidden" | string "veryHidden"
4347 sml_CT_CustomWorkbookViews =
4348     element customWorkbookView { sml_CT_CustomWorkbookView }+
4349 sml_CT_CustomWorkbookView =
4350     attribute name { s_ST_Xstring },
4351     attribute guid { s_ST_Guid },
4352
4353     ## default value: false
4354     attribute autoUpdate { xsd:boolean }?,
4355     attribute mergeInterval { xsd:unsignedInt }?,
4356
4357     ## default value: false
4358     attribute changesSavedWin { xsd:boolean }?,
4359
4360     ## default value: false
4361     attribute onlySync { xsd:boolean }?,
4362
4363     ## default value: false
4364     attribute personalView { xsd:boolean }?,
4365
4366     ## default value: true
4367     attribute includePrintSettings { xsd:boolean }?,
4368
4369     ## default value: true
4370     attribute includeHiddenRowCol { xsd:boolean }?,
4371
4372     ## default value: false
4373     attribute maximized { xsd:boolean }?,
4374
4375     ## default value: false
4376     attribute minimized { xsd:boolean }?,
4377
4378     ## default value: true
4379     attribute showHorizontalScroll { xsd:boolean }?,
4380
4381     ## default value: true
4382     attribute showVerticalScroll { xsd:boolean }?,
4383
4384     ## default value: true
4385     attribute showSheetTabs { xsd:boolean }?,
4386

```



```

4387 ## default value: 0
4388 attribute xWindow { xsd:int }?,
4389
4390 ## default value: 0
4391 attribute yWindow { xsd:int }?,
4392 attribute windowWidth { xsd:unsignedInt },
4393 attribute windowHeight { xsd:unsignedInt },
4394
4395 ## default value: 600
4396 attribute tabRatio { xsd:unsignedInt }?,
4397 attribute activeSheetId { xsd:unsignedInt },
4398
4399 ## default value: true
4400 attribute showFormulaBar { xsd:boolean }?,
4401
4402 ## default value: true
4403 attribute showStatusbar { xsd:boolean }?,
4404
4405 ## default value: commIndicator
4406 attribute showComments { sml_ST_Comments }?,
4407
4408 ## default value: all
4409 attribute showObjects { sml_ST_Objects }?,
4410 element extLst { sml_CT_ExtensionList }?
4411 sml_ST_Comments =
4412     string "commNone"
4413     | string "commIndicator"
4414     | string "commIndAndComment"
4415 sml_ST_Objects = string "all" | string "placeholders" | string "none"
4416 sml_CT_Sheets = element sheet { sml_CT_Sheet }+
4417 sml_CT_Sheet =
4418     attribute name { s_ST_Xstring },
4419     attribute sheetId { xsd:unsignedInt },
4420
4421 ## default value: visible
4422 attribute state { sml_ST_SheetState }?,
4423 r_id
4424 sml_ST_SheetState =
4425     string "visible" | string "hidden" | string "veryHidden"
4426 sml_CT_WorkbookPr =
4427
4428 ## default value: false
4429 attribute date1904 { xsd:boolean }?,
4430
4431 ## default value: all
4432 attribute showObjects { sml_ST_Objects }?,
4433
4434 ## default value: true
4435 attribute showBorderUnselectedTables { xsd:boolean }?,
4436
4437 ## default value: false
4438 attribute filterPrivacy { xsd:boolean }?,
4439

```

```

4440 ## default value: false
4441 attribute promptedSolutions { xsd:boolean }?,
4442
4443 ## default value: true
4444 attribute showInkAnnotation { xsd:boolean }?,
4445
4446 ## default value: false
4447 attribute backupFile { xsd:boolean }?,
4448
4449 ## default value: true
4450 attribute saveExternalLinkValues { xsd:boolean }?,
4451
4452 ## default value: userSet
4453 attribute updateLinks { sml_ST_UpdateLinks }?,
4454 attribute codeName { xsd:string }?,
4455
4456 ## default value: false
4457 attribute hidePivotFieldList { xsd:boolean }?,
4458
4459 ## default value: false
4460 attribute showPivotChartFilter { xsd:boolean }?,
4461
4462 ## default value: false
4463 attribute allowRefreshQuery { xsd:boolean }?,
4464
4465 ## default value: false
4466 attribute publishItems { xsd:boolean }?,
4467
4468 ## default value: false
4469 attribute checkCompatibility { xsd:boolean }?,
4470
4471 ## default value: true
4472 attribute autoCompressPictures { xsd:boolean }?,
4473
4474 ## default value: false
4475 attribute refreshAllConnections { xsd:boolean }?,
4476 attribute defaultThemeVersion { xsd:unsignedInt }?
4477 sml_ST_UpdateLinks = string "userSet" | string "never" | string "always"
4478 sml_CT_SmartTagPr =
4479
4480 ## default value: false
4481 attribute embed { xsd:boolean }?,
4482
4483 ## default value: all
4484 attribute show { sml_ST_SmartTagShow }?
4485 sml_ST_SmartTagShow =
4486   string "all" | string "none" | string "noIndicator"
4487 sml_CT_SmartTagTypes = element smartTagType { sml_CT_SmartTagType }*
4488 sml_CT_SmartTagType =
4489   attribute namespaceUri { s_ST_Xstring }?,
4490   attribute name { s_ST_Xstring }?,
4491   attribute url { s_ST_Xstring }?
4492 sml_CT_FileRecoveryPr =

```

```

4493
4494   ## default value: true
4495   attribute autoRecover { xsd:boolean }?,
4496
4497   ## default value: false
4498   attribute crashSave { xsd:boolean }?,
4499
4500   ## default value: false
4501   attribute dataExtractLoad { xsd:boolean }?,
4502
4503   ## default value: false
4504   attribute repairLoad { xsd:boolean }?
4505 sml_CT_CalcPr =
4506   attribute calcId { xsd:unsignedInt }?,
4507
4508   ## default value: auto
4509   attribute calcMode { sml_ST_CalcMode }?,
4510
4511   ## default value: false
4512   attribute fullCalcOnLoad { xsd:boolean }?,
4513
4514   ## default value: A1
4515   attribute refMode { sml_ST_RefMode }?,
4516
4517   ## default value: false
4518   attribute iterate { xsd:boolean }?,
4519
4520   ## default value: 100
4521   attribute iterateCount { xsd:unsignedInt }?,
4522
4523   ## default value: 0.001
4524   attribute iterateDelta { xsd:double }?,
4525
4526   ## default value: true
4527   attribute fullPrecision { xsd:boolean }?,
4528
4529   ## default value: true
4530   attribute calcCompleted { xsd:boolean }?,
4531
4532   ## default value: true
4533   attribute calcOnSave { xsd:boolean }?,
4534
4535   ## default value: true
4536   attribute concurrentCalc { xsd:boolean }?,
4537   attribute concurrentManualCount { xsd:unsignedInt }?,
4538   attribute forceFullCalc { xsd:boolean }?
4539 sml_ST_CalcMode = string "manual" | string "auto" | string "autoNoTable"
4540 sml_ST_RefMode = string "A1" | string "R1C1"
4541 sml_CT_DefinedNames = element definedName { sml_CT_DefinedName }*
4542 sml_CT_DefinedName =
4543   sml_ST_Formula,
4544   attribute name { s_ST_Xstring },
4545   attribute comment { s_ST_Xstring }?,

```

```

4546     attribute customMenu { s_ST_Xstring }?,
4547     attribute description { s_ST_Xstring }?,
4548     attribute help { s_ST_Xstring }?,
4549     attribute statusBar { s_ST_Xstring }?,
4550     attribute localSheetId { xsd:unsignedInt }?,
4551
4552     ## default value: false
4553     attribute hidden { xsd:boolean }?,
4554
4555     ## default value: false
4556     attribute function { xsd:boolean }?,
4557
4558     ## default value: false
4559     attribute vbProcedure { xsd:boolean }?,
4560
4561     ## default value: false
4562     attribute xlm { xsd:boolean }?,
4563     attribute functionGroupId { xsd:unsignedInt }?,
4564     attribute shortcutKey { s_ST_Xstring }?,
4565
4566     ## default value: false
4567     attribute publishToServer { xsd:boolean }?,
4568
4569     ## default value: false
4570     attribute workbookParameter { xsd:boolean }?
4571 sml_CT_ExternalReferences =
4572     element externalReference { sml_CT_ExternalReference }+
4573 sml_CT_ExternalReference = r_id
4574 sml_CT_SheetBackgroundPicture = r_id
4575 sml_CT_PivotCaches = element pivotCache { sml_CT_PivotCache }+
4576 sml_CT_PivotCache =
4577     attribute cacheId { xsd:unsignedInt },
4578     r_id
4579 sml_CT_FileSharing =
4580
4581     ## default value: false
4582     attribute readOnlyRecommended { xsd:boolean }?,
4583     attribute userName { s_ST_Xstring }?,
4584     attribute reservationPassword { sml_ST_UnsignedShortHex }?,
4585     attribute algorithmName { s_ST_Xstring }?,
4586     attribute hashValue { xsd:base64Binary }?,
4587     attribute saltValue { xsd:base64Binary }?,
4588     attribute spinCount { xsd:unsignedInt }?
4589 sml_CT_OleSize = attribute ref { sml_ST_Ref }
4590 sml_CT_WorkbookProtection =
4591     attribute workbookPassword { sml_ST_UnsignedShortHex }?,
4592     attribute workbookPasswordCharacterSet { xsd:string }?,
4593     attribute revisionsPassword { sml_ST_UnsignedShortHex }?,
4594     attribute revisionsPasswordCharacterSet { xsd:string }?,
4595
4596     ## default value: false
4597     attribute lockStructure { xsd:boolean }?,
4598

```

```

4599 ## default value: false
4600 attribute lockWindows { xsd:boolean }?,
4601
4602 ## default value: false
4603 attribute lockRevision { xsd:boolean }?,
4604 attribute revisionsAlgorithmName { s_ST_Xstring }?,
4605 attribute revisionsHashValue { xsd:base64Binary }?,
4606 attribute revisionsSaltValue { xsd:base64Binary }?,
4607 attribute revisionsSpinCount { xsd:unsignedInt }?,
4608 attribute workbookAlgorithmName { s_ST_Xstring }?,
4609 attribute workbookHashValue { xsd:base64Binary }?,
4610 attribute workbookSaltValue { xsd:base64Binary }?,
4611 attribute workbookSpinCount { xsd:unsignedInt }?
4612 sml_CT_WebPublishing =
4613
4614 ## default value: true
4615 attribute css { xsd:boolean }?,
4616
4617 ## default value: true
4618 attribute thicket { xsd:boolean }?,
4619
4620 ## default value: true
4621 attribute longFileNames { xsd:boolean }?,
4622
4623 ## default value: false
4624 attribute vml { xsd:boolean }?,
4625
4626 ## default value: false
4627 attribute allowPng { xsd:boolean }?,
4628
4629 ## default value: 800x600
4630 attribute targetScreenSize { sml_ST_TargetScreenSize }?,
4631
4632 ## default value: 96
4633 attribute dpi { xsd:unsignedInt }?,
4634 attribute codePage { xsd:unsignedInt }?,
4635 attribute characterSet { xsd:string }?
4636 sml_ST_TargetScreenSize =
4637   string "544x376"
4638   | string "640x480"
4639   | string "720x512"
4640   | string "800x600"
4641   | string "1024x768"
4642   | string "1152x882"
4643   | string "1152x900"
4644   | string "1280x1024"
4645   | string "1600x1200"
4646   | string "1800x1440"
4647   | string "1920x1200"
4648 sml_CT_FunctionGroups =
4649
4650 ## default value: 16
4651 attribute builtInGroupCount { xsd:unsignedInt }?,

```

```

4652     element functionGroup { sml_CT_FunctionGroup }*
4653 sml_CT_FunctionGroup = attribute name { s_ST_Xstring }?
4654 sml_CT_WebPublishObjects =
4655     attribute count { xsd:unsignedInt }?,
4656     element webPublishObject { sml_CT_WebPublishObject }+
4657 sml_CT_WebPublishObject =
4658     attribute id { xsd:unsignedInt },
4659     attribute divId { s_ST_Xstring },
4660     attribute sourceObject { s_ST_Xstring }?,
4661     attribute destinationFile { s_ST_Xstring },
4662     attribute title { s_ST_Xstring }?,
4663
4664     ## default value: false
4665     attribute autoRepublish { xsd:boolean }?

```

B.2.1 Part Schemas

B.2.1.1 Calculation Chain Part

This schema is available in the file SpreadsheetML_Calculation_Chain.rnc.

```

1  include "sml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "any.rnc"
4  include "shared-commonSimpleTypes.rnc"
5  include "dml-spreadsheetDrawing.rnc"
6  include "dml-main.rnc"
7  include "dml-diagram.rnc"
8  include "dml-lockedCanvas.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_calcChain

```

B.2.1.2 Chartsheet Part

This schema is available in the file SpreadsheetML_Chartsheet.rnc.

```

1  include "sml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "any.rnc"
4  include "shared-commonSimpleTypes.rnc"
5  include "dml-spreadsheetDrawing.rnc"
6  include "dml-main.rnc"
7  include "dml-diagram.rnc"
8  include "dml-lockedCanvas.rnc"
9  include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_chartsheet

```

B.2.1.3 Comments Part

This schema is available in the file SpreadsheetML_Comments.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_comments

```

B.2.1.4 Connections Part

This schema is available in the file SpreadsheetML_Connections.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_connections

```

B.2.1.5 Custom XML Mappings Part

This schema is available in the file SpreadsheetML_Custom_XML_Mappings.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_MapInfo

```

B.2.1.6 Dialogsheet Part

This schema is available in the file SpreadsheetML_Dialogsheet.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"

```

```

4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_dialogsheet

```

B.2.1.7 Drawing Part

This schema is available in the file SpreadsheetML_Drawing.rnc.

```

1 include "dml-spreadsheetDrawing.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = xdr_wsDr

```

B.2.1.8 External Workbook References Part

This schema is available in the file SpreadsheetML_External_Workbook_References.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_externalLink

```

B.2.1.9 Metadata Part

This schema is available in the file SpreadsheetML_Metadata.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"

```



```

8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_metadata

```

B.2.1.10 Pivot Table Part

This schema is available in the file SpreadsheetML_Pivot_Table.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_pivotTableDefinition

```

B.2.1.11 Pivot Table Cache Definition Part

This schema is available in the file SpreadsheetML_Pivot_Table_Cache_Definition.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_pivotCacheDefinition

```

B.2.1.12 Pivot Table Cache Records Part

This schema is available in the file SpreadsheetML_Pivot_Table_Cache_Records.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"

```

```

11 include "dml-picture.rnc"
12 start = sml_pivotCacheRecords

```

B.2.1.13 Query Table Part

This schema is available in the file SpreadsheetML_Query_Table.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_queryTable

```

B.2.1.14 Shared String Table Part

This schema is available in the file SpreadsheetML_Shared_String_Table.rnc.

```

15 include "sml.rnc"
16 include "shared-relationshipReference.rnc"
17 include "any.rnc"
18 include "shared-commonSimpleTypes.rnc"
19 include "dml-spreadsheetDrawing.rnc"
20 include "dml-main.rnc"
21 include "dml-diagram.rnc"
22 include "dml-lockedCanvas.rnc"
23 include "dml-chart.rnc"
24 include "dml-chartDrawing.rnc"
25 include "dml-picture.rnc"
26 include "dml-compatibility.rnc"
27 start = sml_sst

```

B.2.1.15 Shared Workbook Revision Headers Part

This schema is available in the file SpreadsheetML_Shared_Workbook_Revision_Headers.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_headers

```

B.2.1.16 Shared Workbook Revision Log Part

This schema is available in the file SpreadsheetML_Shared_Workbook_Revision_Log.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_revisions

```

B.2.1.17 Shared Workbook User Data Part

This schema is available in the file SpreadsheetML_Shared_Workbook_User_Data.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_users

```

B.2.1.18 Single Cell Table Definitions Part

This schema is available in the file SpreadsheetML_Single_Cell_Table_Definitions.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_singleXmlCells

```

B.2.1.19 Styles Part

This schema is available in the file SpreadsheetML_Styles.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_styleSheet

```

B.2.1.20 Table Definitions Part

This schema is available in the file SpreadsheetML_Table_Definitions.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_table

```

B.2.1.21 Volatile Dependencies Part

This schema is available in the file SpreadsheetML_Volatile_Dependencies.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_volTypes

```

B.2.1.22 Workbook Part

This schema is available in the file SpreadsheetML_Workbook.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"

```

```

4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_workbook

```

B.2.1.23 Worksheet Part

This schema is available in the file SpreadsheetML_Worksheet.rnc.

```

1 include "sml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "any.rnc"
4 include "shared-commonSimpleTypes.rnc"
5 include "dml-spreadsheetDrawing.rnc"
6 include "dml-main.rnc"
7 include "dml-diagram.rnc"
8 include "dml-lockedCanvas.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 start = sml_worksheet

```

B.3 PresentationML

This schema is available in the file pml.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/presentationml/2006/main"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace o = "urn:schemas-microsoft-com:office:office"
5 namespace p =
6   "http://schemas.openxmlformats.org/presentationml/2006/main"
7 namespace r =
8   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9 namespace s =
10  "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11 namespace v = "urn:schemas-microsoft-com:vml"
12 namespace w10 = "urn:schemas-microsoft-com:office:word"
13 namespace x = "urn:schemas-microsoft-com:office:excel"
14
15 p_ST_TransitionSideDirectionType = "l" | "u" | "r" | "d"
16 p_ST_TransitionCornerDirectionType = "lu" | "ru" | "ld" | "rd"
17 p_ST_TransitionInOutDirectionType = "out" | "in"
18 p_CT_SideDirectionTransition =
19
20   ## default value: l
21   attribute dir { p_ST_TransitionSideDirectionType }?
22 p_CT_CornerDirectionTransition =

```

```

23
24   ## default value: lu
25   attribute dir { p_ST_TransitionCornerDirectionType }?
26 p_ST_TransitionEightDirectionType =
27   p_ST_TransitionSideDirectionType | p_ST_TransitionCornerDirectionType
28 p_CT_EightDirectionTransition =
29
30   ## default value: l
31   attribute dir { p_ST_TransitionEightDirectionType }?
32 p_CT_OrientationTransition =
33
34   ## default value: horz
35   attribute dir { p_ST_Direction }?
36 p_CT_InOutTransition =
37
38   ## default value: out
39   attribute dir { p_ST_TransitionInOutDirectionType }?
40 p_CT_OptionalBlackTransition =
41
42   ## default value: false
43   attribute thruBlk { xsd:boolean }?
44 p_CT_SplitTransition =
45
46   ## default value: horz
47   attribute orient { p_ST_Direction }?,
48
49   ## default value: out
50   attribute dir { p_ST_TransitionInOutDirectionType }?
51 p_CT_WheelTransition =
52
53   ## default value: 4
54   attribute spokes { xsd:unsignedInt }?
55 p_CT_TransitionStartSoundAction =
56
57   ## default value: false
58   attribute loop { xsd:boolean }?,
59   element snd { a_CT_EmbeddedWAVAudioFile }
60 p_CT_TransitionSoundAction =
61   element stSnd { p_CT_TransitionStartSoundAction }
62   | element endSnd { p_CT_Empty }
63 p_ST_TransitionSpeed = "slow" | "med" | "fast"
64 p_CT_SlideTransition =
65
66   ## default value: fast
67   attribute spd { p_ST_TransitionSpeed }?,
68
69   ## default value: true
70   attribute advClick { xsd:boolean }?,
71   attribute advTm { xsd:unsignedInt }?,
72   (element blinds { p_CT_OrientationTransition }
73   | element checker { p_CT_OrientationTransition }
74   | element circle { p_CT_Empty }
75   | element dissolve { p_CT_Empty }

```

```

76 | element comb { p_CT_OrientationTransition }
77 | element cover { p_CT_EightDirectionTransition }
78 | element cut { p_CT_OptionalBlackTransition }
79 | element diamond { p_CT_Empty }
80 | element fade { p_CT_OptionalBlackTransition }
81 | element newsflash { p_CT_Empty }
82 | element plus { p_CT_Empty }
83 | element pull { p_CT_EightDirectionTransition }
84 | element push { p_CT_SideDirectionTransition }
85 | element random { p_CT_Empty }
86 | element randomBar { p_CT_OrientationTransition }
87 | element split { p_CT_SplitTransition }
88 | element strips { p_CT_CornerDirectionTransition }
89 | element wedge { p_CT_Empty }
90 | element wheel { p_CT_WheelTransition }
91 | element wipe { p_CT_SideDirectionTransition }
92 | element zoom { p_CT_InOutTransition }}?,
93 element sndAc { p_CT_TransitionSoundAction }?,
94 element extLst { p_CT_ExtensionListModify }?
95 p_ST_TLTimeIndefinite = "indefinite"
96 p_ST_TLTime = xsd:unsignedInt | p_ST_TLTimeIndefinite
97 p_ST_TLTimeNodeID = xsd:unsignedInt
98 p_CT_TLIterateIntervalTime = attribute val { p_ST_TLTime }
99 p_CT_TLIterateIntervalPercentage =
100 attribute val { a_ST_PositivePercentage }
101 p_ST_IterateType = "el" | "wd" | "lt"
102 p_CT_TLIterateData =
103
104 ## default value: el
105 attribute type { p_ST_IterateType }?,
106
107 ## default value: false
108 attribute backwards { xsd:boolean }?,
109 (element tmAbs { p_CT_TLIterateIntervalTime }
110 | element tmPct { p_CT_TLIterateIntervalPercentage })
111 p_CT_TLSubShapeId = attribute spid { a_ST_ShapeID }
112 p_CT_TLTextTargetElement =
113 (element charRg { p_CT_IndexRange }
114 | element pRg { p_CT_IndexRange }}?)
115 p_ST_TLChartSubelementType =
116 "gridLegend" | "series" | "category" | "ptInSeries" | "ptInCategory"
117 p_CT_TLOleChartTargetElement =
118 attribute type { p_ST_TLChartSubelementType },
119
120 ## default value: 0
121 attribute lvl { xsd:unsignedInt }?
122 p_CT_TLShapeTargetElement =
123 attribute spid { a_ST_DrawingElementId },
124 (element bg { p_CT_Empty }
125 | element subSp { p_CT_TLSubShapeId }
126 | element oleChartEl { p_CT_TLOleChartTargetElement }
127 | element txEl { p_CT_TLTextTargetElement }
128 | element graphicEl { a_CT_AnimationElementChoice }}?)

```

```

129 p_CT_TLTimeTargetElement =
130     element sldTgt { p_CT_Empty }
131     | element sndTgt { a_CT_EmbeddedWAVAudioFile }
132     | element spTgt { p_CT_TLShapeTargetElement }
133     | element inkTgt { p_CT_TLSubShapeId }
134 p_CT_TLTriggerTimeNodeID = attribute val { p_ST_TLTimeNodeID }
135 p_ST_TLTriggerRuntimeNode = "first" | "last" | "all"
136 p_CT_TLTriggerRuntimeNode = attribute val { p_ST_TLTriggerRuntimeNode }
137 p_ST_TLTriggerEvent =
138     "onBegin"
139     | "onEnd"
140     | "begin"
141     | "end"
142     | "onClick"
143     | "onDb1Click"
144     | "onMouseOver"
145     | "onMouseOut"
146     | "onNext"
147     | "onPrev"
148     | "onStopAudio"
149 p_CT_TLTimeCondition =
150     attribute evt { p_ST_TLTriggerEvent }?,
151     attribute delay { p_ST_TLTime }?,
152     (element tgtEl { p_CT_TLTimeTargetElement }
153     | element tn { p_CT_TLTriggerTimeNodeID }
154     | element rtn { p_CT_TLTriggerRuntimeNode })?
155 p_CT_TLTimeConditionList = element cond { p_CT_TLTimeCondition }+
156 p_CT_TimeNodeList =
157     (element par { p_CT_TLTimeNodeParallel }
158     | element seq { p_CT_TLTimeNodeSequence }
159     | element excl { p_CT_TLTimeNodeExclusive }
160     | element anim { p_CT_TLAnimateBehavior }
161     | element animClr { p_CT_TLAnimateColorBehavior }
162     | element animEffect { p_CT_TLAnimateEffectBehavior }
163     | element animMotion { p_CT_TLAnimateMotionBehavior }
164     | element animRot { p_CT_TLAnimateRotationBehavior }
165     | element animScale { p_CT_TLAnimateScaleBehavior }
166     | element cmd { p_CT_TLCommandBehavior }
167     | element set { p_CT_TLSetBehavior }
168     | element audio { p_CT_TLMediaNodeAudio }
169     | element video { p_CT_TLMediaNodeVideo })+
170 p_ST_TLTimeNodePresetClassType =
171     "entr" | "exit" | "emph" | "path" | "verb" | "mediacall"
172 p_ST_TLTimeNodeRestartType = "always" | "whenNotActive" | "never"
173 p_ST_TLTimeNodeFillType = "remove" | "freeze" | "hold" | "transition"
174 p_ST_TLTimeNodeSyncType = "canSlip" | "locked"
175 p_ST_TLTimeNodeMasterRelation = "sameClick" | "lastClick" | "nextClick"
176 p_ST_TLTimeNodeType =
177     "clickEffect"
178     | "withEffect"
179     | "afterEffect"
180     | "mainSeq"
181     | "interactiveSeq"

```



```

182 | "clickPar"
183 | "withGroup"
184 | "afterGroup"
185 | "tmRoot"
186 p_CT_TLCommonTimeNodeData =
187   attribute id { p_ST_TLTimeNodeID }?,
188   attribute presetID { xsd:int }?,
189   attribute presetClass { p_ST_TLTimeNodePresetClassType }?,
190   attribute presetSubtype { xsd:int }?,
191   attribute dur { p_ST_TLTime }?,
192
193   ## default value: 1000
194   attribute repeatCount { p_ST_TLTime }?,
195   attribute repeatDur { p_ST_TLTime }?,
196
197   ## default value: 100%
198   attribute spd { a_ST_Percentage }?,
199
200   ## default value: 0%
201   attribute accel { a_ST_PositiveFixedPercentage }?,
202
203   ## default value: 0%
204   attribute decel { a_ST_PositiveFixedPercentage }?,
205
206   ## default value: false
207   attribute autoRev { xsd:boolean }?,
208   attribute restart { p_ST_TLTimeNodeRestartType }?,
209   attribute fill { p_ST_TLTimeNodeFillType }?,
210   attribute syncBehavior { p_ST_TLTimeNodeSyncType }?,
211   attribute tmFilter { xsd:string }?,
212   attribute evtFilter { xsd:string }?,
213   attribute display { xsd:boolean }?,
214   attribute masterRel { p_ST_TLTimeNodeMasterRelation }?,
215   attribute bldLvl { xsd:int }?,
216   attribute grpId { xsd:unsignedInt }?,
217   attribute afterEffect { xsd:boolean }?,
218   attribute nodeType { p_ST_TLTimeNodeType }?,
219   attribute nodePh { xsd:boolean }?,
220   element stCondLst { p_CT_TLTimeConditionList }?,
221   element endCondLst { p_CT_TLTimeConditionList }?,
222   element endSync { p_CT_TLTimeCondition }?,
223   element iterate { p_CT_TLIterateData }?,
224   element childTnLst { p_CT_TimeNodeList }?,
225   element subTnLst { p_CT_TimeNodeList }?
226 p_CT_TLTimeNodeParallel = element cTn { p_CT_TLCommonTimeNodeData }
227 p_ST_TLNextActionType = "none" | "seek"
228 p_ST_TLPreviousActionType = "none" | "skipTimed"
229 p_CT_TLTimeNodeSequence =
230   attribute concurrent { xsd:boolean }?,
231   attribute prevAc { p_ST_TLPreviousActionType }?,
232   attribute nextAc { p_ST_TLNextActionType }?,
233   element cTn { p_CT_TLCommonTimeNodeData },
234   element prevCondLst { p_CT_TLTimeConditionList }?,

```

```

235     element nextCondLst { p_CT_TLTimeConditionList }?
236 p_CT_TLTimeNodeExclusive = element cTn { p_CT_TLCommonTimeNodeData }
237 p_CT_TLBehaviorAttributeNameList = element attrName { xsd:string }+
238 p_ST_TLBehaviorAdditiveType = "base" | "sum" | "repl" | "mult" | "none"
239 p_ST_TLBehaviorAccumulateType = "none" | "always"
240 p_ST_TLBehaviorTransformType = "pt" | "img"
241 p_ST_TLBehaviorOverrideType = "normal" | "childStyle"
242 p_CT_TLCommonBehaviorData =
243     attribute additive { p_ST_TLBehaviorAdditiveType }?,
244     attribute accumulate { p_ST_TLBehaviorAccumulateType }?,
245     attribute xfrmType { p_ST_TLBehaviorTransformType }?,
246     attribute from { xsd:string }?,
247     attribute to { xsd:string }?,
248     attribute by { xsd:string }?,
249     attribute rctx { xsd:string }?,
250     attribute override { p_ST_TLBehaviorOverrideType }?,
251     element cTn { p_CT_TLCommonTimeNodeData },
252     element tgtEl { p_CT_TLTimeTargetElement },
253     element attrNameLst { p_CT_TLBehaviorAttributeNameList }?
254 p_CT_TLAnimVariantBooleanVal = attribute val { xsd:boolean }
255 p_CT_TLAnimVariantIntegerVal = attribute val { xsd:int }
256 p_CT_TLAnimVariantFloatVal = attribute val { xsd:float }
257 p_CT_TLAnimVariantStringVal = attribute val { xsd:string }
258 p_CT_TLAnimVariant =
259     element boolVal { p_CT_TLAnimVariantBooleanVal }
260     | element intVal { p_CT_TLAnimVariantIntegerVal }
261     | element fltVal { p_CT_TLAnimVariantFloatVal }
262     | element strVal { p_CT_TLAnimVariantStringVal }
263     | element clrVal { a_CT_Color }
264 p_ST_TLTimeAnimateValueTime =
265     a_ST_PositiveFixedPercentage | p_ST_TLTimeIndefinite
266 p_CT_TLTimeAnimateValue =
267
268     ## default value: indefinite
269     attribute tm { p_ST_TLTimeAnimateValueTime }?,
270     attribute fmla { xsd:string }?,
271     element val { p_CT_TLAnimVariant }?
272 p_CT_TLTimeAnimateValueList = element tav { p_CT_TLTimeAnimateValue }*
273 p_ST_TLAnimateBehaviorCalcMode = "discrete" | "lin" | "fmla"
274 p_ST_TLAnimateBehaviorValueType = "str" | "num" | "clr"
275 p_CT_TLAnimateBehavior =
276     attribute by { xsd:string }?,
277     attribute from { xsd:string }?,
278     attribute to { xsd:string }?,
279     attribute calcmode { p_ST_TLAnimateBehaviorCalcMode }?,
280     attribute valueType { p_ST_TLAnimateBehaviorValueType }?,
281     element cBhvr { p_CT_TLCommonBehaviorData },
282     element tavLst { p_CT_TLTimeAnimateValueList }?
283 p_CT_TLByRgbColorTransform =
284     attribute r { a_ST_FixedPercentage },
285     attribute g { a_ST_FixedPercentage },
286     attribute b { a_ST_FixedPercentage }
287 p_CT_TLByHslColorTransform =

```

```

288     attribute h { a_ST_Angle },
289     attribute s { a_ST_FixedPercentage },
290     attribute l { a_ST_FixedPercentage }
291 p_CT_TLByAnimateColorTransform =
292     element rgb { p_CT_TLByRgbColorTransform }
293     | element hsl { p_CT_TLByHslColorTransform }
294 p_ST_TLAnimateColorSpace = "rgb" | "hsl"
295 p_ST_TLAnimateColorDirection = "cw" | "ccw"
296 p_CT_TLAnimateColorBehavior =
297     attribute clrSpc { p_ST_TLAnimateColorSpace }?,
298     attribute dir { p_ST_TLAnimateColorDirection }?,
299     element cBhvr { p_CT_TLCommonBehaviorData },
300     element by { p_CT_TLByAnimateColorTransform }?,
301     element from { a_CT_Color }?,
302     element to { a_CT_Color }?
303 p_ST_TLAnimateEffectTransition = "in" | "out" | "none"
304 p_CT_TLAnimateEffectBehavior =
305     attribute transition { p_ST_TLAnimateEffectTransition }?,
306     attribute filter { xsd:string }?,
307     attribute prLst { xsd:string }?,
308     element cBhvr { p_CT_TLCommonBehaviorData },
309     element progress { p_CT_TLAnimVariant }?
310 p_ST_TLAnimateMotionBehaviorOrigin = "parent" | "layout"
311 p_ST_TLAnimateMotionPathEditMode = "relative" | "fixed"
312 p_CT_TLPoint =
313     attribute x { a_ST_Percentage },
314     attribute y { a_ST_Percentage }
315 p_CT_TLAnimateMotionBehavior =
316     attribute origin { p_ST_TLAnimateMotionBehaviorOrigin }?,
317     attribute path { xsd:string }?,
318     attribute pathEditMode { p_ST_TLAnimateMotionPathEditMode }?,
319     attribute rAng { a_ST_Angle }?,
320     attribute ptsTypes { xsd:string }?,
321     element cBhvr { p_CT_TLCommonBehaviorData },
322     element by { p_CT_TLPoint }?,
323     element from { p_CT_TLPoint }?,
324     element to { p_CT_TLPoint }?,
325     element rCtr { p_CT_TLPoint }?
326 p_CT_TLAnimateRotationBehavior =
327     attribute by { a_ST_Angle }?,
328     attribute from { a_ST_Angle }?,
329     attribute to { a_ST_Angle }?,
330     element cBhvr { p_CT_TLCommonBehaviorData }
331 p_CT_TLAnimateScaleBehavior =
332     attribute zoomContents { xsd:boolean }?,
333     element cBhvr { p_CT_TLCommonBehaviorData },
334     element by { p_CT_TLPoint }?,
335     element from { p_CT_TLPoint }?,
336     element to { p_CT_TLPoint }?
337 p_ST_TLCommandType = "evt" | "call" | "verb"
338 p_CT_TLCommandBehavior =
339     attribute type { p_ST_TLCommandType }?,
340     attribute cmd { xsd:string }?,

```

```

341     element cBhvr { p_CT_TLCommonBehaviorData }
342 p_CT_TLSetBehavior =
343     element cBhvr { p_CT_TLCommonBehaviorData },
344     element to { p_CT_TLAnimVariant }?
345 p_CT_TLCommonMediaNodeData =
346
347     ## default value: 50%
348     attribute vol { a_ST_PositiveFixedPercentage }?,
349
350     ## default value: false
351     attribute mute { xsd:boolean }?,
352
353     ## default value: 1
354     attribute numSld { xsd:unsignedInt }?,
355
356     ## default value: true
357     attribute showWhenStopped { xsd:boolean }?,
358     element cTn { p_CT_TLCommonTimeNodeData },
359     element tgtEl { p_CT_TLTimeTargetElement }
360 p_CT_TLMediaNodeAudio =
361
362     ## default value: false
363     attribute isNarration { xsd:boolean }?,
364     element cMediaNode { p_CT_TLCommonMediaNodeData }
365 p_CT_TLMediaNodeVideo =
366
367     ## default value: false
368     attribute fullScrn { xsd:boolean }?,
369     element cMediaNode { p_CT_TLCommonMediaNodeData }
370 p_AG_TLBuild =
371     attribute spid { a_ST_DrawingElementId },
372     attribute grpId { xsd:unsignedInt },
373
374     ## default value: false
375     attribute uiExpand { xsd:boolean }?
376 p_CT_TLTemplate =
377
378     ## default value: 0
379     attribute lvl { xsd:unsignedInt }?,
380     element tnLst { p_CT_TimeNodeList }
381 p_CT_TLTemplateList = element tpl { p_CT_TLTemplate }*
382 p_ST_TLParaBuildType = "allAtOnce" | "p" | "cust" | "whole"
383 p_CT_TLBuildParagraph =
384     p_AG_TLBuild,
385
386     ## default value: whole
387     attribute build { p_ST_TLParaBuildType }?,
388
389     ## default value: 1
390     attribute bldLvl { xsd:unsignedInt }?,
391
392     ## default value: false
393     attribute animBg { xsd:boolean }?,

```

```

394
395   ## default value: true
396   attribute autoUpdateAnimBg { xsd:boolean }?,
397
398   ## default value: false
399   attribute rev { xsd:boolean }?,
400
401   ## default value: indefinite
402   attribute advAuto { p_ST_TLTime }?,
403   element tplLst { p_CT_TLTemplateList }?
404 p_ST_TLDiagramBuildType =
405   "whole"
406   | "depthByNode"
407   | "depthByBranch"
408   | "breadthByNode"
409   | "breadthByLvl"
410   | "cw"
411   | "cwIn"
412   | "cwOut"
413   | "ccw"
414   | "ccwIn"
415   | "ccwOut"
416   | "inByRing"
417   | "outByRing"
418   | "up"
419   | "down"
420   | "allAtOnce"
421   | "cust"
422 p_CT_TLBuildDiagram =
423   p_AG_TLBuild,
424
425   ## default value: whole
426   attribute bld { p_ST_TLDiagramBuildType }?
427 p_ST_TLOleChartBuildType =
428   "allAtOnce" | "series" | "category" | "seriesEl" | "categoryEl"
429 p_CT_TLOleBuildChart =
430   p_AG_TLBuild,
431
432   ## default value: allAtOnce
433   attribute bld { p_ST_TLOleChartBuildType }?,
434
435   ## default value: true
436   attribute animBg { xsd:boolean }?
437 p_CT_TLGraphicalObjectBuild =
438   p_AG_TLBuild,
439   (element bldAsOne { p_CT_Empty }
440    | element bldSub { a_CT_AnimationGraphicalObjectBuildProperties })
441 p_CT_BuildList =
442   (element bldP { p_CT_TLBuildParagraph }
443    | element bldDgm { p_CT_TLBuildDiagram }
444    | element bldOleChart { p_CT_TLOleBuildChart }
445    | element bldGraphic { p_CT_TLGraphicalObjectBuild })+
446 p_CT_SlideTiming =

```

```

447     element tnLst { p_CT_TimeNodeList }?,
448     element bldLst { p_CT_BuildList }?,
449     element extLst { p_CT_ExtensionListModify }?
450 p_CT_Empty = empty
451 p_ST_Name = xsd:string
452 p_ST_Direction = "horz" | "vert"
453 p_ST_Index = xsd:unsignedInt
454 p_CT_IndexRange =
455     attribute st { p_ST_Index },
456     attribute end { p_ST_Index }
457 p_CT_SlideRelationshipListEntry = r_id
458 p_CT_SlideRelationshipList =
459     element sld { p_CT_SlideRelationshipListEntry }*
460 p_CT_CustomShowId = attribute id { xsd:unsignedInt }
461 p_EG_SlideListChoice =
462     element sldAll { p_CT_Empty }
463     | element sldRg { p_CT_IndexRange }
464     | element custShow { p_CT_CustomShowId }
465 p_CT_CustomerData = r_id
466 p_CT_TagsData = r_id
467 p_CT_CustomerDataList =
468     (element custData { p_CT_CustomerData }*,
469     element tags { p_CT_TagsData }?)?
470 p_CT_Extension =
471     attribute uri { xsd:token },
472     p_CT_Extension_any*
473 p_CT_Extension_any =
474     element * - (o:* | v:* | w10:* | x:*) {
475         anyAttribute*,
476         mixed { anyElement* }
477     }
478 p_EG_ExtensionList = element ext { p_CT_Extension }*
479 p_CT_ExtensionList = p_EG_ExtensionList?
480 p_CT_ExtensionListModify =
481
482     ## default value: false
483     attribute mod { xsd:boolean }?,
484     p_EG_ExtensionList?
485 p_CT_CommentAuthor =
486     attribute id { xsd:unsignedInt },
487     attribute name { p_ST_Name },
488     attribute initials { p_ST_Name },
489     attribute lastIdx { xsd:unsignedInt },
490     attribute clrIdx { xsd:unsignedInt },
491     element extLst { p_CT_ExtensionList }?
492 p_CT_CommentAuthorList = element cmAuthor { p_CT_CommentAuthor }*
493 p_cmAuthorLst = element cmAuthorLst { p_CT_CommentAuthorList }
494 p_CT_Comment =
495     attribute authorId { xsd:unsignedInt },
496     attribute dt { xsd:dateTime }?,
497     attribute idx { p_ST_Index },
498     element pos { a_CT_Point2D },
499     element text { xsd:string },

```

```

500     element extLst { p_CT_ExtensionListModify }?
501 p_CT_CommentList = element cm { p_CT_Comment }*
502 p_cmLst = element cmLst { p_CT_CommentList }
503 p_AG_Ole =
504     attribute name { xsd:string }?,
505
506     ## default value: false
507     attribute showAsIcon { xsd:boolean }?,
508     r_id?,
509     attribute imgW { a_ST_PositiveCoordinate32 }?,
510     attribute imgH { a_ST_PositiveCoordinate32 }?
511 p_ST_OleObjectFollowColorScheme = "none" | "full" | "textAndBackground"
512 p_CT_OleObjectEmbed =
513
514     ## default value: none
515     attribute followColorScheme { p_ST_OleObjectFollowColorScheme }?,
516     element extLst { p_CT_ExtensionList }?
517 p_CT_OleObjectLink =
518
519     ## default value: false
520     attribute updateAutomatic { xsd:boolean }?,
521     element extLst { p_CT_ExtensionList }?
522 p_CT_OleObject =
523     p_AG_Ole,
524     attribute progId { xsd:string }?,
525     (element embed { p_CT_OleObjectEmbed }
526      | element link { p_CT_OleObjectLink }
527     ),
528     (attribute spid { a_ST_ShapeID } | element pic { p_CT_Picture })
529 p_oleObj = element oleObj { p_CT_OleObject }
530 p_CT_Control =
531     p_AG_Ole,
532     element extLst { p_CT_ExtensionList }?,
533     (attribute spid { a_ST_ShapeID } | element pic { p_CT_Picture })
534 p_CT_ControlList = element control { p_CT_Control }*
535 p_ST_SlideId =
536     xsd:unsignedInt { minInclusive = "256" maxExclusive = "2147483648" }
537 p_CT_SlideIdListEntry =
538     attribute id { p_ST_SlideId },
539     r_id,
540     element extLst { p_CT_ExtensionList }?
541 p_CT_SlideIdList = element sldId { p_CT_SlideIdListEntry }*
542 p_ST_SlideMasterId = xsd:unsignedInt { minInclusive = "2147483648" }
543 p_CT_SlideMasterIdListEntry =
544     attribute id { p_ST_SlideMasterId }?,
545     r_id,
546     element extLst { p_CT_ExtensionList }?
547 p_CT_SlideMasterIdList =
548     element sldMasterId { p_CT_SlideMasterIdListEntry }*
549 p_CT_NotesMasterIdListEntry =
550     r_id,
551     element extLst { p_CT_ExtensionList }?
552 p_CT_NotesMasterIdList =

```

```

553     element notesMasterId { p_CT_NotesMasterIdListEntry }?
554 p_CT_HandoutMasterIdListEntry =
555     r_id,
556     element extLst { p_CT_ExtensionList }?
557 p_CT_HandoutMasterIdList =
558     element handoutMasterId { p_CT_HandoutMasterIdListEntry }?
559 p_CT_EmbeddedFontDataId = r_id
560 p_CT_EmbeddedFontListEntry =
561     element font { a_CT_TextFont },
562     element regular { p_CT_EmbeddedFontDataId }?,
563     element bold { p_CT_EmbeddedFontDataId }?,
564     element italic { p_CT_EmbeddedFontDataId }?,
565     element boldItalic { p_CT_EmbeddedFontDataId }?
566 p_CT_EmbeddedFontList =
567     element embeddedFont { p_CT_EmbeddedFontListEntry }*
568 p_CT_SmartTags = r_id
569 p_CT_CustomShow =
570     attribute name { p_ST_Name },
571     attribute id { xsd:unsignedInt },
572     element sldLst { p_CT_SlideRelationshipList },
573     element extLst { p_CT_ExtensionList }?
574 p_CT_CustomShowList = element custShow { p_CT_CustomShow }*
575 p_ST_PhotoAlbumLayout =
576     "fitToSlide"
577     | "1pic"
578     | "2pic"
579     | "4pic"
580     | "1picTitle"
581     | "2picTitle"
582     | "4picTitle"
583 p_ST_PhotoAlbumFrameShape =
584     "frameStyle1"
585     | "frameStyle2"
586     | "frameStyle3"
587     | "frameStyle4"
588     | "frameStyle5"
589     | "frameStyle6"
590     | "frameStyle7"
591 p_CT_PhotoAlbum =
592
593     ## default value: false
594     attribute bw { xsd:boolean }?,
595
596     ## default value: false
597     attribute showCaptions { xsd:boolean }?,
598
599     ## default value: fitToSlide
600     attribute layout { p_ST_PhotoAlbumLayout }?,
601
602     ## default value: frameStyle1
603     attribute frame { p_ST_PhotoAlbumFrameShape }?,
604     element extLst { p_CT_ExtensionList }?
605 p_ST_SlideSizeCoordinate =

```



```

606     xsd:int {
607         minInclusive = "914400"
608         maxInclusive = "51206400"
609     }
610 p_ST_SlideSizeType =
611     "screen4x3"
612     | "letter"
613     | "A4"
614     | "35mm"
615     | "overhead"
616     | "banner"
617     | "custom"
618     | "ledger"
619     | "A3"
620     | "B4ISO"
621     | "B5ISO"
622     | "B4JIS"
623     | "B5JIS"
624     | "hagakiCard"
625     | "screen16x9"
626     | "screen16x10"
627 p_CT_SlideSize =
628     attribute cx { p_ST_SlideSizeCoordinate },
629     attribute cy { p_ST_SlideSizeCoordinate },
630
631     ## default value: custom
632     attribute type { p_ST_SlideSizeType }?
633 p_CT_Kinsoku =
634     attribute lang { xsd:string }?,
635     attribute invalStChars { xsd:string },
636     attribute invalEndChars { xsd:string }
637 p_ST_BookmarkIdSeed =
638     xsd:unsignedInt { minInclusive = "1" maxExclusive = "2147483648" }
639 p_CT_ModifyVerifier =
640     attribute algorithmName { xsd:string }?,
641     attribute hashValue { xsd:base64Binary }?,
642     attribute saltValue { xsd:base64Binary }?,
643     attribute spinValue { xsd:unsignedInt }?,
644     attribute cryptProviderType { s_ST_CryptProv }?,
645     attribute cryptAlgorithmClass { s_ST_AlgClass }?,
646     attribute cryptAlgorithmType { s_ST_AlgType }?,
647     attribute cryptAlgorithmSid { xsd:unsignedInt }?,
648     attribute spinCount { xsd:unsignedInt }?,
649     attribute saltData { xsd:base64Binary }?,
650     attribute hashData { xsd:base64Binary }?,
651     attribute cryptProvider { xsd:string }?,
652     attribute algIdExt { xsd:unsignedInt }?,
653     attribute algIdExtSource { xsd:string }?,
654     attribute cryptProviderTypeExt { xsd:unsignedInt }?,
655     attribute cryptProviderTypeExtSource { xsd:string }?
656 p_CT_Presentation =
657
658     ## default value: 50%

```

```

659 attribute serverZoom { a_ST_Percentage }?,
660
661 ## default value: 1
662 attribute firstSlideNum { xsd:int }?,
663
664 ## default value: true
665 attribute showSpecialPlsOnTitleSld { xsd:boolean }?,
666
667 ## default value: false
668 attribute rtl { xsd:boolean }?,
669
670 ## default value: false
671 attribute removePersonalInfoOnSave { xsd:boolean }?,
672
673 ## default value: false
674 attribute compatMode { xsd:boolean }?,
675
676 ## default value: true
677 attribute strictFirstAndLastChars { xsd:boolean }?,
678
679 ## default value: false
680 attribute embedTrueTypeFonts { xsd:boolean }?,
681
682 ## default value: false
683 attribute saveSubsetFonts { xsd:boolean }?,
684
685 ## default value: true
686 attribute autoCompressPictures { xsd:boolean }?,
687
688 ## default value: 1
689 attribute bookmarkIdSeed { p_ST_BookmarkIdSeed }?,
690 attribute conformance { s_ST_ConformanceClass }?,
691 element sldMasterIdLst { p_CT_SlideMasterIdList }?,
692 element notesMasterIdLst { p_CT_NotesMasterIdList }?,
693 element handoutMasterIdLst { p_CT_HandoutMasterIdList }?,
694 element sldIdLst { p_CT_SlideIdList }?,
695 element sldSz { p_CT_SlideSize }?,
696 element notesSz { a_CT_PositiveSize2D },
697 element smartTags { p_CT_SmartTags }?,
698 element embeddedFontLst { p_CT_EmbeddedFontList }?,
699 element custShowLst { p_CT_CustomShowList }?,
700 element photoAlbum { p_CT_PhotoAlbum }?,
701 element custDataLst { p_CT_CustomerDataList }?,
702 element kinsoku { p_CT_Kinsoku }?,
703 element defaultTextStyle { a_CT_TextListStyle }?,
704 element modifyVerifier { p_CT_ModifyVerifier }?,
705 element extLst { p_CT_ExtensionList }?
706 p_presentation = element presentation { p_CT_Presentation }
707 p_CT_HtmlPublishProperties =
708
709 ## default value: true
710 attribute showSpeakerNotes { xsd:boolean }?,
711 attribute target { xsd:string }?,

```

```

712     attribute title { xsd:string }?,
713     r_id,
714     p_EG_SlideListChoice,
715     element extLst { p_CT_ExtensionList }?
716 p_ST_WebColorType =
717     "none"
718     | "browser"
719     | "presentationText"
720     | "presentationAccent"
721     | "whiteTextOnBlack"
722     | "blackTextOnWhite"
723 p_ST_WebScreenSize =
724     "544x376"
725     | "640x480"
726     | "720x512"
727     | "800x600"
728     | "1024x768"
729     | "1152x882"
730     | "1152x900"
731     | "1280x1024"
732     | "1600x1200"
733     | "1800x1400"
734     | "1920x1200"
735 p_ST_WebEncoding = xsd:string
736 p_CT_WebProperties =
737
738     ## default value: false
739     attribute showAnimation { xsd:boolean }?,
740
741     ## default value: true
742     attribute resizeGraphics { xsd:boolean }?,
743
744     ## default value: false
745     attribute allowPng { xsd:boolean }?,
746
747     ## default value: false
748     attribute relyOnVml { xsd:boolean }?,
749
750     ## default value: true
751     attribute organizeInFolders { xsd:boolean }?,
752
753     ## default value: true
754     attribute useLongFileNames { xsd:boolean }?,
755
756     ## default value: 800x600
757     attribute imgSz { p_ST_WebScreenSize }?,
758     attribute encoding { p_ST_WebEncoding }?,
759
760     ## default value: whiteTextOnBlack
761     attribute clr { p_ST_WebColorType }?,
762     element extLst { p_CT_ExtensionList }?
763 p_ST_PrintWhat =
764     "slides"

```

```

765 | "handouts1"
766 | "handouts2"
767 | "handouts3"
768 | "handouts4"
769 | "handouts6"
770 | "handouts9"
771 | "notes"
772 | "outline"
773 p_ST_PrintColorMode = "bw" | "gray" | "clr"
774 p_CT_PrintProperties =
775
776   ## default value: slides
777   attribute prnWhat { p_ST_PrintWhat }?,
778
779   ## default value: clr
780   attribute clrMode { p_ST_PrintColorMode }?,
781
782   ## default value: false
783   attribute hiddenSlides { xsd:boolean }?,
784
785   ## default value: false
786   attribute scaleToFitPaper { xsd:boolean }?,
787
788   ## default value: false
789   attribute frameSlides { xsd:boolean }?,
790   element extLst { p_CT_ExtensionList }?
791 p_CT_ShowInfoBrowse =
792
793   ## default value: true
794   attribute showScrollbar { xsd:boolean }?
795 p_CT_ShowInfoKiosk =
796
797   ## default value: 300000
798   attribute restart { xsd:unsignedInt }?
799 p_EG_ShowType =
800   element present { p_CT_Empty }
801   | element browse { p_CT_ShowInfoBrowse }
802   | element kiosk { p_CT_ShowInfoKiosk }
803 p_CT_ShowProperties =
804
805   ## default value: false
806   attribute loop { xsd:boolean }?,
807
808   ## default value: false
809   attribute showNarration { xsd:boolean }?,
810
811   ## default value: true
812   attribute showAnimation { xsd:boolean }?,
813
814   ## default value: true
815   attribute useTimings { xsd:boolean }?,
816   (p_EG_ShowType?,
817   p_EG_SlidelistChoice?,

```

```

818     element penClr { a_CT_Color }?,
819     element extLst { p_CT_ExtensionList }?))?
820 p_CT_PresentationProperties =
821     element htmlPubPr { p_CT_HtmlPublishProperties }?,
822     element webPr { p_CT_WebProperties }?,
823     element prnPr { p_CT_PrintProperties }?,
824     element showPr { p_CT_ShowProperties }?,
825     element clrMru { a_CT_ColorMRU }?,
826     element extLst { p_CT_ExtensionList }?
827 p_presentationPr =
828     element presentationPr { p_CT_PresentationProperties }
829 p_CT_HeaderFooter =
830
831     ## default value: true
832     attribute sldNum { xsd:boolean }?,
833
834     ## default value: true
835     attribute hdr { xsd:boolean }?,
836
837     ## default value: true
838     attribute ftr { xsd:boolean }?,
839
840     ## default value: true
841     attribute dt { xsd:boolean }?,
842     element extLst { p_CT_ExtensionListModify }?
843 p_ST_PlaceholderType =
844     "title"
845     | "body"
846     | "ctrTitle"
847     | "subTitle"
848     | "dt"
849     | "sldNum"
850     | "ftr"
851     | "hdr"
852     | "obj"
853     | "chart"
854     | "tbl"
855     | "clipArt"
856     | "dgm"
857     | "media"
858     | "sldImg"
859     | "pic"
860 p_ST_PlaceholderSize = "full" | "half" | "quarter"
861 p_CT_Placeholder =
862
863     ## default value: obj
864     attribute type { p_ST_PlaceholderType }?,
865
866     ## default value: horz
867     attribute orient { p_ST_Direction }?,
868
869     ## default value: full
870     attribute sz { p_ST_PlaceholderSize }?,

```

```

871
872   ## default value: 0
873   attribute idx { xsd:unsignedInt }?,
874
875   ## default value: false
876   attribute hasCustomPrompt { xsd:boolean }?,
877   element extLst { p_CT_ExtensionListModify }?
878 p_CT_ApplicationNonVisualDrawingProps =
879
880   ## default value: false
881   attribute isPhoto { xsd:boolean }?,
882
883   ## default value: false
884   attribute userDrawn { xsd:boolean }?,
885   element ph { p_CT_Placeholder }?,
886   a_EG_Media?,
887   element custDataLst { p_CT_CustomerDataList }?,
888   element extLst { p_CT_ExtensionList }?
889 p_CT_ShapeNonVisual =
890   element cNvPr { a_CT_NonVisualDrawingProps },
891   element cNvSpPr { a_CT_NonVisualDrawingShapeProps },
892   element nvPr { p_CT_ApplicationNonVisualDrawingProps }
893 p_CT_Shape =
894
895   ## default value: false
896   attribute useBgFill { xsd:boolean }?,
897   element nvSpPr { p_CT_ShapeNonVisual },
898   element spPr { a_CT_ShapeProperties },
899   element style { a_CT_ShapeStyle }?,
900   element txBdy { a_CT_TextBody }?,
901   element extLst { p_CT_ExtensionListModify }?
902 p_CT_ConnectorNonVisual =
903   element cNvPr { a_CT_NonVisualDrawingProps },
904   element cNvCxnSpPr { a_CT_NonVisualConnectorProperties },
905   element nvPr { p_CT_ApplicationNonVisualDrawingProps }
906 p_CT_Connector =
907   element nvCxnSpPr { p_CT_ConnectorNonVisual },
908   element spPr { a_CT_ShapeProperties },
909   element style { a_CT_ShapeStyle }?,
910   element extLst { p_CT_ExtensionListModify }?
911 p_CT_PictureNonVisual =
912   element cNvPr { a_CT_NonVisualDrawingProps },
913   element cNvPicPr { a_CT_NonVisualPictureProperties },
914   element nvPr { p_CT_ApplicationNonVisualDrawingProps }
915 p_CT_Picture =
916   element nvPicPr { p_CT_PictureNonVisual },
917   element blipFill { a_CT_BlipFillProperties },
918   element spPr { a_CT_ShapeProperties },
919   element style { a_CT_ShapeStyle }?,
920   element extLst { p_CT_ExtensionListModify }?
921 p_CT_GraphicalObjectFrameNonVisual =
922   element cNvPr { a_CT_NonVisualDrawingProps },
923   element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties },

```

```

924     element nvPr { p_CT_ApplicationNonVisualDrawingProps }
925 p_CT_GraphicalObjectFrame =
926     attribute bwMode { a_ST_BlackWhiteMode }?,
927     element nvGraphicFramePr { p_CT_GraphicalObjectFrameNonVisual },
928     element xfrm { a_CT_Transform2D },
929     a_graphic,
930     element extLst { p_CT_ExtensionListModify }?
931 p_CT_GroupShapeNonVisual =
932     element cNvPr { a_CT_NonVisualDrawingProps },
933     element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps },
934     element nvPr { p_CT_ApplicationNonVisualDrawingProps }
935 p_CT_GroupShape =
936     element nvGrpSpPr { p_CT_GroupShapeNonVisual },
937     element grpSpPr { a_CT_GroupShapeProperties },
938     (element sp { p_CT_Shape }
939     | element grpSp { p_CT_GroupShape }
940     | element graphicFrame { p_CT_GraphicalObjectFrame }
941     | element cxnSp { p_CT_Connector }
942     | element pic { p_CT_Picture }
943     | element contentPart { p_CT_Rel })*,
944     element extLst { p_CT_ExtensionListModify }?
945 p_CT_Rel = r_id
946 p_EG_TopLevelSlide = element clrMap { a_CT_ColorMapping }
947 p_EG_ChildSlide = element clrMapOvr { a_CT_ColorMappingOverride }?
948 p_AG_ChildSlide =
949
950     ## default value: true
951     attribute showMasterSp { xsd:boolean }?,
952
953     ## default value: true
954     attribute showMasterPhAnim { xsd:boolean }?
955 p_CT_BackgroundProperties =
956
957     ## default value: false
958     attribute shadeToTitle { xsd:boolean }?,
959     a_EG_FillProperties,
960     a_EG_EffectProperties?,
961     element extLst { p_CT_ExtensionList }?
962 p_EG_Background =
963     element bgPr { p_CT_BackgroundProperties }
964     | element bgRef { a_CT_StyleMatrixReference }
965 p_CT_Background =
966
967     ## default value: white
968     attribute bwMode { a_ST_BlackWhiteMode }?,
969     p_EG_Background
970 p_CT_CommonSlideData =
971     attribute name { xsd:string }?,
972     element bg { p_CT_Background }?,
973     element spTree { p_CT_GroupShape },
974     element custDataLst { p_CT_CustomerDataList }?,
975     element controls { p_CT_ControlList }?,
976     element extLst { p_CT_ExtensionList }?

```

```

977 p_CT_Slide =
978     p_AG_ChildSlide,
979
980     ## default value: true
981     attribute show { xsd:boolean }?,
982     element cSld { p_CT_CommonSlideData },
983     p_EG_ChildSlide?,
984     element transition { p_CT_SlideTransition }?,
985     element timing { p_CT_SlideTiming }?,
986     element extLst { p_CT_ExtensionListModify }?
987 p_sld = element sld { p_CT_Slide }
988 p_ST_SlideLayoutType =
989     "title"
990     | "tx"
991     | "twoColTx"
992     | "tbl"
993     | "txAndChart"
994     | "chartAndTx"
995     | "dgm"
996     | "chart"
997     | "txAndClipArt"
998     | "clipArtAndTx"
999     | "titleOnly"
1000    | "blank"
1001    | "txAndObj"
1002    | "objAndTx"
1003    | "objOnly"
1004    | "obj"
1005    | "txAndMedia"
1006    | "mediaAndTx"
1007    | "objOverTx"
1008    | "txOverObj"
1009    | "txAndTwoObj"
1010    | "twoObjAndTx"
1011    | "twoObjOverTx"
1012    | "fourObj"
1013    | "vertTx"
1014    | "clipArtAndVertTx"
1015    | "vertTitleAndTx"
1016    | "vertTitleAndTxOverChart"
1017    | "twoObj"
1018    | "objAndTwoObj"
1019    | "twoObjAndObj"
1020    | "cust"
1021    | "secHead"
1022    | "twoTxTwoObj"
1023    | "objTx"
1024    | "picTx"
1025 p_CT_SlideLayout =
1026     p_AG_ChildSlide,
1027     attribute matchingName { xsd:string }?,
1028
1029     ## default value: cust

```



```

1030 attribute type { p_ST_SlideLayoutType }?,
1031
1032 ## default value: false
1033 attribute preserve { xsd:boolean }?,
1034
1035 ## default value: false
1036 attribute userDrawn { xsd:boolean }?,
1037 element cSld { p_CT_CommonSlideData },
1038 p_EG_ChildSlide?,
1039 element transition { p_CT_SlideTransition }?,
1040 element timing { p_CT_SlideTiming }?,
1041 element hf { p_CT_HeaderFooter }?,
1042 element extLst { p_CT_ExtensionListModify }?
1043 p_sldLayout = element sldLayout { p_CT_SlideLayout }
1044 p_CT_SlideMasterTextStyles =
1045     element titleStyle { a_CT_TextListStyle }?,
1046     element bodyStyle { a_CT_TextListStyle }?,
1047     element otherStyle { a_CT_TextListStyle }?,
1048     element extLst { p_CT_ExtensionList }?
1049 p_ST_SlideLayoutId = xsd:unsignedInt { minInclusive = "2147483648" }
1050 p_CT_SlideLayoutIdListEntry =
1051     attribute id { p_ST_SlideLayoutId }?,
1052     r_id,
1053     element extLst { p_CT_ExtensionList }?
1054 p_CT_SlideLayoutIdList =
1055     element sldLayoutId { p_CT_SlideLayoutIdListEntry }*
1056 p_CT_SlideMaster =
1057
1058 ## default value: false
1059 attribute preserve { xsd:boolean }?,
1060 element cSld { p_CT_CommonSlideData },
1061 p_EG_TopLevelSlide,
1062 element sldLayoutIdList { p_CT_SlideLayoutIdList }?,
1063 element transition { p_CT_SlideTransition }?,
1064 element timing { p_CT_SlideTiming }?,
1065 element hf { p_CT_HeaderFooter }?,
1066 element txStyles { p_CT_SlideMasterTextStyles }?,
1067 element extLst { p_CT_ExtensionListModify }?
1068 p_sldMaster = element sldMaster { p_CT_SlideMaster }
1069 p_CT_HandoutMaster =
1070     element cSld { p_CT_CommonSlideData },
1071     p_EG_TopLevelSlide,
1072     element hf { p_CT_HeaderFooter }?,
1073     element extLst { p_CT_ExtensionListModify }?
1074 p_handoutMaster = element handoutMaster { p_CT_HandoutMaster }
1075 p_CT_NotesMaster =
1076     element cSld { p_CT_CommonSlideData },
1077     p_EG_TopLevelSlide,
1078     element hf { p_CT_HeaderFooter }?,
1079     element notesStyle { a_CT_TextListStyle }?,
1080     element extLst { p_CT_ExtensionListModify }?
1081 p_notesMaster = element notesMaster { p_CT_NotesMaster }
1082 p_CT_NotesSlide =

```

```

1083   p_AG_ChildSlide,
1084   element cSld { p_CT_CommonSlideData },
1085   p_EG_ChildSlide?,
1086   element extLst { p_CT_ExtensionListModify }?
1087 p_notes = element notes { p_CT_NotesSlide }
1088 p_CT_SlideSyncProperties =
1089   attribute serverSldId { xsd:string },
1090   attribute serverSldModifiedTime { xsd:dateTime },
1091   attribute clientInsertedTime { xsd:dateTime },
1092   element extLst { p_CT_ExtensionList }?
1093 p_sldSyncPr = element sldSyncPr { p_CT_SlideSyncProperties }
1094 p_CT_StringTag =
1095   attribute name { xsd:string },
1096   attribute val { xsd:string }
1097 p_CT_TagList = element tag { p_CT_StringTag }*
1098 p_tagLst = element tagLst { p_CT_TagList }
1099 p_ST_SplitterBarState = "minimized" | "restored" | "maximized"
1100 p_ST_ViewType =
1101   "sldView"
1102   | "sldMasterView"
1103   | "notesView"
1104   | "handoutView"
1105   | "notesMasterView"
1106   | "outlineView"
1107   | "sldSorterView"
1108   | "sldThumbnailView"
1109 p_CT_NormalViewPortion =
1110   attribute sz { a_ST_PositiveFixedPercentage },
1111
1112   ## default value: true
1113   attribute autoAdjust { xsd:boolean }?
1114 p_CT_NormalViewProperties =
1115
1116   ## default value: true
1117   attribute showOutlineIcons { xsd:boolean }?,
1118
1119   ## default value: false
1120   attribute snapVertSplitter { xsd:boolean }?,
1121
1122   ## default value: restored
1123   attribute vertBarState { p_ST_SplitterBarState }?,
1124
1125   ## default value: restored
1126   attribute horzBarState { p_ST_SplitterBarState }?,
1127
1128   ## default value: false
1129   attribute preferSingleView { xsd:boolean }?,
1130   element restoredLeft { p_CT_NormalViewPortion },
1131   element restoredTop { p_CT_NormalViewPortion },
1132   element extLst { p_CT_ExtensionList }?
1133 p_CT_CommonViewProperties =
1134
1135   ## default value: false

```

```

1136     attribute varScale { xsd:boolean }?,
1137     element scale { a_CT_Scale2D },
1138     element origin { a_CT_Point2D }
1139 p_CT_NotesTextViewProperties =
1140     element cViewPr { p_CT_CommonViewProperties },
1141     element extLst { p_CT_ExtensionList }?
1142 p_CT_OutlineViewSlideEntry =
1143     r_id,
1144
1145     ## default value: false
1146     attribute collapse { xsd:boolean }?
1147 p_CT_OutlineViewSlideList = element sld { p_CT_OutlineViewSlideEntry }*
1148 p_CT_OutlineViewProperties =
1149     element cViewPr { p_CT_CommonViewProperties },
1150     element sldLst { p_CT_OutlineViewSlideList }?,
1151     element extLst { p_CT_ExtensionList }?
1152 p_CT_SlideSorterViewProperties =
1153
1154     ## default value: true
1155     attribute showFormatting { xsd:boolean }?,
1156     element cViewPr { p_CT_CommonViewProperties },
1157     element extLst { p_CT_ExtensionList }?
1158 p_CT_Guide =
1159
1160     ## default value: vert
1161     attribute orient { p_ST_Direction }?,
1162
1163     ## default value: 0
1164     attribute pos { a_ST_Coordinate32 }?
1165 p_CT_GuideList = element guide { p_CT_Guide }*
1166 p_CT_CommonSlideViewProperties =
1167
1168     ## default value: true
1169     attribute snapToGrid { xsd:boolean }?,
1170
1171     ## default value: false
1172     attribute snapToObjects { xsd:boolean }?,
1173
1174     ## default value: false
1175     attribute showGuides { xsd:boolean }?,
1176     element cViewPr { p_CT_CommonViewProperties },
1177     element guideLst { p_CT_GuideList }?
1178 p_CT_SlideViewProperties =
1179     element cSldViewPr { p_CT_CommonSlideViewProperties },
1180     element extLst { p_CT_ExtensionList }?
1181 p_CT_NotesViewProperties =
1182     element cSldViewPr { p_CT_CommonSlideViewProperties },
1183     element extLst { p_CT_ExtensionList }?
1184 p_CT_ViewProperties =
1185
1186     ## default value: sldView
1187     attribute lastView { p_ST_ViewType }?,
1188

```

```

1189  ## default value: true
1190  attribute showComments { xsd:boolean }?,
1191  (element normalViewPr { p_CT_NormalViewProperties }?,
1192   element slideViewPr { p_CT_SlideViewProperties }?,
1193   element outlineViewPr { p_CT_OutlineViewProperties }?,
1194   element notesTextViewPr { p_CT_NotesTextViewProperties }?,
1195   element sorterViewPr { p_CT_SlideSorterViewProperties }?,
1196   element notesViewPr { p_CT_NotesViewProperties }?,
1197   element gridSpacing { a_CT_PositiveSize2D }?,
1198   element extLst { p_CT_ExtensionList }?)?
1199  p_viewPr = element viewPr { p_CT_ViewProperties }

```

B.3.1 Part Schemas

B.3.1.1 Comment Authors Part

This schema is available in the file PresentationML_Comment_Authors.rnc.

```

1  include "pml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-main.rnc"
4  include "dml-diagram.rnc"
5  include "shared-commonSimpleTypes.rnc"
6  include "dml-lockedCanvas.rnc"
7  include "any.rnc"
8  include "dml-chart.rnc"
9  include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_cmAuthorLst

```

B.3.1.2 Comments Part

This schema is available in the file PresentationML_Comments.rnc.

```

1  include "pml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-main.rnc"
4  include "dml-diagram.rnc"
5  include "shared-commonSimpleTypes.rnc"
6  include "dml-lockedCanvas.rnc"
7  include "any.rnc"
8  include "dml-chart.rnc"
9  include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_cmLst

```

B.3.1.3 Handout Master Part

This schema is available in the file PresentationML_Handout_Master.rnc.

```

1  include "pml.rnc"
2  include "shared-relationshipReference.rnc"
3  include "dml-main.rnc"
4  include "dml-diagram.rnc"

```

```

5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_handoutMaster

```

B.3.1.4 Notes Master Part

This schema is available in the file PresentationML_Notes_Master.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_notesMaster

```

B.3.1.5 Notes Slide Part

This schema is available in the file PresentationML_Notes_Slide.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_notes

```

B.3.1.6 Presentation Part

This schema is available in the file PresentationML_Presentation.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"

```

```
11 start = p_presentation
```

B.3.1.7 Presentation Properties Part

This schema is available in the file PresentationML_Presentation_Properties.rnc.

```
1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_presentationPr
```

B.3.1.8 Slide Part

This schema is available in the file PresentationML_Slide.rnc.

```
1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_sld
```

B.3.1.9 Slide Layout Part

This schema is available in the file PresentationML_Slide_Layout.rnc.

```
1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_sldLayout
```

B.3.1.10 Slide Master Part

This schema is available in the file PresentationML_Slide_Master.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_sldMaster

```

B.3.1.11 Slide Synchronization Data Part

This schema is available in the file PresentationML_Slide_Synchronization_Data.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_sldSyncPr

```

B.3.1.12 User Defined Tags Part

This schema is available in the file PresentationML_User-Defined_Tags.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_tagLst

```

B.3.1.13 View Properties Part

This schema is available in the file PresentationML_View_Properties.rnc.

```

1 include "pml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"

```

```

7 include "any.rnc"
8 include "dml-chart.rnc"
9 include "dml-chartDrawing.rnc"
10 include "dml-picture.rnc"
11 start = p_viewPr

```

B.4 DrawingML - Framework

B.4.1 DrawingML - Main

This schema is available in the file dml-main.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/main"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace o = "urn:schemas-microsoft-com:office:office"
5 namespace r =
6   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
7 namespace s =
8   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
9 namespace v = "urn:schemas-microsoft-com:vml"
10 namespace w10 = "urn:schemas-microsoft-com:office:word"
11 namespace x = "urn:schemas-microsoft-com:office:excel"
12
13 a_CT_AudioFile =
14   r_link,
15   attribute contentType { xsd:string }?,
16   element extLst { a_CT_OfficeArtExtensionList }?
17 a_CT_VideoFile =
18   r_link,
19   attribute contentType { xsd:string }?,
20   element extLst { a_CT_OfficeArtExtensionList }?
21 a_CT_QuickTimeFile =
22   r_link,
23   element extLst { a_CT_OfficeArtExtensionList }?
24 a_CT_AudioCDTime =
25   attribute track { xsd:unsignedByte },
26
27   ## default value: 0
28   attribute time { xsd:unsignedInt }?
29 a_CT_AudioCD =
30   element st { a_CT_AudioCDTime },
31   element end { a_CT_AudioCDTime },
32   element extLst { a_CT_OfficeArtExtensionList }?
33 a_EG_Media =
34   element audioCd { a_CT_AudioCD }
35   | element wavAudioFile { a_CT_EmbeddedWAVAudioFile }
36   | element audioFile { a_CT_AudioFile }
37   | element videoFile { a_CT_VideoFile }
38   | element quickTimeFile { a_CT_QuickTimeFile }
39 a_videoFile = element videoFile { a_CT_VideoFile }
40 a_ST_StyleMatrixColumnIndex = xsd:unsignedInt
41 a_ST_FontCollectionIndex = "major" | "minor" | "none"

```



```

42 a_ST_ColorSchemeIndex =
43     "dk1"
44     | "lt1"
45     | "dk2"
46     | "lt2"
47     | "accent1"
48     | "accent2"
49     | "accent3"
50     | "accent4"
51     | "accent5"
52     | "accent6"
53     | "hlink"
54     | "folHlink"
55 a_CT_ColorScheme =
56     attribute name { xsd:string },
57     element dk1 { a_CT_Color },
58     element lt1 { a_CT_Color },
59     element dk2 { a_CT_Color },
60     element lt2 { a_CT_Color },
61     element accent1 { a_CT_Color },
62     element accent2 { a_CT_Color },
63     element accent3 { a_CT_Color },
64     element accent4 { a_CT_Color },
65     element accent5 { a_CT_Color },
66     element accent6 { a_CT_Color },
67     element hlink { a_CT_Color },
68     element folHlink { a_CT_Color },
69     element extLst { a_CT_OfficeArtExtensionList }?
70 a_CT_CustomColor =
71     attribute name { xsd:string }?,
72     a_EG_ColorChoice
73 a_CT_SupplementalFont =
74     attribute script { xsd:string },
75     attribute typeface { a_ST_TextTypeface }
76 a_CT_CustomColorList = element custClr { a_CT_CustomColor }*
77 a_CT_FontCollection =
78     element latin { a_CT_TextFont },
79     element ea { a_CT_TextFont },
80     element cs { a_CT_TextFont },
81     element font { a_CT_SupplementalFont }*,
82     element extLst { a_CT_OfficeArtExtensionList }?
83 a_CT_EffectStyleItem =
84     a_EG_EffectProperties,
85     element scene3d { a_CT_Scene3D }?,
86     element sp3d { a_CT_Shape3D }?
87 a_CT_FontScheme =
88     attribute name { xsd:string },
89     element majorFont { a_CT_FontCollection },
90     element minorFont { a_CT_FontCollection },
91     element extLst { a_CT_OfficeArtExtensionList }?
92 a_CT_FillStyleList = a_EG_FillProperties+
93 a_CT_LineStyleList = element ln { a_CT_LineProperties }+
94 a_CT_EffectStyleList = element effectStyle { a_CT_EffectStyleItem }+

```

```

95 a_CT_BackgroundFillStyleList = a_EG_FillProperties+
96 a_CT_StyleMatrix =
97   attribute name { xsd:string }?,
98   element fillStyleLst { a_CT_FillStyleList },
99   element lnStyleLst { a_CT_LineStyleList },
100  element effectStyleLst { a_CT_EffectStyleList },
101  element bgFillStyleLst { a_CT_BackgroundFillStyleList }
102 a_CT_BaseStyles =
103   element clrScheme { a_CT_ColorScheme },
104   element fontScheme { a_CT_FontScheme },
105   element fmtScheme { a_CT_StyleMatrix },
106   element extLst { a_CT_OfficeArtExtensionList }?
107 a_CT_OfficeArtExtension =
108   attribute uri { xsd:token },
109   a_CT_OfficeArtExtension_any*
110 a_CT_OfficeArtExtension_any =
111   element * - (o:* | v:* | w10:* | x:*) {
112     anyAttribute*,
113     mixed { anyElement* }
114   }
115 a_ST_Coordinate = a_ST_CoordinateUnqualified | s_ST_UniversalMeasure
116 a_ST_CoordinateUnqualified =
117   xsd:long {
118     minInclusive = "-27273042329600"
119     maxInclusive = "27273042316900"
120   }
121 a_ST_Coordinate32 = a_ST_Coordinate32Unqualified | s_ST_UniversalMeasure
122 a_ST_Coordinate32Unqualified = xsd:int
123 a_ST_PositiveCoordinate =
124   xsd:long { minInclusive = "0" maxInclusive = "27273042316900" }
125 a_ST_PositiveCoordinate32 = xsd:int { minInclusive = "0" }
126 a_ST_Angle = xsd:int
127 a_CT_Angle = attribute val { a_ST_Angle }
128 a_ST_FixedAngle =
129   xsd:int { minExclusive = "-5400000" maxExclusive = "5400000" }
130 a_ST_PositiveFixedAngle =
131   xsd:int { minInclusive = "0" maxExclusive = "21600000" }
132 a_CT_PositiveFixedAngle = attribute val { a_ST_PositiveFixedAngle }
133 a_ST_Percentage = a_ST_PercentageDecimal | s_ST_Percentage
134 a_ST_PercentageDecimal = xsd:int
135 a_CT_Percentage = attribute val { a_ST_Percentage }
136 a_ST_PositivePercentage =
137   a_ST_PositivePercentageDecimal | s_ST_PositivePercentage
138 a_ST_PositivePercentageDecimal = xsd:int { minInclusive = "0" }
139 a_CT_PositivePercentage = attribute val { a_ST_PositivePercentage }
140 a_ST_FixedPercentage =
141   a_ST_FixedPercentageDecimal | s_ST_FixedPercentage
142 a_ST_FixedPercentageDecimal =
143   xsd:int { minInclusive = "-100000" maxInclusive = "100000" }
144 a_CT_FixedPercentage = attribute val { a_ST_FixedPercentage }
145 a_ST_PositiveFixedPercentage =
146   a_ST_PositiveFixedPercentageDecimal | s_ST_PositiveFixedPercentage
147 a_ST_PositiveFixedPercentageDecimal =

```

```

148     xsd:int { minInclusive = "0" maxInclusive = "100000" }
149 a_CT_PositiveFixedPercentage =
150     attribute val { a_ST_PositiveFixedPercentage }
151 a_CT_Ratio =
152     attribute n { xsd:long },
153     attribute d { xsd:long }
154 a_CT_Point2D =
155     attribute x { a_ST_Coordinate },
156     attribute y { a_ST_Coordinate }
157 a_CT_PositiveSize2D =
158     attribute cx { a_ST_PositiveCoordinate },
159     attribute cy { a_ST_PositiveCoordinate }
160 a_CT_ComplementTransform = empty
161 a_CT_InverseTransform = empty
162 a_CT_GrayscaleTransform = empty
163 a_CT_GammaTransform = empty
164 a_CT_InverseGammaTransform = empty
165 a_EG_ColorTransform =
166     element tint { a_CT_PositiveFixedPercentage }
167     | element shade { a_CT_PositiveFixedPercentage }
168     | element comp { a_CT_ComplementTransform }
169     | element inv { a_CT_InverseTransform }
170     | element gray { a_CT_GrayscaleTransform }
171     | element alpha { a_CT_PositiveFixedPercentage }
172     | element alphaOff { a_CT_FixedPercentage }
173     | element alphaMod { a_CT_PositivePercentage }
174     | element hue { a_CT_PositiveFixedAngle }
175     | element hueOff { a_CT_Angle }
176     | element hueMod { a_CT_PositivePercentage }
177     | element sat { a_CT_Percentage }
178     | element satOff { a_CT_Percentage }
179     | element satMod { a_CT_Percentage }
180     | element lum { a_CT_Percentage }
181     | element lumOff { a_CT_Percentage }
182     | element lumMod { a_CT_Percentage }
183     | element red { a_CT_Percentage }
184     | element redOff { a_CT_Percentage }
185     | element redMod { a_CT_Percentage }
186     | element green { a_CT_Percentage }
187     | element greenOff { a_CT_Percentage }
188     | element greenMod { a_CT_Percentage }
189     | element blue { a_CT_Percentage }
190     | element blueOff { a_CT_Percentage }
191     | element blueMod { a_CT_Percentage }
192     | element gamma { a_CT_GammaTransform }
193     | element invGamma { a_CT_InverseGammaTransform }
194 a_CT_ScRgbColor =
195     attribute r { a_ST_Percentage },
196     attribute g { a_ST_Percentage },
197     attribute b { a_ST_Percentage },
198     a_EG_ColorTransform*
199 a_CT_SRgbColor =
200     attribute val { s_ST_HexColorRGB },

```

```

201   a_EG_ColorTransform*
202   a_CT_HslColor =
203     attribute hue { a_ST_PositiveFixedAngle },
204     attribute sat { a_ST_Percentage },
205     attribute lum { a_ST_Percentage },
206     a_EG_ColorTransform*
207   a_ST_SystemColorVal =
208     "scrollBar"
209     | "background"
210     | "activeCaption"
211     | "inactiveCaption"
212     | "menu"
213     | "window"
214     | "windowFrame"
215     | "menuText"
216     | "windowText"
217     | "captionText"
218     | "activeBorder"
219     | "inactiveBorder"
220     | "appWorkspace"
221     | "highlight"
222     | "highlightText"
223     | "btnFace"
224     | "btnShadow"
225     | "grayText"
226     | "btnText"
227     | "inactiveCaptionText"
228     | "btnHighlight"
229     | "3dDkShadow"
230     | "3dLight"
231     | "infoText"
232     | "infoBk"
233     | "hotLight"
234     | "gradientActiveCaption"
235     | "gradientInactiveCaption"
236     | "menuHighlight"
237     | "menuBar"
238   a_CT_SystemColor =
239     attribute val { a_ST_SystemColorVal },
240     attribute lastClr { s_ST_HexColorRGB }?,
241     a_EG_ColorTransform*
242   a_ST_SchemeColorVal =
243     "bg1"
244     | "tx1"
245     | "bg2"
246     | "tx2"
247     | "accent1"
248     | "accent2"
249     | "accent3"
250     | "accent4"
251     | "accent5"
252     | "accent6"
253     | "hlink"

```

```

254 | "folHlink"
255 | "phClr"
256 | "dk1"
257 | "lt1"
258 | "dk2"
259 | "lt2"
260 a_CT_SchemeColor =
261   attribute val { a_ST_SchemeColorVal },
262   a_EG_ColorTransform*
263 a_ST_PresetColorVal =
264   "aliceBlue"
265   | "antiqueWhite"
266   | "aqua"
267   | "aquamarine"
268   | "azure"
269   | "beige"
270   | "bisque"
271   | "black"
272   | "blanchedAlmond"
273   | "blue"
274   | "blueViolet"
275   | "brown"
276   | "burlyWood"
277   | "cadetBlue"
278   | "chartreuse"
279   | "chocolate"
280   | "coral"
281   | "cornflowerBlue"
282   | "cornsilk"
283   | "crimson"
284   | "cyan"
285   | "darkBlue"
286   | "darkCyan"
287   | "darkGoldenrod"
288   | "darkGray"
289   | "darkGrey"
290   | "darkGreen"
291   | "darkKhaki"
292   | "darkMagenta"
293   | "darkOliveGreen"
294   | "darkOrange"
295   | "darkOrchid"
296   | "darkRed"
297   | "darkSalmon"
298   | "darkSeaGreen"
299   | "darkSlateBlue"
300   | "darkSlateGray"
301   | "darkSlateGrey"
302   | "darkTurquoise"
303   | "darkViolet"
304   | "dkBlue"
305   | "dkCyan"
306   | "dkGoldenrod"

```

307	"dkGray"
308	"dkGrey"
309	"dkGreen"
310	"dkKhaki"
311	"dkMagenta"
312	"dkOliveGreen"
313	"dkOrange"
314	"dkOrchid"
315	"dkRed"
316	"dkSalmon"
317	"dkSeaGreen"
318	"dkSlateBlue"
319	"dkSlateGray"
320	"dkSlateGrey"
321	"dkTurquoise"
322	"dkViolet"
323	"deepPink"
324	"deepSkyBlue"
325	"dimGray"
326	"dimGrey"
327	"dodgerBlue"
328	"firebrick"
329	"floralWhite"
330	"forestGreen"
331	"fuchsia"
332	"gainsboro"
333	"ghostWhite"
334	"gold"
335	"goldenrod"
336	"gray"
337	"grey"
338	"green"
339	"greenYellow"
340	"honeydew"
341	"hotPink"
342	"indianRed"
343	"indigo"
344	"ivory"
345	"khaki"
346	"lavender"
347	"lavenderBlush"
348	"lawnGreen"
349	"lemonChiffon"
350	"lightBlue"
351	"lightCoral"
352	"lightCyan"
353	"lightGoldenrodYellow"
354	"lightGray"
355	"lightGrey"
356	"lightGreen"
357	"lightPink"
358	"lightSalmon"
359	"lightSeaGreen"

```

360 | "lightSkyBlue"
361 | "lightSlateGray"
362 | "lightSlateGrey"
363 | "lightSteelBlue"
364 | "lightYellow"
365 | "ltBlue"
366 | "ltCoral"
367 | "ltCyan"
368 | "ltGoldenrodYellow"
369 | "ltGray"
370 | "ltGrey"
371 | "ltGreen"
372 | "ltPink"
373 | "ltSalmon"
374 | "ltSeaGreen"
375 | "ltSkyBlue"
376 | "ltSlateGray"
377 | "ltSlateGrey"
378 | "ltSteelBlue"
379 | "ltYellow"
380 | "lime"
381 | "limeGreen"
382 | "linen"
383 | "magenta"
384 | "maroon"
385 | "medAquamarine"
386 | "medBlue"
387 | "medOrchid"
388 | "medPurple"
389 | "medSeaGreen"
390 | "medSlateBlue"
391 | "medSpringGreen"
392 | "medTurquoise"
393 | "medVioletRed"
394 | "mediumAquamarine"
395 | "mediumBlue"
396 | "mediumOrchid"
397 | "mediumPurple"
398 | "mediumSeaGreen"
399 | "mediumSlateBlue"
400 | "mediumSpringGreen"
401 | "mediumTurquoise"
402 | "mediumVioletRed"
403 | "midnightBlue"
404 | "mintCream"
405 | "mistyRose"
406 | "moccasin"
407 | "navajoWhite"
408 | "navy"
409 | "oldLace"
410 | "olive"
411 | "oliveDrab"
412 | "orange"

```

```

413 | "orangeRed"
414 | "orchid"
415 | "paleGoldenrod"
416 | "paleGreen"
417 | "paleTurquoise"
418 | "paleVioletRed"
419 | "papayaWhip"
420 | "peachPuff"
421 | "peru"
422 | "pink"
423 | "plum"
424 | "powderBlue"
425 | "purple"
426 | "red"
427 | "rosyBrown"
428 | "royalBlue"
429 | "saddleBrown"
430 | "salmon"
431 | "sandyBrown"
432 | "seaGreen"
433 | "seaShell"
434 | "sienna"
435 | "silver"
436 | "skyBlue"
437 | "slateBlue"
438 | "slateGray"
439 | "slateGrey"
440 | "snow"
441 | "springGreen"
442 | "steelBlue"
443 | "tan"
444 | "teal"
445 | "thistle"
446 | "tomato"
447 | "turquoise"
448 | "violet"
449 | "wheat"
450 | "white"
451 | "whiteSmoke"
452 | "yellow"
453 | "yellowGreen"
454 a_CT_PresetColor =
455   attribute val { a_ST_PresetColorVal },
456   a_EG_ColorTransform*
457 a_EG_OfficeArtExtensionList = element ext { a_CT_OfficeArtExtension }*
458 a_CT_OfficeArtExtensionList = a_EG_OfficeArtExtensionList
459 a_CT_Scale2D =
460   element sx { a_CT_Ratio },
461   element sy { a_CT_Ratio }
462 a_CT_Transform2D =
463
464   ## default value: 0
465   attribute rot { a_ST_Angle }?,

```



```

466
467     ## default value: false
468     attribute flipH { xsd:boolean }?,
469
470     ## default value: false
471     attribute flipV { xsd:boolean }?,
472     element off { a_CT_Point2D }?,
473     element ext { a_CT_PositiveSize2D }?
474 a_CT_GroupTransform2D =
475
476     ## default value: 0
477     attribute rot { a_ST_Angle }?,
478
479     ## default value: false
480     attribute flipH { xsd:boolean }?,
481
482     ## default value: false
483     attribute flipV { xsd:boolean }?,
484     element off { a_CT_Point2D }?,
485     element ext { a_CT_PositiveSize2D }?,
486     element chOff { a_CT_Point2D }?,
487     element chExt { a_CT_PositiveSize2D }?
488 a_CT_Point3D =
489     attribute x { a_ST_Coordinate },
490     attribute y { a_ST_Coordinate },
491     attribute z { a_ST_Coordinate }
492 a_CT_Vector3D =
493     attribute dx { a_ST_Coordinate },
494     attribute dy { a_ST_Coordinate },
495     attribute dz { a_ST_Coordinate }
496 a_CT_SphereCoords =
497     attribute lat { a_ST_PositiveFixedAngle },
498     attribute lon { a_ST_PositiveFixedAngle },
499     attribute rev { a_ST_PositiveFixedAngle }
500 a_CT_RelativeRect =
501
502     ## default value: 0%
503     attribute l { a_ST_Percentage }?,
504
505     ## default value: 0%
506     attribute t { a_ST_Percentage }?,
507
508     ## default value: 0%
509     attribute r { a_ST_Percentage }?,
510
511     ## default value: 0%
512     attribute b { a_ST_Percentage }?
513 a_ST_RectAlignment =
514     "tl" | "t" | "tr" | "l" | "ctr" | "r" | "bl" | "b" | "br"
515 a_EG_ColorChoice =
516     element scrGbClr { a_CT_ScRgbColor }
517     | element srGbClr { a_CT_SRgbColor }
518     | element hslClr { a_CT_HslColor }

```

```

519 | element sysClr { a_CT_SystemColor }
520 | element schemeClr { a_CT_SchemeColor }
521 | element prstClr { a_CT_PresetColor }
522 a_CT_Color = a_EG_ColorChoice
523 a_CT_ColorMRU = a_EG_ColorChoice*
524 a_ST_BlackWhiteMode =
525     "clr"
526     | "auto"
527     | "gray"
528     | "ltGray"
529     | "invGray"
530     | "grayWhite"
531     | "blackGray"
532     | "blackWhite"
533     | "black"
534     | "white"
535     | "hidden"
536 a_AG_Blob = r_embed?, r_link?
537 a_CT_EmbeddedWAVAudioFile =
538     r_embed,
539     attribute name { xsd:string }?
540 a_CT_Hyperlink =
541     r_id?,
542     attribute invalidUrl { xsd:string }?,
543     attribute action { xsd:string }?,
544     attribute tgtFrame { xsd:string }?,
545     attribute tooltip { xsd:string }?,
546
547     ## default value: true
548     attribute history { xsd:boolean }?,
549
550     ## default value: false
551     attribute highlightClick { xsd:boolean }?,
552
553     ## default value: false
554     attribute endSnd { xsd:boolean }?,
555     element snd { a_CT_EmbeddedWAVAudioFile }?,
556     element extLst { a_CT_OfficeArtExtensionList }?
557 a_ST_DrawingElementId = xsd:unsignedInt
558 a_AG_Locking =
559
560     ## default value: false
561     attribute noGrp { xsd:boolean }?,
562
563     ## default value: false
564     attribute noSelect { xsd:boolean }?,
565
566     ## default value: false
567     attribute noRot { xsd:boolean }?,
568
569     ## default value: false
570     attribute noChangeAspect { xsd:boolean }?,
571

```

```

572 ## default value: false
573 attribute noMove { xsd:boolean }?,
574
575 ## default value: false
576 attribute noResize { xsd:boolean }?,
577
578 ## default value: false
579 attribute noEditPoints { xsd:boolean }?,
580
581 ## default value: false
582 attribute noAdjustHandles { xsd:boolean }?,
583
584 ## default value: false
585 attribute noChangeArrowheads { xsd:boolean }?,
586
587 ## default value: false
588 attribute noChangeShapeType { xsd:boolean }?
589 a_CT_ConnectorLocking =
590   a_AG_Locking,
591   element extLst { a_CT_OfficeArtExtensionList }?
592 a_CT_ShapeLocking =
593   a_AG_Locking,
594
595 ## default value: false
596 attribute noTextEdit { xsd:boolean }?,
597   element extLst { a_CT_OfficeArtExtensionList }?
598 a_CT_PictureLocking =
599   a_AG_Locking,
600
601 ## default value: false
602 attribute noCrop { xsd:boolean }?,
603   element extLst { a_CT_OfficeArtExtensionList }?
604 a_CT_GroupLocking =
605
606 ## default value: false
607 attribute noGrp { xsd:boolean }?,
608
609 ## default value: false
610 attribute noUngroup { xsd:boolean }?,
611
612 ## default value: false
613 attribute noSelect { xsd:boolean }?,
614
615 ## default value: false
616 attribute noRot { xsd:boolean }?,
617
618 ## default value: false
619 attribute noChangeAspect { xsd:boolean }?,
620
621 ## default value: false
622 attribute noMove { xsd:boolean }?,
623
624 ## default value: false

```

```

625     attribute noResize { xsd:boolean }?,
626     element extLst { a_CT_OfficeArtExtensionList }?
627 a_CT_GraphicalObjectFrameLocking =
628
629     ## default value: false
630     attribute noGrp { xsd:boolean }?,
631
632     ## default value: false
633     attribute noDrilldown { xsd:boolean }?,
634
635     ## default value: false
636     attribute noSelect { xsd:boolean }?,
637
638     ## default value: false
639     attribute noChangeAspect { xsd:boolean }?,
640
641     ## default value: false
642     attribute noMove { xsd:boolean }?,
643
644     ## default value: false
645     attribute noResize { xsd:boolean }?,
646     element extLst { a_CT_OfficeArtExtensionList }?
647 a_CT_ContentPartLocking =
648     a_AG_Locking,
649     element extLst { a_CT_OfficeArtExtensionList }?
650 a_CT_NonVisualDrawingProps =
651     attribute id { a_ST_DrawingElementId },
652     attribute name { xsd:string },
653     attribute descr { xsd:string }?,
654
655     ## default value: false
656     attribute hidden { xsd:boolean }?,
657     attribute title { xsd:string }?,
658     element hlinkClick { a_CT_Hyperlink }?,
659     element hlinkHover { a_CT_Hyperlink }?,
660     element extLst { a_CT_OfficeArtExtensionList }?
661 a_CT_NonVisualDrawingShapeProps =
662
663     ## default value: false
664     attribute txBox { xsd:boolean }?,
665     element spLocks { a_CT_ShapeLocking }?,
666     element extLst { a_CT_OfficeArtExtensionList }?
667 a_CT_NonVisualConnectorProperties =
668     element cxnSpLocks { a_CT_ConnectorLocking }?,
669     element stCxn { a_CT_Connection }?,
670     element endCxn { a_CT_Connection }?,
671     element extLst { a_CT_OfficeArtExtensionList }?
672 a_CT_NonVisualPictureProperties =
673
674     ## default value: true
675     attribute preferRelativeResize { xsd:boolean }?,
676     element picLocks { a_CT_PictureLocking }?,
677     element extLst { a_CT_OfficeArtExtensionList }?

```

```

678 a_CT_NonVisualGroupDrawingShapeProps =
679     element grpSpLocks { a_CT_GroupLocking }?,
680     element extLst { a_CT_OfficeArtExtensionList }?
681 a_CT_NonVisualGraphicFrameProperties =
682     element graphicFrameLocks { a_CT_GraphicalObjectFrameLocking }?,
683     element extLst { a_CT_OfficeArtExtensionList }?
684 a_CT_NonVisualContentPartProperties =
685
686     ## default value: true
687     attribute isComment { xsd:boolean }?,
688     element cpLocks { a_CT_ContentPartLocking }?,
689     element extLst { a_CT_OfficeArtExtensionList }?
690 a_CT_GraphicalObjectData =
691     attribute uri { xsd:token },
692     a_CT_GraphicalObjectData_any*
693 a_CT_GraphicalObjectData_any =
694     element * - (o:* | v:* | w10:* | x:*) {
695         anyAttribute*,
696         mixed { anyElement* }
697     }
698 a_CT_GraphicalObject = element graphicData { a_CT_GraphicalObjectData }
699 a_graphic = element graphic { a_CT_GraphicalObject }
700 a_ST_ChartBuildStep =
701     "category"
702     | "ptInCategory"
703     | "series"
704     | "ptInSeries"
705     | "allPts"
706     | "gridLegend"
707 a_ST_DgmBuildStep = "sp" | "bg"
708 a_CT_AnimationDgmElement =
709
710     ## default value: {00000000-0000-0000-0000-000000000000}
711     attribute id { s_ST_Guid }?,
712
713     ## default value: sp
714     attribute bldStep { a_ST_DgmBuildStep }?
715 a_CT_AnimationChartElement =
716
717     ## default value: -1
718     attribute seriesIdx { xsd:int }?,
719
720     ## default value: -1
721     attribute categoryIdx { xsd:int }?,
722     attribute bldStep { a_ST_ChartBuildStep }
723 a_CT_AnimationElementChoice =
724     element dgm { a_CT_AnimationDgmElement }
725     | element chart { a_CT_AnimationChartElement }
726 a_ST_AnimationBuildType = "allAtOnce"
727 a_ST_AnimationDgmOnlyBuildType = "one" | "lvlOne" | "lvlAtOnce"
728 a_ST_AnimationDgmBuildType =
729     a_ST_AnimationBuildType | a_ST_AnimationDgmOnlyBuildType
730 a_CT_AnimationDgmBuildProperties =

```

```

731
732   ## default value: allAtOnce
733   attribute bld { a_ST_AnimationDgmBuildType }?,
734
735   ## default value: false
736   attribute rev { xsd:boolean }?
737 a_ST_AnimationChartOnlyBuildType =
738   "series" | "category" | "seriesEl" | "categoryEl"
739 a_ST_AnimationChartBuildType =
740   a_ST_AnimationBuildType | a_ST_AnimationChartOnlyBuildType
741 a_CT_AnimationChartBuildProperties =
742
743   ## default value: allAtOnce
744   attribute bld { a_ST_AnimationChartBuildType }?,
745
746   ## default value: true
747   attribute animBg { xsd:boolean }?
748 a_CT_AnimationGraphicalObjectBuildProperties =
749   element bldDgm { a_CT_AnimationDgmBuildProperties }
750   | element bldChart { a_CT_AnimationChartBuildProperties }
751 a_CT_BackgroundFormatting = a_EG_FillProperties?, a_EG_EffectProperties?
752 a_CT_WholeE2oFormatting =
753   element ln { a_CT_LineProperties }?,
754   a_EG_EffectProperties?
755 a_CT_GvmlUseShapeRectangle = empty
756 a_CT_GvmlTextShape =
757   element txBody { a_CT_TextBody },
758   (element useSpRect { a_CT_GvmlUseShapeRectangle }
759   | element xfrm { a_CT_Transform2D } ),
760   element extLst { a_CT_OfficeArtExtensionList }?
761 a_CT_GvmlShapeNonVisual =
762   element cNvPr { a_CT_NonVisualDrawingProps },
763   element cNvSpPr { a_CT_NonVisualDrawingShapeProps }
764 a_CT_GvmlShape =
765   element nvSpPr { a_CT_GvmlShapeNonVisual },
766   element spPr { a_CT_ShapeProperties },
767   element txSp { a_CT_GvmlTextShape }?,
768   element style { a_CT_ShapeStyle }?,
769   element extLst { a_CT_OfficeArtExtensionList }?
770 a_CT_GvmlConnectorNonVisual =
771   element cNvPr { a_CT_NonVisualDrawingProps },
772   element cNvCxnSpPr { a_CT_NonVisualConnectorProperties }
773 a_CT_GvmlConnector =
774   element nvCxnSpPr { a_CT_GvmlConnectorNonVisual },
775   element spPr { a_CT_ShapeProperties },
776   element style { a_CT_ShapeStyle }?,
777   element extLst { a_CT_OfficeArtExtensionList }?
778 a_CT_GvmlPictureNonVisual =
779   element cNvPr { a_CT_NonVisualDrawingProps },
780   element cNvPicPr { a_CT_NonVisualPictureProperties }
781 a_CT_GvmlPicture =
782   element nvPicPr { a_CT_GvmlPictureNonVisual },
783   element blipFill { a_CT_BlipFillProperties },

```

```

784     element spPr { a_CT_ShapeProperties },
785     element style { a_CT_ShapeStyle }?,
786     element extLst { a_CT_OfficeArtExtensionList }?
787 a_CT_GvmlGraphicFrameNonVisual =
788     element cNvPr { a_CT_NonVisualDrawingProps },
789     element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }
790 a_CT_GvmlGraphicalObjectFrame =
791     element nvGraphicFramePr { a_CT_GvmlGraphicFrameNonVisual },
792     a_graphic,
793     element xfrm { a_CT_Transform2D },
794     element extLst { a_CT_OfficeArtExtensionList }?
795 a_CT_GvmlGroupShapeNonVisual =
796     element cNvPr { a_CT_NonVisualDrawingProps },
797     element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps }
798 a_CT_GvmlGroupShape =
799     element nvGrpSpPr { a_CT_GvmlGroupShapeNonVisual },
800     element grpSpPr { a_CT_GroupShapeProperties },
801     (element txSp { a_CT_GvmlTextShape }
802     | element sp { a_CT_GvmlShape }
803     | element cxnSp { a_CT_GvmlConnector }
804     | element pic { a_CT_GvmlPicture }
805     | element graphicFrame { a_CT_GvmlGraphicalObjectFrame }
806     | element grpSp { a_CT_GvmlGroupShape })*,
807     element extLst { a_CT_OfficeArtExtensionList }?
808 a_ST_PresetCameraType =
809     "legacyObliqueTopLeft"
810     | "legacyObliqueTop"
811     | "legacyObliqueTopRight"
812     | "legacyObliqueLeft"
813     | "legacyObliqueFront"
814     | "legacyObliqueRight"
815     | "legacyObliqueBottomLeft"
816     | "legacyObliqueBottom"
817     | "legacyObliqueBottomRight"
818     | "legacyPerspectiveTopLeft"
819     | "legacyPerspectiveTop"
820     | "legacyPerspectiveTopRight"
821     | "legacyPerspectiveLeft"
822     | "legacyPerspectiveFront"
823     | "legacyPerspectiveRight"
824     | "legacyPerspectiveBottomLeft"
825     | "legacyPerspectiveBottom"
826     | "legacyPerspectiveBottomRight"
827     | "orthographicFront"
828     | "isometricTopUp"
829     | "isometricTopDown"
830     | "isometricBottomUp"
831     | "isometricBottomDown"
832     | "isometricLeftUp"
833     | "isometricLeftDown"
834     | "isometricRightUp"
835     | "isometricRightDown"
836     | "isometricOffAxis1Left"

```

```

837 | "isometricOffAxis1Right"
838 | "isometricOffAxis1Top"
839 | "isometricOffAxis2Left"
840 | "isometricOffAxis2Right"
841 | "isometricOffAxis2Top"
842 | "isometricOffAxis3Left"
843 | "isometricOffAxis3Right"
844 | "isometricOffAxis3Bottom"
845 | "isometricOffAxis4Left"
846 | "isometricOffAxis4Right"
847 | "isometricOffAxis4Bottom"
848 | "obliqueTopLeft"
849 | "obliqueTop"
850 | "obliqueTopRight"
851 | "obliqueLeft"
852 | "obliqueRight"
853 | "obliqueBottomLeft"
854 | "obliqueBottom"
855 | "obliqueBottomRight"
856 | "perspectiveFront"
857 | "perspectiveLeft"
858 | "perspectiveRight"
859 | "perspectiveAbove"
860 | "perspectiveBelow"
861 | "perspectiveAboveLeftFacing"
862 | "perspectiveAboveRightFacing"
863 | "perspectiveContrastingLeftFacing"
864 | "perspectiveContrastingRightFacing"
865 | "perspectiveHeroicLeftFacing"
866 | "perspectiveHeroicRightFacing"
867 | "perspectiveHeroicExtremeLeftFacing"
868 | "perspectiveHeroicExtremeRightFacing"
869 | "perspectiveRelaxed"
870 | "perspectiveRelaxedModerately"
871 a_ST_FOVAngle = xsd:int { minInclusive = "0" maxInclusive = "10800000" }
872 a_CT_Camera =
873   attribute prst { a_ST_PresetCameraType },
874   attribute fov { a_ST_FOVAngle }?,
875
876   ## default value: 100%
877   attribute zoom { a_ST_PositivePercentage }?,
878   element rot { a_CT_SphereCoords }?
879 a_ST_LightRigDirection =
880   "tl" | "t" | "tr" | "l" | "r" | "bl" | "b" | "br"
881 a_ST_LightRigType =
882   "legacyFlat1"
883   | "legacyFlat2"
884   | "legacyFlat3"
885   | "legacyFlat4"
886   | "legacyNormal1"
887   | "legacyNormal2"
888   | "legacyNormal3"
889   | "legacyNormal4"

```



```

890 | "legacyHarsh1"
891 | "legacyHarsh2"
892 | "legacyHarsh3"
893 | "legacyHarsh4"
894 | "threePt"
895 | "balanced"
896 | "soft"
897 | "harsh"
898 | "flood"
899 | "contrasting"
900 | "morning"
901 | "sunrise"
902 | "sunset"
903 | "chilly"
904 | "freezing"
905 | "flat"
906 | "twoPt"
907 | "glow"
908 | "brightRoom"
909 a_CT_LightRig =
910     attribute rig { a_ST_LightRigType },
911     attribute dir { a_ST_LightRigDirection },
912     element rot { a_CT_SphereCoords }?
913 a_CT_Scene3D =
914     element camera { a_CT_Camera },
915     element lightRig { a_CT_LightRig },
916     element backdrop { a_CT_Backdrop }?,
917     element extLst { a_CT_OfficeArtExtensionList }?
918 a_CT_Backdrop =
919     element anchor { a_CT_Point3D },
920     element norm { a_CT_Vector3D },
921     element up { a_CT_Vector3D },
922     element extLst { a_CT_OfficeArtExtensionList }?
923 a_ST_BevelPresetType =
924     "relaxedInset"
925     | "circle"
926     | "slope"
927     | "cross"
928     | "angle"
929     | "softRound"
930     | "convex"
931     | "coolSlant"
932     | "divot"
933     | "riblet"
934     | "hardEdge"
935     | "artDeco"
936 a_CT_Bevel =
937
938     ## default value: 76200
939     attribute w { a_ST_PositiveCoordinate }?,
940
941     ## default value: 76200
942     attribute h { a_ST_PositiveCoordinate }?,

```

```

943
944     ## default value: circle
945     attribute prst { a_ST_BevelPresetType }?
946 a_ST_PresetMaterialType =
947     "legacyMatte"
948     | "legacyPlastic"
949     | "legacyMetal"
950     | "legacyWireframe"
951     | "matte"
952     | "plastic"
953     | "metal"
954     | "warmMatte"
955     | "translucentPowder"
956     | "powder"
957     | "dkEdge"
958     | "softEdge"
959     | "clear"
960     | "flat"
961     | "softmetal"
962 a_CT_Shape3D =
963
964     ## default value: 0
965     attribute z { a_ST_Coordinate }?,
966
967     ## default value: 0
968     attribute extrusionH { a_ST_PositiveCoordinate }?,
969
970     ## default value: 0
971     attribute contourW { a_ST_PositiveCoordinate }?,
972
973     ## default value: warmMatte
974     attribute prstMaterial { a_ST_PresetMaterialType }?,
975     element bevelT { a_CT_Bevel }?,
976     element bevelB { a_CT_Bevel }?,
977     element extrusionClr { a_CT_Color }?,
978     element contourClr { a_CT_Color }?,
979     element extLst { a_CT_OfficeArtExtensionList }?
980 a_CT_FlatText =
981
982     ## default value: 0
983     attribute z { a_ST_Coordinate }?
984 a_EG_Text3D =
985     element sp3d { a_CT_Shape3D }
986     | element flatTx { a_CT_FlatText }
987 a_CT_AlphaBiLevelEffect =
988     attribute thresh { a_ST_PositiveFixedPercentage }
989 a_CT_AlphaCeilingEffect = empty
990 a_CT_AlphaFloorEffect = empty
991 a_CT_AlphaInverseEffect = a_EG_ColorChoice?
992 a_CT_AlphaModulateFixedEffect =
993
994     ## default value: 100%
995     attribute amt { a_ST_PositivePercentage }?

```

```

996 a_CT_AlphaOutsetEffect =
997
998     ## default value: 0
999     attribute rad { a_ST_Coordinate }?
1000 a_CT_AlphaReplaceEffect = attribute a { a_ST_PositiveFixedPercentage }
1001 a_CT_BiLevelEffect = attribute thresh { a_ST_PositiveFixedPercentage }
1002 a_CT_BlurEffect =
1003
1004     ## default value: 0
1005     attribute rad { a_ST_PositiveCoordinate }?,
1006
1007     ## default value: true
1008     attribute grow { xsd:boolean }?
1009 a_CT_ColorChangeEffect =
1010
1011     ## default value: true
1012     attribute useA { xsd:boolean }?,
1013     element clrFrom { a_CT_Color },
1014     element clrTo { a_CT_Color }
1015 a_CT_ColorReplaceEffect = a_EG_ColorChoice
1016 a_CT_DuotoneEffect = a_EG_ColorChoice+
1017 a_CT_GlowEffect =
1018
1019     ## default value: 0
1020     attribute rad { a_ST_PositiveCoordinate }?,
1021     a_EG_ColorChoice
1022 a_CT_GrayscaleEffect = empty
1023 a_CT_HSLEffect =
1024
1025     ## default value: 0
1026     attribute hue { a_ST_PositiveFixedAngle }?,
1027
1028     ## default value: 0%
1029     attribute sat { a_ST_FixedPercentage }?,
1030
1031     ## default value: 0%
1032     attribute lum { a_ST_FixedPercentage }?
1033 a_CT_InnerShadowEffect =
1034
1035     ## default value: 0
1036     attribute blurRad { a_ST_PositiveCoordinate }?,
1037
1038     ## default value: 0
1039     attribute dist { a_ST_PositiveCoordinate }?,
1040
1041     ## default value: 0
1042     attribute dir { a_ST_PositiveFixedAngle }?,
1043     a_EG_ColorChoice
1044 a_CT_LuminanceEffect =
1045
1046     ## default value: 0%
1047     attribute bright { a_ST_FixedPercentage }?,
1048

```

```

1049     ## default value: 0%
1050     attribute contrast { a_ST_FixedPercentage }?
1051 a_CT_OuterShadowEffect =
1052
1053     ## default value: 0
1054     attribute blurRad { a_ST_PositiveCoordinate }?,
1055
1056     ## default value: 0
1057     attribute dist { a_ST_PositiveCoordinate }?,
1058
1059     ## default value: 0
1060     attribute dir { a_ST_PositiveFixedAngle }?,
1061
1062     ## default value: 100%
1063     attribute sx { a_ST_Percentage }?,
1064
1065     ## default value: 100%
1066     attribute sy { a_ST_Percentage }?,
1067
1068     ## default value: 0
1069     attribute kx { a_ST_FixedAngle }?,
1070
1071     ## default value: 0
1072     attribute ky { a_ST_FixedAngle }?,
1073
1074     ## default value: b
1075     attribute algn { a_ST_RectAlignment }?,
1076
1077     ## default value: true
1078     attribute rotWithShape { xsd:boolean }?,
1079     a_EG_ColorChoice
1080 a_ST_PresetShadowVal =
1081     "shdw1"
1082     | "shdw2"
1083     | "shdw3"
1084     | "shdw4"
1085     | "shdw5"
1086     | "shdw6"
1087     | "shdw7"
1088     | "shdw8"
1089     | "shdw9"
1090     | "shdw10"
1091     | "shdw11"
1092     | "shdw12"
1093     | "shdw13"
1094     | "shdw14"
1095     | "shdw15"
1096     | "shdw16"
1097     | "shdw17"
1098     | "shdw18"
1099     | "shdw19"
1100     | "shdw20"
1101 a_CT_PresetShadowEffect =

```

```

1102 attribute prst { a_ST_PresetShadowVal },
1103
1104 ## default value: 0
1105 attribute dist { a_ST_PositiveCoordinate }?,
1106
1107 ## default value: 0
1108 attribute dir { a_ST_PositiveFixedAngle }?,
1109 a_EG_ColorChoice
1110 a_CT_ReflectionEffect =
1111
1112 ## default value: 0
1113 attribute blurRad { a_ST_PositiveCoordinate }?,
1114
1115 ## default value: 100%
1116 attribute stA { a_ST_PositiveFixedPercentage }?,
1117
1118 ## default value: 0%
1119 attribute stPos { a_ST_PositiveFixedPercentage }?,
1120
1121 ## default value: 0%
1122 attribute endA { a_ST_PositiveFixedPercentage }?,
1123
1124 ## default value: 100%
1125 attribute endPos { a_ST_PositiveFixedPercentage }?,
1126
1127 ## default value: 0
1128 attribute dist { a_ST_PositiveCoordinate }?,
1129
1130 ## default value: 0
1131 attribute dir { a_ST_PositiveFixedAngle }?,
1132
1133 ## default value: 5400000
1134 attribute fadeDir { a_ST_PositiveFixedAngle }?,
1135
1136 ## default value: 100%
1137 attribute sx { a_ST_Percentage }?,
1138
1139 ## default value: 100%
1140 attribute sy { a_ST_Percentage }?,
1141
1142 ## default value: 0
1143 attribute kx { a_ST_FixedAngle }?,
1144
1145 ## default value: 0
1146 attribute ky { a_ST_FixedAngle }?,
1147
1148 ## default value: b
1149 attribute algn { a_ST_RectAlignment }?,
1150
1151 ## default value: true
1152 attribute rotWithShape { xsd:boolean }?
1153 a_CT_RelativeOffsetEffect =
1154

```

```

1155     ## default value: 0%
1156     attribute tx { a_ST_Percentage }?,
1157
1158     ## default value: 0%
1159     attribute ty { a_ST_Percentage }?
1160 a_CT_SoftEdgesEffect = attribute rad { a_ST_PositiveCoordinate }
1161 a_CT_TintEffect =
1162
1163     ## default value: 0
1164     attribute hue { a_ST_PositiveFixedAngle }?,
1165
1166     ## default value: 0%
1167     attribute amt { a_ST_FixedPercentage }?
1168 a_CT_TransformEffect =
1169
1170     ## default value: 100%
1171     attribute sx { a_ST_Percentage }?,
1172
1173     ## default value: 100%
1174     attribute sy { a_ST_Percentage }?,
1175
1176     ## default value: 0
1177     attribute kx { a_ST_FixedAngle }?,
1178
1179     ## default value: 0
1180     attribute ky { a_ST_FixedAngle }?,
1181
1182     ## default value: 0
1183     attribute tx { a_ST_Coordinate }?,
1184
1185     ## default value: 0
1186     attribute ty { a_ST_Coordinate }?
1187 a_CT_NoFillProperties = empty
1188 a_CT_SolidColorFillProperties = a_EG_ColorChoice?
1189 a_CT_LinearShadeProperties =
1190     attribute ang { a_ST_PositiveFixedAngle }?,
1191     attribute scaled { xsd:boolean }?
1192 a_ST_PathShadeType = "shape" | "circle" | "rect"
1193 a_CT_PathShadeProperties =
1194     attribute path { a_ST_PathShadeType }?,
1195     element fillToRect { a_CT_RelativeRect }?
1196 a_EG_ShadeProperties =
1197     element lin { a_CT_LinearShadeProperties }
1198     | element path { a_CT_PathShadeProperties }
1199 a_ST_TileFlipMode = "none" | "x" | "y" | "xy"
1200 a_CT_GradientStop =
1201     attribute pos { a_ST_PositiveFixedPercentage },
1202     a_EG_ColorChoice
1203 a_CT_GradientStopList = element gs { a_CT_GradientStop }+
1204 a_CT_GradientFillProperties =
1205     attribute flip { a_ST_TileFlipMode }?,
1206     attribute rotWithShape { xsd:boolean }?,
1207     element gsLst { a_CT_GradientStopList }?,

```

```

1208   a_EG_ShadeProperties?,
1209   element tileRect { a_CT_RelativeRect }?
1210 a_CT_TileInfoProperties =
1211   attribute tx { a_ST_Coordinate }?,
1212   attribute ty { a_ST_Coordinate }?,
1213   attribute sx { a_ST_Percentage }?,
1214   attribute sy { a_ST_Percentage }?,
1215   attribute flip { a_ST_TileFlipMode }?,
1216   attribute algn { a_ST_RectAlignment }?
1217 a_CT_StretchInfoProperties = element fillRect { a_CT_RelativeRect }?
1218 a_EG_FillModeProperties =
1219   element tile { a_CT_TileInfoProperties }
1220   | element stretch { a_CT_StretchInfoProperties }
1221 a_ST_BlipCompression = "email" | "screen" | "print" | "hqprint" | "none"
1222 a_CT_Blip =
1223   a_AG_Blob,
1224
1225   ## default value: none
1226   attribute cstate { a_ST_BlipCompression }?,
1227   (element alphaBiLevel { a_CT_AlphaBiLevelEffect }
1228   | element alphaCeiling { a_CT_AlphaCeilingEffect }
1229   | element alphaFloor { a_CT_AlphaFloorEffect }
1230   | element alphaInv { a_CT_AlphaInverseEffect }
1231   | element alphaMod { a_CT_AlphaModulateEffect }
1232   | element alphaModFix { a_CT_AlphaModulateFixedEffect }
1233   | element alphaRepl { a_CT_AlphaReplaceEffect }
1234   | element biLevel { a_CT_BiLevelEffect }
1235   | element blur { a_CT_BlurEffect }
1236   | element clrChange { a_CT_ColorChangeEffect }
1237   | element clrRepl { a_CT_ColorReplaceEffect }
1238   | element duotone { a_CT_DuotoneEffect }
1239   | element fillOverlay { a_CT_FillOverlayEffect }
1240   | element grayscl { a_CT_GrayscaleEffect }
1241   | element hsl { a_CT_HSLEffect }
1242   | element lum { a_CT_LuminanceEffect }
1243   | element tint { a_CT_TintEffect })*,
1244   element extLst { a_CT_OfficeArtExtensionList }?
1245 a_CT_BlipFillProperties =
1246   attribute dpi { xsd:unsignedInt }?,
1247   attribute rotWithShape { xsd:boolean }?,
1248   element blip { a_CT_Blip }?,
1249   element srcRect { a_CT_RelativeRect }?,
1250   a_EG_FillModeProperties?
1251 a_ST_PresetPatternVal =
1252   "pct5"
1253   | "pct10"
1254   | "pct20"
1255   | "pct25"
1256   | "pct30"
1257   | "pct40"
1258   | "pct50"
1259   | "pct60"
1260   | "pct70"

```

```

1261 | "pct75"
1262 | "pct80"
1263 | "pct90"
1264 | "horz"
1265 | "vert"
1266 | "ltHorz"
1267 | "ltVert"
1268 | "dkHorz"
1269 | "dkVert"
1270 | "narHorz"
1271 | "narVert"
1272 | "dashHorz"
1273 | "dashVert"
1274 | "cross"
1275 | "dnDiag"
1276 | "upDiag"
1277 | "ltDnDiag"
1278 | "ltUpDiag"
1279 | "dkDnDiag"
1280 | "dkUpDiag"
1281 | "wdDnDiag"
1282 | "wdUpDiag"
1283 | "dashDnDiag"
1284 | "dashUpDiag"
1285 | "diagCross"
1286 | "smCheck"
1287 | "lgCheck"
1288 | "smGrid"
1289 | "lgGrid"
1290 | "dotGrid"
1291 | "smConfetti"
1292 | "lgConfetti"
1293 | "horzBrick"
1294 | "diagBrick"
1295 | "solidDmnd"
1296 | "openDmnd"
1297 | "dotDmnd"
1298 | "plaid"
1299 | "sphere"
1300 | "weave"
1301 | "divot"
1302 | "shingle"
1303 | "wave"
1304 | "trellis"
1305 | "zigZag"
1306 a_CT_PatternFillProperties =
1307   attribute prst { a_ST_PresetPatternVal }?,
1308   element fgClr { a_CT_Color }?,
1309   element bgClr { a_CT_Color }?
1310 a_CT_GroupFillProperties = empty
1311 a_EG_FillProperties =
1312   element noFill { a_CT_NoFillProperties }
1313   | element solidFill { a_CT_SolidColorFillProperties }

```



```

1314 | element gradFill { a_CT_GradientFillProperties }
1315 | element blipFill { a_CT_BlipFillProperties }
1316 | element pattFill { a_CT_PatternFillProperties }
1317 | element grpFill { a_CT_GroupFillProperties }
1318 a_CT_FillProperties = a_EG_FillProperties
1319 a_CT_FillEffect = a_EG_FillProperties
1320 a_ST_BlendMode = "over" | "mult" | "screen" | "darken" | "lighten"
1321 a_CT_FillOverlayEffect =
1322     attribute blend { a_ST_BlendMode },
1323     a_EG_FillProperties
1324 a_CT_EffectReference = attribute ref { xsd:token }
1325 a_EG_Effect =
1326     element cont { a_CT_EffectContainer }
1327     | element effect { a_CT_EffectReference }
1328     | element alphaBiLevel { a_CT_AlphaBiLevelEffect }
1329     | element alphaCeiling { a_CT_AlphaCeilingEffect }
1330     | element alphaFloor { a_CT_AlphaFloorEffect }
1331     | element alphaInv { a_CT_AlphaInverseEffect }
1332     | element alphaMod { a_CT_AlphaModulateEffect }
1333     | element alphaModFix { a_CT_AlphaModulateFixedEffect }
1334     | element alphaOutset { a_CT_AlphaOutsetEffect }
1335     | element alphaRepl { a_CT_AlphaReplaceEffect }
1336     | element biLevel { a_CT_BiLevelEffect }
1337     | element blend { a_CT_BlendEffect }
1338     | element blur { a_CT_BlurEffect }
1339     | element clrChange { a_CT_ColorChangeEffect }
1340     | element clrRepl { a_CT_ColorReplaceEffect }
1341     | element duotone { a_CT_DuotoneEffect }
1342     | element fill { a_CT_FillEffect }
1343     | element fillOverlay { a_CT_FillOverlayEffect }
1344     | element glow { a_CT_GlowEffect }
1345     | element grayscl { a_CT_GrayscaleEffect }
1346     | element hsl { a_CT_HSLEffect }
1347     | element innerShdw { a_CT_InnerShadowEffect }
1348     | element lum { a_CT_LuminanceEffect }
1349     | element outerShdw { a_CT_OuterShadowEffect }
1350     | element prstShdw { a_CT_PresetShadowEffect }
1351     | element reflection { a_CT_ReflectionEffect }
1352     | element relOff { a_CT_RelativeOffsetEffect }
1353     | element softEdge { a_CT_SoftEdgesEffect }
1354     | element tint { a_CT_TintEffect }
1355     | element xfrm { a_CT_TransformEffect }
1356 a_ST_EffectContainerType = "sib" | "tree"
1357 a_CT_EffectContainer =
1358
1359     ## default value: sib
1360     attribute type { a_ST_EffectContainerType }?,
1361     attribute name { xsd:token }?,
1362     a_EG_Effect*
1363 a_CT_AlphaModulateEffect = element cont { a_CT_EffectContainer }
1364 a_CT_BlendEffect =
1365     attribute blend { a_ST_BlendMode },
1366     element cont { a_CT_EffectContainer }

```

```

1367 a_CT_EffectList =
1368     element blur { a_CT_BlurEffect }?,
1369     element fillOverlay { a_CT_FillOverlayEffect }?,
1370     element glow { a_CT_GlowEffect }?,
1371     element innerShdw { a_CT_InnerShadowEffect }?,
1372     element outerShdw { a_CT_OuterShadowEffect }?,
1373     element prstShdw { a_CT_PresetShadowEffect }?,
1374     element reflection { a_CT_ReflectionEffect }?,
1375     element softEdge { a_CT_SoftEdgesEffect }?
1376 a_EG_EffectProperties =
1377     element effectLst { a_CT_EffectList }
1378     | element effectDag { a_CT_EffectContainer }
1379 a_CT_EffectProperties = a_EG_EffectProperties
1380 a_blip = element blip { a_CT_Blip }
1381 a_ST_ShapeType =
1382     "line"
1383     | "lineInv"
1384     | "triangle"
1385     | "rtTriangle"
1386     | "rect"
1387     | "diamond"
1388     | "parallelogram"
1389     | "trapezoid"
1390     | "nonIsoscelesTrapezoid"
1391     | "pentagon"
1392     | "hexagon"
1393     | "heptagon"
1394     | "octagon"
1395     | "decagon"
1396     | "dodecagon"
1397     | "star4"
1398     | "star5"
1399     | "star6"
1400     | "star7"
1401     | "star8"
1402     | "star10"
1403     | "star12"
1404     | "star16"
1405     | "star24"
1406     | "star32"
1407     | "roundRect"
1408     | "round1Rect"
1409     | "round2SameRect"
1410     | "round2DiagRect"
1411     | "snipRoundRect"
1412     | "snip1Rect"
1413     | "snip2SameRect"
1414     | "snip2DiagRect"
1415     | "plaque"
1416     | "ellipse"
1417     | "teardrop"
1418     | "homePlate"
1419     | "chevron"

```

1420	"pieWedge"
1421	"pie"
1422	"blockArc"
1423	"donut"
1424	"noSmoking"
1425	"rightArrow"
1426	"leftArrow"
1427	"upArrow"
1428	"downArrow"
1429	"stripedRightArrow"
1430	"notchedRightArrow"
1431	"bentUpArrow"
1432	"leftRightArrow"
1433	"upDownArrow"
1434	"leftUpArrow"
1435	"leftRightUpArrow"
1436	"quadArrow"
1437	"leftArrowCallout"
1438	"rightArrowCallout"
1439	"upArrowCallout"
1440	"downArrowCallout"
1441	"leftRightArrowCallout"
1442	"upDownArrowCallout"
1443	"quadArrowCallout"
1444	"bentArrow"
1445	"uturnArrow"
1446	"circularArrow"
1447	"leftCircularArrow"
1448	"leftRightCircularArrow"
1449	"curvedRightArrow"
1450	"curvedLeftArrow"
1451	"curvedUpArrow"
1452	"curvedDownArrow"
1453	"swooshArrow"
1454	"cube"
1455	"can"
1456	"lightningBolt"
1457	"heart"
1458	"sun"
1459	"moon"
1460	"smileyFace"
1461	"irregularSeal1"
1462	"irregularSeal2"
1463	"foldedCorner"
1464	"bevel"
1465	"frame"
1466	"halfFrame"
1467	"corner"
1468	"diagStripe"
1469	"chord"
1470	"arc"
1471	"leftBracket"
1472	"rightBracket"

1473	"leftBrace"
1474	"rightBrace"
1475	"bracketPair"
1476	"bracePair"
1477	"straightConnector1"
1478	"bentConnector2"
1479	"bentConnector3"
1480	"bentConnector4"
1481	"bentConnector5"
1482	"curvedConnector2"
1483	"curvedConnector3"
1484	"curvedConnector4"
1485	"curvedConnector5"
1486	"callout1"
1487	"callout2"
1488	"callout3"
1489	"accentCallout1"
1490	"accentCallout2"
1491	"accentCallout3"
1492	"borderCallout1"
1493	"borderCallout2"
1494	"borderCallout3"
1495	"accentBorderCallout1"
1496	"accentBorderCallout2"
1497	"accentBorderCallout3"
1498	"wedgeRectCallout"
1499	"wedgeRoundRectCallout"
1500	"wedgeEllipseCallout"
1501	"cloudCallout"
1502	"cloud"
1503	"ribbon"
1504	"ribbon2"
1505	"ellipseRibbon"
1506	"ellipseRibbon2"
1507	"leftRightRibbon"
1508	"verticalScroll"
1509	"horizontalScroll"
1510	"wave"
1511	"doubleWave"
1512	"plus"
1513	"flowChartProcess"
1514	"flowChartDecision"
1515	"flowChartInputOutput"
1516	"flowChartPredefinedProcess"
1517	"flowChartInternalStorage"
1518	"flowChartDocument"
1519	"flowChartMultidocument"
1520	"flowChartTerminator"
1521	"flowChartPreparation"
1522	"flowChartManualInput"
1523	"flowChartManualOperation"
1524	"flowChartConnector"
1525	"flowChartPunchedCard"

```

1526 | "flowChartPunchedTape"
1527 | "flowChartSummingJunction"
1528 | "flowChartOr"
1529 | "flowChartCollate"
1530 | "flowChartSort"
1531 | "flowChartExtract"
1532 | "flowChartMerge"
1533 | "flowChartOfflineStorage"
1534 | "flowChartOnlineStorage"
1535 | "flowChartMagneticTape"
1536 | "flowChartMagneticDisk"
1537 | "flowChartMagneticDrum"
1538 | "flowChartDisplay"
1539 | "flowChartDelay"
1540 | "flowChartAlternateProcess"
1541 | "flowChartOffpageConnector"
1542 | "actionButtonBlank"
1543 | "actionButtonHome"
1544 | "actionButtonHelp"
1545 | "actionButtonInformation"
1546 | "actionButtonForwardNext"
1547 | "actionButtonBackPrevious"
1548 | "actionButtonEnd"
1549 | "actionButtonBeginning"
1550 | "actionButtonReturn"
1551 | "actionButtonDocument"
1552 | "actionButtonSound"
1553 | "actionButtonMovie"
1554 | "gear6"
1555 | "gear9"
1556 | "funnel"
1557 | "mathPlus"
1558 | "mathMinus"
1559 | "mathMultiply"
1560 | "mathDivide"
1561 | "mathEqual"
1562 | "mathNotEqual"
1563 | "cornerTabs"
1564 | "squareTabs"
1565 | "plaqueTabs"
1566 | "chartX"
1567 | "chartStar"
1568 | "chartPlus"
1569 a_ST_TextShapeType =
1570     "textNoShape"
1571     | "textPlain"
1572     | "textStop"
1573     | "textTriangle"
1574     | "textTriangleInverted"
1575     | "textChevron"
1576     | "textChevronInverted"
1577     | "textRingInside"
1578     | "textRingOutside"

```

```

1579 | "textArchUp"
1580 | "textArchDown"
1581 | "textCircle"
1582 | "textButton"
1583 | "textArchUpPour"
1584 | "textArchDownPour"
1585 | "textCirclePour"
1586 | "textButtonPour"
1587 | "textCurveUp"
1588 | "textCurveDown"
1589 | "textCanUp"
1590 | "textCanDown"
1591 | "textWave1"
1592 | "textWave2"
1593 | "textDoubleWave1"
1594 | "textWave4"
1595 | "textInflate"
1596 | "textDeflate"
1597 | "textInflateBottom"
1598 | "textDeflateBottom"
1599 | "textInflateTop"
1600 | "textDeflateTop"
1601 | "textDeflateInflate"
1602 | "textDeflateInflateDeflate"
1603 | "textFadeRight"
1604 | "textFadeLeft"
1605 | "textFadeUp"
1606 | "textFadeDown"
1607 | "textSlantUp"
1608 | "textSlantDown"
1609 | "textCascadeUp"
1610 | "textCascadeDown"
1611 a_ST_GeomGuideName = xsd:token
1612 a_ST_GeomGuideFormula = xsd:string
1613 a_CT_GeomGuide =
1614     attribute name { a_ST_GeomGuideName },
1615     attribute fmla { a_ST_GeomGuideFormula }
1616 a_CT_GeomGuideList = element gd { a_CT_GeomGuide }*
1617 a_ST_AdjCoordinate = a_ST_Coordinate | a_ST_GeomGuideName
1618 a_ST_AdjAngle = a_ST_Angle | a_ST_GeomGuideName
1619 a_CT_AdjPoint2D =
1620     attribute x { a_ST_AdjCoordinate },
1621     attribute y { a_ST_AdjCoordinate }
1622 a_CT_GeomRect =
1623     attribute l { a_ST_AdjCoordinate },
1624     attribute t { a_ST_AdjCoordinate },
1625     attribute r { a_ST_AdjCoordinate },
1626     attribute b { a_ST_AdjCoordinate }
1627 a_CT_XYAdjustHandle =
1628     attribute gdRefX { a_ST_GeomGuideName }?,
1629     attribute minX { a_ST_AdjCoordinate }?,
1630     attribute maxX { a_ST_AdjCoordinate }?,
1631     attribute gdRefY { a_ST_GeomGuideName }?,

```

```

1632     attribute minY { a_ST_AdjCoordinate }?,
1633     attribute maxY { a_ST_AdjCoordinate }?,
1634     element pos { a_CT_AdjPoint2D }
1635 a_CT_PolarAdjustHandle =
1636     attribute gdRefR { a_ST_GeomGuideName }?,
1637     attribute minR { a_ST_AdjCoordinate }?,
1638     attribute maxR { a_ST_AdjCoordinate }?,
1639     attribute gdRefAng { a_ST_GeomGuideName }?,
1640     attribute minAng { a_ST_AdjAngle }?,
1641     attribute maxAng { a_ST_AdjAngle }?,
1642     element pos { a_CT_AdjPoint2D }
1643 a_CT_ConnectionSite =
1644     attribute ang { a_ST_AdjAngle },
1645     element pos { a_CT_AdjPoint2D }
1646 a_CT_AdjustHandleList =
1647     (element ahXY { a_CT_XYAdjustHandle }
1648      | element ahPolar { a_CT_PolarAdjustHandle })*
1649 a_CT_ConnectionSitelist = element cxn { a_CT_ConnectionSite }*
1650 a_CT_Connection =
1651     attribute id { a_ST_DrawingElementId },
1652     attribute idx { xsd:unsignedInt }
1653 a_CT_Path2DMoveTo = element pt { a_CT_AdjPoint2D }
1654 a_CT_Path2DLineTo = element pt { a_CT_AdjPoint2D }
1655 a_CT_Path2DArcTo =
1656     attribute wR { a_ST_AdjCoordinate },
1657     attribute hR { a_ST_AdjCoordinate },
1658     attribute stAng { a_ST_AdjAngle },
1659     attribute swAng { a_ST_AdjAngle }
1660 a_CT_Path2DQuadBezierTo = element pt { a_CT_AdjPoint2D }+
1661 a_CT_Path2DCubicBezierTo = element pt { a_CT_AdjPoint2D }+
1662 a_CT_Path2DClose = empty
1663 a_ST_PathFillMode =
1664     "none" | "norm" | "lighten" | "lightenLess" | "darken" | "darkenLess"
1665 a_CT_Path2D =
1666
1667     ## default value: 0
1668     attribute w { a_ST_PositiveCoordinate }?,
1669
1670     ## default value: 0
1671     attribute h { a_ST_PositiveCoordinate }?,
1672
1673     ## default value: norm
1674     attribute fill { a_ST_PathFillMode }?,
1675
1676     ## default value: true
1677     attribute stroke { xsd:boolean }?,
1678
1679     ## default value: true
1680     attribute extrusionOk { xsd:boolean }?,
1681     (element close { a_CT_Path2DClose }
1682      | element moveTo { a_CT_Path2DMoveTo }
1683      | element lnTo { a_CT_Path2DLineTo }
1684      | element arcTo { a_CT_Path2DArcTo }

```

```

1685 | element quadBezTo { a_CT_Path2DQuadBezierTo }
1686 | element cubicBezTo { a_CT_Path2DCubicBezierTo })*
1687 a_CT_Path2DList = element path { a_CT_Path2D }*
1688 a_CT_PresetGeometry2D =
1689   attribute prst { a_ST_ShapeType },
1690   element avLst { a_CT_GeomGuideList }?
1691 a_CT_PresetTextShape =
1692   attribute prst { a_ST_TextShapeType },
1693   element avLst { a_CT_GeomGuideList }?
1694 a_CT_CustomGeometry2D =
1695   element avLst { a_CT_GeomGuideList }?,
1696   element gdLst { a_CT_GeomGuideList }?,
1697   element ahLst { a_CT_AdjustHandleList }?,
1698   element cxnLst { a_CT_ConnectionSiteList }?,
1699   element rect { a_CT_GeomRect }?,
1700   element pathLst { a_CT_Path2DList }
1701 a_EG_Geometry =
1702   element custGeom { a_CT_CustomGeometry2D }
1703   | element prstGeom { a_CT_PresetGeometry2D }
1704 a_EG_TextGeometry =
1705   element custGeom { a_CT_CustomGeometry2D }
1706   | element prstTxWarp { a_CT_PresetTextShape }
1707 a_ST_LineEndType =
1708   "none" | "triangle" | "stealth" | "diamond" | "oval" | "arrow"
1709 a_ST_LineEndWidth = "sm" | "med" | "lg"
1710 a_ST_LineEndLength = "sm" | "med" | "lg"
1711 a_CT_LineEndProperties =
1712   attribute type { a_ST_LineEndType }?,
1713   attribute w { a_ST_LineEndWidth }?,
1714   attribute len { a_ST_LineEndLength }?
1715 a_EG_LineFillProperties =
1716   element noFill { a_CT_NoFillProperties }
1717   | element solidFill { a_CT_SolidColorFillProperties }
1718   | element gradFill { a_CT_GradientFillProperties }
1719   | element pattFill { a_CT_PatternFillProperties }
1720 a_CT_LineJoinBevel = empty
1721 a_CT_LineJoinRound = empty
1722 a_CT_LineJoinMiterProperties =
1723   attribute lim { a_ST_PositivePercentage }?
1724 a_EG_LineJoinProperties =
1725   element round { a_CT_LineJoinRound }
1726   | element bevel { a_CT_LineJoinBevel }
1727   | element miter { a_CT_LineJoinMiterProperties }
1728 a_ST_PresetLineDashVal =
1729   "solid"
1730   | "dot"
1731   | "dash"
1732   | "lgDash"
1733   | "dashDot"
1734   | "lgDashDot"
1735   | "lgDashDotDot"
1736   | "sysDash"
1737   | "sysDot"

```



```

1738 | "sysDashDot"
1739 | "sysDashDotDot"
1740 a_CT_PresetLineDashProperties =
1741   attribute val { a_ST_PresetLineDashVal }?
1742 a_CT_DashStop =
1743   attribute d { a_ST_PositivePercentage },
1744   attribute sp { a_ST_PositivePercentage }
1745 a_CT_DashStopList = element ds { a_CT_DashStop }*
1746 a_EG_LineDashProperties =
1747   element prstDash { a_CT_PresetLineDashProperties }
1748   | element custDash { a_CT_DashStopList }
1749 a_ST_LineCap = "rnd" | "sq" | "flat"
1750 a_ST_LineWidth =
1751   xsd:int { minInclusive = "0" maxInclusive = "20116800" }
1752 a_ST_PenAlignment = "ctr" | "in"
1753 a_ST_CompoundLine = "sng" | "dbl" | "thickThin" | "thinThick" | "tri"
1754 a_CT_LineProperties =
1755   attribute w { a_ST_LineWidth }?,
1756   attribute cap { a_ST_LineCap }?,
1757   attribute cmpd { a_ST_CompoundLine }?,
1758   attribute algn { a_ST_PenAlignment }?,
1759   a_EG_LineFillProperties?,
1760   a_EG_LineDashProperties?,
1761   a_EG_LineJoinProperties?,
1762   element headEnd { a_CT_LineEndProperties }?,
1763   element tailEnd { a_CT_LineEndProperties }?,
1764   element extLst { a_CT_OfficeArtExtensionList }?
1765 a_ST_ShapeID = xsd:token
1766 a_CT_ShapeProperties =
1767   attribute bwMode { a_ST_BlackWhiteMode }?,
1768   element xfrm { a_CT_Transform2D }?,
1769   a_EG_Geometry?,
1770   a_EG_FillProperties?,
1771   element ln { a_CT_LineProperties }?,
1772   a_EG_EffectProperties?,
1773   element scene3d { a_CT_Scene3D }?,
1774   element sp3d { a_CT_Shape3D }?,
1775   element extLst { a_CT_OfficeArtExtensionList }?
1776 a_CT_GroupShapeProperties =
1777   attribute bwMode { a_ST_BlackWhiteMode }?,
1778   element xfrm { a_CT_GroupTransform2D }?,
1779   a_EG_FillProperties?,
1780   a_EG_EffectProperties?,
1781   element scene3d { a_CT_Scene3D }?,
1782   element extLst { a_CT_OfficeArtExtensionList }?
1783 a_CT_StyleMatrixReference =
1784   attribute idx { a_ST_StyleMatrixColumnIndex },
1785   a_EG_ColorChoice?
1786 a_CT_FontReference =
1787   attribute idx { a_ST_FontCollectionIndex },
1788   a_EG_ColorChoice?
1789 a_CT_ShapeStyle =
1790   element lnRef { a_CT_StyleMatrixReference },

```

```

1791     element fillRef { a_CT_StyleMatrixReference },
1792     element effectRef { a_CT_StyleMatrixReference },
1793     element fontRef { a_CT_FontReference }
1794 a_CT_DefaultShapeDefinition =
1795     element spPr { a_CT_ShapeProperties },
1796     element bodyPr { a_CT_TextBodyProperties },
1797     element lstStyle { a_CT_TextListStyle },
1798     element style { a_CT_ShapeStyle }?,
1799     element extLst { a_CT_OfficeArtExtensionList }?
1800 a_CT_ObjectStyleDefaults =
1801     element spDef { a_CT_DefaultShapeDefinition }?,
1802     element lnDef { a_CT_DefaultShapeDefinition }?,
1803     element txDef { a_CT_DefaultShapeDefinition }?,
1804     element extLst { a_CT_OfficeArtExtensionList }?
1805 a_CT_EmptyElement = empty
1806 a_CT_ColorMapping =
1807     attribute bg1 { a_ST_ColorSchemeIndex },
1808     attribute tx1 { a_ST_ColorSchemeIndex },
1809     attribute bg2 { a_ST_ColorSchemeIndex },
1810     attribute tx2 { a_ST_ColorSchemeIndex },
1811     attribute accent1 { a_ST_ColorSchemeIndex },
1812     attribute accent2 { a_ST_ColorSchemeIndex },
1813     attribute accent3 { a_ST_ColorSchemeIndex },
1814     attribute accent4 { a_ST_ColorSchemeIndex },
1815     attribute accent5 { a_ST_ColorSchemeIndex },
1816     attribute accent6 { a_ST_ColorSchemeIndex },
1817     attribute hlink { a_ST_ColorSchemeIndex },
1818     attribute folHlink { a_ST_ColorSchemeIndex },
1819     element extLst { a_CT_OfficeArtExtensionList }?
1820 a_CT_ColorMappingOverride =
1821     element masterClrMapping { a_CT_EmptyElement }
1822     | element overrideClrMapping { a_CT_ColorMapping }
1823 a_CT_ColorSchemeAndMapping =
1824     element clrScheme { a_CT_ColorScheme },
1825     element clrMap { a_CT_ColorMapping }?
1826 a_CT_ColorSchemeList =
1827     element extraClrScheme { a_CT_ColorSchemeAndMapping }*
1828 a_CT_OfficeStyleSheet =
1829     attribute name { xsd:string }?,
1830     element themeElements { a_CT_BaseStyles },
1831     element objectDefaults { a_CT_ObjectStyleDefaults }?,
1832     element extraClrSchemeLst { a_CT_ColorSchemeList }?,
1833     element custClrLst { a_CT_CustomColorList }?,
1834     element extLst { a_CT_OfficeArtExtensionList }?
1835 a_CT_BaseStylesOverride =
1836     element clrScheme { a_CT_ColorScheme }?,
1837     element fontScheme { a_CT_FontScheme }?,
1838     element fmtScheme { a_CT_StyleMatrix }?
1839 a_CT_ClipboardStyleSheet =
1840     element themeElements { a_CT_BaseStyles },
1841     element clrMap { a_CT_ColorMapping }
1842 a_theme = element theme { a_CT_OfficeStyleSheet }
1843 a_themeOverride = element themeOverride { a_CT_BaseStylesOverride }

```

```

1844 a_themeManager = element themeManager { a_CT_EmptyElement }
1845 a_CT_TableCellProperties =
1846
1847     ## default value: 91440
1848     attribute marL { a_ST_Coordinate32 }?,
1849
1850     ## default value: 91440
1851     attribute marR { a_ST_Coordinate32 }?,
1852
1853     ## default value: 45720
1854     attribute marT { a_ST_Coordinate32 }?,
1855
1856     ## default value: 45720
1857     attribute marB { a_ST_Coordinate32 }?,
1858
1859     ## default value: horz
1860     attribute vert { a_ST_TextVerticalType }?,
1861
1862     ## default value: t
1863     attribute anchor { a_ST_TextAnchoringType }?,
1864
1865     ## default value: false
1866     attribute anchorCtr { xsd:boolean }?,
1867
1868     ## default value: clip
1869     attribute horzOverflow { a_ST_TextHorzOverflowType }?,
1870     element lnL { a_CT_LineProperties }?,
1871     element lnR { a_CT_LineProperties }?,
1872     element lnT { a_CT_LineProperties }?,
1873     element lnB { a_CT_LineProperties }?,
1874     element lnTlToBr { a_CT_LineProperties }?,
1875     element lnBlToTr { a_CT_LineProperties }?,
1876     element cell3D { a_CT_Cell3D }?,
1877     a_EG_FillProperties?,
1878     element headers { a_CT_Headers }?,
1879     element extLst { a_CT_OfficeArtExtensionList }?
1880 a_CT_Headers = element header { xsd:string }*
1881 a_CT_TableCol =
1882     attribute w { a_ST_Coordinate },
1883     element extLst { a_CT_OfficeArtExtensionList }?
1884 a_CT_TableGrid = element gridCol { a_CT_TableCol }*
1885 a_CT_TableCell =
1886
1887     ## default value: 1
1888     attribute rowSpan { xsd:int }?,
1889
1890     ## default value: 1
1891     attribute gridSpan { xsd:int }?,
1892
1893     ## default value: false
1894     attribute hMerge { xsd:boolean }?,
1895
1896     ## default value: false

```

```

1897 attribute vMerge { xsd:boolean }?,
1898 attribute id { xsd:string }?,
1899 element txBdy { a_CT_TextBody }?,
1900 element tcPr { a_CT_TableCellProperties }?,
1901 element extLst { a_CT_OfficeArtExtensionList }?
1902 a_CT_TableRow =
1903     attribute h { a_ST_Coordinate },
1904     element tc { a_CT_TableCell }*,
1905     element extLst { a_CT_OfficeArtExtensionList }?
1906 a_CT_TableProperties =
1907
1908     ## default value: false
1909     attribute rtl { xsd:boolean }?,
1910
1911     ## default value: false
1912     attribute firstRow { xsd:boolean }?,
1913
1914     ## default value: false
1915     attribute firstCol { xsd:boolean }?,
1916
1917     ## default value: false
1918     attribute lastRow { xsd:boolean }?,
1919
1920     ## default value: false
1921     attribute lastCol { xsd:boolean }?,
1922
1923     ## default value: false
1924     attribute bandRow { xsd:boolean }?,
1925
1926     ## default value: false
1927     attribute bandCol { xsd:boolean }?,
1928     a_EG_FillProperties?,
1929     a_EG_EffectProperties?,
1930     (element tableStyle { a_CT_TableStyle }
1931      | element tableStyleId { s_ST_Guid })?,
1932     element extLst { a_CT_OfficeArtExtensionList }?
1933 a_CT_Table =
1934     element tblPr { a_CT_TableProperties }?,
1935     element tblGrid { a_CT_TableGrid },
1936     element tr { a_CT_TableRow }*
1937 a_tbl = element tbl { a_CT_Table }
1938 a_CT_Cell3D =
1939
1940     ## default value: plastic
1941     attribute prstMaterial { a_ST_PresetMaterialType }?,
1942     element bevel { a_CT_Bevel },
1943     element lightRig { a_CT_LightRig }?,
1944     element extLst { a_CT_OfficeArtExtensionList }?
1945 a_EG_ThemeableFillStyle =
1946     element fill { a_CT_FillProperties }
1947     | element fillRef { a_CT_StyleMatrixReference }
1948 a_CT_ThemeableLineStyle =
1949     element ln { a_CT_LineProperties }

```

```

1950 | element lnRef { a_CT_StyleMatrixReference }
1951 a_EG_ThemeableEffectStyle =
1952   element effect { a_CT_EffectProperties }
1953   | element effectRef { a_CT_StyleMatrixReference }
1954 a_EG_ThemeableFontStyles =
1955   element font { a_CT_FontCollection }
1956   | element fontRef { a_CT_FontReference }
1957 a_ST_OnOffStyleType = "on" | "off" | "def"
1958 a_CT_TableStyleTextStyle =
1959
1960   ## default value: def
1961   attribute b { a_ST_OnOffStyleType }?,
1962
1963   ## default value: def
1964   attribute i { a_ST_OnOffStyleType }?,
1965   a_EG_ThemeableFontStyles?,
1966   a_EG_ColorChoice?,
1967   element extLst { a_CT_OfficeArtExtensionList }?
1968 a_CT_TableCellBorderStyle =
1969   element left { a_CT_ThemeableLineStyle }?,
1970   element right { a_CT_ThemeableLineStyle }?,
1971   element top { a_CT_ThemeableLineStyle }?,
1972   element bottom { a_CT_ThemeableLineStyle }?,
1973   element insideH { a_CT_ThemeableLineStyle }?,
1974   element insideV { a_CT_ThemeableLineStyle }?,
1975   element tl2br { a_CT_ThemeableLineStyle }?,
1976   element tr2bl { a_CT_ThemeableLineStyle }?,
1977   element extLst { a_CT_OfficeArtExtensionList }?
1978 a_CT_TableBackgroundStyle =
1979   a_EG_ThemeableFillStyle?, a_EG_ThemeableEffectStyle?
1980 a_CT_TableStyleCellStyle =
1981   element tcBdr { a_CT_TableCellBorderStyle }?,
1982   a_EG_ThemeableFillStyle?,
1983   element cell3D { a_CT_Cell3D }?
1984 a_CT_TablePartStyle =
1985   element tcTxStyle { a_CT_TableStyleTextStyle }?,
1986   element tcStyle { a_CT_TableStyleCellStyle }?
1987 a_CT_TableStyle =
1988   attribute styleId { s_ST_Guid },
1989   attribute styleName { xsd:string },
1990   element tblBg { a_CT_TableBackgroundStyle }?,
1991   element wholeTbl { a_CT_TablePartStyle }?,
1992   element band1H { a_CT_TablePartStyle }?,
1993   element band2H { a_CT_TablePartStyle }?,
1994   element band1V { a_CT_TablePartStyle }?,
1995   element band2V { a_CT_TablePartStyle }?,
1996   element lastCol { a_CT_TablePartStyle }?,
1997   element firstCol { a_CT_TablePartStyle }?,
1998   element lastRow { a_CT_TablePartStyle }?,
1999   element seCell { a_CT_TablePartStyle }?,
2000   element swCell { a_CT_TablePartStyle }?,
2001   element firstRow { a_CT_TablePartStyle }?,
2002   element neCell { a_CT_TablePartStyle }?,

```

```

2003     element nwCell { a_CT_TablePartStyle }?,
2004     element extLst { a_CT_OfficeArtExtensionList }?
2005 a_CT_TableStyleList =
2006     attribute def { s_ST_Guid },
2007     element tblStyle { a_CT_TableStyle }*
2008 a_tblStyleLst = element tblStyleLst { a_CT_TableStyleList }
2009 a_CT_TextParagraph =
2010     element pPr { a_CT_TextParagraphProperties }?,
2011     a_EG_TextRun*,
2012     element endParaPr { a_CT_TextCharacterProperties }?
2013 a_ST_TextAnchoringType = "t" | "ctr" | "b" | "just" | "dist"
2014 a_ST_TextVertOverflowType = "overflow" | "ellipsis" | "clip"
2015 a_ST_TextHorzOverflowType = "overflow" | "clip"
2016 a_ST_TextVerticalType =
2017     "horz"
2018     | "vert"
2019     | "vert270"
2020     | "wordArtVert"
2021     | "eaVert"
2022     | "mongolianVert"
2023     | "wordArtVertRtl"
2024 a_ST_TextWrappingType = "none" | "square"
2025 a_ST_TextColumnCount =
2026     xsd:int { minInclusive = "1" maxInclusive = "16" }
2027 a_CT_TextListStyle =
2028     element defPPr { a_CT_TextParagraphProperties }?,
2029     element lvl1pPr { a_CT_TextParagraphProperties }?,
2030     element lvl2pPr { a_CT_TextParagraphProperties }?,
2031     element lvl3pPr { a_CT_TextParagraphProperties }?,
2032     element lvl4pPr { a_CT_TextParagraphProperties }?,
2033     element lvl5pPr { a_CT_TextParagraphProperties }?,
2034     element lvl6pPr { a_CT_TextParagraphProperties }?,
2035     element lvl7pPr { a_CT_TextParagraphProperties }?,
2036     element lvl8pPr { a_CT_TextParagraphProperties }?,
2037     element lvl9pPr { a_CT_TextParagraphProperties }?,
2038     element extLst { a_CT_OfficeArtExtensionList }?
2039 a_ST_TextFontScalePercentOrPercentString =
2040     a_ST_TextFontScalePercent | s_ST_Percentage
2041 a_ST_TextFontScalePercent =
2042     xsd:int { minInclusive = "1000" maxInclusive = "100000" }
2043 a_CT_TextNormalAutofit =
2044
2045     ## default value: 100%
2046     attribute fontScale { a_ST_TextFontScalePercentOrPercentString }?,
2047
2048     ## default value: 0%
2049     attribute lnSpcReduction { a_ST_TextSpacingPercentOrPercentString }?
2050 a_CT_TextShapeAutofit = empty
2051 a_CT_TextNoAutofit = empty
2052 a_EG_TextAutofit =
2053     element noAutofit { a_CT_TextNoAutofit }
2054     | element normAutofit { a_CT_TextNormalAutofit }
2055     | element spAutoFit { a_CT_TextShapeAutofit }

```

```

2056 a_CT_TextBodyProperties =
2057     attribute rot { a_ST_Angle }?,
2058     attribute spcFirstLastPara { xsd:boolean }?,
2059     attribute vertOverflow { a_ST_TextVertOverflowType }?,
2060     attribute horzOverflow { a_ST_TextHorzOverflowType }?,
2061     attribute vert { a_ST_TextVerticalType }?,
2062     attribute wrap { a_ST_TextWrappingType }?,
2063     attribute lIns { a_ST_Coordinate32 }?,
2064     attribute tIns { a_ST_Coordinate32 }?,
2065     attribute rIns { a_ST_Coordinate32 }?,
2066     attribute bIns { a_ST_Coordinate32 }?,
2067     attribute numCol { a_ST_TextColumnCount }?,
2068     attribute spcCol { a_ST_PositiveCoordinate32 }?,
2069     attribute rtlCol { xsd:boolean }?,
2070     attribute fromWordArt { xsd:boolean }?,
2071     attribute anchor { a_ST_TextAnchoringType }?,
2072     attribute anchorCtr { xsd:boolean }?,
2073     attribute forceAA { xsd:boolean }?,
2074
2075     ## default value: false
2076     attribute upright { xsd:boolean }?,
2077     attribute compatLnSpc { xsd:boolean }?,
2078     element prstTxWarp { a_CT_PresetTextShape }?,
2079     a_EG_TextAutofit?,
2080     element scene3d { a_CT_Scene3D }?,
2081     a_EG_Text3D?,
2082     element extLst { a_CT_OfficeArtExtensionList }?
2083 a_CT_TextBody =
2084     element bodyPr { a_CT_TextBodyProperties },
2085     element lstStyle { a_CT_TextListStyle }?,
2086     element p { a_CT_TextParagraph }+
2087 a_ST_TextBulletStartAtNum =
2088     xsd:int { minInclusive = "1" maxInclusive = "32767" }
2089 a_ST_TextAutonumberScheme =
2090     "alphaLcParenBoth"
2091     | "alphaUcParenBoth"
2092     | "alphaLcParenR"
2093     | "alphaUcParenR"
2094     | "alphaLcPeriod"
2095     | "alphaUcPeriod"
2096     | "arabicParenBoth"
2097     | "arabicParenR"
2098     | "arabicPeriod"
2099     | "arabicPlain"
2100     | "romanLcParenBoth"
2101     | "romanUcParenBoth"
2102     | "romanLcParenR"
2103     | "romanUcParenR"
2104     | "romanLcPeriod"
2105     | "romanUcPeriod"
2106     | "circleNumDbPlain"
2107     | "circleNumWdBlackPlain"
2108     | "circleNumWdWhitePlain"

```

```

2109 | "arabicDbPeriod"
2110 | "arabicDbPlain"
2111 | "ea1ChsPeriod"
2112 | "ea1ChsPlain"
2113 | "ea1ChtPeriod"
2114 | "ea1ChtPlain"
2115 | "ea1JpnChsDbPeriod"
2116 | "ea1JpnKorPlain"
2117 | "ea1JpnKorPeriod"
2118 | "arabic1Minus"
2119 | "arabic2Minus"
2120 | "hebrew2Minus"
2121 | "thaiAlphaPeriod"
2122 | "thaiAlphaParenR"
2123 | "thaiAlphaParenBoth"
2124 | "thaiNumPeriod"
2125 | "thaiNumParenR"
2126 | "thaiNumParenBoth"
2127 | "hindiAlphaPeriod"
2128 | "hindiNumPeriod"
2129 | "hindiNumParenR"
2130 | "hindiAlpha1Period"
2131 a_CT_TextBulletColorFollowText = empty
2132 a_EG_TextBulletColor =
2133   element buClrTx { a_CT_TextBulletColorFollowText }
2134   | element buClr { a_CT_Color }
2135 a_ST_TextBulletSize = a_ST_TextBulletSizePercent | a_ST_TextBulletSizeDecimal
2136 a_ST_TextBulletSizePercent =
2137   xsd:string {
2138     pattern = "0*((2[5-9])|([3-9][0-9])|([1-3][0-9][0-9])|400)%"
2139   }
2140 a_ST_TextBulletSizeDecimal = xsd:int { minInclusive = "25000" maxInclusive = "400000" }
2141 a_CT_TextBulletSizeFollowText = empty
2142 a_CT_TextBulletSizePercent =
2143   attribute val { a_ST_TextBulletSizePercent }
2144 a_CT_TextBulletSizePoint = attribute val { a_ST_TextFontSize }
2145 a_EG_TextBulletSize =
2146   element buSzTx { a_CT_TextBulletSizeFollowText }
2147   | element buSzPct { a_CT_TextBulletSizePercent }
2148   | element buSzPts { a_CT_TextBulletSizePoint }
2149 a_CT_TextBulletTypefaceFollowText = empty
2150 a_EG_TextBulletTypeface =
2151   element buFontTx { a_CT_TextBulletTypefaceFollowText }
2152   | element buFont { a_CT_TextFont }
2153 a_CT_TextAutonumberBullet =
2154   attribute type { a_ST_TextAutonumberScheme },
2155
2156   ## default value: 1
2157   attribute startAt { a_ST_TextBulletStartAtNum }?
2158 a_CT_TextCharBullet = attribute char { xsd:string }
2159 a_CT_TextBlipBullet = element blip { a_CT_Blip }
2160 a_CT_TextNoBullet = empty
2161 a_EG_TextBullet =

```



```

2162 element buNone { a_CT_TextNoBullet }
2163 | element buAutoNum { a_CT_TextAutonumberBullet }
2164 | element buChar { a_CT_TextCharBullet }
2165 | element buBlip { a_CT_TextBlipBullet }
2166 a_ST_TextPoint = a_ST_TextPointUnqualified | s_ST_UniversalMeasure
2167 a_ST_TextPointUnqualified =
2168   xsd:int { minInclusive = "-400000" maxInclusive = "400000" }
2169 a_ST_TextNonNegativePoint =
2170   xsd:int { minInclusive = "0" maxInclusive = "400000" }
2171 a_ST_TextFontSize =
2172   xsd:int { minInclusive = "100" maxInclusive = "400000" }
2173 a_ST_TextTypeface = xsd:string
2174 a_ST_PitchFamily =
2175   xsd:byte "00" | xsd:byte "01" | xsd:byte "02" | xsd:byte "16" |
2176   xsd:byte "17" | xsd:byte "18" | xsd:byte "32" | xsd:byte "33" |
2177   xsd:byte "34" | xsd:byte "48" | xsd:byte "49" | xsd:byte "50" |
2178   xsd:byte "64" | xsd:byte "65" | xsd:byte "66" | xsd:byte "80" |
2179   xsd:byte "81" | xsd:byte "82"
2180 a_CT_TextFont =
2181   attribute typeface { a_ST_TextTypeface },
2182   attribute panose { s_ST_Panose }?,
2183
2184   ## default value: 0
2185   attribute pitchFamily { a_ST_PitchFamily }?,
2186
2187   ## default value: 1
2188   attribute charset { xsd:byte }?
2189 a_ST_TextUnderlineType =
2190   "none"
2191   | "words"
2192   | "sng"
2193   | "dbl"
2194   | "heavy"
2195   | "dotted"
2196   | "dottedHeavy"
2197   | "dash"
2198   | "dashHeavy"
2199   | "dashLong"
2200   | "dashLongHeavy"
2201   | "dotDash"
2202   | "dotDashHeavy"
2203   | "dotDotDash"
2204   | "dotDotDashHeavy"
2205   | "wavy"
2206   | "wavyHeavy"
2207   | "wavyDb1"
2208 a_CT_TextUnderlineLineFollowText = empty
2209 a_CT_TextUnderlineFillFollowText = empty
2210 a_CT_TextUnderlineFillGroupWrapper = a_EG_FillProperties
2211 a_EG_TextUnderlineLine =
2212   element uLnTx { a_CT_TextUnderlineLineFollowText }
2213   | element uLn { a_CT_LineProperties }?
2214 a_EG_TextUnderlineFill =

```

```

2215     element uFillTx { a_CT_TextUnderlineFillFollowText }
2216     | element uFill { a_CT_TextUnderlineFillGroupWrapper }
2217 a_ST_TextStrikeType = "noStrike" | "sngStrike" | "dblStrike"
2218 a_ST_TextCapsType = "none" | "small" | "all"
2219 a_CT_TextCharacterProperties =
2220     attribute kumimoji { xsd:boolean }?,
2221     attribute lang { s_ST_Lang }?,
2222     attribute altLang { s_ST_Lang }?,
2223     attribute sz { a_ST_TextFontSize }?,
2224     attribute b { xsd:boolean }?,
2225     attribute i { xsd:boolean }?,
2226     attribute u { a_ST_TextUnderlineType }?,
2227     attribute strike { a_ST_TextStrikeType }?,
2228     attribute kern { a_ST_TextNonNegativePoint }?,
2229     attribute cap { a_ST_TextCapsType }?,
2230     attribute spc { a_ST_TextPoint }?,
2231     attribute normalizeH { xsd:boolean }?,
2232     attribute baseline { a_ST_Percentage }?,
2233     attribute noProof { xsd:boolean }?,
2234
2235     ## default value: true
2236     attribute dirty { xsd:boolean }?,
2237
2238     ## default value: false
2239     attribute err { xsd:boolean }?,
2240
2241     ## default value: true
2242     attribute smtClean { xsd:boolean }?,
2243
2244     ## default value: 0
2245     attribute smtId { xsd:unsignedInt }?,
2246     attribute bmk { xsd:string }?,
2247     element ln { a_CT_LineProperties }?,
2248     a_EG_FillProperties?,
2249     a_EG_EffectProperties?,
2250     element highlight { a_CT_Color }?,
2251     a_EG_TextUnderlineLine?,
2252     a_EG_TextUnderlineFill?,
2253     element latin { a_CT_TextFont }?,
2254     element ea { a_CT_TextFont }?,
2255     element cs { a_CT_TextFont }?,
2256     element sym { a_CT_TextFont }?,
2257     element hlinkClick { a_CT_Hyperlink }?,
2258     element hlinkMouseOver { a_CT_Hyperlink }?,
2259     element rtl { a_CT_Boolean }?,
2260     element extLst { a_CT_OfficeArtExtensionList }?
2261 a_CT_Boolean =
2262
2263     ## default value: 0
2264     attribute val { s_ST_OnOff }?
2265 a_ST_TextSpacingPoint =
2266     xsd:int { minInclusive = "0" maxInclusive = "158400" }
2267 a_ST_TextSpacingPercentOrPercentString =

```

```

2268   a_ST_TextSpacingPercent | s_ST_Percentage
2269 a_ST_TextSpacingPercent =
2270   xsd:int { minInclusive = "0" maxInclusive = "13200000" }
2271 a_CT_TextSpacingPercent =
2272   attribute val { a_ST_TextSpacingPercentOrPercentString }
2273 a_CT_TextSpacingPoint = attribute val { a_ST_TextSpacingPoint }
2274 a_ST_TextMargin =
2275   xsd:int { minInclusive = "0" maxInclusive = "51206400" }
2276 a_ST_TextIndent =
2277   xsd:int { minInclusive = "-51206400" maxInclusive = "51206400" }
2278 a_ST_TextTabAlignType = "l" | "ctr" | "r" | "dec"
2279 a_CT_TextTabStop =
2280   attribute pos { a_ST_Coordinate32 }?,
2281   attribute algn { a_ST_TextTabAlignType }?
2282 a_CT_TextTabStopList = element tab { a_CT_TextTabStop }*
2283 a_CT_TextLineBreak = element rPr { a_CT_TextCharacterProperties }?
2284 a_CT_TextSpacing =
2285   element spcPct { a_CT_TextSpacingPercent }
2286   | element spcPts { a_CT_TextSpacingPoint }
2287 a_ST_TextAlignType =
2288   "l" | "ctr" | "r" | "just" | "justLow" | "dist" | "thaiDist"
2289 a_ST_TextFontAlignType = "auto" | "t" | "ctr" | "base" | "b"
2290 a_ST_TextIndentLevelType =
2291   xsd:int { minInclusive = "0" maxInclusive = "8" }
2292 a_CT_TextParagraphProperties =
2293   attribute marL { a_ST_TextMargin }?,
2294   attribute marR { a_ST_TextMargin }?,
2295   attribute lvl { a_ST_TextIndentLevelType }?,
2296   attribute indent { a_ST_TextIndent }?,
2297   attribute algn { a_ST_TextAlignType }?,
2298   attribute defTabSz { a_ST_Coordinate32 }?,
2299   attribute rtl { xsd:boolean }?,
2300   attribute eaLnBrk { xsd:boolean }?,
2301   attribute fontAlgn { a_ST_TextFontAlignType }?,
2302   attribute latinLnBrk { xsd:boolean }?,
2303   attribute hangingPunct { xsd:boolean }?,
2304   element lnSpc { a_CT_TextSpacing }?,
2305   element spcBef { a_CT_TextSpacing }?,
2306   element spcAft { a_CT_TextSpacing }?,
2307   a_EG_TextBulletColor?,
2308   a_EG_TextBulletSize?,
2309   a_EG_TextBulletTypeface?,
2310   a_EG_TextBullet?,
2311   element tabLst { a_CT_TextTabStopList }?,
2312   element defRPr { a_CT_TextCharacterProperties }?,
2313   element extLst { a_CT_OfficeArtExtensionList }?
2314 a_CT_TextField =
2315   attribute id { s_ST_Guid },
2316   attribute type { xsd:string }?,
2317   element rPr { a_CT_TextCharacterProperties }?,
2318   element pPr { a_CT_TextParagraphProperties }?,
2319   element t { xsd:string }?
2320 a_EG_TextRun =

```

```

2321   element r { a_CT_RegularTextRun }
2322   | element br { a_CT_TextLineBreak }
2323   | element fld { a_CT_TextField }
2324   a_CT_RegularTextRun =
2325     element rPr { a_CT_TextCharacterProperties }?,
2326     element t { xsd:string }

```

B.4.1.1 Part Schemas

B.4.1.1.1 Table Styles Part

This schema is available in the file DrawingML_Table_Styles.rnc.

```

1   include "dml-main.rnc"
2   include "shared-relationshipReference.rnc"
3   include "dml-diagram.rnc"
4   include "shared-commonSimpleTypes.rnc"
5   include "dml-lockedCanvas.rnc"
6   include "any.rnc"
7   include "dml-chart.rnc"
8   include "dml-chartDrawing.rnc"
9   include "dml-picture.rnc"
10  start = a_tblStyleLst

```

B.4.1.1.2 Theme Part

This schema is available in the file DrawingML_Theme.rnc.

```

1   include "dml-main.rnc"
2   include "shared-relationshipReference.rnc"
3   include "dml-diagram.rnc"
4   include "shared-commonSimpleTypes.rnc"
5   include "dml-lockedCanvas.rnc"
6   include "any.rnc"
7   include "dml-chart.rnc"
8   include "dml-chartDrawing.rnc"
9   include "dml-picture.rnc"
10  start = a_theme

```

B.4.1.1.3 Theme Override Part

This schema is available in the file DrawingML_Theme_Override.rnc.

```

1   include "dml-main.rnc"
2   include "shared-relationshipReference.rnc"
3   include "dml-diagram.rnc"
4   include "shared-commonSimpleTypes.rnc"
5   include "dml-lockedCanvas.rnc"
6   include "any.rnc"
7   include "dml-chart.rnc"
8   include "dml-chartDrawing.rnc"
9   include "dml-picture.rnc"
10  start = a_themeOverride

```

B.4.2 DrawingML - Picture

This schema is available in the file dml-picture.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/picture"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace dpct =
5   "http://schemas.openxmlformats.org/drawingml/2006/picture"
6 namespace o = "urn:schemas-microsoft-com:office:office"
7 namespace v = "urn:schemas-microsoft-com:vm1"
8 namespace w10 = "urn:schemas-microsoft-com:office:word"
9 namespace x = "urn:schemas-microsoft-com:office:excel"
10
11 dpct_CT_PictureNonVisual =
12   element cNvPr { a_CT_NonVisualDrawingProps },
13   element cNvPicPr { a_CT_NonVisualPictureProperties }
14 dpct_CT_Picture =
15   element nvPicPr { dpct_CT_PictureNonVisual },
16   element blipFill { a_CT_BlipFillProperties },
17   element spPr { a_CT_ShapeProperties }
18 dpct_pic = element pic { dpct_CT_Picture }

```

B.4.3 DrawingML - Locked Canvas

This schema is available in the file dml-lockedCanvas.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace dlckcnv =
5   "http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas"
6 namespace o = "urn:schemas-microsoft-com:office:office"
7 namespace r =
8   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9 namespace v = "urn:schemas-microsoft-com:vm1"
10 namespace w10 = "urn:schemas-microsoft-com:office:word"
11 namespace x = "urn:schemas-microsoft-com:office:excel"
12
13 dlckcnv_lockedCanvas = element lockedCanvas { a_CT_Gvm1GroupShape }

```

B.4.4 DrawingML - Wordprocessing Drawing

This schema is available in the file dml-wordprocessingDrawing.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace o = "urn:schemas-microsoft-com:office:office"
5 namespace dpct = "http://schemas.openxmlformats.org/drawingml/2006/picture"
6 namespace r = http://schemas.openxmlformats.org/officeDocument/2006/relationships
7 namespace v = "urn:schemas-microsoft-com:vm1"
8 namespace w =

```

```

9      "http://schemas.openxmlformats.org/wordprocessingml/2006/main"
10 namespace w10 = "urn:schemas-microsoft-com:office:word"
11 namespace wp =
12     "http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
13 namespace x = "urn:schemas-microsoft-com:office:excel"
14
15 wp_CT_EffectExtent =
16     attribute l { a_ST_Coordinate },
17     attribute t { a_ST_Coordinate },
18     attribute r { a_ST_Coordinate },
19     attribute b { a_ST_Coordinate }
20 wp_ST_WrapDistance = xsd:unsignedInt
21 wp_CT_Inline =
22     attribute distT { wp_ST_WrapDistance }?,
23     attribute distB { wp_ST_WrapDistance }?,
24     attribute distL { wp_ST_WrapDistance }?,
25     attribute distR { wp_ST_WrapDistance }?,
26     element extent { a_CT_PositiveSize2D },
27     element effectExtent { wp_CT_EffectExtent }?,
28     element docPr { a_CT_NonVisualDrawingProps },
29     element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }?,
30     a_graphic
31 wp_ST_WrapText = "bothSides" | "left" | "right" | "largest"
32 wp_CT_WrapPath =
33     attribute edited { xsd:boolean }?,
34     element start { a_CT_Point2D },
35     element lineTo { a_CT_Point2D }+
36 wp_CT_WrapNone = empty
37 wp_CT_WrapSquare =
38     attribute wrapText { wp_ST_WrapText },
39     attribute distT { wp_ST_WrapDistance }?,
40     attribute distB { wp_ST_WrapDistance }?,
41     attribute distL { wp_ST_WrapDistance }?,
42     attribute distR { wp_ST_WrapDistance }?,
43     element effectExtent { wp_CT_EffectExtent }?
44 wp_CT_WrapTight =
45     attribute wrapText { wp_ST_WrapText },
46     attribute distL { wp_ST_WrapDistance }?,
47     attribute distR { wp_ST_WrapDistance }?,
48     element wrapPolygon { wp_CT_WrapPath }
49 wp_CT_WrapThrough =
50     attribute wrapText { wp_ST_WrapText },
51     attribute distL { wp_ST_WrapDistance }?,
52     attribute distR { wp_ST_WrapDistance }?,
53     element wrapPolygon { wp_CT_WrapPath }
54 wp_CT_WrapTopBottom =
55     attribute distT { wp_ST_WrapDistance }?,
56     attribute distB { wp_ST_WrapDistance }?,
57     element effectExtent { wp_CT_EffectExtent }?
58 wp_EG_WrapType =
59     element wrapNone { wp_CT_WrapNone }
60     | element wrapSquare { wp_CT_WrapSquare }
61     | element wrapTight { wp_CT_WrapTight }

```

```

62 | element wrapThrough { wp_CT_WrapThrough }
63 | element wrapTopAndBottom { wp_CT_WrapTopBottom }
64 wp_ST_PositionOffset = xsd:int
65 wp_ST_AlignH = "left" | "right" | "center" | "inside" | "outside"
66 wp_ST_RelFromH =
67   "margin"
68   | "page"
69   | "column"
70   | "character"
71   | "leftMargin"
72   | "rightMargin"
73   | "insideMargin"
74   | "outsideMargin"
75 wp_CT_PosH =
76   attribute relativeFrom { wp_ST_RelFromH },
77   (element align { wp_ST_AlignH }
78     | element posOffset { wp_ST_PositionOffset })
79 wp_ST_AlignV = "top" | "bottom" | "center" | "inside" | "outside"
80 wp_ST_RelFromV =
81   "margin"
82   | "page"
83   | "paragraph"
84   | "line"
85   | "topMargin"
86   | "bottomMargin"
87   | "insideMargin"
88   | "outsideMargin"
89 wp_CT_PosV =
90   attribute relativeFrom { wp_ST_RelFromV },
91   (element align { wp_ST_AlignV }
92     | element posOffset { wp_ST_PositionOffset })
93 wp_CT_Anchor =
94   attribute distT { wp_ST_WrapDistance }?,
95   attribute distB { wp_ST_WrapDistance }?,
96   attribute distL { wp_ST_WrapDistance }?,
97   attribute distR { wp_ST_WrapDistance }?,
98   attribute simplePos { xsd:boolean }?,
99   attribute relativeHeight { xsd:unsignedInt },
100  attribute behindDoc { xsd:boolean },
101  attribute locked { xsd:boolean },
102  attribute layoutInCell { xsd:boolean },
103  attribute hidden { xsd:boolean }?,
104  attribute allowOverlap { xsd:boolean },
105  element simplePos { a_CT_Point2D },
106  element positionH { wp_CT_PosH },
107  element positionV { wp_CT_PosV },
108  element extent { a_CT_PositiveSize2D },
109  element effectExtent { wp_CT_EffectExtent }?,
110  wp_EG_WrapType,
111  element docPr { a_CT_NonVisualDrawingProps },
112  element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }?,
113  a_graphic
114 wp_CT_TxbxContent = w_EG_BlockLevelElts+

```

```

115 wp_CT_TextboxInfo =
116
117     ## default value: 0
118     attribute id { xsd:unsignedShort }?,
119     element txbxContent { wp_CT_TxbxContent },
120     element extLst { a_CT_OfficeArtExtensionList }?
121 wp_CT_LinkedTextboxInformation =
122     attribute id { xsd:unsignedShort },
123     attribute seq { xsd:unsignedShort },
124     element extLst { a_CT_OfficeArtExtensionList }?
125 wp_CT_WordprocessingShape =
126
127     ## default value: false
128     attribute normalEastAsianFlow { xsd:boolean }?,
129     element cNvPr { a_CT_NonVisualDrawingProps }?,
130     (element cNvSpPr { a_CT_NonVisualDrawingShapeProps }
131      | element cNvCnPr { a_CT_NonVisualConnectorProperties } ),
132     element spPr { a_CT_ShapeProperties },
133     element style { a_CT_ShapeStyle }?,
134     element extLst { a_CT_OfficeArtExtensionList }?,
135     (element txbx { wp_CT_TextboxInfo }
136      | element linkedTxbx { wp_CT_LinkedTextboxInformation } )?,
137     element bodyPr { a_CT_TextBodyProperties }
138 wp_CT_GraphicFrame =
139     element cNvPr { a_CT_NonVisualDrawingProps },
140     element cNvFrPr { a_CT_NonVisualGraphicFrameProperties },
141     element xfrm { a_CT_Transform2D },
142     a_graphic,
143     element extLst { a_CT_OfficeArtExtensionList }?
144 wp_CT_WordprocessingContentPartNonVisual =
145     element cNvPr { a_CT_NonVisualDrawingProps }?,
146     element cNvContentPartPr { a_CT_NonVisualContentPartProperties }?
147 wp_CT_WordprocessingContentPart =
148     attribute bwMode { a_ST_BlackWhiteMode }?,
149     r_id,
150     element nvContentPartPr { wp_CT_WordprocessingContentPartNonVisual }?,
151     element xfrm { a_CT_Transform2D }?,
152     element extLst { a_CT_OfficeArtExtensionList }?
153 wp_CT_WordprocessingGroup =
154     element cNvPr { a_CT_NonVisualDrawingProps }?,
155     element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps },
156     element grpSpPr { a_CT_GroupShapeProperties },
157     (wp_wsp
158      | element grpSp { wp_CT_WordprocessingGroup }
159      | element graphicFrame { wp_CT_GraphicFrame }
160      | dpct_pic
161      | element contentPart { wp_CT_WordprocessingContentPart } )*,
162     element extLst { a_CT_OfficeArtExtensionList }?
163 wp_CT_WordprocessingCanvas =
164     element bg { a_CT_BackgroundFormatting }?,
165     element whole { a_CT_WholeE2oFormatting }?,
166     (wp_wsp
167      | dpct_pic

```



```

168 | element contentPart { wp_CT_WordprocessingContentPart }
169 | wp_wgp
170 | element graphicFrame { wp_CT_GraphicFrame })*,
171 | element extLst { a_CT_OfficeArtExtensionList }?
172 wp_wpc = element wpc { wp_CT_WordprocessingCanvas }
173 wp_wgp = element wgp { wp_CT_WordprocessingGroup }
174 wp_wsp = element wsp { wp_CT_WordprocessingShape }
175 wp_inline = element inline { wp_CT_Inline }
176 wp_anchor = element anchor { wp_CT_Anchor }

```

B.4.5 DrawingML - Spreadsheet Drawing

This schema is available in the file dml-spreadsheetDrawing.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace o = "urn:schemas-microsoft-com:office:office"
5 namespace r =
6   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
7 namespace v = "urn:schemas-microsoft-com:vml"
8 namespace w10 = "urn:schemas-microsoft-com:office:word"
9 namespace x = "urn:schemas-microsoft-com:office:excel"
10 namespace xdr =
11   "http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
12
13 xdr_from = element from { xdr_CT_Marker }
14 xdr_to = element to { xdr_CT_Marker }
15 xdr_CT_AnchorClientData =
16
17   ## default value: true
18   attribute fLocksWithSheet { xsd:boolean }?,
19
20   ## default value: true
21   attribute fPrintsWithSheet { xsd:boolean }?
22 xdr_CT_ShapeNonVisual =
23   element cNvPr { a_CT_NonVisualDrawingProps },
24   element cNvSpPr { a_CT_NonVisualDrawingShapeProps }
25 xdr_CT_Shape =
26   attribute macro { xsd:string }?,
27   attribute textlink { xsd:string }?,
28
29   ## default value: true
30   attribute fLocksText { xsd:boolean }?,
31
32   ## default value: false
33   attribute fPublished { xsd:boolean }?,
34   element nvSpPr { xdr_CT_ShapeNonVisual },
35   element spPr { a_CT_ShapeProperties },
36   element style { a_CT_ShapeStyle }?,
37   element txBody { a_CT_TextBody }?
38 xdr_CT_ConnectorNonVisual =
39   element cNvPr { a_CT_NonVisualDrawingProps },

```

```

40     element cNvCxnSpPr { a_CT_NonVisualConnectorProperties }
41 xdr_CT_Connector =
42     attribute macro { xsd:string }?,
43
44     ## default value: false
45     attribute fPublished { xsd:boolean }?,
46     element nvCxnSpPr { xdr_CT_ConnectorNonVisual },
47     element spPr { a_CT_ShapeProperties },
48     element style { a_CT_ShapeStyle }?
49 xdr_CT_PictureNonVisual =
50     element cNvPr { a_CT_NonVisualDrawingProps },
51     element cNvPicPr { a_CT_NonVisualPictureProperties }
52 xdr_CT_Picture =
53     attribute macro { xsd:string }?,
54
55     ## default value: false
56     attribute fPublished { xsd:boolean }?,
57     element nvPicPr { xdr_CT_PictureNonVisual },
58     element blipFill { a_CT_BlipFillProperties },
59     element spPr { a_CT_ShapeProperties },
60     element style { a_CT_ShapeStyle }?
61 xdr_CT_GraphicalObjectFrameNonVisual =
62     element cNvPr { a_CT_NonVisualDrawingProps },
63     element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }
64 xdr_CT_GraphicalObjectFrame =
65     attribute macro { xsd:string }?,
66
67     ## default value: false
68     attribute fPublished { xsd:boolean }?,
69     element nvGraphicFramePr { xdr_CT_GraphicalObjectFrameNonVisual },
70     element xfrm { a_CT_Transform2D },
71     a_graphic
72 xdr_CT_GroupShapeNonVisual =
73     element cNvPr { a_CT_NonVisualDrawingProps },
74     element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps }
75 xdr_CT_GroupShape =
76     element nvGrpSpPr { xdr_CT_GroupShapeNonVisual },
77     element grpSpPr { a_CT_GroupShapeProperties },
78     (element sp { xdr_CT_Shape }
79     | element grpSp { xdr_CT_GroupShape }
80     | element graphicFrame { xdr_CT_GraphicalObjectFrame }
81     | element cxnSp { xdr_CT_Connector }
82     | element pic { xdr_CT_Picture })*
83 xdr_EG_ObjectChoices =
84     element sp { xdr_CT_Shape }
85     | element grpSp { xdr_CT_GroupShape }
86     | element graphicFrame { xdr_CT_GraphicalObjectFrame }
87     | element cxnSp { xdr_CT_Connector }
88     | element pic { xdr_CT_Picture }
89     | element contentPart { xdr_CT_Rel }
90 xdr_CT_Rel = r_id
91 xdr_ST_ColID = xsd:int { minInclusive = "0" }
92 xdr_ST_RowID = xsd:int { minInclusive = "0" }

```

```

93 xdr_CT_Marker =
94     element col { xdr_ST_ColID },
95     element colOff { a_ST_Coordinate },
96     element row { xdr_ST_RowID },
97     element rowOff { a_ST_Coordinate }
98 xdr_ST_EditAs = "twoCell" | "oneCell" | "absolute"
99 xdr_CT_TwoCellAnchor =
100
101     ## default value: twoCell
102     attribute editAs { xdr_ST_EditAs }?,
103     element from { xdr_CT_Marker },
104     element to { xdr_CT_Marker },
105     xdr_EG_ObjectChoices,
106     element clientData { xdr_CT_AnchorClientData }
107 xdr_CT_OneCellAnchor =
108     element from { xdr_CT_Marker },
109     element ext { a_CT_PositiveSize2D },
110     xdr_EG_ObjectChoices,
111     element clientData { xdr_CT_AnchorClientData }
112 xdr_CT_AbsoluteAnchor =
113     element pos { a_CT_Point2D },
114     element ext { a_CT_PositiveSize2D },
115     xdr_EG_ObjectChoices,
116     element clientData { xdr_CT_AnchorClientData }
117 xdr_EG_Anchor =
118     element twoCellAnchor { xdr_CT_TwoCellAnchor }
119     | element oneCellAnchor { xdr_CT_OneCellAnchor }
120     | element absoluteAnchor { xdr_CT_AbsoluteAnchor }
121 xdr_CT_Drawing = xdr_EG_Anchor*
122 xdr_wsDr = element wsDr { xdr_CT_Drawing }

```

B.5 DrawingML - Components

B.5.1 DrawingML - Chart

This schema is available in the file dml-chart.rnc.

```

1 default namespace =
2     "http://schemas.openxmlformats.org/drawingml/2006/chart"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace cdr =
5     "http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
6 namespace dchrt =
7     "http://schemas.openxmlformats.org/drawingml/2006/chart"
8 namespace o = "urn:schemas-microsoft-com:office:office"
9 namespace r =
10     "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
11 namespace s =
12     "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
13 namespace v = "urn:schemas-microsoft-com:vml"
14 namespace w10 = "urn:schemas-microsoft-com:office:word"
15 namespace x = "urn:schemas-microsoft-com:office:excel"
16

```

```

17 dchrt_CT_Boolean =
18
19     ## default value: true
20     attribute val { xsd:boolean }?
21 dchrt_CT_Double = attribute val { xsd:double }
22 dchrt_CT_UnsignedInt = attribute val { xsd:unsignedInt }
23 dchrt_CT_RelId = r_id
24 dchrt_CT_Extension =
25     attribute uri { xsd:token }?,
26     dchrt_CT_Extension_any
27 dchrt_CT_Extension_any =
28     element * - (o:* | v:* | w10:* | x:*) {
29         anyAttribute*,
30         mixed { anyElement* }
31     }
32 dchrt_CT_ExtensionList = element ext { dchrt_CT_Extension }*
33 dchrt_CT_NumVal =
34     attribute idx { xsd:unsignedInt },
35     attribute formatCode { s_ST_Xstring }?,
36     element v { s_ST_Xstring }
37 dchrt_CT_NumData =
38     element formatCode { s_ST_Xstring }?,
39     element ptCount { dchrt_CT_UnsignedInt }?,
40     element pt { dchrt_CT_NumVal }*,
41     element extLst { dchrt_CT_ExtensionList }?
42 dchrt_CT_NumRef =
43     element f { xsd:string },
44     element numCache { dchrt_CT_NumData }?,
45     element extLst { dchrt_CT_ExtensionList }?
46 dchrt_CT_NumDataSource =
47     element numRef { dchrt_CT_NumRef }
48     | element numLit { dchrt_CT_NumData }
49 dchrt_CT_StrVal =
50     attribute idx { xsd:unsignedInt },
51     element v { s_ST_Xstring }
52 dchrt_CT_StrData =
53     element ptCount { dchrt_CT_UnsignedInt }?,
54     element pt { dchrt_CT_StrVal }*,
55     element extLst { dchrt_CT_ExtensionList }?
56 dchrt_CT_StrRef =
57     element f { xsd:string },
58     element strCache { dchrt_CT_StrData }?,
59     element extLst { dchrt_CT_ExtensionList }?
60 dchrt_CT_Tx =
61     element strRef { dchrt_CT_StrRef }
62     | element rich { a_CT_TextBody }
63 dchrt_CT_TextLanguageID = attribute val { s_ST_Lang }
64 dchrt_CT_Lvl = element pt { dchrt_CT_StrVal }*
65 dchrt_CT_MultiLvlStrData =
66     element ptCount { dchrt_CT_UnsignedInt }?,
67     element lvl { dchrt_CT_Lvl }*,
68     element extLst { dchrt_CT_ExtensionList }?
69 dchrt_CT_MultiLvlStrRef =

```

```

70     element f { xsd:string },
71     element multiLvlStrCache { dchrt_CT_MultiLvlStrData }?,
72     element extLst { dchrt_CT_ExtensionList }?
73 dchrt_CT_AxDataSource =
74     element multiLvlStrRef { dchrt_CT_MultiLvlStrRef }
75     | element numRef { dchrt_CT_NumRef }
76     | element numLit { dchrt_CT_NumData }
77     | element strRef { dchrt_CT_StrRef }
78     | element strLit { dchrt_CT_StrData }
79 dchrt_CT_SerTx =
80     element strRef { dchrt_CT_StrRef }
81     | element v { s_ST_Xstring }
82 dchrt_ST_LayoutTarget = string "inner" | string "outer"
83 dchrt_CT_LayoutTarget =
84
85     ## default value: outer
86     attribute val { dchrt_ST_LayoutTarget }?
87 dchrt_ST_LayoutMode = string "edge" | string "factor"
88 dchrt_CT_LayoutMode =
89
90     ## default value: factor
91     attribute val { dchrt_ST_LayoutMode }?
92 dchrt_CT_ManualLayout =
93     element layoutTarget { dchrt_CT_LayoutTarget }?,
94     element xMode { dchrt_CT_LayoutMode }?,
95     element yMode { dchrt_CT_LayoutMode }?,
96     element wMode { dchrt_CT_LayoutMode }?,
97     element hMode { dchrt_CT_LayoutMode }?,
98     element x { dchrt_CT_Double }?,
99     element y { dchrt_CT_Double }?,
100    element w { dchrt_CT_Double }?,
101    element h { dchrt_CT_Double }?,
102    element extLst { dchrt_CT_ExtensionList }?
103 dchrt_CT_Layout =
104     element manualLayout { dchrt_CT_ManualLayout }?,
105     element extLst { dchrt_CT_ExtensionList }?
106 dchrt_CT_Title =
107     element tx { dchrt_CT_Tx }?,
108     element layout { dchrt_CT_Layout }?,
109     element overlay { dchrt_CT_Boolean }?,
110     element spPr { a_CT_ShapeProperties }?,
111     element txPr { a_CT_TextBody }?,
112     element extLst { dchrt_CT_ExtensionList }?
113 dchrt_ST_RotX = xsd:byte { minInclusive = "-90" maxInclusive = "90" }
114 dchrt_CT_RotX =
115
116     ## default value: 0
117     attribute val { dchrt_ST_RotX }?
118 dchrt_ST_HPercent =
119 dchrt_ST_HPercentWithSymbol | dchrt_ST_HPercentUShort
120 dchrt_ST_HPercentWithSymbol =
121     xsd:string {
122         pattern = "0*(([5-9])|([1-9][0-9])|([1-4][0-9][0-9])|500)%"

```

```

123 }
124 dchrt_ST_HPercentUShort =
125   xsd:unsignedShort { minInclusive = "5" maxInclusive = "500" }
126 dchrt_CT_HPercent =
127
128   ## default value: 100%
129   attribute val { dchrt_ST_HPercent }?
130 dchrt_ST_RotY =
131   xsd:unsignedShort { minInclusive = "0" maxInclusive = "360" }
132 dchrt_CT_RotY =
133
134   ## default value: 0
135   attribute val { dchrt_ST_RotY }?
136 dchrt_ST_DepthPercent =
137 dchrt_ST_DepthPercentWithSymbol | dchrt_ST_DepthPercentUShort
138 dchrt_ST_DepthPercentWithSymbol =
139   xsd:string {
140     pattern = "0*(([2-9][0-9])|([1-9][0-9][0-9])|(1[0-9][0-9][0-9])|2000)%"
141   }
142 dchrt_ST_DepthPercentUShort =
143   xsd:unsignedShort { minInclusive = "20" maxInclusive = "2000" }
144 dchrt_CT_DepthPercent =
145
146   ## default value: 100%
147   attribute val { dchrt_ST_DepthPercent }?
148 dchrt_ST_Perspective =
149   xsd:unsignedByte { minInclusive = "0" maxInclusive = "240" }
150 dchrt_CT_Perspective =
151
152   ## default value: 30
153   attribute val { dchrt_ST_Perspective }?
154 dchrt_CT_View3D =
155   element rotX { dchrt_CT_RotX }?,
156   element hPercent { dchrt_CT_HPercent }?,
157   element rotY { dchrt_CT_RotY }?,
158   element depthPercent { dchrt_CT_DepthPercent }?,
159   element rAngAx { dchrt_CT_Boolean }?,
160   element perspective { dchrt_CT_Perspective }?,
161   element extLst { dchrt_CT_ExtensionList }?
162 dchrt_CT_Surface =
163   element thickness { dchrt_CT_Thickness }?,
164   element spPr { a_CT_ShapeProperties }?,
165   element pictureOptions { dchrt_CT_PictureOptions }?,
166   element extLst { dchrt_CT_ExtensionList }?
167 dchrt_ST_Thickness = dchrt_ST_ThicknessPercent | xsd:unsignedInt
168 dchrt_ST_ThicknessPercent = xsd:string { pattern = "([0-9]+)%" }
169 dchrt_CT_Thickness = attribute val { dchrt_ST_Thickness }
170 dchrt_CT_DTable =
171   element showHorzBorder { dchrt_CT_Boolean }?,
172   element showVertBorder { dchrt_CT_Boolean }?,
173   element showOutline { dchrt_CT_Boolean }?,
174   element showKeys { dchrt_CT_Boolean }?,
175   element spPr { a_CT_ShapeProperties }?,

```

```

176     element txPr { a_CT_TextBody }?,
177     element extLst { dchrt_CT_ExtensionList }?
178 dchrt_ST_GapAmount =
179 dchrt_ST_GapAmountPercent | dchrt_ST_GapAmountUShort
180 dchrt_ST_GapAmountPercent =
181     xsd:string {
182         pattern = "0*(([0-9])|([1-9][0-9])|([1-4][0-9][0-9])|500)%"
183     }
184 dchrt_ST_GapAmountUShort =
185     xsd:unsignedShort { minInclusive = "0" maxInclusive = "500" }
186 dchrt_CT_GapAmount =
187
188     ## default value: 150%
189     attribute val { dchrt_ST_GapAmount }?
190 dchrt_ST_Overlap =
191     dchrt_ST_OverlapPercent | dchrt_ST_OverlapByte
192 dchrt_ST_OverlapPercent =
193     xsd:string { pattern = "(-?0*(([0-9])|([1-9][0-9])|100))%" }
194 dchrt_ST_OverlapByte =
195     xsd:byte { minInclusive = "-100" maxInclusive = "100" }
196 dchrt_CT_Overlap =
197
198     ## default value: 0%
199     attribute val { dchrt_ST_Overlap }?
200 dchrt_ST_BubbleScale =
201     dchrt_ST_BubbleScalePercent | dchrt_ST_BubbleScaleUInt
202 dchrt_ST_BubbleScalePercent =
203     xsd:string {
204         pattern = "0*(([0-9])|([1-9][0-9])|([1-2][0-9][0-9])|300)%"
205     }
206 dchrt_ST_BubbleScaleUInt =
207     xsd:unsignedInt { minInclusive = "0" maxInclusive = "300" }
208 dchrt_CT_BubbleScale =
209
210     ## default value: 100%
211     attribute val { dchrt_ST_BubbleScale }?
212 dchrt_ST_SizeRepresents = string "area" | string "w"
213 dchrt_CT_SizeRepresents =
214
215     ## default value: area
216     attribute val { dchrt_ST_SizeRepresents }?
217 dchrt_ST_FirstSliceAng =
218     xsd:unsignedShort { minInclusive = "0" maxInclusive = "360" }
219 dchrt_CT_FirstSliceAng =
220
221     ## default value: 0
222     attribute val { dchrt_ST_FirstSliceAng }?
223 dchrt_ST_HoleSize =
224     dchrt_ST_HoleSizePercent | dchrt_ST_HoleSizeUByte
225 dchrt_ST_HoleSizePercent = xsd:string { pattern = "0*([1-9]|([1-8][0-9])|90)%" }
226 dchrt_ST_HoleSizeUByte = xsd:unsignedByte { minInclusive = "1" maxInclusive = "90" }
227 dchrt_CT_HoleSize =
228

```

```

229   ## default value: 10%
230   attribute val { dchrt_ST_HoleSize }?
231 dchrt_ST_SplitType =
232   string "auto"
233   | string "cust"
234   | string "percent"
235   | string "pos"
236   | string "val"
237 dchrt_CT_SplitType =
238
239   ## default value: auto
240   attribute val { dchrt_ST_SplitType }?
241 dchrt_CT_CustSplit = element secondPiePt { dchrt_CT_UnsignedInt }*
242 dchrt_ST_SecondPieSize =
243 dchrt_ST_SecondPieSizePercent | dchrt_ST_SecondPieSizeUShort
244 dchrt_ST_SecondPieSizePercent =
245   xsd:string { pattern = "0*(([5-9])|([1-9][0-9])|(1[0-9][0-9])|200)%" }
246 dchrt_ST_SecondPieSizeUShort =
247   xsd:unsignedShort { minInclusive = "5" maxInclusive = "200" }
248 dchrt_CT_SecondPieSize =
249
250   ## default value: 75%
251   attribute val { dchrt_ST_SecondPieSize }?
252 dchrt_CT_NumFmt =
253   attribute formatCode { s_ST_Xstring },
254   attribute sourceLinked { xsd:boolean }?
255 dchrt_ST_LblAlgn = string "ctr" | string "l" | string "r"
256 dchrt_CT_LblAlgn = attribute val { dchrt_ST_LblAlgn }
257 dchrt_ST_DLblPos =
258   string "bestFit"
259   | string "b"
260   | string "ctr"
261   | string "inBase"
262   | string "inEnd"
263   | string "l"
264   | string "outEnd"
265   | string "r"
266   | string "t"
267 dchrt_CT_DLblPos = attribute val { dchrt_ST_DLblPos }
268 dchrt_EG_DLblShared =
269   element numFmt { dchrt_CT_NumFmt }?,
270   element spPr { a_CT_ShapeProperties }?,
271   element txPr { a_CT_TextBody }?,
272   element dLblPos { dchrt_CT_DLblPos }?,
273   element showLegendKey { dchrt_CT_Boolean }?,
274   element showVal { dchrt_CT_Boolean }?,
275   element showCatName { dchrt_CT_Boolean }?,
276   element showSerName { dchrt_CT_Boolean }?,
277   element showPercent { dchrt_CT_Boolean }?,
278   element showBubbleSize { dchrt_CT_Boolean }?,
279   element separator { xsd:string }?
280 dchrt_Group_DLbl =
281   element layout { dchrt_CT_Layout }?,

```



```

282     element tx { dchrt_CT_Tx }?,
283     dchrt_EG_DLblShared
284 dchrt_CT_DLbl =
285     element idx { dchrt_CT_UnsignedInt },
286     (element delete { dchrt_CT_Boolean }
287      | dchrt_Group_DLbl),
288     element extLst { dchrt_CT_ExtensionList }?
289 dchrt_Group_DLbls =
290     dchrt_EG_DLblShared,
291     element showLeaderLines { dchrt_CT_Boolean }?,
292     element leaderLines { dchrt_CT_ChartLines }?
293 dchrt_CT_DLbls =
294     element dlbl { dchrt_CT_DLbl }*,
295     (element delete { dchrt_CT_Boolean }
296      | dchrt_Group_DLbls),
297     element extLst { dchrt_CT_ExtensionList }?
298 dchrt_ST_MarkerStyle =
299     string "circle"
300     | string "dash"
301     | string "diamond"
302     | string "dot"
303     | string "none"
304     | string "picture"
305     | string "plus"
306     | string "square"
307     | string "star"
308     | string "triangle"
309     | string "x"
310     | string "auto"
311 dchrt_CT_MarkerStyle = attribute val { dchrt_ST_MarkerStyle }
312 dchrt_ST_MarkerSize =
313     xsd:unsignedByte { minInclusive = "2" maxInclusive = "72" }
314 dchrt_CT_MarkerSize =
315
316     ## default value: 5
317     attribute val { dchrt_ST_MarkerSize }?
318 dchrt_CT_Marker =
319     element symbol { dchrt_CT_MarkerStyle }?,
320     element size { dchrt_CT_MarkerSize }?,
321     element spPr { a_CT_ShapeProperties }?,
322     element extLst { dchrt_CT_ExtensionList }?
323 dchrt_CT_DPt =
324     element idx { dchrt_CT_UnsignedInt },
325     element invertIfNegative { dchrt_CT_Boolean }?,
326     element marker { dchrt_CT_Marker }?,
327     element bubble3D { dchrt_CT_Boolean }?,
328     element explosion { dchrt_CT_UnsignedInt }?,
329     element spPr { a_CT_ShapeProperties }?,
330     element pictureOptions { dchrt_CT_PictureOptions }?,
331     element extLst { dchrt_CT_ExtensionList }?
332 dchrt_ST_TrendlineType =
333     string "exp"
334     | string "linear"

```

```

335 | string "log"
336 | string "movingAvg"
337 | string "poly"
338 | string "power"
339 dchrt_CT_TrendlineType =
340
341     ## default value: linear
342     attribute val { dchrt_ST_TrendlineType }?
343 dchrt_ST_Order =
344     xsd:unsignedByte { minInclusive = "2" maxInclusive = "6" }
345 dchrt_CT_Order =
346
347     ## default value: 2
348     attribute val { dchrt_ST_Order }?
349 dchrt_ST_Period =
350     xsd:unsignedInt { minInclusive = "2" }
351 dchrt_CT_Period =
352
353     ## default value: 2
354     attribute val { dchrt_ST_Period }?
355 dchrt_CT_TrendlineLbl =
356     element layout { dchrt_CT_Layout }?,
357     element tx { dchrt_CT_Tx }?,
358     element numFmt { dchrt_CT_NumFmt }?,
359     element spPr { a_CT_ShapeProperties }?,
360     element txPr { a_CT_TextBody }?,
361     element extLst { dchrt_CT_ExtensionList }?
362 dchrt_CT_Trendline =
363     element name { xsd:string }?,
364     element spPr { a_CT_ShapeProperties }?,
365     element trendlineType { dchrt_CT_TrendlineType },
366     element order { dchrt_CT_Order }?,
367     element period { dchrt_CT_Period }?,
368     element forward { dchrt_CT_Double }?,
369     element backward { dchrt_CT_Double }?,
370     element intercept { dchrt_CT_Double }?,
371     element dispRSqr { dchrt_CT_Boolean }?,
372     element dispEq { dchrt_CT_Boolean }?,
373     element trendlineLbl { dchrt_CT_TrendlineLbl }?,
374     element extLst { dchrt_CT_ExtensionList }?
375 dchrt_ST_ErrDir = string "x" | string "y"
376 dchrt_CT_ErrDir = attribute val { dchrt_ST_ErrDir }
377 dchrt_ST_ErrBarType = string "both" | string "minus" | string "plus"
378 dchrt_CT_ErrBarType =
379
380     ## default value: both
381     attribute val { dchrt_ST_ErrBarType }?
382 dchrt_ST_ErrValType =
383     string "cust"
384     | string "fixedVal"
385     | string "percentage"
386     | string "stdDev"
387     | string "stdErr"

```

```

388 dchrt_CT_ErrValType =
389
390   ## default value: fixedVal
391   attribute val { dchrt_ST_ErrValType }?
392 dchrt_CT_ErrBars =
393   element errDir { dchrt_CT_ErrDir }?,
394   element errBarType { dchrt_CT_ErrBarType },
395   element errValType { dchrt_CT_ErrValType },
396   element noEndCap { dchrt_CT_Boolean }?,
397   element plus { dchrt_CT_NumDataSource }?,
398   element minus { dchrt_CT_NumDataSource }?,
399   element val { dchrt_CT_Double }?,
400   element spPr { a_CT_ShapeProperties }?,
401   element extLst { dchrt_CT_ExtensionList }?
402 dchrt_CT_UpDownBar = element spPr { a_CT_ShapeProperties }?
403 dchrt_CT_UpDownBars =
404   element gapWidth { dchrt_CT_GapAmount }?,
405   element upBars { dchrt_CT_UpDownBar }?,
406   element downBars { dchrt_CT_UpDownBar }?,
407   element extLst { dchrt_CT_ExtensionList }?
408 dchrt_EG_SerShared =
409   element idx { dchrt_CT_UnsignedInt },
410   element order { dchrt_CT_UnsignedInt },
411   element tx { dchrt_CT_SerTx }?,
412   element spPr { a_CT_ShapeProperties }?
413 dchrt_CT_LineSer =
414   dchrt_EG_SerShared,
415   element marker { dchrt_CT_Marker }?,
416   element dPt { dchrt_CT_DPt }*,
417   element dLbls { dchrt_CT_DLbls }?,
418   element trendline { dchrt_CT_Trendline }*,
419   element errBars { dchrt_CT_ErrBars }?,
420   element cat { dchrt_CT_AxDataSource }?,
421   element val { dchrt_CT_NumDataSource }?,
422   element smooth { dchrt_CT_Boolean }?,
423   element extLst { dchrt_CT_ExtensionList }?
424 dchrt_CT_ScatterSer =
425   dchrt_EG_SerShared,
426   element marker { dchrt_CT_Marker }?,
427   element dPt { dchrt_CT_DPt }*,
428   element dLbls { dchrt_CT_DLbls }?,
429   element trendline { dchrt_CT_Trendline }*,
430   element errBars { dchrt_CT_ErrBars }*,
431   element xVal { dchrt_CT_AxDataSource }?,
432   element yVal { dchrt_CT_NumDataSource }?,
433   element smooth { dchrt_CT_Boolean }?,
434   element extLst { dchrt_CT_ExtensionList }?
435 dchrt_CT_RadarSer =
436   dchrt_EG_SerShared,
437   element marker { dchrt_CT_Marker }?,
438   element dPt { dchrt_CT_DPt }*,
439   element dLbls { dchrt_CT_DLbls }?,
440   element cat { dchrt_CT_AxDataSource }?,

```

```

441     element val { dchrt_CT_NumDataSource }?,
442     element extLst { dchrt_CT_ExtensionList }?
443 dchrt_CT_BarSer =
444     dchrt_EG_SerShared,
445     element invertIfNegative { dchrt_CT_Boolean }?,
446     element pictureOptions { dchrt_CT_PictureOptions }?,
447     element dPt { dchrt_CT_DPt }*,
448     element dLbls { dchrt_CT_DLbLs }?,
449     element trendline { dchrt_CT_Trendline }*,
450     element errBars { dchrt_CT_ErrBars }?,
451     element cat { dchrt_CT_AxDataSource }?,
452     element val { dchrt_CT_NumDataSource }?,
453     element shape { dchrt_CT_Shape }?,
454     element extLst { dchrt_CT_ExtensionList }?
455 dchrt_CT_AreaSer =
456     dchrt_EG_SerShared,
457     element pictureOptions { dchrt_CT_PictureOptions }?,
458     element dPt { dchrt_CT_DPt }*,
459     element dLbls { dchrt_CT_DLbLs }?,
460     element trendline { dchrt_CT_Trendline }*,
461     element errBars { dchrt_CT_ErrBars }*,
462     element cat { dchrt_CT_AxDataSource }?,
463     element val { dchrt_CT_NumDataSource }?,
464     element extLst { dchrt_CT_ExtensionList }?
465 dchrt_CT_PieSer =
466     dchrt_EG_SerShared,
467     element explosion { dchrt_CT_UnsignedInt }?,
468     element dPt { dchrt_CT_DPt }*,
469     element dLbls { dchrt_CT_DLbLs }?,
470     element cat { dchrt_CT_AxDataSource }?,
471     element val { dchrt_CT_NumDataSource }?,
472     element extLst { dchrt_CT_ExtensionList }?
473 dchrt_CT_BubbleSer =
474     dchrt_EG_SerShared,
475     element invertIfNegative { dchrt_CT_Boolean }?,
476     element dPt { dchrt_CT_DPt }*,
477     element dLbls { dchrt_CT_DLbLs }?,
478     element trendline { dchrt_CT_Trendline }*,
479     element errBars { dchrt_CT_ErrBars }*,
480     element xVal { dchrt_CT_AxDataSource }?,
481     element yVal { dchrt_CT_NumDataSource }?,
482     element bubbleSize { dchrt_CT_NumDataSource }?,
483     element bubble3D { dchrt_CT_Boolean }?,
484     element extLst { dchrt_CT_ExtensionList }?
485 dchrt_CT_SurfaceSer =
486     dchrt_EG_SerShared,
487     element cat { dchrt_CT_AxDataSource }?,
488     element val { dchrt_CT_NumDataSource }?,
489     element extLst { dchrt_CT_ExtensionList }?
490 dchrt_ST_Grouping =
491     string "percentStacked" | string "standard" | string "stacked"
492 dchrt_CT_Grouping =
493

```

```

494   ## default value: standard
495   attribute val { dchrt_ST_Grouping }?
496 dchrt_CT_ChartLines = element spPr { a_CT_ShapeProperties }?
497 dchrt_EG_LineChartShared =
498   element grouping { dchrt_CT_Grouping },
499   element varyColors { dchrt_CT_Boolean }?,
500   element ser { dchrt_CT_LineSer }*,
501   element dLbIs { dchrt_CT_DLbIs }?,
502   element dropLines { dchrt_CT_ChartLines }?
503 dchrt_CT_LineChart =
504   dchrt_EG_LineChartShared,
505   element hiLowLines { dchrt_CT_ChartLines }?,
506   element upDownBars { dchrt_CT_UpDownBars }?,
507   element marker { dchrt_CT_Boolean }?,
508   element smooth { dchrt_CT_Boolean }?,
509   element axId { dchrt_CT_UnsignedInt }+,
510   element extLst { dchrt_CT_ExtensionList }?
511 dchrt_CT_Line3DChart =
512   dchrt_EG_LineChartShared,
513   element gapDepth { dchrt_CT_GapAmount }?,
514   element axId { dchrt_CT_UnsignedInt }+,
515   element extLst { dchrt_CT_ExtensionList }?
516 dchrt_CT_StockChart =
517   element ser { dchrt_CT_LineSer }+,
518   element dLbIs { dchrt_CT_DLbIs }?,
519   element dropLines { dchrt_CT_ChartLines }?,
520   element hiLowLines { dchrt_CT_ChartLines }?,
521   element upDownBars { dchrt_CT_UpDownBars }?,
522   element axId { dchrt_CT_UnsignedInt }+,
523   element extLst { dchrt_CT_ExtensionList }?
524 dchrt_ST_ScatterStyle =
525   string "none"
526   | string "line"
527   | string "lineMarker"
528   | string "marker"
529   | string "smooth"
530   | string "smoothMarker"
531 dchrt_CT_ScatterStyle =
532
533   ## default value: marker
534   attribute val { dchrt_ST_ScatterStyle }?
535 dchrt_CT_ScatterChart =
536   element scatterStyle { dchrt_CT_ScatterStyle },
537   element varyColors { dchrt_CT_Boolean }?,
538   element ser { dchrt_CT_ScatterSer }*,
539   element dLbIs { dchrt_CT_DLbIs }?,
540   element axId { dchrt_CT_UnsignedInt }+,
541   element extLst { dchrt_CT_ExtensionList }?
542 dchrt_ST_RadarStyle =
543   string "standard" | string "marker" | string "filled"
544 dchrt_CT_RadarStyle =
545
546   ## default value: standard

```

```

547     attribute val { dchrt_ST_RadarStyle }?
548 dchrt_CT_RadarChart =
549     element radarStyle { dchrt_CT_RadarStyle },
550     element varyColors { dchrt_CT_Boolean }?,
551     element ser { dchrt_CT_RadarSer }*,
552     element dLbIs { dchrt_CT_DLbIs }?,
553     element axId { dchrt_CT_UnsignedInt }+,
554     element extLst { dchrt_CT_ExtensionList }?
555 dchrt_ST_BarGrouping =
556     string "percentStacked"
557     | string "clustered"
558     | string "standard"
559     | string "stacked"
560 dchrt_CT_BarGrouping =
561
562     ## default value: clustered
563     attribute val { dchrt_ST_BarGrouping }?
564 dchrt_ST_BarDir = string "bar" | string "col"
565 dchrt_CT_BarDir =
566
567     ## default value: col
568     attribute val { dchrt_ST_BarDir }?
569 dchrt_ST_Shape =
570     string "cone"
571     | string "coneToMax"
572     | string "box"
573     | string "cylinder"
574     | string "pyramid"
575     | string "pyramidToMax"
576 dchrt_CT_Shape =
577
578     ## default value: box
579     attribute val { dchrt_ST_Shape }?
580 dchrt_EG_BarChartShared =
581     element barDir { dchrt_CT_BarDir },
582     element grouping { dchrt_CT_BarGrouping }?,
583     element varyColors { dchrt_CT_Boolean }?,
584     element ser { dchrt_CT_BarSer }*,
585     element dLbIs { dchrt_CT_DLbIs }?
586 dchrt_CT_BarChart =
587     dchrt_EG_BarChartShared,
588     element gapWidth { dchrt_CT_GapAmount }?,
589     element overlap { dchrt_CT_Overlap }?,
590     element serLines { dchrt_CT_ChartLines }*,
591     element axId { dchrt_CT_UnsignedInt }+,
592     element extLst { dchrt_CT_ExtensionList }?
593 dchrt_CT_Bar3DChart =
594     dchrt_EG_BarChartShared,
595     element gapWidth { dchrt_CT_GapAmount }?,
596     element gapDepth { dchrt_CT_GapAmount }?,
597     element shape { dchrt_CT_Shape }?,
598     element axId { dchrt_CT_UnsignedInt }+,
599     element extLst { dchrt_CT_ExtensionList }?

```

```

600 dchrt_EG_AreaChartShared =
601     element grouping { dchrt_CT_Grouping }?,
602     element varyColors { dchrt_CT_Boolean }?,
603     element ser { dchrt_CT_AreaSer }*,
604     element dLbIs { dchrt_CT_DLbIs }?,
605     element dropLines { dchrt_CT_ChartLines }?
606 dchrt_CT_AreaChart =
607     dchrt_EG_AreaChartShared,
608     element axId { dchrt_CT_UnsignedInt }+,
609     element extLst { dchrt_CT_ExtensionList }?
610 dchrt_CT_Area3DChart =
611     dchrt_EG_AreaChartShared,
612     element gapDepth { dchrt_CT_GapAmount }?,
613     element axId { dchrt_CT_UnsignedInt }+,
614     element extLst { dchrt_CT_ExtensionList }?
615 dchrt_EG_PieChartShared =
616     element varyColors { dchrt_CT_Boolean }?,
617     element ser { dchrt_CT_PieSer }*,
618     element dLbIs { dchrt_CT_DLbIs }?
619 dchrt_CT_PieChart =
620     dchrt_EG_PieChartShared,
621     element firstSliceAng { dchrt_CT_FirstSliceAng }?,
622     element extLst { dchrt_CT_ExtensionList }?
623 dchrt_CT_Pie3DChart =
624     dchrt_EG_PieChartShared,
625     element extLst { dchrt_CT_ExtensionList }?
626 dchrt_CT_DoughnutChart =
627     dchrt_EG_PieChartShared,
628     element firstSliceAng { dchrt_CT_FirstSliceAng }?,
629     element holeSize { dchrt_CT_HoleSize }?,
630     element extLst { dchrt_CT_ExtensionList }?
631 dchrt_ST_OfPieType = string "pie" | string "bar"
632 dchrt_CT_OfPieType =
633
634     ## default value: pie
635     attribute val { dchrt_ST_OfPieType }?
636 dchrt_CT_OfPieChart =
637     element ofPieType { dchrt_CT_OfPieType },
638     dchrt_EG_PieChartShared,
639     element gapWidth { dchrt_CT_GapAmount }?,
640     element splitType { dchrt_CT_SplitType }?,
641     element splitPos { dchrt_CT_Double }?,
642     element custSplit { dchrt_CT_CustSplit }?,
643     element secondPieSize { dchrt_CT_SecondPieSize }?,
644     element serLines { dchrt_CT_ChartLines }*,
645     element extLst { dchrt_CT_ExtensionList }?
646 dchrt_CT_BubbleChart =
647     element varyColors { dchrt_CT_Boolean }?,
648     element ser { dchrt_CT_BubbleSer }*,
649     element dLbIs { dchrt_CT_DLbIs }?,
650     element bubble3D { dchrt_CT_Boolean }?,
651     element bubbleScale { dchrt_CT_BubbleScale }?,
652     element showNegBubbles { dchrt_CT_Boolean }?,

```

```

653     element sizeRepresents { dchrt_CT_SizeRepresents }?,
654     element axId { dchrt_CT_UnsignedInt }+,
655     element extLst { dchrt_CT_ExtensionList }?
656 dchrt_CT_BandFmt =
657     element idx { dchrt_CT_UnsignedInt },
658     element spPr { a_CT_ShapeProperties }?
659 dchrt_CT_BandFmts = element bandFmt { dchrt_CT_BandFmt }*
660 dchrt_EG_SurfaceChartShared =
661     element wireframe { dchrt_CT_Boolean }?,
662     element ser { dchrt_CT_SurfaceSer }*,
663     element bandFmts { dchrt_CT_BandFmts }?
664 dchrt_CT_SurfaceChart =
665     dchrt_EG_SurfaceChartShared,
666     element axId { dchrt_CT_UnsignedInt }+,
667     element extLst { dchrt_CT_ExtensionList }?
668 dchrt_CT_Surface3DChart =
669     dchrt_EG_SurfaceChartShared,
670     element axId { dchrt_CT_UnsignedInt }+,
671     element extLst { dchrt_CT_ExtensionList }?
672 dchrt_ST_AxPos = string "b" | string "l" | string "r" | string "t"
673 dchrt_CT_AxPos = attribute val { dchrt_ST_AxPos }
674 dchrt_ST_Crosses = string "autoZero" | string "max" | string "min"
675 dchrt_CT_Crosses = attribute val { dchrt_ST_Crosses }
676 dchrt_ST_CrossBetween = string "between" | string "midCat"
677 dchrt_CT_CrossBetween = attribute val { dchrt_ST_CrossBetween }
678 dchrt_ST_TickMark =
679     string "cross" | string "in" | string "none" | string "out"
680 dchrt_CT_TickMark =
681
682     ## default value: cross
683     attribute val { dchrt_ST_TickMark }?
684 dchrt_ST_TickLblPos =
685     string "high" | string "low" | string "nextTo" | string "none"
686 dchrt_CT_TickLblPos =
687
688     ## default value: nextTo
689     attribute val { dchrt_ST_TickLblPos }?
690 dchrt_ST_Skip = xsd:unsignedInt { minInclusive = "1" }
691 dchrt_CT_Skip = attribute val { dchrt_ST_Skip }
692 dchrt_ST_TimeUnit = string "days" | string "months" | string "years"
693 dchrt_CT_TimeUnit =
694
695     ## default value: days
696     attribute val { dchrt_ST_TimeUnit }?
697 dchrt_ST_AxisUnit = xsd:double { minExclusive = "0" }
698 dchrt_CT_AxisUnit = attribute val { dchrt_ST_AxisUnit }
699 dchrt_ST_BuiltInUnit =
700     string "hundreds"
701     | string "thousands"
702     | string "tenThousands"
703     | string "hundredThousands"
704     | string "millions"
705     | string "tenMillions"

```



```

706 | string "hundredMillions"
707 | string "billions"
708 | string "trillions"
709 dchrt_CT_BuiltInUnit =
710
711   ## default value: thousands
712   attribute val { dchrt_ST_BuiltInUnit }?
713 dchrt_ST_PictureFormat =
714   string "stretch" | string "stack" | string "stackScale"
715 dchrt_CT_PictureFormat = attribute val { dchrt_ST_PictureFormat }
716 dchrt_ST_PictureStackUnit = xsd:double { minExclusive = "0" }
717 dchrt_CT_PictureStackUnit = attribute val { dchrt_ST_PictureStackUnit }
718 dchrt_CT_PictureOptions =
719   element applyToFront { dchrt_CT_Boolean }?,
720   element applyToSides { dchrt_CT_Boolean }?,
721   element applyToEnd { dchrt_CT_Boolean }?,
722   element pictureFormat { dchrt_CT_PictureFormat }?,
723   element pictureStackUnit { dchrt_CT_PictureStackUnit }?
724 dchrt_CT_DispUnitsLbl =
725   element layout { dchrt_CT_Layout }?,
726   element tx { dchrt_CT_Tx }?,
727   element spPr { a_CT_ShapeProperties }?,
728   element txPr { a_CT_TextBody }?
729 dchrt_CT_DispUnits =
730   (element custUnit { dchrt_CT_Double }
731    | element builtInUnit { dchrt_CT_BuiltInUnit }?),
732   element dispUnitsLbl { dchrt_CT_DispUnitsLbl }?,
733   element extLst { dchrt_CT_ExtensionList }?
734 dchrt_ST_Orientation = string "maxMin" | string "minMax"
735 dchrt_CT_Orientation =
736
737   ## default value: minMax
738   attribute val { dchrt_ST_Orientation }?
739 dchrt_ST_LogBase =
740   xsd:double { minInclusive = "2" maxInclusive = "1000" }
741 dchrt_CT_LogBase = attribute val { dchrt_ST_LogBase }
742 dchrt_CT_Scaling =
743   element logBase { dchrt_CT_LogBase }?,
744   element orientation { dchrt_CT_Orientation }?,
745   element max { dchrt_CT_Double }?,
746   element min { dchrt_CT_Double }?,
747   element extLst { dchrt_CT_ExtensionList }?
748 dchrt_ST_LblOffset =
749 dchrt_ST_LblOffsetPercent | dchrt_ST_LblOffsetUShort
750 dchrt_ST_LblOffsetPercent =
751   xsd:string {
752     pattern = "0*([0-9])|([1-9][0-9])|([1-9][0-9][0-9])|1000)%"
753   }
754 dchrt_ST_LblOffsetUShort =
755   xsd:unsignedShort { minInclusive = "0" maxInclusive = "1000" }
756 dchrt_CT_LblOffset =
757
758   ## default value: 100%

```

```

759   attribute val { dchrt_ST_LblOffset }?
760   dchrt_EG_AxShared =
761     element axId { dchrt_CT_UnsignedInt },
762     element scaling { dchrt_CT_Scaling },
763     element delete { dchrt_CT_Boolean }?,
764     element axPos { dchrt_CT_AxPos },
765     element majorGridlines { dchrt_CT_ChartLines }?,
766     element minorGridlines { dchrt_CT_ChartLines }?,
767     element title { dchrt_CT_Title }?,
768     element numFmt { dchrt_CT_NumFmt }?,
769     element majorTickMark { dchrt_CT_TickMark }?,
770     element minorTickMark { dchrt_CT_TickMark }?,
771     element tickLblPos { dchrt_CT_TickLblPos }?,
772     element spPr { a_CT_ShapeProperties }?,
773     element txPr { a_CT_TextBody }?,
774     element crossAx { dchrt_CT_UnsignedInt },
775     (element crosses { dchrt_CT_Crosses }
776      | element crossesAt { dchrt_CT_Double })?
777   dchrt_CT_CatAx =
778     dchrt_EG_AxShared,
779     element auto { dchrt_CT_Boolean }?,
780     element lblAlgn { dchrt_CT_LblAlgn }?,
781     element lblOffset { dchrt_CT_LblOffset }?,
782     element tickLblSkip { dchrt_CT_Skip }?,
783     element tickMarkSkip { dchrt_CT_Skip }?,
784     element noMultiLvlLbl { dchrt_CT_Boolean }?,
785     element extLst { dchrt_CT_ExtensionList }?
786   dchrt_CT_DateAx =
787     dchrt_EG_AxShared,
788     element auto { dchrt_CT_Boolean }?,
789     element lblOffset { dchrt_CT_LblOffset }?,
790     element baseTimeUnit { dchrt_CT_TimeUnit }?,
791     element majorUnit { dchrt_CT_AxisUnit }?,
792     element majorTimeUnit { dchrt_CT_TimeUnit }?,
793     element minorUnit { dchrt_CT_AxisUnit }?,
794     element minorTimeUnit { dchrt_CT_TimeUnit }?,
795     element extLst { dchrt_CT_ExtensionList }?
796   dchrt_CT_SerAx =
797     dchrt_EG_AxShared,
798     element tickLblSkip { dchrt_CT_Skip }?,
799     element tickMarkSkip { dchrt_CT_Skip }?,
800     element extLst { dchrt_CT_ExtensionList }?
801   dchrt_CT_ValAx =
802     dchrt_EG_AxShared,
803     element crossBetween { dchrt_CT_CrossBetween }?,
804     element majorUnit { dchrt_CT_AxisUnit }?,
805     element minorUnit { dchrt_CT_AxisUnit }?,
806     element dispUnits { dchrt_CT_DispatchUnits }?,
807     element extLst { dchrt_CT_ExtensionList }?
808   dchrt_CT_PlotArea =
809     element layout { dchrt_CT_Layout }?,
810     (element areaChart { dchrt_CT_AreaChart }
811      | element area3DChart { dchrt_CT_Area3DChart }

```

```

812 | element lineChart { dchrt_CT_LineChart }
813 | element line3DChart { dchrt_CT_Line3DChart }
814 | element stockChart { dchrt_CT_StockChart }
815 | element radarChart { dchrt_CT_RadarChart }
816 | element scatterChart { dchrt_CT_ScatterChart }
817 | element pieChart { dchrt_CT_PieChart }
818 | element pie3DChart { dchrt_CT_Pie3DChart }
819 | element doughnutChart { dchrt_CT_DoughnutChart }
820 | element barChart { dchrt_CT_BarChart }
821 | element bar3DChart { dchrt_CT_Bar3DChart }
822 | element ofPieChart { dchrt_CT_OfPieChart }
823 | element surfaceChart { dchrt_CT_SurfaceChart }
824 | element surface3DChart { dchrt_CT_Surface3DChart }
825 | element bubbleChart { dchrt_CT_BubbleChart }},
826 (element valAx { dchrt_CT_ValAx }
827 | element catAx { dchrt_CT_CatAx }
828 | element dateAx { dchrt_CT_DateAx }
829 | element serAx { dchrt_CT_SerAx })*,
830 element dTable { dchrt_CT_DTable }?,
831 element spPr { a_CT_ShapeProperties }?,
832 element extLst { dchrt_CT_ExtensionList }?
833 dchrt_CT_PivotFmt =
834 | element idx { dchrt_CT_UnsignedInt },
835 | element spPr { a_CT_ShapeProperties }?,
836 | element txPr { a_CT_TextBody }?,
837 | element marker { dchrt_CT_Marker }?,
838 | element dLbl { dchrt_CT_DLbl }?,
839 | element extLst { dchrt_CT_ExtensionList }?
840 dchrt_CT_PivotFmts = element pivotFmt { dchrt_CT_PivotFmt }*
841 dchrt_ST_LegendPos =
842 | string "b" | string "tr" | string "l" | string "r" | string "t"
843 dchrt_CT_LegendPos =
844
845 ## default value: r
846 attribute val { dchrt_ST_LegendPos }?
847 dchrt_EG_LegendEntryData = element txPr { a_CT_TextBody }?
848 dchrt_CT_LegendEntry =
849 | element idx { dchrt_CT_UnsignedInt },
850 | (element delete { dchrt_CT_Boolean }
851 | | dchrt_EG_LegendEntryData),
852 | element extLst { dchrt_CT_ExtensionList }?
853 dchrt_CT_Legend =
854 | element legendPos { dchrt_CT_LegendPos }?,
855 | element legendEntry { dchrt_CT_LegendEntry }*,
856 | element layout { dchrt_CT_Layout }?,
857 | element overlay { dchrt_CT_Boolean }?,
858 | element spPr { a_CT_ShapeProperties }?,
859 | element txPr { a_CT_TextBody }?,
860 | element extLst { dchrt_CT_ExtensionList }?
861 dchrt_ST_DisbBlanksAs = string "span" | string "gap" | string "zero"
862 dchrt_CT_DisbBlanksAs =
863
864 ## default value: zero

```

```

865     attribute val { dchrt_ST_DispBlanksAs }?
866 dchrt_CT_Chart =
867     element title { dchrt_CT_Title }?,
868     element autoTitleDeleted { dchrt_CT_Boolean }?,
869     element pivotFmts { dchrt_CT_PivotFmts }?,
870     element view3D { dchrt_CT_View3D }?,
871     element floor { dchrt_CT_Surface }?,
872     element sideWall { dchrt_CT_Surface }?,
873     element backWall { dchrt_CT_Surface }?,
874     element plotArea { dchrt_CT_PlotArea },
875     element legend { dchrt_CT_Legend }?,
876     element plotVisOnly { dchrt_CT_Boolean }?,
877     element dispBlanksAs { dchrt_CT_DispBlanksAs }?,
878     element showDLblsOverMax { dchrt_CT_Boolean }?,
879     element extLst { dchrt_CT_ExtensionList }?
880 dchrt_ST_Style =
881     xsd:unsignedByte { minInclusive = "1" maxInclusive = "48" }
882 dchrt_CT_Style = attribute val { dchrt_ST_Style }
883 dchrt_CT_PivotSource =
884     element name { s_ST_Xstring },
885     element fmtId { dchrt_CT_UnsignedInt },
886     element extLst { dchrt_CT_ExtensionList }*
887 dchrt_CT_Protection =
888     element chartObject { dchrt_CT_Boolean }?,
889     element data { dchrt_CT_Boolean }?,
890     element formatting { dchrt_CT_Boolean }?,
891     element selection { dchrt_CT_Boolean }?,
892     element userInterface { dchrt_CT_Boolean }?
893 dchrt_CT_HeaderFooter =
894
895     ## default value: true
896     attribute alignWithMargins { xsd:boolean }?,
897
898     ## default value: false
899     attribute differentOddEven { xsd:boolean }?,
900
901     ## default value: false
902     attribute differentFirst { xsd:boolean }?,
903     element oddHeader { s_ST_Xstring }?,
904     element oddFooter { s_ST_Xstring }?,
905     element evenHeader { s_ST_Xstring }?,
906     element evenFooter { s_ST_Xstring }?,
907     element firstHeader { s_ST_Xstring }?,
908     element firstFooter { s_ST_Xstring }?
909 dchrt_CT_PageMargins =
910     attribute l { xsd:double },
911     attribute r { xsd:double },
912     attribute t { xsd:double },
913     attribute b { xsd:double },
914     attribute header { xsd:double },
915     attribute footer { xsd:double }
916 dchrt_ST_PageSetupOrientation =
917     string "default" | string "portrait" | string "landscape"

```

```

918 dchrt_CT_ExternalData =
919     r_id,
920     element autoUpdate { dchrt_CT_Boolean }?
921 dchrt_CT_PageSetup =
922
923     ## default value: 1
924     attribute paperSize { xsd:unsignedInt }?,
925     attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
926     attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
927
928     ## default value: 1
929     attribute firstPageNumber { xsd:unsignedInt }?,
930
931     ## default value: default
932     attribute orientation { dchrt_ST_PageSetupOrientation }?,
933
934     ## default value: false
935     attribute blackAndWhite { xsd:boolean }?,
936
937     ## default value: false
938     attribute draft { xsd:boolean }?,
939
940     ## default value: false
941     attribute useFirstPageNumber { xsd:boolean }?,
942
943     ## default value: 600
944     attribute horizontalDpi { xsd:int }?,
945
946     ## default value: 600
947     attribute verticalDpi { xsd:int }?,
948
949     ## default value: 1
950     attribute copies { xsd:unsignedInt }?
951 dchrt_CT_PrintSettings =
952     element headerFooter { dchrt_CT_HeaderFooter }?,
953     element pageMargins { dchrt_CT_PageMargins }?,
954     element pageSetup { dchrt_CT_PageSetup }?,
955     element legacyDrawingHT { dchrt_CT_RelId }?
956 dchrt_CT_ChartSpace =
957     element date1904 { dchrt_CT_Boolean }?,
958     element lang { dchrt_CT_TextLanguageID }?,
959     element roundedCorners { dchrt_CT_Boolean }?,
960     element style { dchrt_CT_Style }?,
961     element clrMapOvr { a_CT_ColorMapping }?,
962     element pivotSource { dchrt_CT_PivotSource }?,
963     element protection { dchrt_CT_Protection }?,
964     element chart { dchrt_CT_Chart },
965     element spPr { a_CT_ShapeProperties }?,
966     element txPr { a_CT_TextBody }?,
967     element externalData { dchrt_CT_ExternalData }?,
968     element printSettings { dchrt_CT_PrintSettings }?,
969     element userShapes { dchrt_CT_RelId }?,
970     element extLst { dchrt_CT_ExtensionList }?

```

```

971 dchrt_chartSpace = element chartSpace { dchrt_CT_ChartSpace }
972 dchrt_userShapes = element userShapes { cdr_CT_Drawing }
973 dchrt_chart = element chart { dchrt_CT_RelId }

```

B.5.1.1 Part Schemas

B.5.1.1.1 Chart Part

This schema is available in the file DrawingML_Chart.rnc.

```

1 include "dml-chart.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-picture.rnc"
9 include "dml-chartDrawing.rnc"
10 start = dchrt_chartSpace

```

B.5.1.1.2 Chart Drawing Part

This schema is available in the file DrawingML_Chart_Drawing.rnc.

```

1 include "dml-chart.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-diagram.rnc"
5 include "shared-commonSimpleTypes.rnc"
6 include "dml-lockedCanvas.rnc"
7 include "any.rnc"
8 include "dml-picture.rnc"
9 include "dml-chartDrawing.rnc"
10 start = dchrt_userShapes

```

B.5.2 DrawingML - Chart Drawing

This schema is available in the file dml-chartDrawing.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace cdr =
5   "http://schemas.openxmlformats.org/drawingml/2006/chartDrawing"
6 namespace o = "urn:schemas-microsoft-com:office:office"
7 namespace v = "urn:schemas-microsoft-com:vml"
8 namespace w10 = "urn:schemas-microsoft-com:office:word"
9 namespace x = "urn:schemas-microsoft-com:office:excel"
10
11 cdr_CT_ShapeNonVisual =
12   element cNvPr { a_CT_NonVisualDrawingProps },
13   element cNvSpPr { a_CT_NonVisualDrawingShapeProps }
14 cdr_CT_Shape =

```

```

15 attribute macro { xsd:string }?,
16 attribute textlink { xsd:string }?,
17
18 ## default value: true
19 attribute fLocksText { xsd:boolean }?,
20
21 ## default value: false
22 attribute fPublished { xsd:boolean }?,
23 element nvSpPr { cdr_CT_ShapeNonVisual },
24 element spPr { a_CT_ShapeProperties },
25 element style { a_CT_ShapeStyle }?,
26 element txBody { a_CT_TextBody }?
27 cdr_CT_ConnectorNonVisual =
28   element cNvPr { a_CT_NonVisualDrawingProps },
29   element cNvCxnSpPr { a_CT_NonVisualConnectorProperties }
30 cdr_CT_Connector =
31   attribute macro { xsd:string }?,
32
33   ## default value: false
34   attribute fPublished { xsd:boolean }?,
35   element nvCxnSpPr { cdr_CT_ConnectorNonVisual },
36   element spPr { a_CT_ShapeProperties },
37   element style { a_CT_ShapeStyle }?
38 cdr_CT_PictureNonVisual =
39   element cNvPr { a_CT_NonVisualDrawingProps },
40   element cNvPicPr { a_CT_NonVisualPictureProperties }
41 cdr_CT_Picture =
42   attribute macro { xsd:string }?,
43
44   ## default value: false
45   attribute fPublished { xsd:boolean }?,
46   element nvPicPr { cdr_CT_PictureNonVisual },
47   element blipFill { a_CT_BlipFillProperties },
48   element spPr { a_CT_ShapeProperties },
49   element style { a_CT_ShapeStyle }?
50 cdr_CT_GraphicFrameNonVisual =
51   element cNvPr { a_CT_NonVisualDrawingProps },
52   element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }
53 cdr_CT_GraphicFrame =
54   attribute macro { xsd:string }?,
55
56   ## default value: false
57   attribute fPublished { xsd:boolean }?,
58   element nvGraphicFramePr { cdr_CT_GraphicFrameNonVisual },
59   element xfrm { a_CT_Transform2D },
60   a_graphic
61 cdr_CT_GroupShapeNonVisual =
62   element cNvPr { a_CT_NonVisualDrawingProps },
63   element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps }
64 cdr_CT_GroupShape =
65   element nvGrpSpPr { cdr_CT_GroupShapeNonVisual },
66   element grpSpPr { a_CT_GroupShapeProperties },
67   (element sp { cdr_CT_Shape }

```

```

68 | element grpSp { cdr_CT_GroupShape }
69 | element graphicFrame { cdr_CT_GraphicFrame }
70 | element cxnSp { cdr_CT_Connector }
71 | element pic { cdr_CT_Picture })*
72 cdr_EG_ObjectChoices =
73   element sp { cdr_CT_Shape }
74   | element grpSp { cdr_CT_GroupShape }
75   | element graphicFrame { cdr_CT_GraphicFrame }
76   | element cxnSp { cdr_CT_Connector }
77   | element pic { cdr_CT_Picture }
78 cdr_ST_MarkerCoordinate =
79   xsd:double { minInclusive = "0.0" maxInclusive = "1.0" }
80 cdr_CT_Marker =
81   element x { cdr_ST_MarkerCoordinate },
82   element y { cdr_ST_MarkerCoordinate }
83 cdr_CT_RelSizeAnchor =
84   element from { cdr_CT_Marker },
85   element to { cdr_CT_Marker },
86   cdr_EG_ObjectChoices
87 cdr_CT_AbsSizeAnchor =
88   element from { cdr_CT_Marker },
89   element ext { a_CT_PositiveSize2D },
90   cdr_EG_ObjectChoices
91 cdr_EG_Anchor =
92   element relSizeAnchor { cdr_CT_RelSizeAnchor }
93   | element absSizeAnchor { cdr_CT_AbsSizeAnchor }
94 cdr_CT_Drawing = cdr_EG_Anchor*

```

B.5.3 DrawingML - Diagrams

This schema is available in the file dml-diagram.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/diagram"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace ddgrm =
5   "http://schemas.openxmlformats.org/drawingml/2006/diagram"
6 namespace o = "urn:schemas-microsoft-com:office:office"
7 namespace r =
8   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
9 namespace s =
10  "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11 namespace v = "urn:schemas-microsoft-com:vml"
12 namespace w10 = "urn:schemas-microsoft-com:office:word"
13 namespace x = "urn:schemas-microsoft-com:office:excel"
14
15 ddgrm_CT_CTName =
16   attribute lang { xsd:string }?,
17   attribute val { xsd:string }
18 ddgrm_CT_CTDescription =
19   attribute lang { xsd:string }?,
20   attribute val { xsd:string }
21 ddgrm_CT_CTCategory =

```



```

22   attribute type { xsd:anyURI },
23   attribute pri { xsd:unsignedInt }
24 ddgrm_CT_CTCategories = element cat { ddgrm_CT_CTCategory }*
25 ddgrm_ST_ClrAppMethod = "span" | "cycle" | "repeat"
26 ddgrm_ST_HueDir = "cw" | "ccw"
27 ddgrm_CT_Colors =
28
29   ## default value: span
30   attribute meth { ddgrm_ST_ClrAppMethod }?,
31
32   ## default value: cw
33   attribute hueDir { ddgrm_ST_HueDir }?,
34   a_EG_ColorChoice*
35 ddgrm_CT_CTStyleLabel =
36   attribute name { xsd:string },
37   element fillClrLst { ddgrm_CT_Colors }?,
38   element linClrLst { ddgrm_CT_Colors }?,
39   element effectClrLst { ddgrm_CT_Colors }?,
40   element txLinClrLst { ddgrm_CT_Colors }?,
41   element txFillClrLst { ddgrm_CT_Colors }?,
42   element txEffectClrLst { ddgrm_CT_Colors }?,
43   element extLst { a_CT_OfficeArtExtensionList }?
44 ddgrm_CT_ColorTransform =
45   attribute uniqueId { xsd:string }?,
46
47   attribute minVer { xsd:string }?,
48   element title { ddgrm_CT_CTName }*,
49   element desc { ddgrm_CT_CTDescription }*,
50   element catLst { ddgrm_CT_CTCategories }?,
51   element styleLbl { ddgrm_CT_CTStyleLabel }*,
52   element extLst { a_CT_OfficeArtExtensionList }?
53 ddgrm_colorsDef = element colorsDef { ddgrm_CT_ColorTransform }
54 ddgrm_CT_ColorTransformHeader =
55   attribute uniqueId { xsd:string },
56
57   attribute minVer { xsd:string }?,
58
59   ## default value: 0
60   attribute resId { xsd:int }?,
61   element title { ddgrm_CT_CTName }+,
62   element desc { ddgrm_CT_CTDescription }+,
63   element catLst { ddgrm_CT_CTCategories }?,
64   element extLst { a_CT_OfficeArtExtensionList }?
65 ddgrm_colorsDefHdr =
66   element colorsDefHdr { ddgrm_CT_ColorTransformHeader }
67 ddgrm_CT_ColorTransformHeaderLst =
68   element colorsDefHdr { ddgrm_CT_ColorTransformHeader }*
69 ddgrm_colorsDefHdrLst =
70   element colorsDefHdrLst { ddgrm_CT_ColorTransformHeaderLst }
71 ddgrm_ST_PtType =
72   "node" | "asst" | "doc" | "pres" | "parTrans" | "sibTrans"
73 ddgrm_CT_Pt =
74   attribute modelId { ddgrm_ST_ModelId },

```

```

75
76   ## default value: node
77   attribute type { ddgrm_ST_PtType }?,
78
79   ## default value: 0
80   attribute cxnId { ddgrm_ST_ModelId }?,
81   element prSet { ddgrm_CT_ElemPropSet }?,
82   element spPr { a_CT_ShapeProperties }?,
83   element t { a_CT_TextBody }?,
84   element extLst { a_CT_OfficeArtExtensionList }?
85 ddgrm_CT_PtList = element pt { ddgrm_CT_Pt }*
86 ddgrm_ST_CxnType =
87   "parOf" | "presOf" | "presParOf" | "unknownRelationship"
88 ddgrm_CT_Cxn =
89   attribute modelId { ddgrm_ST_ModelId },
90
91   ## default value: parOf
92   attribute type { ddgrm_ST_CxnType }?,
93   attribute srcId { ddgrm_ST_ModelId },
94   attribute destId { ddgrm_ST_ModelId },
95   attribute srcOrd { xsd:unsignedInt },
96   attribute destOrd { xsd:unsignedInt },
97
98   ## default value: 0
99   attribute parTransId { ddgrm_ST_ModelId }?,
100
101   ## default value: 0
102   attribute sibTransId { ddgrm_ST_ModelId }?,
103   attribute presId { xsd:string }?,
104   element extLst { a_CT_OfficeArtExtensionList }?
105 ddgrm_CT_CxnList = element cxn { ddgrm_CT_Cxn }*
106 ddgrm_CT_DataModel =
107   element ptLst { ddgrm_CT_PtList },
108   element cxnLst { ddgrm_CT_CxnList }?,
109   element bg { a_CT_BackgroundFormatting }?,
110   element whole { a_CT_WholeE2oFormatting }?,
111   element extLst { a_CT_OfficeArtExtensionList }?
112 ddgrm_dataModel = element dataModel { ddgrm_CT_DataModel }
113 ddgrm_AG_IteratorAttributes =
114
115   ## default value: none
116   attribute axis { ddgrm_ST_AxisTypes }?,
117
118   ## default value: all
119   attribute ptType { ddgrm_ST_ElementTypes }?,
120
121   ## default value: true
122   attribute hideLastTrans { ddgrm_ST_Booleans }?,
123
124   ## default value: 1
125   attribute st { ddgrm_ST_Ints }?,
126
127   ## default value: 0

```

```

128     attribute cnt { ddgrm_ST_UnsignedInts }?,
129
130     ## default value: 1
131     attribute step { ddgrm_ST_Ints }?
132 ddgrm_AG_ConstraintAttributes =
133     attribute type { ddgrm_ST_ConstraintType },
134
135     ## default value: self
136     attribute for { ddgrm_ST_ConstraintRelationship }?,
137     attribute forName { xsd:string }?,
138
139     ## default value: all
140     attribute ptType { ddgrm_ST_ElementType }?
141 ddgrm_AG_ConstraintRefAttributes =
142
143     ## default value: none
144     attribute refType { ddgrm_ST_ConstraintType }?,
145
146     ## default value: self
147     attribute refFor { ddgrm_ST_ConstraintRelationship }?,
148     attribute refForName { xsd:string }?,
149
150     ## default value: all
151     attribute refPtType { ddgrm_ST_ElementType }?
152 ddgrm_CT_Constraint =
153     ddgrm_AG_ConstraintAttributes,
154     ddgrm_AG_ConstraintRefAttributes,
155
156     ## default value: none
157     attribute op { ddgrm_ST_BoolOperator }?,
158
159     ## default value: 0
160     attribute val { xsd:double }?,
161
162     ## default value: 1
163     attribute fact { xsd:double }?,
164     element extLst { a_CT_OfficeArtExtensionList }?
165 ddgrm_CT_Constraints = element constr { ddgrm_CT_Constraint }*
166 ddgrm_CT_NumericRule =
167     ddgrm_AG_ConstraintAttributes,
168
169     ## default value: NaN
170     attribute val { xsd:double }?,
171
172     ## default value: NaN
173     attribute fact { xsd:double }?,
174
175     ## default value: NaN
176     attribute max { xsd:double }?,
177     element extLst { a_CT_OfficeArtExtensionList }?
178 ddgrm_CT_Rules = element rule { ddgrm_CT_NumericRule }*
179 ddgrm_CT_PresentationOf =
180     ddgrm_AG_IteratorAttributes,

```

```

181     element extLst { a_CT_OfficeArtExtensionList }?
182 ddgrm_ST_LayoutShapeType = a_ST_ShapeType | ddgrm_ST_OutputShapeType
183 ddgrm_ST_Index1 = xsd:unsignedInt { minInclusive = "1" }
184 ddgrm_CT_Adj =
185     attribute idx { ddgrm_ST_Index1 },
186     attribute val { xsd:double }
187 ddgrm_CT_AdjLst = element adj { ddgrm_CT_Adj }*
188 ddgrm_CT_Shape =
189
190     ## default value: 0
191     attribute rot { xsd:double }?,
192
193     ## default value: none
194     attribute type { ddgrm_ST_LayoutShapeType }?,
195     r_blip?,
196
197     ## default value: 0
198     attribute zOrderOff { xsd:int }?,
199
200     ## default value: false
201     attribute hideGeom { xsd:boolean }?,
202
203     ## default value: false
204     attribute lkTxEntry { xsd:boolean }?,
205
206     ## default value: false
207     attribute blipPhldr { xsd:boolean }?,
208     element adjLst { ddgrm_CT_AdjLst }?,
209     element extLst { a_CT_OfficeArtExtensionList }?
210 ddgrm_CT_Parameter =
211     attribute type { ddgrm_ST_ParameterId },
212     attribute val { ddgrm_ST_ParameterVal }
213 ddgrm_CT_Algorithm =
214     attribute type { ddgrm_ST_AlgorithmType },
215
216     ## default value: 0
217     attribute rev { xsd:unsignedInt }?,
218     element param { ddgrm_CT_Parameter }*,
219     element extLst { a_CT_OfficeArtExtensionList }?
220 ddgrm_CT_LayoutNode =
221     attribute name { xsd:string }?,
222     attribute styleLbl { xsd:string }?,
223
224     ## default value: b
225     attribute chOrder { ddgrm_ST_ChildOrderType }?,
226     attribute moveWith { xsd:string }?,
227     (element alg { ddgrm_CT_Algorithm }?
228     | element shape { ddgrm_CT_Shape }?
229     | element presOf { ddgrm_CT_PresentationOf }?
230     | element constrLst { ddgrm_CT_Constraints }?
231     | element ruleLst { ddgrm_CT_Rules }?
232     | element varLst { ddgrm_CT_LayoutVariablePropertySet }?
233     | element forEach { ddgrm_CT_ForEach }

```

```

234 | element layoutNode { ddgrm_CT_LayoutNode }
235 | element choose { ddgrm_CT_Choose }
236 | element extLst { a_CT_OfficeArtExtensionList }?)*
237 ddgrm_CT_ForEach =
238   attribute name { xsd:string }?,
239   attribute ref { xsd:string }?,
240   ddgrm_AG_IteratorAttributes,
241   (element alg { ddgrm_CT_Algorithm }?
242   | element shape { ddgrm_CT_Shape }?
243   | element presOf { ddgrm_CT_PresentationOf }?
244   | element constrLst { ddgrm_CT_Constraints }?
245   | element ruleLst { ddgrm_CT_Rules }?
246   | element forEach { ddgrm_CT_ForEach }
247   | element layoutNode { ddgrm_CT_LayoutNode }
248   | element choose { ddgrm_CT_Choose }
249   | element extLst { a_CT_OfficeArtExtensionList }?)*
250 ddgrm_CT_When =
251   attribute name { xsd:string }?,
252   ddgrm_AG_IteratorAttributes,
253   attribute func { ddgrm_ST_FunctionType },
254
255   ## default value: none
256   attribute arg { ddgrm_ST_FunctionArgument }?,
257   attribute op { ddgrm_ST_FunctionOperator },
258   attribute val { ddgrm_ST_FunctionValue },
259   (element alg { ddgrm_CT_Algorithm }?
260   | element shape { ddgrm_CT_Shape }?
261   | element presOf { ddgrm_CT_PresentationOf }?
262   | element constrLst { ddgrm_CT_Constraints }?
263   | element ruleLst { ddgrm_CT_Rules }?
264   | element forEach { ddgrm_CT_ForEach }
265   | element layoutNode { ddgrm_CT_LayoutNode }
266   | element choose { ddgrm_CT_Choose }
267   | element extLst { a_CT_OfficeArtExtensionList }?)*
268 ddgrm_CT_Otherwise =
269   attribute name { xsd:string }?,
270   (element alg { ddgrm_CT_Algorithm }?
271   | element shape { ddgrm_CT_Shape }?
272   | element presOf { ddgrm_CT_PresentationOf }?
273   | element constrLst { ddgrm_CT_Constraints }?
274   | element ruleLst { ddgrm_CT_Rules }?
275   | element forEach { ddgrm_CT_ForEach }
276   | element layoutNode { ddgrm_CT_LayoutNode }
277   | element choose { ddgrm_CT_Choose }
278   | element extLst { a_CT_OfficeArtExtensionList }?)*
279 ddgrm_CT_Choose =
280   attribute name { xsd:string }?,
281   element if { ddgrm_CT_When }+,
282   element else { ddgrm_CT_Otherwise }?
283 ddgrm_CT_SampleData =
284
285   ## default value: false
286   attribute useDef { xsd:boolean }?,

```

```

287     element dataModel { ddgrm_CT_DataModel }?
288 ddgrm_CT_Category =
289     attribute type { xsd:anyURI },
290     attribute pri { xsd:unsignedInt }
291 ddgrm_CT_Categories = element cat { ddgrm_CT_Category }*
292 ddgrm_CT_Name =
293     attribute lang { xsd:string }?,
294     attribute val { xsd:string }
295 ddgrm_CT_Description =
296     attribute lang { xsd:string }?,
297     attribute val { xsd:string }
298 ddgrm_CT_DiagramDefinition =
299     attribute uniqueId { xsd:string }?,
300
301     attribute minVer { xsd:string }?,
302     attribute defStyle { xsd:string }?,
303     element title { ddgrm_CT_Name }*,
304     element desc { ddgrm_CT_Description }*,
305     element catLst { ddgrm_CT_Categories }?,
306     element sampData { ddgrm_CT_SampleData }?,
307     element styleData { ddgrm_CT_SampleData }?,
308     element clrData { ddgrm_CT_SampleData }?,
309     element layoutNode { ddgrm_CT_LayoutNode },
310     element extLst { a_CT_OfficeArtExtensionList }?
311 ddgrm_layoutDef = element layoutDef { ddgrm_CT_DiagramDefinition }
312 ddgrm_CT_DiagramDefinitionHeader =
313     attribute uniqueId { xsd:string },
314
315     attribute minVer { xsd:string }?,
316     attribute defStyle { xsd:string }?,
317
318     ## default value: 0
319     attribute resId { xsd:int }?,
320     element title { ddgrm_CT_Name }+,
321     element desc { ddgrm_CT_Description }+,
322     element catLst { ddgrm_CT_Categories }?,
323     element extLst { a_CT_OfficeArtExtensionList }?
324 ddgrm_layoutDefHdr =
325     element layoutDefHdr { ddgrm_CT_DiagramDefinitionHeader }
326 ddgrm_CT_DiagramDefinitionHeaderLst =
327     element layoutDefHdr { ddgrm_CT_DiagramDefinitionHeader }*
328 ddgrm_layoutDefHdrLst =
329     element layoutDefHdrLst { ddgrm_CT_DiagramDefinitionHeaderLst }
330 ddgrm_CT_RelIds = r_dm, r_lo, r_qs, r_cs
331 ddgrm_relIds = element relIds { ddgrm_CT_RelIds }
332 ddgrm_ST_ParameterVal =
333     ddgrm_ST_DiagramHorizontalAlignment
334     | ddgrm_ST_VerticalAlignment
335     | ddgrm_ST_ChildDirection
336     | ddgrm_ST_ChildAlignment
337     | ddgrm_ST_SecondaryChildAlignment
338     | ddgrm_ST_LinearDirection
339     | ddgrm_ST_SecondaryLinearDirection

```

```

340 | ddgrm_ST_StartingElement
341 | ddgrm_ST_BendPoint
342 | ddgrm_ST_ConnectorRouting
343 | ddgrm_ST_ArrowheadStyle
344 | ddgrm_ST_ConnectorDimension
345 | ddgrm_ST_RotationPath
346 | ddgrm_ST_CenterShapeMapping
347 | ddgrm_ST_NodeHorizontalAlignment
348 | ddgrm_ST_NodeVerticalAlignment
349 | ddgrm_ST_FallbackDimension
350 | ddgrm_ST_TextDirection
351 | ddgrm_ST_PyramidAccentPosition
352 | ddgrm_ST_PyramidAccentTextMargin
353 | ddgrm_ST_TextBlockDirection
354 | ddgrm_ST_TextAnchorHorizontal
355 | ddgrm_ST_TextAnchorVertical
356 | ddgrm_ST_DiagramTextAlignment
357 | ddgrm_ST_AutoTextRotation
358 | ddgrm_ST_GrowDirection
359 | ddgrm_ST_FlowDirection
360 | ddgrm_ST_ContinueDirection
361 | ddgrm_ST_Breakpoint
362 | ddgrm_ST_Offset
363 | ddgrm_ST_HierarchyAlignment
364 | xsd:int
365 | xsd:double
366 | xsd:boolean
367 | xsd:string
368 | ddgrm_ST_ConnectorPoint
369 ddgrm_ST_ModelId = xsd:int | s_ST_Guid
370 ddgrm_ST_PrSetCustVal = s_ST_Percentage | xsd:int
371 ddgrm_CT_ElemPropSet =
372     attribute presAssocID { ddgrm_ST_ModelId }?,
373     attribute presName { xsd:string }?,
374     attribute presStyleLbl { xsd:string }?,
375     attribute presStyleIdx { xsd:int }?,
376     attribute presStyleCnt { xsd:int }?,
377     attribute loTypeId { xsd:string }?,
378     attribute loCatId { xsd:string }?,
379     attribute qsTypeId { xsd:string }?,
380     attribute qsCatId { xsd:string }?,
381     attribute csTypeId { xsd:string }?,
382     attribute csCatId { xsd:string }?,
383     attribute coherent3DOff { xsd:boolean }?,
384     attribute phldrT { xsd:string }?,
385     attribute phldr { xsd:boolean }?,
386     attribute custAng { xsd:int }?,
387     attribute custFlipVert { xsd:boolean }?,
388     attribute custFlipHor { xsd:boolean }?,
389     attribute custSzX { xsd:int }?,
390     attribute custSzY { xsd:int }?,
391     attribute custScaleX { ddgrm_ST_PrSetCustVal }?,
392     attribute custScaleY { ddgrm_ST_PrSetCustVal }?,

```

```

393     attribute custT { xsd:boolean }?,
394     attribute custLinFactX { ddgrm_ST_PrSetCustVal }?,
395     attribute custLinFactY { ddgrm_ST_PrSetCustVal }?,
396     attribute custLinFactNeighborX { ddgrm_ST_PrSetCustVal }?,
397     attribute custLinFactNeighborY { ddgrm_ST_PrSetCustVal }?,
398     attribute custRadScaleRad { ddgrm_ST_PrSetCustVal }?,
399     attribute custRadScaleInc { ddgrm_ST_PrSetCustVal }?,
400     element presLayoutVars { ddgrm_CT_LayoutVariablePropertySet }?,
401     element style { a_CT_ShapeStyle }?
402 ddgrm_ST_Direction = "norm" | "rev"
403 ddgrm_ST_HierBranchStyle = "l" | "r" | "hang" | "std" | "init"
404 ddgrm_ST_AnimOneStr = "none" | "one" | "branch"
405 ddgrm_ST_AnimLvlStr = "none" | "lvl" | "ctr"
406 ddgrm_CT_OrgChart =
407
408     ## default value: false
409     attribute val { xsd:boolean }?
410 ddgrm_ST_NodeCount = xsd:int { minInclusive = "-1" }
411 ddgrm_CT_ChildMax =
412
413     ## default value: -1
414     attribute val { ddgrm_ST_NodeCount }?
415 ddgrm_CT_ChildPref =
416
417     ## default value: -1
418     attribute val { ddgrm_ST_NodeCount }?
419 ddgrm_CT_BulletEnabled =
420
421     ## default value: false
422     attribute val { xsd:boolean }?
423 ddgrm_CT_Direction =
424
425     ## default value: norm
426     attribute val { ddgrm_ST_Direction }?
427 ddgrm_CT_HierBranchStyle =
428
429     ## default value: std
430     attribute val { ddgrm_ST_HierBranchStyle }?
431 ddgrm_CT_AnimOne =
432
433     ## default value: one
434     attribute val { ddgrm_ST_AnimOneStr }?
435 ddgrm_CT_AnimLvl =
436
437     ## default value: none
438     attribute val { ddgrm_ST_AnimLvlStr }?
439 ddgrm_ST_ResizeHandlesStr = "exact" | "rel"
440 ddgrm_CT_ResizeHandles =
441
442     ## default value: rel
443     attribute val { ddgrm_ST_ResizeHandlesStr }?
444 ddgrm_CT_LayoutVariablePropertySet =
445     element orgChart { ddgrm_CT_OrgChart }?,

```



```

446     element chMax { ddgrm_CT_ChildMax }?,
447     element chPref { ddgrm_CT_ChildPref }?,
448     element bulletEnabled { ddgrm_CT_BulletEnabled }?,
449     element dir { ddgrm_CT_Direction }?,
450     element hierBranch { ddgrm_CT_HierBranchStyle }?,
451     element animOne { ddgrm_CT_AnimOne }?,
452     element animLvl { ddgrm_CT_AnimLvl }?,
453     element resizeHandles { ddgrm_CT_ResizeHandles }?
454 ddgrm_CT_SDName =
455     attribute lang { xsd:string }?,
456     attribute val { xsd:string }
457 ddgrm_CT_SDDescription =
458     attribute lang { xsd:string }?,
459     attribute val { xsd:string }
460 ddgrm_CT_SDCategory =
461     attribute type { xsd:anyURI },
462     attribute pri { xsd:unsignedInt }
463 ddgrm_CT_SDCategories = element cat { ddgrm_CT_SDCategory }*
464 ddgrm_CT_TextProps = a_EG_Text3D?
465 ddgrm_CT_StyleLabel =
466     attribute name { xsd:string },
467     element scene3d { a_CT_Scene3D }?,
468     element sp3d { a_CT_Shape3D }?,
469     element txPr { ddgrm_CT_TextProps }?,
470     element style { a_CT_ShapeStyle }?,
471     element extLst { a_CT_OfficeArtExtensionList }?
472 ddgrm_CT_StyleDefinition =
473     attribute uniqueId { xsd:string }?,
474
475     attribute minVer { xsd:string }?,
476     element title { ddgrm_CT_SDName }*,
477     element desc { ddgrm_CT_SDDescription }*,
478     element catLst { ddgrm_CT_SDCategories }?,
479     element scene3d { a_CT_Scene3D }?,
480     element styleLbl { ddgrm_CT_StyleLabel }+,
481     element extLst { a_CT_OfficeArtExtensionList }?
482 ddgrm_styleDef = element styleDef { ddgrm_CT_StyleDefinition }
483 ddgrm_CT_StyleDefinitionHeader =
484     attribute uniqueId { xsd:string },
485
486     attribute minVer { xsd:string }?,
487
488     ## default value: 0
489     attribute resId { xsd:int }?,
490     element title { ddgrm_CT_SDName }+,
491     element desc { ddgrm_CT_SDDescription }+,
492     element catLst { ddgrm_CT_SDCategories }?,
493     element extLst { a_CT_OfficeArtExtensionList }?
494 ddgrm_styleDefHdr =
495     element styleDefHdr { ddgrm_CT_StyleDefinitionHeader }
496 ddgrm_CT_StyleDefinitionHeaderLst =
497     element styleDefHdr { ddgrm_CT_StyleDefinitionHeader }*
498 ddgrm_styleDefHdrLst =

```

```

499     element styleDefHdrLst { ddgrm_CT_StyleDefinitionHeaderLst }
500 ddgrm_ST_AlgorithmType =
501     "composite"
502     | "conn"
503     | "cycle"
504     | "hierChild"
505     | "hierRoot"
506     | "pyra"
507     | "lin"
508     | "sp"
509     | "tx"
510     | "snake"
511 ddgrm_ST_AxisType =
512     "self"
513     | "ch"
514     | "des"
515     | "desOrSelf"
516     | "par"
517     | "ancst"
518     | "ancstOrSelf"
519     | "followSib"
520     | "precedSib"
521     | "follow"
522     | "preced"
523     | "root"
524     | "none"
525 ddgrm_ST_AxisTypes = list { ddgrm_ST_AxisType* }
526 ddgrm_ST_BoolOperator = "none" | "equ" | "gte" | "lte"
527 ddgrm_ST_ChildOrderType = "b" | "t"
528 ddgrm_ST_ConstraintType =
529     "none"
530     | "alignOff"
531     | "begMarg"
532     | "bendDist"
533     | "begPad"
534     | "b"
535     | "bMarg"
536     | "bOff"
537     | "ctrX"
538     | "ctrXOff"
539     | "ctrY"
540     | "ctrYOff"
541     | "connDist"
542     | "diam"
543     | "endMarg"
544     | "endPad"
545     | "h"
546     | "hArH"
547     | "hOff"
548     | "l"
549     | "lMarg"
550     | "lOff"
551     | "r"

```

```

552 | "rMarg"
553 | "rOff"
554 | "primFontSz"
555 | "pyraAcctRatio"
556 | "secFontSz"
557 | "sibSp"
558 | "secSibSp"
559 | "sp"
560 | "stemThick"
561 | "t"
562 | "tMarg"
563 | "tOff"
564 | "userA"
565 | "userB"
566 | "userC"
567 | "userD"
568 | "userE"
569 | "userF"
570 | "userG"
571 | "userH"
572 | "userI"
573 | "userJ"
574 | "userK"
575 | "userL"
576 | "userM"
577 | "userN"
578 | "userO"
579 | "userP"
580 | "userQ"
581 | "userR"
582 | "userS"
583 | "userT"
584 | "userU"
585 | "userV"
586 | "userW"
587 | "userX"
588 | "userY"
589 | "userZ"
590 | "w"
591 | "wArH"
592 | "wOff"
593 ddgrm_ST_ConstraintRelationship = "self" | "ch" | "des"
594 ddgrm_ST_ElementType =
595     "all"
596     | "doc"
597     | "node"
598     | "norm"
599     | "nonNorm"
600     | "asst"
601     | "nonAsst"
602     | "parTrans"
603     | "pres"
604     | "sibTrans"

```

```

605 ddgrm_ST_ElementTypes = list { ddgrm_ST_ElementType* }
606 ddgrm_ST_ParameterId =
607     "horzAlign"
608     | "vertAlign"
609     | "chDir"
610     | "chAlign"
611     | "secChAlign"
612     | "linDir"
613     | "secLinDir"
614     | "stElem"
615     | "bendPt"
616     | "connRout"
617     | "begSty"
618     | "endSty"
619     | "dim"
620     | "rotPath"
621     | "ctrShpMap"
622     | "nodeHorzAlign"
623     | "nodeVertAlign"
624     | "fallback"
625     | "txDir"
626     | "pyraAcctPos"
627     | "pyraAcctTxMar"
628     | "txBldir"
629     | "txAnchorHorz"
630     | "txAnchorVert"
631     | "txAnchorHorzCh"
632     | "txAnchorVertCh"
633     | "parTxLTRAlign"
634     | "parTxRTLAlign"
635     | "shpTxLTRAlignCh"
636     | "shpTxRTLAlignCh"
637     | "autoTxRot"
638     | "grDir"
639     | "flowDir"
640     | "contDir"
641     | "bkpt"
642     | "off"
643     | "hierAlign"
644     | "bkPtFixedVal"
645     | "stBulletLvl"
646     | "stAng"
647     | "spanAng"
648     | "ar"
649     | "lnSpPar"
650     | "lnSpAfParP"
651     | "lnSpCh"
652     | "lnSpAfChP"
653     | "rtShortDist"
654     | "alignTx"
655     | "pyraLvlNode"
656     | "pyraAcctBkgdNode"
657     | "pyraAcctTxNode"

```

```

658 | "srcNode"
659 | "dstNode"
660 | "begPts"
661 | "endPts"
662 ddgrm_ST_Ints = list { xsd:int* }
663 ddgrm_ST_UnsignedInts = list { xsd:unsignedInt* }
664 ddgrm_ST_Booleans = list { xsd:boolean* }
665 ddgrm_ST_FunctionType =
666   "cnt"
667   | "pos"
668   | "revPos"
669   | "posEven"
670   | "posOdd"
671   | "var"
672   | "depth"
673   | "maxDepth"
674 ddgrm_ST_FunctionOperator = "equ" | "neq" | "gt" | "lt" | "gte" | "lte"
675 ddgrm_ST_DiagramHorizontalAlignment = "l" | "ctr" | "r" | "none"
676 ddgrm_ST_VerticalAlignment = "t" | "mid" | "b" | "none"
677 ddgrm_ST_ChildDirection = "horz" | "vert"
678 ddgrm_ST_ChildAlignment = "t" | "b" | "l" | "r"
679 ddgrm_ST_SecondaryChildAlignment = "none" | "t" | "b" | "l" | "r"
680 ddgrm_ST_LinearDirection = "fromL" | "fromR" | "fromT" | "fromB"
681 ddgrm_ST_SecondaryLinearDirection =
682   "none" | "fromL" | "fromR" | "fromT" | "fromB"
683 ddgrm_ST_StartingElement = "node" | "trans"
684 ddgrm_ST_RotationPath = "none" | "alongPath"
685 ddgrm_ST_CenterShapeMapping = "none" | "fNode"
686 ddgrm_ST_BendPoint = "beg" | "def" | "end"
687 ddgrm_ST_ConnectorRouting = "stra" | "bend" | "curve" | "longCurve"
688 ddgrm_ST_ArrowheadStyle = "auto" | "arr" | "noArr"
689 ddgrm_ST_ConnectorDimension = "1D" | "2D" | "cust"
690 ddgrm_ST_ConnectorPoint =
691   "auto"
692   | "bCtr"
693   | "ctr"
694   | "midL"
695   | "midR"
696   | "tCtr"
697   | "bL"
698   | "bR"
699   | "tL"
700   | "tR"
701   | "radial"
702 ddgrm_ST_NodeHorizontalAlignment = "l" | "ctr" | "r"
703 ddgrm_ST_NodeVerticalAlignment = "t" | "mid" | "b"
704 ddgrm_ST_FallbackDimension = "1D" | "2D"
705 ddgrm_ST_TextDirection = "fromT" | "fromB"
706 ddgrm_ST_PyramidAccentPosition = "bef" | "aft"
707 ddgrm_ST_PyramidAccentTextMargin = "step" | "stack"
708 ddgrm_ST_TextBlockDirection = "horz" | "vert"
709 ddgrm_ST_TextAnchorHorizontal = "none" | "ctr"
710 ddgrm_ST_TextAnchorVertical = "t" | "mid" | "b"

```

```

711 ddgrm_ST_DiagramTextAlignment = "l" | "ctr" | "r"
712 ddgrm_ST_AutoTextRotation = "none" | "upr" | "grav"
713 ddgrm_ST_GrowDirection = "tL" | "tR" | "bL" | "bR"
714 ddgrm_ST_FlowDirection = "row" | "col"
715 ddgrm_ST_ContinueDirection = "revDir" | "sameDir"
716 ddgrm_ST_Breakpoint = "endCnv" | "bal" | "fixed"
717 ddgrm_ST_Offset = "ctr" | "off"
718 ddgrm_ST_HierarchyAlignment =
719     "tL"
720     | "tR"
721     | "tCtrCh"
722     | "tCtrDes"
723     | "bL"
724     | "bR"
725     | "bCtrCh"
726     | "bCtrDes"
727     | "lT"
728     | "lB"
729     | "lCtrCh"
730     | "lCtrDes"
731     | "rT"
732     | "rB"
733     | "rCtrCh"
734     | "rCtrDes"
735 ddgrm_ST_FunctionValue =
736     xsd:int
737     | xsd:boolean
738     | ddgrm_ST_Direction
739     | ddgrm_ST_HierBranchStyle
740     | ddgrm_ST_AnimOneStr
741     | ddgrm_ST_AnimLvlStr
742     | ddgrm_ST_ResizeHandlesStr
743 ddgrm_ST_VariableType =
744     "none"
745     | "orgChart"
746     | "chMax"
747     | "chPref"
748     | "bulEnabled"
749     | "dir"
750     | "hierBranch"
751     | "animOne"
752     | "animLvl"
753     | "resizeHandles"
754 ddgrm_ST_FunctionArgument = ddgrm_ST_VariableType
755 ddgrm_ST_OutputShapeType = "none" | "conn"

```

B.5.3.1 Part Schemas

B.5.3.1.1 Diagram Colors Part

This schema is available in the file DrawingML_Diagram_Colors.rnc.

```
1 include "dml-diagram.rnc"
```

```

2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-lockedCanvas.rnc"
5 include "any.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-chart.rnc"
8 include "dml-chartDrawing.rnc"
9 include "dml-picture.rnc"
10 start = ddgrm_colorsDef

```

B.5.3.1.2 Diagram Data Part

This schema is available in the file DrawingML_Diagram_Data.rnc.

```

1 include "dml-diagram.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-lockedCanvas.rnc"
5 include "any.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-chart.rnc"
8 include "dml-chartDrawing.rnc"
9 include "dml-picture.rnc"
10 start = ddgrm_dataModel

```

B.5.3.1.3 Diagram Layout Definitions Part

This schema is available in the file DrawingML_Diagram_Layout_Definition.rnc.

```

1 include "dml-diagram.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-main.rnc"
4 include "dml-lockedCanvas.rnc"
5 include "any.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-chart.rnc"
8 include "dml-chartDrawing.rnc"
9 include "dml-picture.rnc"
10 start = ddgrm_layoutDef

```

B.5.3.1.4 Diagram Style Part

This schema is available in the file DrawingML_Diagram_Style.rnc.

```

include "dml-diagram.rnc"
include "shared-relationshipReference.rnc"
include "dml-main.rnc"
include "dml-lockedCanvas.rnc"
include "any.rnc"
include "shared-commonSimpleTypes.rnc"
include "dml-chart.rnc"
include "dml-chartDrawing.rnc"
include "dml-picture.rnc"
start = ddgrm_styleDef

```

B.6 VML

B.6.1 VML - Main

This schema is available in the file vml-main.rnc.

```

1 namespace o = "urn:schemas-microsoft-com:office:office"
2 namespace pvml = "urn:schemas-microsoft-com:office:powerpoint"
3 namespace r =
4   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
5 namespace s =
6   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
7 default namespace v = "urn:schemas-microsoft-com:vml"
8 namespace w =
9   "http://schemas.openxmlformats.org/wordprocessingml/2006/main"
10 namespace w10 = "urn:schemas-microsoft-com:office:word"
11 namespace x = "urn:schemas-microsoft-com:office:excel"
12
13 v_AG_Id = attribute id { xsd:string }?
14 v_AG_Style = attribute style { xsd:string }?
15 v_AG_Type = attribute type { xsd:string }?
16 v_AG_Adj = attribute adj { xsd:string }?
17 v_AG_Path = attribute path { xsd:string }?
18 v_AG_Fill =
19   attribute filled { s_ST_TrueFalse }?,
20   attribute fillcolor { s_ST_ColorType }?
21 v_AG_ChromaKey = attribute chromakey { s_ST_ColorType }?
22 v_AG_Ext = attribute v:ext { v_ST_Ext }?
23 v_AG_CoreAttributes =
24   v_AG_Id,
25   v_AG_Style,
26   attribute href { xsd:string }?,
27   attribute target { xsd:string }?,
28   attribute class { xsd:string }?,
29   attribute title { xsd:string }?,
30   attribute alt { xsd:string }?,
31   attribute coordsize { xsd:string }?,
32   attribute coordorigin { xsd:string }?,
33   attribute wrapcoords { xsd:string }?,
34   attribute print { s_ST_TrueFalse }?
35 v_AG_ShapeAttributes =
36   v_AG_ChromaKey,
37   v_AG_Fill,
38   attribute opacity { xsd:string }?,
39   attribute stroked { s_ST_TrueFalse }?,
40   attribute strokecolor { s_ST_ColorType }?,
41   attribute strokeweight { xsd:string }?,
42   attribute insetpen { s_ST_TrueFalse }?
43 v_AG_OfficeCoreAttributes =
44   o_spid?,
45   o_oned?,
46   o_regroupid?,
47   o_doubleclicknotify?,

```



```

48   o_button?,
49   o_userhidden?,
50   o_bullet?,
51   o_hr?,
52   o_hrstd?,
53   o_hrnoshade?,
54   o_hrpct?,
55   o_hralign?,
56   o_allowincell?,
57   o_allowoverlap?,
58   o_userdrawn?,
59   o_bordertopcolor?,
60   o_borderleftcolor?,
61   o_borderbottomcolor?,
62   o_borderrightcolor?,
63   o_dgmlayout?,
64   o_dgmnodekind?,
65   o_dgmlayoutmru?,
66   o_insetmode?
67 v_AG_OfficeShapeAttributes =
68   o_spt?,
69   o_connectortype?,
70   o_bwmode?,
71   o_bwpure?,
72   o_bwnormal?,
73   o_forcedash?,
74   o_oleicon?,
75   o_ole?,
76   o_preferrelative?,
77   o_cliptowrap?,
78   o_clip?
79 v_AG_AllCoreAttributes = v_AG_CoreAttributes, v_AG_OfficeCoreAttributes
80 v_AG_AllShapeAttributes =
81   v_AG_ShapeAttributes, v_AG_OfficeShapeAttributes
82 v_AG_ImageAttributes =
83   attribute src { xsd:string }?,
84   attribute cropleft { xsd:string }?,
85   attribute croptop { xsd:string }?,
86   attribute cropright { xsd:string }?,
87   attribute cropbottom { xsd:string }?,
88   attribute gain { xsd:string }?,
89   attribute blacklevel { xsd:string }?,
90   attribute gamma { xsd:string }?,
91   attribute grayscale { s_ST_TrueFalse }?,
92   attribute bilevel { s_ST_TrueFalse }?
93 v_AG_StrokeAttributes =
94   attribute on { s_ST_TrueFalse }?,
95   attribute weight { xsd:string }?,
96   attribute color { s_ST_ColorType }?,
97   attribute opacity { xsd:string }?,
98   attribute linestyle { v_ST_StrokeLineStyle }?,
99   attribute miterlimit { xsd:decimal }?,
100  attribute joinstyle { v_ST_StrokeJoinStyle }?,

```

```

101 attribute endcap { v_ST_StrokeEndCap }?,
102 attribute dashstyle { xsd:string }?,
103 attribute filltype { v_ST_FillType }?,
104 attribute src { xsd:string }?,
105 attribute imageaspect { v_ST_ImageAspect }?,
106 attribute imagesize { xsd:string }?,
107 attribute imagealignshape { s_ST_TrueFalse }?,
108 attribute color2 { s_ST_ColorType }?,
109 attribute startarrow { v_ST_StrokeArrowType }?,
110 attribute startarrowwidth { v_ST_StrokeArrowWidth }?,
111 attribute startarrowlength { v_ST_StrokeArrowLength }?,
112 attribute endarrow { v_ST_StrokeArrowType }?,
113 attribute endarrowwidth { v_ST_StrokeArrowWidth }?,
114 attribute endarrowlength { v_ST_StrokeArrowLength }?,
115 o_href?,
116 o_althref?,
117 o_title?,
118 o_forcedash?,
119 r_id?,
120 attribute insetpen { s_ST_TrueFalse }?,
121 o_relid?
122 v_EG_ShapeElements =
123   v_path
124   | v_formulas
125   | v_handles
126   | v_fill
127   | v_stroke
128   | v_shadow
129   | v_textbox
130   | v_textpath
131   | v_imagedata
132   | o_skew
133   | o_extrusion
134   | o_callout
135   | o_lock
136   | o_clippath
137   | o_signatureline
138   | w10_wrap
139   | w10_anchorlock
140   | w10_bordertop
141   | w10_borderbottom
142   | w10_borderleft
143   | w10_borderright
144   | x_ClientData?
145   | pvml_textdata?
146 v_shape = element shape { v_CT_Shape }
147 v_shapetype = element shapetype { v_CT_Shapetype }
148 v_group = element group { v_CT_Group }
149 v_background = element background { v_CT_Background }
150 v_CT_Shape =
151   v_AG_AllCoreAttributes,
152   v_AG_AllShapeAttributes,
153   v_AG_Type,

```

```

154     v_AG_Adj,
155     v_AG_Path,
156     o_gfxdata?,
157     attribute equationxml { xsd:string }?,
158     (v_EG_ShapeElements | o_ink | pvml_iscomment | o_equationxml)+
159 v_CT_Shapetype =
160     v_AG_AllCoreAttributes,
161     v_AG_AllShapeAttributes,
162     v_AG_Adj,
163     v_AG_Path,
164     o_master?,
165     v_EG_ShapeElements*,
166     o_complex?
167 v_CT_Group =
168     v_AG_AllCoreAttributes,
169     v_AG_Fill,
170     attribute editas { v_ST_EditAs }?,
171     o_tableproperties?,
172     o_tablelimits?,
173     (v_EG_ShapeElements
174         | v_group
175         | v_shape
176         | v_shapetype
177         | v_arc
178         | v_curve
179         | v_image
180         | v_line
181         | v_oval
182         | v_polyline
183         | v_rect
184         | v_roundrect
185         | o_diagram)+
186 v_CT_Background =
187     v_AG_Id,
188     v_AG_Fill,
189     o_bwmode?,
190     o_bwpure?,
191     o_bwnormal?,
192     o_targetssize?,
193     v_fill?
194 v_fill = element fill { v_CT_Fill }
195 v_formulas = element formulas { v_CT_Formulas }
196 v_handles = element handles { v_CT_Handles }
197 v_imagedata = element imagedata { v_CT_ImageData }
198 v_path = element path { v_CT_Path }
199 v_textbox = element textbox { v_CT_Textbox }
200 v_shadow = element shadow { v_CT_Shadow }
201 v_stroke = element stroke { v_CT_Stroke }
202 v_textpath = element textpath { v_CT_TextPath }
203 v_CT_Fill =
204     v_AG_Id,
205     attribute type { v_ST_FillType }?,
206     attribute on { s_ST_TrueFalse }?,

```

```

207 attribute color { s_ST_ColorType }?,
208 attribute opacity { xsd:string }?,
209 attribute color2 { s_ST_ColorType }?,
210 attribute src { xsd:string }?,
211 o_href?,
212 o_althref?,
213 attribute size { xsd:string }?,
214 attribute origin { xsd:string }?,
215 attribute position { xsd:string }?,
216 attribute aspect { v_ST_ImageAspect }?,
217 attribute colors { xsd:string }?,
218 attribute angle { xsd:decimal }?,
219 attribute alignshape { s_ST_TrueFalse }?,
220 attribute focus { xsd:string }?,
221 attribute focussize { xsd:string }?,
222 attribute focusposition { xsd:string }?,
223 attribute method { v_ST_FillMethod }?,
224 o_detectmouseclick?,
225 o_title?,
226 o_opacity2?,
227 attribute recolor { s_ST_TrueFalse }?,
228 attribute rotate { s_ST_TrueFalse }?,
229 r_id?,
230 o_relid?,
231 o_fill?
232 v_CT_Formulas = element f { v_CT_F }*
233 v_CT_F = attribute eqn { xsd:string }?
234 v_CT_Handles = element h { v_CT_H }*
235 v_CT_H =
236 attribute position { xsd:string }?,
237 attribute polar { xsd:string }?,
238 attribute map { xsd:string }?,
239 attribute invx { s_ST_TrueFalse }?,
240 attribute invy { s_ST_TrueFalse }?,
241 attribute switch { s_ST_TrueFalseBlank }?,
242 attribute xrange { xsd:string }?,
243 attribute yrange { xsd:string }?,
244 attribute radiusrange { xsd:string }?
245 v_CT_ImageData =
246 v_AG_Id,
247 v_AG_ImageAttributes,
248 v_AG_ChromaKey,
249 attribute embosscolor { s_ST_ColorType }?,
250 attribute recolorTarget { s_ST_ColorType }?,
251 o_href?,
252 o_althref?,
253 o_title?,
254 o_oleid?,
255 o_detectmouseclick?,
256 o_movie?,
257 o_relid?,
258 r_id?,
259 r_pict?,

```

```

260   r_href?
261 v_CT_Path =
262   v_AG_Id,
263   attribute v { xsd:string }?,
264   attribute limo { xsd:string }?,
265   attribute textboxrect { xsd:string }?,
266   attribute fillok { s_ST_TrueFalse }?,
267   attribute strokeok { s_ST_TrueFalse }?,
268   attribute shadowok { s_ST_TrueFalse }?,
269   attribute arrowok { s_ST_TrueFalse }?,
270   attribute gradientshapeok { s_ST_TrueFalse }?,
271   attribute textpathok { s_ST_TrueFalse }?,
272   attribute insetpenok { s_ST_TrueFalse }?,
273   o_connecttype?,
274   o_connectlocs?,
275   o_connectangles?,
276   o_extrusionok?
277 v_CT_Shadow =
278   v_AG_Id,
279   attribute on { s_ST_TrueFalse }?,
280   attribute type { v_ST_ShadowType }?,
281   attribute obscured { s_ST_TrueFalse }?,
282   attribute color { s_ST_ColorType }?,
283   attribute opacity { xsd:string }?,
284   attribute offset { xsd:string }?,
285   attribute color2 { s_ST_ColorType }?,
286   attribute offset2 { xsd:string }?,
287   attribute origin { xsd:string }?,
288   attribute matrix { xsd:string }?
289 v_CT_Stroke =
290   v_AG_Id,
291   v_AG_StrokeAttributes,
292   o_left?,
293   o_top?,
294   o_right?,
295   o_bottom?,
296   o_column?
297 v_CT_Textbox =
298   v_AG_Id,
299   v_AG_Style,
300   attribute inset { xsd:string }?,
301   o_singleclick?,
302   o_insetmode?,
303   (w_txbxContent? | anyHTMLElementAsLocalElement)
304 anyHTMLElementAsLocalElement = element local:* { anyAttribute*, text?,
305   anyHTMLElementAsLocalElement* }
306 v_CT_TextPath =
307   v_AG_Id,
308   v_AG_Style,
309   attribute on { s_ST_TrueFalse }?,
310   attribute fitshape { s_ST_TrueFalse }?,
311   attribute fitpath { s_ST_TrueFalse }?,
312   attribute trim { s_ST_TrueFalse }?,

```

```

313     attribute xscale { s_ST_TrueFalse }?,
314     attribute string { xsd:string }?
315 v_arc = element arc { v_CT_Arc }
316 v_curve = element curve { v_CT_Curve }
317 v_image = element image { v_CT_Image }
318 v_line = element line { v_CT_Line }
319 v_oval = element oval { v_CT_Oval }
320 v_polyline = element polyline { v_CT_PolyLine }
321 v_rect = element rect { v_CT_Rect }
322 v_roundrect = element roundrect { v_CT_RoundRect }
323 v_CT_Arc =
324     v_AG_AllCoreAttributes,
325     v_AG_AllShapeAttributes,
326     attribute startAngle { xsd:decimal }?,
327     attribute endAngle { xsd:decimal }?,
328     v_EG_ShapeElements*
329 v_CT_Curve =
330     v_AG_AllCoreAttributes,
331     v_AG_AllShapeAttributes,
332     attribute from { xsd:string }?,
333     attribute control1 { xsd:string }?,
334     attribute control2 { xsd:string }?,
335     attribute to { xsd:string }?,
336     v_EG_ShapeElements*
337 v_CT_Image =
338     v_AG_AllCoreAttributes,
339     v_AG_AllShapeAttributes,
340     v_AG_ImageAttributes,
341     v_EG_ShapeElements*
342 v_CT_Line =
343     v_AG_AllCoreAttributes,
344     v_AG_AllShapeAttributes,
345     attribute from { xsd:string }?,
346     attribute to { xsd:string }?,
347     v_EG_ShapeElements*
348 v_CT_Oval =
349     v_AG_AllCoreAttributes,
350     v_AG_AllShapeAttributes,
351     (v_EG_ShapeElements)*
352 v_CT_PolyLine =
353     v_AG_AllCoreAttributes,
354     v_AG_AllShapeAttributes,
355     attribute points { xsd:string }?,
356     (v_EG_ShapeElements | o_ink)*
357 v_CT_Rect =
358     v_AG_AllCoreAttributes,
359     v_AG_AllShapeAttributes,
360     (v_EG_ShapeElements)*
361 v_CT_RoundRect =
362     v_AG_AllCoreAttributes,
363     v_AG_AllShapeAttributes,
364     attribute arcsize { xsd:string }?,
365     (v_EG_ShapeElements)*

```

```

366 v_ST_Ext = string "view" | string "edit" | string "backwardCompatible"
367 v_ST_FillType =
368     string "solid"
369     | string "gradient"
370     | string "gradientRadial"
371     | string "tile"
372     | string "pattern"
373     | string "frame"
374 v_ST_FillMethod =
375     string "none"
376     | string "linear"
377     | string "sigma"
378     | string "any"
379     | string "linear sigma"
380 v_ST_ShadowType =
381     string "single"
382     | string "double"
383     | string "emboss"
384     | string "perspective"
385 v_ST_StrokeLineStyle =
386     string "single"
387     | string "thinThin"
388     | string "thinThick"
389     | string "thickThin"
390     | string "thickBetweenThin"
391 v_ST_StrokeJoinStyle = string "round" | string "bevel" | string "miter"
392 v_ST_StrokeEndCap = string "flat" | string "square" | string "round"
393 v_ST_StrokeArrowLength =
394     string "short" | string "medium" | string "long"
395 v_ST_StrokeArrowWidth =
396     string "narrow" | string "medium" | string "wide"
397 v_ST_StrokeArrowType =
398     string "none"
399     | string "block"
400     | string "classic"
401     | string "oval"
402     | string "diamond"
403     | string "open"
404 v_ST_ImageAspect = string "ignore" | string "atMost" | string "atLeast"
405 v_ST_EditAs =
406     string "canvas"
407     | string "orgchart"
408     | string "radial"
409     | string "cycle"
410     | string "stacked"
411     | string "venn"
412     | string "bullseye"

```

B.6.2 VML - Office Drawing

This schema is available in the file vml-officeDrawing.rnc.

```

1 default namespace o = "urn:schemas-microsoft-com:office:office"

```

```

2 namespace r =
3   "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace v = "urn:schemas-microsoft-com:vml"
7 namespace w10 = "urn:schemas-microsoft-com:office:word"
8 namespace x = "urn:schemas-microsoft-com:office:excel"
9
10 o_bwmode = attribute o:bwmode { o_ST_BWMode }
11 o_bwpure = attribute o:bwpure { o_ST_BWMode }
12 o_bwnormal = attribute o:bwnormal { o_ST_BWMode }
13 o_targetscreenSize = attribute o:targetscreenSize { o_ST_ScreenSize }
14 o_insetmode =
15
16   ## default value: custom
17   attribute o:insetmode { o_ST_InsetMode }
18 o_spt = attribute o:spt { xsd:float }
19 o_wrapcoords = attribute o:wrapcoords { xsd:string }
20 o_oned = attribute o:oned { s_ST_TrueFalse }
21 o_regroupid = attribute o:regroupid { xsd:integer }
22 o_doubleclicknotify = attribute o:doubleclicknotify { s_ST_TrueFalse }
23 o_connectortype =
24
25   ## default value: straight
26   attribute o:connectortype { o_ST_ConnectorType }
27 o_button = attribute o:button { s_ST_TrueFalse }
28 o_userhidden = attribute o:userhidden { s_ST_TrueFalse }
29 o_forcedash = attribute o:forcedash { s_ST_TrueFalse }
30 o_oleicon = attribute o:oleicon { s_ST_TrueFalse }
31 o_ole = attribute o:ole { s_ST_TrueFalseBlank }
32 o_preferrelative = attribute o:preferrelative { s_ST_TrueFalse }
33 o_cliptowrap = attribute o:cliptowrap { s_ST_TrueFalse }
34 o_clip = attribute o:clip { s_ST_TrueFalse }
35 o_bullet = attribute o:bullet { s_ST_TrueFalse }
36 o_hr = attribute o:hr { s_ST_TrueFalse }
37 o_hrstd = attribute o:hrstd { s_ST_TrueFalse }
38 o_hrnoshade = attribute o:hrnoshade { s_ST_TrueFalse }
39 o_hrpct = attribute o:hrpct { xsd:float }
40 o_hralign =
41
42   ## default value: left
43   attribute o:hralign { o_ST_HrAlign }
44 o_allowincell = attribute o:allowincell { s_ST_TrueFalse }
45 o_allowoverlap = attribute o:allowoverlap { s_ST_TrueFalse }
46 o_userdrawn = attribute o:userdrawn { s_ST_TrueFalse }
47 o_bordertopcolor = attribute o:bordertopcolor { xsd:string }
48 o_borderleftcolor = attribute o:borderleftcolor { xsd:string }
49 o_borderbottomcolor = attribute o:borderbottomcolor { xsd:string }
50 o_borderrightcolor = attribute o:borderrightcolor { xsd:string }
51 o_connecttype = attribute o:connecttype { o_ST_ConnectType }
52 o_connectlocs = attribute o:connectlocs { xsd:string }
53 o_connectangles = attribute o:connectangles { xsd:string }
54 o_master = attribute o:master { xsd:string }

```



```

55 o_extrusionok = attribute o:extrusionok { s_ST_TrueFalse }
56 o_href = attribute o:href { xsd:string }
57 o_althref = attribute o:althref { xsd:string }
58 o_title = attribute o:title { xsd:string }
59 o_singleclick = attribute o:singleclick { s_ST_TrueFalse }
60 o_oleid = attribute o:oleid { xsd:float }
61 o_detectmouseclick = attribute o:detectmouseclick { s_ST_TrueFalse }
62 o_movie = attribute o:movie { xsd:float }
63 o_spid = attribute o:spid { xsd:string }
64 o_opacity2 = attribute o:opacity2 { xsd:string }
65 o_relid = attribute o:relid { r_ST_RelationshipId }
66 o_dgmlayout = attribute o:dgmlayout { o_ST_DiagramLayout }
67 o_dgmnodekind = attribute o:dgmnodekind { xsd:integer }
68 o_dgmlayoutmru = attribute o:dgmlayoutmru { o_ST_DiagramLayout }
69 o_gfxdata = attribute o:gfxdata { xsd:base64Binary }
70 o_tableproperties = attribute o:tableproperties { xsd:string }
71 o_tablelimits = attribute o:tablelimits { xsd:string }
72 o_shapedefaults = element shapedefaults { o_CT_ShapeDefaults }
73 o_shapelayout = element shapelayout { o_CT_Shapelayout }
74 o_signatureline = element signatureline { o_CT_SignatureLine }
75 o_ink = element ink { o_CT_Ink }
76 o_diagram = element diagram { o_CT_Diagram }
77 o_equationxml = element equationxml { o_CT_EquationXml }
78 o_CT_ShapeDefaults =
79   v_AG_Ext,
80   attribute spidmax { xsd:integer }?,
81   attribute style { xsd:string }?,
82   attribute fill { s_ST_TrueFalse }?,
83   attribute fillcolor { s_ST_ColorType }?,
84   attribute stroke { s_ST_TrueFalse }?,
85   attribute strokecolor { s_ST_ColorType }?,
86   attribute o:allowincell { s_ST_TrueFalse }?,
87   (v_fill?
88     & v_stroke?
89     & v_textbox?
90     & v_shadow?
91     & o_skew?
92     & o_extrusion?
93     & o_callout?
94     & o_lock?
95     & element colormru { o_CT_ColorMru }?
96     & element colormenu { o_CT_ColorMenu }?)?
97 o_CT_Ink =
98   attribute i { xsd:string }?,
99   attribute annotation { s_ST_TrueFalse }?,
100   attribute contentType { o_ST_ContentType }?,
101   empty
102 o_CT_SignatureLine =
103   v_AG_Ext,
104   attribute issignatureline { s_ST_TrueFalse }?,
105   attribute id { s_ST_Guid }?,
106   attribute provid { s_ST_Guid }?,
107   attribute signinginstructionsset { s_ST_TrueFalse }?,

```

```

108     attribute allowcomments { s_ST_TrueFalse }?,
109     attribute showsigndate { s_ST_TrueFalse }?,
110     attribute o:suggestedesigner { xsd:string }?,
111     attribute o:suggestedesigner2 { xsd:string }?,
112     attribute o:suggestedesigneremail { xsd:string }?,
113     attribute signinginstructions { xsd:string }?,
114     attribute addlxml { xsd:string }?,
115     attribute sigprovurl { xsd:string }?
116 o_CT_ShapeLayout =
117     v_AG_Ext,
118     (element idmap { o_CT_IdMap }?
119      & element regrouptable { o_CT_RegroupTable }?
120      & element rules { o_CT_Rules }?)
121 o_CT_IdMap =
122     v_AG_Ext,
123     attribute data { xsd:string }?
124 o_CT_RegroupTable =
125     v_AG_Ext,
126     element entry { o_CT_Entry }*
127 o_CT_Entry =
128     attribute new { xsd:int }?,
129     attribute old { xsd:int }?
130 o_CT_Rules =
131     v_AG_Ext,
132     element r { o_CT_R }*
133 o_CT_R =
134     attribute id { xsd:string },
135     attribute type { o_ST_RType }?,
136     attribute how { o_ST_How }?,
137     attribute idref { xsd:string }?,
138     element proxy { o_CT_Proxy }*
139 o_CT_Proxy =
140
141     ## default value: false
142     attribute start { s_ST_TrueFalseBlank }?,
143
144     ## default value: false
145     attribute end { s_ST_TrueFalseBlank }?,
146     attribute idref { xsd:string }?,
147     attribute connectloc { xsd:int }?
148 o_CT_Diagram =
149     v_AG_Ext,
150     attribute dgmstyle { xsd:integer }?,
151     attribute autoformat { s_ST_TrueFalse }?,
152     attribute reverse { s_ST_TrueFalse }?,
153     attribute autolayout { s_ST_TrueFalse }?,
154     attribute dgmscalex { xsd:integer }?,
155     attribute dgmscaley { xsd:integer }?,
156     attribute dgmfontsize { xsd:integer }?,
157     attribute constrainbounds { xsd:string }?,
158     attribute dgmbasetextscale { xsd:integer }?,
159     element relationtable { o_CT_RelationTable }?
160 o_CT_EquationXml =

```

```

161   attribute contentType { o_ST_AlternateMathContentType }?,
162   o_CT_EquationXml_any
163 o_CT_EquationXml_any =
164   element * - (o:* | v:* | w10:* | x:*) {
165     anyAttribute*,
166     mixed { anyElement* }
167   }
168 o_ST_AlternateMathContentType = xsd:string
169 o_CT_RelationTable =
170   v_AG_Ext,
171   element rel { o_CT_Relation }*
172 o_CT_Relation =
173   v_AG_Ext,
174   attribute idsrc { xsd:string }?,
175   attribute iddest { xsd:string }?,
176   attribute idcntr { xsd:string }?
177 o_CT_ColorMru =
178   v_AG_Ext,
179   attribute colors { xsd:string }?
180 o_CT_ColorMenu =
181   v_AG_Ext,
182   attribute strokecolor { s_ST_ColorType }?,
183   attribute fillcolor { s_ST_ColorType }?,
184   attribute shadowcolor { s_ST_ColorType }?,
185   attribute extrusioncolor { s_ST_ColorType }?
186 o_skew = element skew { o_CT_Skew }
187 o_extrusion = element extrusion { o_CT_Extrusion }
188 o_callout = element callout { o_CT_Callout }
189 o_lock = element lock { o_CT_Lock }
190 o_OLEObject = element OLEObject { o_CT_OLEObject }
191 o_complex = element complex { o_CT_Complex }
192 o_left = element left { o_CT_StrokeChild }
193 o_top = element top { o_CT_StrokeChild }
194 o_right = element right { o_CT_StrokeChild }
195 o_bottom = element bottom { o_CT_StrokeChild }
196 o_column = element column { o_CT_StrokeChild }
197 o_clippath = element clippath { o_CT_ClipPath }
198 o_fill = element fill { o_CT_Fill }
199 o_CT_Skew =
200   v_AG_Ext,
201   attribute id { xsd:string }?,
202   attribute on { s_ST_TrueFalse }?,
203   attribute offset { xsd:string }?,
204   attribute origin { xsd:string }?,
205   attribute matrix { xsd:string }?
206 o_CT_Extrusion =
207   v_AG_Ext,
208   attribute on { s_ST_TrueFalse }?,
209
210   ## default value: parallel
211   attribute type { o_ST_ExtrusionType }?,
212
213   ## default value: solid

```

```

214 attribute render { o_ST_ExtrusionRender }?,
215 attribute viewpointorigin { xsd:string }?,
216 attribute viewpoint { xsd:string }?,
217
218 ## default value: XY
219 attribute plane { o_ST_ExtrusionPlane }?,
220 attribute skewangle { xsd:float }?,
221 attribute skewamt { xsd:string }?,
222 attribute foredepth { xsd:string }?,
223 attribute backdepth { xsd:string }?,
224 attribute orientation { xsd:string }?,
225 attribute orientationangle { xsd:float }?,
226 attribute lockrotationcenter { s_ST_TrueFalse }?,
227 attribute autorotationcenter { s_ST_TrueFalse }?,
228 attribute rotationcenter { xsd:string }?,
229 attribute rotationangle { xsd:string }?,
230 attribute colormode { o_ST_ColorMode }?,
231 attribute color { s_ST_ColorType }?,
232 attribute shininess { xsd:float }?,
233 attribute specularity { xsd:string }?,
234 attribute diffusity { xsd:string }?,
235 attribute metal { s_ST_TrueFalse }?,
236 attribute edge { xsd:string }?,
237 attribute facet { xsd:string }?,
238 attribute lightface { s_ST_TrueFalse }?,
239 attribute brightness { xsd:string }?,
240 attribute lightposition { xsd:string }?,
241 attribute lightlevel { xsd:string }?,
242 attribute lightharsh { s_ST_TrueFalse }?,
243 attribute lightposition2 { xsd:string }?,
244 attribute lightlevel2 { xsd:string }?,
245 attribute lightharsh2 { s_ST_TrueFalse }?
246 o_CT_Callout =
247   v_AG_Ext,
248   attribute on { s_ST_TrueFalse }?,
249   attribute type { xsd:string }?,
250   attribute gap { xsd:string }?,
251   attribute angle { o_ST_Angle }?,
252   attribute dropauto { s_ST_TrueFalse }?,
253   attribute drop { o_ST_CalloutDrop }?,
254   attribute distance { xsd:string }?,
255
256 ## default value: f
257 attribute lengthspecified { s_ST_TrueFalse }?,
258 attribute length { xsd:string }?,
259 attribute accentbar { s_ST_TrueFalse }?,
260 attribute textborder { s_ST_TrueFalse }?,
261 attribute minusx { s_ST_TrueFalse }?,
262 attribute minusy { s_ST_TrueFalse }?
263 o_CT_Lock =
264   v_AG_Ext,
265   attribute position { s_ST_TrueFalse }?,
266   attribute selection { s_ST_TrueFalse }?,

```

```

267 attribute grouping { s_ST_TrueFalse }?,
268 attribute ungrouping { s_ST_TrueFalse }?,
269 attribute rotation { s_ST_TrueFalse }?,
270 attribute cropping { s_ST_TrueFalse }?,
271 attribute verticies { s_ST_TrueFalse }?,
272 attribute adjusthandles { s_ST_TrueFalse }?,
273 attribute text { s_ST_TrueFalse }?,
274 attribute aspectratio { s_ST_TrueFalse }?,
275 attribute shapetype { s_ST_TrueFalse }?
276 o_CT_OLEObject =
277   attribute Type { o_ST_OLEType }?,
278   attribute ProgID { xsd:string }?,
279   attribute ShapeID { xsd:string }?,
280   attribute DrawAspect { o_ST_OLEDrawAspect }?,
281   attribute ObjectID { xsd:string }?,
282   r_id?,
283   attribute UpdateMode { o_ST_OLEUpdateMode }?,
284   element LinkType { o_ST_OLELinkType }?,
285   element LockedField { s_ST_TrueFalseBlank }?,
286   element FieldCodes { xsd:string }?
287 o_CT_Complex = v_AG_Ext
288 o_CT_StrokeChild =
289   v_AG_Ext,
290   attribute on { s_ST_TrueFalse }?,
291   attribute weight { xsd:string }?,
292   attribute color { s_ST_ColorType }?,
293   attribute color2 { s_ST_ColorType }?,
294   attribute opacity { xsd:string }?,
295   attribute linestyle { v_ST_StrokeLineStyle }?,
296   attribute miterlimit { xsd:decimal }?,
297   attribute joinstyle { v_ST_StrokeJoinStyle }?,
298   attribute endcap { v_ST_StrokeEndCap }?,
299   attribute dashstyle { xsd:string }?,
300   attribute insetpen { s_ST_TrueFalse }?,
301   attribute filltype { v_ST_FillType }?,
302   attribute src { xsd:string }?,
303   attribute imageaspect { v_ST_ImageAspect }?,
304   attribute imagesize { xsd:string }?,
305   attribute imagealignshape { s_ST_TrueFalse }?,
306   attribute startarrow { v_ST_StrokeArrowType }?,
307   attribute startarrowwidth { v_ST_StrokeArrowWidth }?,
308   attribute startarrowlength { v_ST_StrokeArrowLength }?,
309   attribute endarrow { v_ST_StrokeArrowType }?,
310   attribute endarrowwidth { v_ST_StrokeArrowWidth }?,
311   attribute endarrowlength { v_ST_StrokeArrowLength }?,
312   o_href?,
313   o_althref?,
314   o_title?,
315   o_forcedash?
316 o_CT_ClipPath = attribute o:v { xsd:string }
317 o_CT_Fill =
318   v_AG_Ext,
319   attribute type { o_ST_FillType }?

```

```

320 o_ST_RType =
321     string "arc" | string "callout" | string "connector" | string "align"
322 o_ST_How =
323     string "top"
324     | string "middle"
325     | string "bottom"
326     | string "left"
327     | string "center"
328     | string "right"
329 o_ST_BWMode =
330     string "color"
331     | string "auto"
332     | string "grayScale"
333     | string "lightGrayscale"
334     | string "inverseGray"
335     | string "grayOutline"
336     | string "highContrast"
337     | string "black"
338     | string "white"
339     | string "hide"
340     | string "undrawn"
341     | string "blackTextAndLines"
342 o_ST_ScreenSize =
343     string "544,376"
344     | string "640,480"
345     | string "720,512"
346     | string "800,600"
347     | string "1024,768"
348     | string "1152,862"
349 o_ST_InsetMode = string "auto" | string "custom"
350 o_ST_ColorMode = string "auto" | string "custom"
351 o_ST_ContentType = xsd:string
352 o_ST_DiagramLayout = "0" | "1" | "2" | "3"
353 o_ST_ExtrusionType = string "perspective" | string "parallel"
354 o_ST_ExtrusionRender =
355     string "solid" | string "wireFrame" | string "boundingCube"
356 o_ST_ExtrusionPlane = string "XY" | string "ZX" | string "YZ"
357 o_ST_Angle =
358     string "any"
359     | string "30"
360     | string "45"
361     | string "60"
362     | string "90"
363     | string "auto"
364 o_ST_CalloutDrop = xsd:string
365 o_ST_CalloutPlacement =
366     string "top" | string "center" | string "bottom" | string "user"
367 o_ST_ConnectorType =
368     string "none" | string "straight" | string "elbow" | string "curved"
369 o_ST_HrAlign = string "left" | string "right" | string "center"
370 o_ST_ConnectType =
371     string "none" | string "rect" | string "segments" | string "custom"
372 o_ST_OLELinkType = xsd:string

```

```

373 o_ST_OLEType = string "Embed" | string "Link"
374 o_ST_OLEDrawAspect = string "Content" | string "Icon"
375 o_ST_OLEUpdateMode = string "Always" | string "OnCall"
376 o_ST_FillType =
377     string "gradientCenter"
378     | string "solid"
379     | string "pattern"
380     | string "tile"
381     | string "frame"
382     | string "gradientUnscaled"
383     | string "gradientRadial"
384     | string "gradient"
385     | string "background"
386 o_any_vml_vml =
387     v_shape
388     | v_shapetype
389     | v_group
390     | v_background
391     | v_fill
392     | v_formulas
393     | v_handles
394     | v_imagedata
395     | v_path
396     | v_textbox
397     | v_shadow
398     | v_stroke
399     | v_textpath
400     | v_arc
401     | v_curve
402     | v_image
403     | v_line
404     | v_oval
405     | v_polyline
406     | v_rect
407     | v_roundrect

```

B.6.3 VML - Wordprocessing Drawing

This schema is available in the file vml-wordprocessingDrawing.rnc.

```

1 default namespace = "urn:schemas-microsoft-com:office:word"
2 namespace o = "urn:schemas-microsoft-com:office:office"
3 namespace v = "urn:schemas-microsoft-com:vml"
4 namespace w10 = "urn:schemas-microsoft-com:office:word"
5 namespace x = "urn:schemas-microsoft-com:office:excel"
6
7 w10_bordertop = element bordertop { w10_CT_Border }
8 w10_borderleft = element borderleft { w10_CT_Border }
9 w10_borderright = element borderright { w10_CT_Border }
10 w10_borderbottom = element borderbottom { w10_CT_Border }
11 w10_CT_Border =
12     attribute type { w10_ST_BorderType }?,
13     attribute width { xsd:positiveInteger }?,

```

```

14     attribute shadow { w10_ST_BorderShadow }?
15 w10_wrap = element wrap { w10_CT_Wrap }
16 w10_CT_Wrap =
17     attribute type { w10_ST_WrapType }?,
18     attribute side { w10_ST_WrapSide }?,
19     attribute anchorx { w10_ST_HorizontalAnchor }?,
20     attribute anchory { w10_ST_VerticalAnchor }?
21 w10_anchorlock = element anchorlock { w10_CT_AnchorLock }
22 w10_CT_AnchorLock = empty
23 w10_ST_BorderType =
24     string "none"
25     | string "single"
26     | string "thick"
27     | string "double"
28     | string "hairline"
29     | string "dot"
30     | string "dash"
31     | string "dotDash"
32     | string "dashDotDot"
33     | string "triple"
34     | string "thinThickSmall"
35     | string "thickThinSmall"
36     | string "thickBetweenThinSmall"
37     | string "thinThick"
38     | string "thickThin"
39     | string "thickBetweenThin"
40     | string "thinThickLarge"
41     | string "thickThinLarge"
42     | string "thickBetweenThinLarge"
43     | string "wave"
44     | string "doubleWave"
45     | string "dashedSmall"
46     | string "dashDotStroked"
47     | string "threeDEmboss"
48     | string "threeDEngrave"
49     | string "HTMLOutset"
50     | string "HTMLInset"
51 w10_ST_BorderShadow =
52     string "t" | string "true" | string "f" | string "false"
53 w10_ST_WrapType =
54     string "topAndBottom"
55     | string "square"
56     | string "none"
57     | string "tight"
58     | string "through"
59 w10_ST_WrapSide =
60     string "both" | string "left" | string "right" | string "largest"
61 w10_ST_HorizontalAnchor =
62     string "margin" | string "page" | string "text" | string "char"
63 w10_ST_VerticalAnchor =
64     string "margin" | string "page" | string "text" | string "line"

```


B.6.4 VML - Spreadsheet Drawing

This schema is available in the file vml-spreadsheetDrawing.rnc.

```

1 default namespace = "urn:schemas-microsoft-com:office:excel"
2 namespace o = "urn:schemas-microsoft-com:office:office"
3 namespace s =
4   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
5 namespace v = "urn:schemas-microsoft-com:vml"
6 namespace w10 = "urn:schemas-microsoft-com:office:word"
7 namespace x = "urn:schemas-microsoft-com:office:excel"
8
9 x_ClientData = element ClientData { x_CT_ClientData }
10 x_CT_ClientData =
11   attribute ObjectType { x_ST_ObjectType },
12   (element MoveWithCells { s_ST_TrueFalseBlank }
13     | element SizeWithCells { s_ST_TrueFalseBlank }
14     | element Anchor { xsd:string }
15     | element Locked { s_ST_TrueFalseBlank }
16     | element DefaultSize { s_ST_TrueFalseBlank }
17     | element PrintObject { s_ST_TrueFalseBlank }
18     | element Disabled { s_ST_TrueFalseBlank }
19     | element AutoFill { s_ST_TrueFalseBlank }
20     | element AutoLine { s_ST_TrueFalseBlank }
21     | element AutoPict { s_ST_TrueFalseBlank }
22     | element FmlaMacro { xsd:string }
23     | element TextHAlign { xsd:string }
24     | element TextVAlign { xsd:string }
25     | element LockText { s_ST_TrueFalseBlank }
26     | element JustLastX { s_ST_TrueFalseBlank }
27     | element SecretEdit { s_ST_TrueFalseBlank }
28     | element Default { s_ST_TrueFalseBlank }
29     | element Help { s_ST_TrueFalseBlank }
30     | element Cancel { s_ST_TrueFalseBlank }
31     | element Dismiss { s_ST_TrueFalseBlank }
32     | element Accel { xsd:integer }
33     | element Accel2 { xsd:integer }
34     | element Row { xsd:integer }
35     | element Column { xsd:integer }
36     | element Visible { s_ST_TrueFalseBlank }
37     | element RowHidden { s_ST_TrueFalseBlank }
38     | element ColHidden { s_ST_TrueFalseBlank }
39     | element VTEdit { xsd:integer }
40     | element MultiLine { s_ST_TrueFalseBlank }
41     | element VScroll { s_ST_TrueFalseBlank }
42     | element ValidIds { s_ST_TrueFalseBlank }
43     | element FmlaRange { xsd:string }
44     | element WidthMin { xsd:integer }
45     | element Sel { xsd:integer }
46     | element NoThreeD2 { s_ST_TrueFalseBlank }
47     | element SelType { xsd:string }
48     | element MultiSel { xsd:string }

```

```

49 | element LCT { xsd:string }
50 | element ListItem { xsd:string }
51 | element DropStyle { xsd:string }
52 | element Colored { s_ST_TrueFalseBlank }
53 | element DropLines { xsd:integer }
54 | element Checked { xsd:integer }
55 | element FmlaLink { xsd:string }
56 | element FmlaPict { xsd:string }
57 | element NoThreeD { s_ST_TrueFalseBlank }
58 | element FirstButton { s_ST_TrueFalseBlank }
59 | element FmlaGroup { xsd:string }
60 | element Val { xsd:integer }
61 | element Min { xsd:integer }
62 | element Max { xsd:integer }
63 | element Inc { xsd:integer }
64 | element Page { xsd:integer }
65 | element Horiz { s_ST_TrueFalseBlank }
66 | element Dx { xsd:integer }
67 | element MapOCX { s_ST_TrueFalseBlank }
68 | element CF { x_ST_CF }
69 | element Camera { s_ST_TrueFalseBlank }
70 | element RecalcAlways { s_ST_TrueFalseBlank }
71 | element AutoScale { s_ST_TrueFalseBlank }
72 | element DDE { s_ST_TrueFalseBlank }
73 | element UIObj { s_ST_TrueFalseBlank }
74 | element ScriptText { xsd:string }
75 | element ScriptExtended { xsd:string }
76 | element ScriptLanguage { xsd:nonNegativeInteger }
77 | element ScriptLocation { xsd:nonNegativeInteger }
78 | element FmlaTxbx { xsd:string })*
79 x_ST_CF = xsd:string
80 x_ST_ObjectType =
81 | string "Button"
82 | string "Checkbox"
83 | string "Dialog"
84 | string "Drop"
85 | string "Edit"
86 | string "GBox"
87 | string "Label"
88 | string "LineA"
89 | string "List"
90 | string "Movie"
91 | string "Note"
92 | string "Pict"
93 | string "Radio"
94 | string "RectA"
95 | string "Scroll"
96 | string "Spin"
97 | string "Shape"
98 | string "Group"
99 | string "Rect"

```

B.6.5 VML - Presentation Drawing

This schema is available in the file vml-presentationDrawing.rnc.

```

1 default namespace = "urn:schemas-microsoft-com:office:powerpoint"
2 namespace o = "urn:schemas-microsoft-com:office:office"
3 namespace pvml = "urn:schemas-microsoft-com:office:powerpoint"
4 namespace v = "urn:schemas-microsoft-com:vml"
5 namespace w10 = "urn:schemas-microsoft-com:office:word"
6 namespace x = "urn:schemas-microsoft-com:office:excel"
7
8 pvml_iscomment = element iscomment { pvml_CT_Empty }
9 pvml_textdata = element textdata { pvml_CT_Rel }
10 pvml_CT_Empty = empty
11 pvml_CT_Rel = attribute id { xsd:string }?

```

B.6.6 Part Schemas

This schema is available in the file VML_Drawing.rnc.

```

1 include "wml.rnc"
2 include "shared-relationshipReference.rnc"
3 include "dml-wordprocessingDrawing.rnc"
4 include "dml-main.rnc"
5 include "dml-diagram.rnc"
6 include "shared-commonSimpleTypes.rnc"
7 include "dml-lockedCanvas.rnc"
8 include "any.rnc"
9 include "dml-chart.rnc"
10 include "dml-chartDrawing.rnc"
11 include "dml-picture.rnc"
12 include "vml-presentationDrawing.rnc"
13 include "xml.rnc"
14 include "shared-customXmlSchemaProperties.rnc"
15 include "vml-officeDrawing.rnc"
16 include "vml-main.rnc"
17 include "vml-spreadsheetDrawing.rnc"
18 include "vml-wordprocessingDrawing.rnc"
19 include "shared-math.rnc"
20 start = element xml {(vml-main | vml-officeDrawing | vml-spreadsheetDrawing |
21   vml-presentationDrawing)* }
22
23 vml-main =
24   v_shape
25   | v_shapetype
26   | v_group
27   | v_background
28   | v_fill
29   | v_formulas
30   | v_handles
31   | v_imagedata
32   | v_path
33   | v_textbox

```

```

34 | v_shadow
35 | v_stroke
36 | v_textpath
37 | v_arc
38 | v_curve
39 | v_image
40 | v_line
41 | v_oval
42 | v_polyline
43 | v_rect
44 | v_roundrect
45
46 vml-officeDrawing =
47   o_shapedefaults
48   | o_shapelayout
49   | o_signatureline
50   | o_ink
51   | o_diagram
52   | o_equationxml
53   | o_skew
54   | o_extrusion
55   | o_callout
56   | o_lock
57   | o_OLEObject
58   | o_complex
59   | o_left
60   | o_top
61   | o_right
62   | o_bottom
63   | o_column
64   | o_clippath
65   | o_fill
66
67 vml-wordprocessingDrawing =
68   w10_bordertop
69   | w10_borderleft
70   | w10_borderright
71   | w10_borderbottom
72   | w10_wrap
73   | w10_anchorlock
74
75 vml-spreadsheetDrawing = x_ClientData
76 vml-presentationDrawing = pvml_iscomment | pvml_textdata

```

B.7 Shared MLs

B.7.1 Math

This schema is available in the file shared-math.rnc.

```

1 default namespace m =
2   "http://schemas.openxmlformats.org/officeDocument/2006/math"
3 namespace o = "urn:schemas-microsoft-com:office:office"

```

```

4 namespace s =
5     "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace v = "urn:schemas-microsoft-com:vml"
7 namespace w =
8     "http://schemas.openxmlformats.org/wordprocessingml/2006/main"
9 namespace w10 = "urn:schemas-microsoft-com:office:word"
10 namespace x = "urn:schemas-microsoft-com:office:excel"
11
12 m_ST_Integer255 =
13     xsd:integer { minInclusive = "1" maxInclusive = "255" }
14 m_CT_Integer255 = attribute m:val { m_ST_Integer255 }
15 m_ST_Integer2 = xsd:integer { minInclusive = "-2" maxInclusive = "2" }
16 m_CT_Integer2 = attribute m:val { m_ST_Integer2 }
17 m_ST_SpacingRule = xsd:integer { minInclusive = "0" maxInclusive = "4" }
18 m_CT_SpacingRule = attribute m:val { m_ST_SpacingRule }
19 m_ST_UnSignedInteger = xsd:unsignedInt
20 m_CT_UnSignedInteger = attribute m:val { m_ST_UnSignedInteger }
21 m_ST_Char = xsd:string { maxLength = "1" }
22 m_CT_Char = attribute m:val { m_ST_Char }
23 m_CT_OnOff = attribute m:val { s_ST_OnOff }?
24 m_CT_String = attribute m:val { s_ST_String }?
25 m_CT_XAlign = attribute m:val { s_ST_XAlign }
26 m_CT_YAlign = attribute m:val { s_ST_YAlign }
27 m_ST_Shp = string "centered" | string "match"
28 m_CT_Shp = attribute m:val { m_ST_Shp }
29 m_ST_FType = string "bar" | string "skw" | string "lin" | string "noBar"
30 m_CT_FType = attribute m:val { m_ST_FType }
31 m_ST_LimLoc = string "undOvr" | string "subSup"
32 m_CT_LimLoc = attribute m:val { m_ST_LimLoc }
33 m_ST_TopBot = string "top" | string "bot"
34 m_CT_TopBot = attribute m:val { m_ST_TopBot }
35 m_ST_Script =
36     string "roman"
37     | string "script"
38     | string "fraktur"
39     | string "double-struck"
40     | string "sans-serif"
41     | string "monospace"
42 m_CT_Script = attribute m:val { m_ST_Script }?
43 m_ST_Style = string "p" | string "b" | string "i" | string "bi"
44 m_CT_Style = attribute m:val { m_ST_Style }?
45 m_CT_ManualBreak = attribute m:alnAt { m_ST_Integer255 }?
46 m_EG_ScriptStyle =
47     element scr { m_CT_Script }?,
48     element sty { m_CT_Style }?
49 m_CT_RPR =
50     element lit { m_CT_OnOff }?,
51     (element nor { m_CT_OnOff }?
52         | m_EG_ScriptStyle),
53     element brk { m_CT_ManualBreak }?,
54     element aln { m_CT_OnOff }?
55 m_CT_Text = s_ST_String, xml_space?
56 m_CT_R =

```

```

57     element rPr { m_CT_RPR }?,
58     w_EG_RPr?,
59     (w_EG_RunInnerContent
60     | element t { m_CT_Text }?)*
61 m_CT_CtrlPr = w_EG_RPrMath?
62 m_CT_AccPr =
63     element chr { m_CT_Char }?,
64     element ctrlPr { m_CT_CtrlPr }?
65 m_CT_Acc =
66     element accPr { m_CT_AccPr }?,
67     element e { m_CT_OMathArg }
68 m_CT_BarPr =
69     element pos { m_CT_TopBot }?,
70     element ctrlPr { m_CT_CtrlPr }?
71 m_CT_Bar =
72     element barPr { m_CT_BarPr }?,
73     element e { m_CT_OMathArg }
74 m_CT_BoxPr =
75     element opEmu { m_CT_OnOff }?,
76     element noBreak { m_CT_OnOff }?,
77     element diff { m_CT_OnOff }?,
78     element brk { m_CT_ManualBreak }?,
79     element aln { m_CT_OnOff }?,
80     element ctrlPr { m_CT_CtrlPr }?
81 m_CT_Box =
82     element boxPr { m_CT_BoxPr }?,
83     element e { m_CT_OMathArg }
84 m_CT_BorderBoxPr =
85     element hideTop { m_CT_OnOff }?,
86     element hideBot { m_CT_OnOff }?,
87     element hideLeft { m_CT_OnOff }?,
88     element hideRight { m_CT_OnOff }?,
89     element strikeH { m_CT_OnOff }?,
90     element strikeV { m_CT_OnOff }?,
91     element strikeBLTR { m_CT_OnOff }?,
92     element strikeTLBR { m_CT_OnOff }?,
93     element ctrlPr { m_CT_CtrlPr }?
94 m_CT_BorderBox =
95     element borderBoxPr { m_CT_BorderBoxPr }?,
96     element e { m_CT_OMathArg }
97 m_CT_DPr =
98     element begChr { m_CT_Char }?,
99     element sepChr { m_CT_Char }?,
100    element endChr { m_CT_Char }?,
101    element grow { m_CT_OnOff }?,
102    element shp { m_CT_Shp }?,
103    element ctrlPr { m_CT_CtrlPr }?
104 m_CT_D =
105     element dPr { m_CT_DPr }?,
106     element e { m_CT_OMathArg }+
107 m_CT_EqArrPr =
108     element baseJc { m_CT_YAlign }?,
109     element maxDist { m_CT_OnOff }?,

```

```

110   element objDist { m_CT_OnOff }?,
111   element rSpRule { m_CT_SpacingRule }?,
112   element rSp { m_CT_UnSignedInteger }?,
113   element ctrlPr { m_CT_CtrlPr }?
114 m_CT_EqArr =
115   element eqArrPr { m_CT_EqArrPr }?,
116   element e { m_CT_OMathArg }+
117 m_CT_FPr =
118   element type { m_CT_FType }?,
119   element ctrlPr { m_CT_CtrlPr }?
120 m_CT_F =
121   element fPr { m_CT_FPr }?,
122   element num { m_CT_OMathArg },
123   element den { m_CT_OMathArg }
124 m_CT_FuncPr = element ctrlPr { m_CT_CtrlPr }?
125 m_CT_Func =
126   element funcPr { m_CT_FuncPr }?,
127   element fName { m_CT_OMathArg },
128   element e { m_CT_OMathArg }
129 m_CT_GroupChrPr =
130   element chr { m_CT_Char }?,
131   element pos { m_CT_TopBot }?,
132   element vertJc { m_CT_TopBot }?,
133   element ctrlPr { m_CT_CtrlPr }?
134 m_CT_GroupChr =
135   element groupChrPr { m_CT_GroupChrPr }?,
136   element e { m_CT_OMathArg }
137 m_CT_LimLowPr = element ctrlPr { m_CT_CtrlPr }?
138 m_CT_LimLow =
139   element limLowPr { m_CT_LimLowPr }?,
140   element e { m_CT_OMathArg },
141   element lim { m_CT_OMathArg }
142 m_CT_LimUppPr = element ctrlPr { m_CT_CtrlPr }?
143 m_CT_LimUpp =
144   element limUppPr { m_CT_LimUppPr }?,
145   element e { m_CT_OMathArg },
146   element lim { m_CT_OMathArg }
147 m_CT_MCPr =
148   element count { m_CT_Integer255 }?,
149   element mcJc { m_CT_XAlign }?
150 m_CT_MC = element mcPr { m_CT_MCPr }?
151 m_CT_MCS = element mc { m_CT_MC }+
152 m_CT_MPr =
153   element baseJc { m_CT_YAlign }?,
154   element plcHide { m_CT_OnOff }?,
155   element rSpRule { m_CT_SpacingRule }?,
156   element cGpRule { m_CT_SpacingRule }?,
157   element rSp { m_CT_UnSignedInteger }?,
158   element cSp { m_CT_UnSignedInteger }?,
159   element cGp { m_CT_UnSignedInteger }?,
160   element mcs { m_CT_MCS }?,
161   element ctrlPr { m_CT_CtrlPr }?
162 m_CT_MR = element e { m_CT_OMathArg }+

```

```

163 m_CT_M =
164     element mPr { m_CT_MPr }?,
165     element mr { m_CT_MR }+
166 m_CT_NaryPr =
167     element chr { m_CT_Char }?,
168     element limLoc { m_CT_LimLoc }?,
169     element grow { m_CT_OnOff }?,
170     element subHide { m_CT_OnOff }?,
171     element supHide { m_CT_OnOff }?,
172     element ctrlPr { m_CT_CtrlPr }?
173 m_CT_Nary =
174     element naryPr { m_CT_NaryPr }?,
175     element sub { m_CT_OMathArg },
176     element sup { m_CT_OMathArg },
177     element e { m_CT_OMathArg }
178 m_CT_PhantPr =
179     element show { m_CT_OnOff }?,
180     element zeroWid { m_CT_OnOff }?,
181     element zeroAsc { m_CT_OnOff }?,
182     element zeroDesc { m_CT_OnOff }?,
183     element transp { m_CT_OnOff }?,
184     element ctrlPr { m_CT_CtrlPr }?
185 m_CT_Phant =
186     element phantPr { m_CT_PhantPr }?,
187     element e { m_CT_OMathArg }
188 m_CT_RadPr =
189     element degHide { m_CT_OnOff }?,
190     element ctrlPr { m_CT_CtrlPr }?
191 m_CT_Rad =
192     element radPr { m_CT_RadPr }?,
193     element deg { m_CT_OMathArg },
194     element e { m_CT_OMathArg }
195 m_CT_SPrePr = element ctrlPr { m_CT_CtrlPr }?
196 m_CT_SPre =
197     element sPrePr { m_CT_SPrePr }?,
198     element sub { m_CT_OMathArg },
199     element sup { m_CT_OMathArg },
200     element e { m_CT_OMathArg }
201 m_CT_SSubPr = element ctrlPr { m_CT_CtrlPr }?
202 m_CT_SSub =
203     element sSubPr { m_CT_SSubPr }?,
204     element e { m_CT_OMathArg },
205     element sub { m_CT_OMathArg }
206 m_CT_SSubSupPr =
207     element alnScr { m_CT_OnOff }?,
208     element ctrlPr { m_CT_CtrlPr }?
209 m_CT_SSubSup =
210     element sSubSupPr { m_CT_SSubSupPr }?,
211     element e { m_CT_OMathArg },
212     element sub { m_CT_OMathArg },
213     element sup { m_CT_OMathArg }
214 m_CT_SSupPr = element ctrlPr { m_CT_CtrlPr }?
215 m_CT_SSup =

```



```

216 element sSupPr { m_CT_SSupPr }?,
217 element e { m_CT_OMathArg },
218 element sup { m_CT_OMathArg }
219 m_EG_OMathMathElements =
220 element acc { m_CT_Acc }
221 | element bar { m_CT_Bar }
222 | element box { m_CT_Box }
223 | element borderBox { m_CT_BorderBox }
224 | element d { m_CT_D }
225 | element eqArr { m_CT_EqArr }
226 | element f { m_CT_F }
227 | element func { m_CT_Func }
228 | element groupChr { m_CT_GroupChr }
229 | element limLow { m_CT_LimLow }
230 | element limUpp { m_CT_LimUpp }
231 | element m { m_CT_M }
232 | element nary { m_CT_Nary }
233 | element phant { m_CT_Phant }
234 | element rad { m_CT_Rad }
235 | element sPre { m_CT_SPre }
236 | element sSub { m_CT_SSub }
237 | element sSubSup { m_CT_SSubSup }
238 | element sSup { m_CT_SSup }
239 | element r { m_CT_R }
240 m_EG_OMathElements = m_EG_OMathMathElements | w_EG_PContentMath
241 m_CT_OMathArgPr = element argSz { m_CT_Integer2 }?
242 m_CT_OMathArg =
243 element argPr { m_CT_OMathArgPr }?,
244 m_EG_OMathElements*,
245 element ctrlPr { m_CT_CtrlPr }?
246 m_ST_Jc =
247 string "left"
248 | string "right"
249 | string "center"
250 | string "centerGroup"
251 m_CT_OMathJc = attribute m:val { m_ST_Jc }?
252 m_CT_OMathParaPr = element jc { m_CT_OMathJc }?
253 m_CT_TwipsMeasure = attribute m:val { s_ST_TwipsMeasure }
254 m_ST_BreakBin = string "before" | string "after" | string "repeat"
255 m_CT_BreakBin = attribute m:val { m_ST_BreakBin }?
256 m_ST_BreakBinSub = string "--" | string "-+" | string "+-"
257 m_CT_BreakBinSub = attribute m:val { m_ST_BreakBinSub }?
258 m_CT_MathPr =
259 element mathFont { m_CT_String }?,
260 element brkBin { m_CT_BreakBin }?,
261 element brkBinSub { m_CT_BreakBinSub }?,
262 element smallFrac { m_CT_OnOff }?,
263 element dispDef { m_CT_OnOff }?,
264 element lMargin { m_CT_TwipsMeasure }?,
265 element rMargin { m_CT_TwipsMeasure }?,
266 element defJc { m_CT_OMathJc }?,
267 element preSp { m_CT_TwipsMeasure }?,
268 element postSp { m_CT_TwipsMeasure }?,

```

```

269   element interSp { m_CT_TwipsMeasure }?,
270   element intraSp { m_CT_TwipsMeasure }?,
271   (element wrapIndent { m_CT_TwipsMeasure }
272     | element wrapRight { m_CT_OnOff })?,
273   element intLim { m_CT_LimLoc }?,
274   element naryLim { m_CT_LimLoc }?
275 m_mathPr = element mathPr { m_CT_MathPr }
276 m_CT_OMathPara =
277   element oMathParaPr { m_CT_OMathParaPr }?,
278   element oMath { m_CT_OMath }+
279 m_CT_OMath = m_EG_OMathElements*
280 m_oMathPara = element oMathPara { m_CT_OMathPara }
281 m_oMath = element oMath { m_CT_OMath }

```

B.7.2 Extended Properties

This schema is available in the file shared-documentPropertiesExtended.rnc.

```

1  default namespace =
2    "http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
3  namespace o = "urn:schemas-microsoft-com:office:office"
4  namespace shdDcEP =
5    "http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
6  namespace v = "urn:schemas-microsoft-com:vml"
7  namespace vt =
8    "http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
9  namespace w10 = "urn:schemas-microsoft-com:office:word"
10 namespace x = "urn:schemas-microsoft-com:office:excel"
11
12 shdDcEP_Properties = element Properties { shdDcEP_CT_Properties }
13 shdDcEP_CT_Properties =
14   element Template { xsd:string }?
15   & element Manager { xsd:string }?
16   & element Company { xsd:string }?
17   & element Pages { xsd:int }?
18   & element Words { xsd:int }?
19   & element Characters { xsd:int }?
20   & element PresentationFormat { xsd:string }?
21   & element Lines { xsd:int }?
22   & element Paragraphs { xsd:int }?
23   & element Slides { xsd:int }?
24   & element Notes { xsd:int }?
25   & element TotalTime { xsd:int }?
26   & element HiddenSlides { xsd:int }?
27   & element MMClips { xsd:int }?
28   & element ScaleCrop { xsd:boolean }?
29   & element HeadingPairs { shdDcEP_CT_VectorVariant }?
30   & element TitlesOfParts { shdDcEP_CT_VectorLpstr }?
31   & element LinksUpToDate { xsd:boolean }?
32   & element CharactersWithSpaces { xsd:int }?
33   & element SharedDoc { xsd:boolean }?
34   & element HyperlinkBase { xsd:string }?
35   & element HLinks { shdDcEP_CT_VectorVariant }?

```

```

36 & element HyperlinksChanged { xsd:boolean }?
37 & element DigSig { shdDcEP_CT_DigSigBlob }?
38 & element Application { xsd:string }?
39 & element AppVersion { xsd:string }?
40 & element DocSecurity { xsd:int }?
41 shdDcEP_CT_VectorVariant = vt_vector
42 shdDcEP_CT_VectorLpstr = vt_vector
43 shdDcEP_CT_DigSigBlob = vt_blob

```

B.7.2.1 Part Schemas

B.7.2.1.1 Extended File Properties Part

This schema is available in the file Shared_Extended_File_Properties.rnc.

```

1 include "shared-documentPropertiesExtended.rnc"
2 include "shared-documentPropertiesVariantTypes.rnc"
3 include "shared-commonSimpleTypes.rnc"
4 start = shdDcEP_Properties

```

B.7.3 Custom Properties

This schema is available in the file shared-documentPropertiesCustom.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/officeDocument/2006/custom-properties"
3 namespace o = "urn:schemas-microsoft-com:office:office"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace shdCstm =
7   "http://schemas.openxmlformats.org/officeDocument/2006/custom-properties"
8 namespace v = "urn:schemas-microsoft-com:xml"
9 namespace vt =
10  "http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
11 namespace w10 = "urn:schemas-microsoft-com:office:word"
12 namespace x = "urn:schemas-microsoft-com:office:excel"
13
14 shdCstm_Properties = element Properties { shdCstm_CT_Properties }
15 shdCstm_CT_Properties = element property { shdCstm_CT_Property }*
16 shdCstm_CT_Property =
17   attribute fmtid { s_ST_Guid },
18   attribute pid { xsd:int },
19   attribute name { xsd:string }?,
20   attribute linkTarget { xsd:string }?,
21   (vt_vector
22     | vt_array
23     | vt_blob
24     | vt_oblob
25     | vt_empty
26     | vt_null
27     | vt_i1
28     | vt_i2
29     | vt_i4
30     | vt_i8

```

```

31 | vt_int
32 | vt_ui1
33 | vt_ui2
34 | vt_ui4
35 | vt_ui8
36 | vt_uint
37 | vt_r4
38 | vt_r8
39 | vt_decimal
40 | vt_lpstr
41 | vt_lpwstr
42 | vt_bstr
43 | vt_date
44 | vt_filetime
45 | vt_bool
46 | vt_cy
47 | vt_error
48 | vt_stream
49 | vt_ostream
50 | vt_storage
51 | vt_ostorage
52 | vt_vstream
53 | vt_clsid)

```

B.7.3.1 Part Schemas

B.7.3.1.1 Custom File Properties Part

This schema is available in the file Shared_Custom_File_Properties.rnc.

```

1 include "shared-documentPropertiesCustom.rnc"
2 include "shared-documentPropertiesVariantTypes.rnc"
3 include "shared-commonSimpleTypes.rnc"
4 start = shdCstm_Properties

```

B.7.4 Variant Types

This schema is available in the file shared-documentPropertiesVariantTypes.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
3 namespace o = "urn:schemas-microsoft-com:office:office"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace v = "urn:schemas-microsoft-com:xml"
7 namespace vt =
8   "http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes"
9 namespace w10 = "urn:schemas-microsoft-com:office:word"
10 namespace x = "urn:schemas-microsoft-com:office:excel"
11
12 vt_ST_VectorBaseType =
13   string "variant"
14   | string "i1"
15   | string "i2"

```

```

16 | string "i4"
17 | string "i8"
18 | string "ui1"
19 | string "ui2"
20 | string "ui4"
21 | string "ui8"
22 | string "r4"
23 | string "r8"
24 | string "lpstr"
25 | string "lpwstr"
26 | string "bstr"
27 | string "date"
28 | string "filetime"
29 | string "bool"
30 | string "cy"
31 | string "error"
32 | string "clsid"
33 vt_ST_ArrayBaseType =
34   string "variant"
35   | string "i1"
36   | string "i2"
37   | string "i4"
38   | string "int"
39   | string "ui1"
40   | string "ui2"
41   | string "ui4"
42   | string "uint"
43   | string "r4"
44   | string "r8"
45   | string "decimal"
46   | string "bstr"
47   | string "date"
48   | string "bool"
49   | string "cy"
50   | string "error"
51 vt_ST_Cy = xsd:string { pattern = "\s*[0-9]*\.[0-9]{4}\s*" }
52 vt_ST_Error = xsd:string { pattern = "\s*0x[0-9A-Za-z]{8}\s*" }
53 vt_CT_Empty = empty
54 vt_CT_Null = empty
55 vt_CT_Vector =
56   attribute baseType { vt_ST_VectorBaseType },
57   attribute size { xsd:unsignedInt },
58   (vt_variant
59     | vt_i1
60     | vt_i2
61     | vt_i4
62     | vt_i8
63     | vt_ui1
64     | vt_ui2
65     | vt_ui4
66     | vt_ui8
67     | vt_r4
68     | vt_r8

```

```

69 | vt_lpstr
70 | vt_lpwstr
71 | vt_bstr
72 | vt_date
73 | vt_filetime
74 | vt_bool
75 | vt_cy
76 | vt_error
77 | vt_clsid)+
78 vt_CT_Array =
79   attribute lBounds { xsd:int },
80   attribute uBounds { xsd:int },
81   attribute baseType { vt_ST_ArrayBaseType },
82   (vt_variant
83     | vt_i1
84     | vt_i2
85     | vt_i4
86     | vt_int
87     | vt_ui1
88     | vt_ui2
89     | vt_ui4
90     | vt_uint
91     | vt_r4
92     | vt_r8
93     | vt_decimal
94     | vt_bstr
95     | vt_date
96     | vt_bool
97     | vt_error
98     | vt_cy)+
99 vt_CT_Variant =
100 vt_variant
101 | vt_vector
102 | vt_array
103 | vt_blob
104 | vt_oblob
105 | vt_empty
106 | vt_null
107 | vt_i1
108 | vt_i2
109 | vt_i4
110 | vt_i8
111 | vt_int
112 | vt_ui1
113 | vt_ui2
114 | vt_ui4
115 | vt_ui8
116 | vt_uint
117 | vt_r4
118 | vt_r8
119 | vt_decimal
120 | vt_lpstr
121 | vt_lpwstr

```

```

122 | vt_bstr
123 | vt_date
124 | vt_filetime
125 | vt_bool
126 | vt_cy
127 | vt_error
128 | vt_stream
129 | vt_ostream
130 | vt_storage
131 | vt_ostorage
132 | vt_vstream
133 | vt_clsid
134 vt_CT_Vstream =
135     xsd:base64Binary,
136     attribute version { s_ST_Guid }?
137 vt_variant = element variant { vt_CT_Variant }
138 vt_vector = element vector { vt_CT_Vector }
139 vt_array = element array { vt_CT_Array }
140 vt_blob = element blob { xsd:base64Binary }
141 vt_oblob = element oblob { xsd:base64Binary }
142 vt_empty = element empty { vt_CT_Empty }
143 vt_null = element null { vt_CT_Null }
144 vt_i1 = element i1 { xsd:byte }
145 vt_i2 = element i2 { xsd:short }
146 vt_i4 = element i4 { xsd:int }
147 vt_i8 = element i8 { xsd:long }
148 vt_int = element int { xsd:int }
149 vt_ui1 = element ui1 { xsd:unsignedByte }
150 vt_ui2 = element ui2 { xsd:unsignedShort }
151 vt_ui4 = element ui4 { xsd:unsignedInt }
152 vt_ui8 = element ui8 { xsd:unsignedLong }
153 vt_uint = element uint { xsd:unsignedInt }
154 vt_r4 = element r4 { xsd:float }
155 vt_r8 = element r8 { xsd:double }
156 vt_decimal = element decimal { xsd:decimal }
157 vt_lpstr = element lpstr { xsd:string }
158 vt_lpwstr = element lpwstr { xsd:string }
159 vt_bstr = element bstr { xsd:string }
160 vt_date = element date { xsd:dateTime }
161 vt_filetime = element filetime { xsd:dateTime }
162 vt_bool = element bool { xsd:boolean }
163 vt_cy = element cy { vt_ST_Cy }
164 vt_error = element error { vt_ST_Error }
165 vt_stream = element stream { xsd:base64Binary }
166 vt_ostream = element ostream { xsd:base64Binary }
167 vt_storage = element storage { xsd:base64Binary }
168 vt_ostorage = element ostorage { xsd:base64Binary }
169 vt_vstream = element vstream { vt_CT_Vstream }
170 vt_clsid = element clsid { s_ST_Guid }

```

B.7.5 Custom XML Data Properties

This schema is available in the file `shared-customXmlDataProperties.rnc`.

```

1 default namespace ds =
2   "http://schemas.openxmlformats.org/officeDocument/2006/customXml"
3 namespace o = "urn:schemas-microsoft-com:office:office"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace v = "urn:schemas-microsoft-com:xml"
7 namespace w10 = "urn:schemas-microsoft-com:office:word"
8 namespace x = "urn:schemas-microsoft-com:office:excel"
9
10 ds_CT_DatastoreSchemaRef = attribute ds:uri { xsd:string }
11 ds_CT_DatastoreSchemaRefs =
12   element schemaRef { ds_CT_DatastoreSchemaRef }*
13 ds_CT_DatastoreItem =
14   attribute ds:itemID { s_ST_Guid },
15   element schemaRefs { ds_CT_DatastoreSchemaRefs }?
16 ds_datastoreItem = element datastoreItem { ds_CT_DatastoreItem }

```

B.7.5.1 Part Schemas

B.7.5.1.1 Custom XML Data Properties Part

This schema is available in the file Shared_Custom_XML_Data_Storage_Properties.rnc.

```

1 include "shared-customXmlDataProperties.rnc"
2 include "shared-commonSimpleTypes.rnc"
3 start = ds_datastoreItem

```

B.7.6 Bibliography

This schema is available in the file shared-bibliography.rnc.

```

1 default namespace =
2   "http://schemas.openxmlformats.org/officeDocument/2006/bibliography"
3 namespace o = "urn:schemas-microsoft-com:office:office"
4 namespace s =
5   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
6 namespace shrdBib =
7   "http://schemas.openxmlformats.org/officeDocument/2006/bibliography"
8 namespace v = "urn:schemas-microsoft-com:xml"
9 namespace w10 = "urn:schemas-microsoft-com:office:word"
10 namespace x = "urn:schemas-microsoft-com:office:excel"
11
12 shrdBib_ST_SourceType =
13   "ArticleInAPeriodical"
14   | "Book"
15   | "BookSection"
16   | "JournalArticle"
17   | "ConferenceProceedings"
18   | "Report"
19   | "SoundRecording"
20   | "Performance"
21   | "Art"
22   | "DocumentFromInternetSite"
23   | "InternetSite"

```



```

24 | "Film"
25 | "Interview"
26 | "Patent"
27 | "ElectronicSource"
28 | "Case"
29 | "Misc"
30 shrdBib_CT_NameListType = element Person { shrdBib_CT_PersonType }+
31 shrdBib_CT_PersonType =
32   element Last { s_ST_String }*,
33   element First { s_ST_String }*,
34   element Middle { s_ST_String }*
35 shrdBib_CT_NameType = element NameList { shrdBib_CT_NameListType }
36 shrdBib_CT_NameOrCorporateType =
37   (element NameList { shrdBib_CT_NameListType }
38    | element Corporate { s_ST_String })?
39 shrdBib_CT_AuthorType =
40   (element Artist { shrdBib_CT_NameType }
41    | element Author { shrdBib_CT_NameOrCorporateType }
42    | element BookAuthor { shrdBib_CT_NameType }
43    | element Compiler { shrdBib_CT_NameType }
44    | element Composer { shrdBib_CT_NameType }
45    | element Conductor { shrdBib_CT_NameType }
46    | element Counsel { shrdBib_CT_NameType }
47    | element Director { shrdBib_CT_NameType }
48    | element Editor { shrdBib_CT_NameType }
49    | element Interviewee { shrdBib_CT_NameType }
50    | element Interviewer { shrdBib_CT_NameType }
51    | element Inventor { shrdBib_CT_NameType }
52    | element Performer { shrdBib_CT_NameOrCorporateType }
53    | element ProducerName { shrdBib_CT_NameType }
54    | element Translator { shrdBib_CT_NameType }
55    | element Writer { shrdBib_CT_NameType })*
56 shrdBib_CT_SourceType =
57   (element AbbreviatedCaseNumber { s_ST_String }
58    | element AlbumTitle { s_ST_String }
59    | element Author { shrdBib_CT_AuthorType }
60    | element BookTitle { s_ST_String }
61    | element Broadcaster { s_ST_String }
62    | element BroadcastTitle { s_ST_String }
63    | element CaseNumber { s_ST_String }
64    | element ChapterNumber { s_ST_String }
65    | element City { s_ST_String }
66    | element Comments { s_ST_String }
67    | element ConferenceName { s_ST_String }
68    | element CountryRegion { s_ST_String }
69    | element Court { s_ST_String }
70    | element Day { s_ST_String }
71    | element DayAccessed { s_ST_String }
72    | element Department { s_ST_String }
73    | element Distributor { s_ST_String }
74    | element Edition { s_ST_String }
75    | element Guid { s_ST_String }
76    | element Institution { s_ST_String }

```

```

77 | element InternetSiteTitle { s_ST_String }
78 | element Issue { s_ST_String }
79 | element JournalName { s_ST_String }
80 | element LCID { s_ST_Lang }
81 | element Medium { s_ST_String }
82 | element Month { s_ST_String }
83 | element MonthAccessed { s_ST_String }
84 | element NumberVolumes { s_ST_String }
85 | element Pages { s_ST_String }
86 | element PatentNumber { s_ST_String }
87 | element PeriodicalTitle { s_ST_String }
88 | element ProductionCompany { s_ST_String }
89 | element PublicationTitle { s_ST_String }
90 | element Publisher { s_ST_String }
91 | element RecordingNumber { s_ST_String }
92 | element RefOrder { s_ST_String }
93 | element Reporter { s_ST_String }
94 | element SourceType { shrdBib_ST_SourceType }
95 | element ShortTitle { s_ST_String }
96 | element StandardNumber { s_ST_String }
97 | element StateProvince { s_ST_String }
98 | element Station { s_ST_String }
99 | element Tag { s_ST_String }
100 | element Theater { s_ST_String }
101 | element ThesisType { s_ST_String }
102 | element Title { s_ST_String }
103 | element Type { s_ST_String }
104 | element URL { s_ST_String }
105 | element Version { s_ST_String }
106 | element Volume { s_ST_String }
107 | element Year { s_ST_String }
108 | element YearAccessed { s_ST_String })*
109 shrdBib_Sources = element Sources { shrdBib_CT_Sources }
110 shrdBib_CT_Sources =
111   attribute SelectedStyle { s_ST_String }?,
112   attribute StyleName { s_ST_String }?,
113   attribute URI { s_ST_String }?,
114   element Source { shrdBib_CT_SourceType }*

```

B.7.6.1 Part Schemas

B.7.6.1.1 Bibliography Part

This schema is available in the file Shared_Bibliography.rnc.

```

1 include "shared-bibliography.rnc"
2 include "shared-commonSimpleTypes.rnc"
3 start = shrdBib_Sources

```

B.7.7 Additional Characteristics

This schema is available in the file shared-additionalCharacteristics.rnc.

```

1 default namespace =

```

```

2  "http://schemas.openxmlformats.org/officeDocument/2006/characteristics"
3  namespace o = "urn:schemas-microsoft-com:office:office"
4  namespace shrdChr =
5    "http://schemas.openxmlformats.org/officeDocument/2006/characteristics"
6  namespace v = "urn:schemas-microsoft-com:vm1"
7  namespace w10 = "urn:schemas-microsoft-com:office:word"
8  namespace x = "urn:schemas-microsoft-com:office:excel"
9
10 shrdChr_CT_AdditionalCharacteristics =
11   element characteristic { shrdChr_CT_Characteristic }*
12 shrdChr_CT_Characteristic =
13   attribute name { xsd:string },
14   attribute relation { shrdChr_ST_Relation },
15   attribute val { xsd:string },
16   attribute vocabulary { xsd:anyURI }?
17 shrdChr_ST_Relation =
18   string "ge" | string "le" | string "gt" | string "lt" | string "eq"
19 shrdChr_additionalCharacteristics =
20   element additionalCharacteristics {
21     shrdChr_CT_AdditionalCharacteristics
22   }

```

B.7.7.1 Part Schemas

B.7.7.1.1 Additional Characteristics Part

This schema is available in the file Shared_Additional_Characteristics.rnc.

```

1  include "shared-additionalCharacteristics.rnc"
2  start = shrdChr_additionalCharacteristics

```

B.7.8 Office Document Relationships

This schema is available in the file shared-relationshipReference.rnc.

```

1  namespace o = "urn:schemas-microsoft-com:office:office"
2  namespace r =
3    "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
4  namespace v = "urn:schemas-microsoft-com:vm1"
5  namespace w10 = "urn:schemas-microsoft-com:office:word"
6  namespace x = "urn:schemas-microsoft-com:office:excel"
7
8  r_ST_RelationshipId = xsd:string
9  r_id = attribute r:id { r_ST_RelationshipId }
10 r_embed = attribute r:embed { r_ST_RelationshipId }
11 r_link = attribute r:link { r_ST_RelationshipId }
12 r_dm = attribute r:dm { r_ST_RelationshipId }
13 r_lo = attribute r:lo { r_ST_RelationshipId }
14 r_qs = attribute r:qs { r_ST_RelationshipId }
15 r_cs = attribute r:cs { r_ST_RelationshipId }
16 r_blip = attribute r:blip { r_ST_RelationshipId }
17 r_pict = attribute r:pict { r_ST_RelationshipId }
18 r_href = attribute r:href { r_ST_RelationshipId }
19 r_topLeft = attribute r:topLeft { r_ST_RelationshipId }

```

```

20 r_topRight = attribute r:topRight { r_ST_RelationshipId }
21 r_bottomLeft = attribute r:bottomLeft { r_ST_RelationshipId }
22 r_bottomRight = attribute r:bottomRight { r_ST_RelationshipId }

```

B.7.9 Shared Simple Types

This schema is available in the file shared-commonSimpleTypes.rnc.

```

1 namespace o = "urn:schemas-microsoft-com:office:office"
2 namespace s =
3   "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
4 namespace v = "urn:schemas-microsoft-com:vm1"
5 namespace w10 = "urn:schemas-microsoft-com:office:word"
6 namespace x = "urn:schemas-microsoft-com:office:excel"
7
8 s_ST_Lang = xsd:string
9 s_ST_HexColorRGB = xsd:hexBinary { length = "3" }
10 s_ST_Panose = xsd:hexBinary { length = "10" }
11 s_ST_CalendarType =
12   string "gregorian"
13   | string "gregorianUs"
14   | string "gregorianMeFrench"
15   | string "gregorianArabic"
16   | string "hijri"
17   | string "hebrew"
18   | string "taiwan"
19   | string "japan"
20   | string "thai"
21   | string "korea"
22   | string "saka"
23   | string "gregorianXlitEnglish"
24   | string "gregorianXlitFrench"
25   | string "none"
26 s_ST_AlzClass = string "hash" | string "custom"
27 s_ST_CryptProv = string "rsaAES" | string "rsaFull" | string "custom"
28 s_ST_AlzType = string "typeAny" | string "custom"
29 s_ST_ColorType = xsd:string
30 s_ST_Guid =
31   xsd:token {
32     pattern =
33       "\{[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}\}"
34   }
35 s_ST_OnOff = xsd:boolean | s_ST_OnOff1
36 s_ST_OnOff1 = string "on" | string "off"
37 s_ST_String = xsd:string
38 s_ST_XmlName = xsd:NCName { minLength = "1" maxLength = "255" }
39 s_ST_TrueFalse =
40   string "t" | string "f" | string "true" | string "false"
41 s_ST_TrueFalseBlank =
42   string "t"
43   | string "f"
44   | string "true"
45   | string "false"

```

```

46 | string ""
47 | string "True"
48 | string "False"
49 s_ST_UnsignedDecimalNumber = xsd:unsignedLong
50 s_ST_TwipsMeasure =
51   s_ST_UnsignedDecimalNumber | s_ST_PositiveUniversalMeasure
52 s_ST_VerticalAlignRun =
53   string "baseline" | string "superscript" | string "subscript"
54 s_ST_Xstring = xsd:string
55 s_ST_XAlign =
56   string "left"
57   | string "center"
58   | string "right"
59   | string "inside"
60   | string "outside"
61 s_ST_YAlign =
62   string "inline"
63   | string "top"
64   | string "center"
65   | string "bottom"
66   | string "inside"
67   | string "outside"
68 s_ST_ConformanceClass = string "strict" | string "transitional"
69 s_ST_UniversalMeasure =
70   xsd:string { pattern = "-?[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)" }
71 s_ST_PositiveUniversalMeasure =
72   xsd:string {
73     pattern = "-?[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)"
74     pattern = "[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)"
75   }
76 s_ST_Percentage = xsd:string { pattern = "-?[0-9]+(\\.[0-9]+)?%" }
77 s_ST_FixedPercentage =
78   xsd:string {
79     pattern = "-?[0-9]+(\\.[0-9]+)?%"
80     pattern = "-?((100)|([0-9][0-9]?))(\\.[0-9][0-9]?)?%"
81   }
82 s_ST_PositivePercentage =
83   xsd:string {
84     pattern = "-?[0-9]+(\\.[0-9]+)?%"
85     pattern = "[0-9]+(\\.[0-9]+)?%"
86   }
87 s_ST_PositiveFixedPercentage =
88   xsd:string {
89     pattern = "-?[0-9]+(\\.[0-9]+)?%"
90     pattern = "((100)|([0-9][0-9]?))(\\.[0-9][0-9]?)?%"
91   }

```

B.8 Custom XML Schema References

This schema is available in the file `shared-customXmlSchemaProperties.rnc`.

```

1 namespace o = "urn:schemas-microsoft-com:office:office"
2 default namespace sl =

```

```

3  "http://schemas.openxmlformats.org/schemaLibrary/2006/main"
4  namespace v = "urn:schemas-microsoft-com:xml"
5  namespace w10 = "urn:schemas-microsoft-com:office:word"
6  namespace x = "urn:schemas-microsoft-com:office:excel"
7
8  sl_CT_Schema =
9    attribute sl:uri { xsd:string }?,
10   attribute sl:manifestLocation { xsd:string }?,
11   attribute sl:schemaLocation { xsd:string }?,
12   attribute sl:schemaLanguage { xsd:token }?
13  sl_CT_SchemaLibrary = element schema { sl_CT_Schema }*
14  sl_schemaLibrary = element schemaLibrary { sl_CT_SchemaLibrary }

```

B.9 Additional Resources

B.9.1 Any

This schema is available in the file any.rnc.

```

1  anyElement = element * { anyAttribute*, text?, anyElement* }
2  anyAttribute = attribute * { text }

```

B.9.2 XML

This schema is available in the file xml.rnc.

```

1  xml_lang = attribute xml:lang { xsd:language | xsd:string "" }
2  xml_space = attribute xml:space { "default" | "preserve" }
3  xml_base = attribute xml:base { xsd:anyURI }
4  xml_id = attribute xml:id { xsd:ID }
5  xml_specialAttrs = xml_base?, xml_lang?, xml_space?, xml_id?

```

End of informative text.

Annex C.

(informative)

Namespace Prefix Mapping in Examples

This Annex is informative.

Throughout ECMA-376, XML syntax is provided to illustrate the concepts being documented. These examples leverage XML namespace prefixes, and, typically, for brevity, do not show the actual namespace mappings. This Annex lists the namespace prefix mappings that are used within these examples.

Prefix	Namespace
a	http://schemas.openxmlformats.org/drawingml/2006/main
b	http://schemas.openxmlformats.org/officeDocument/2006/bibliography
cp	http://schemas.openxmlformats.org/package/2006/metadata/core-properties
cdr	http://schemas.openxmlformats.org/drawingml/2006/chartDrawing
dc	http://purl.org/dc/elements/1.1/
dcmitype	http://purl.org/dc/dcmitype/
dcterms	http://purl.org/dc/terms/
ds	http://schemas.openxmlformats.org/officeDocument/2006/customXml
m	http://schemas.openxmlformats.org/officeDocument/2006/math
o	urn:schemas-microsoft-com:office:office
p	http://schemas.openxmlformats.org/presentationml/2006/main
pic	http://schemas.openxmlformats.org/drawingml/2006/picture
pvm1	urn:schemas-microsoft-com:office:powerpoint
r	http://schemas.openxmlformats.org/officeDocument/2006/relationships
sl	http://schemas.openxmlformats.org/schemaLibrary/2006/main
v	urn:schemas-microsoft-com:vml
ve	http://schemas.openxmlformats.org/markup-compatibility/2006
vt	http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes
w	http://schemas.openxmlformats.org/wordprocessingml/2006/main
w10	urn:schemas-microsoft-com:office:word

Prefix	Namespace
wp	http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing
x	urn:schemas-microsoft-com:office:excel
xdr	http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance

If no namespace prefix is specified, it should be assumed that that element or attribute is contained within the namespace defined by the parent subclause. For example, unprefix elements in Part 1, §18 are contained in the <http://schemas.openxmlformats.org/spreadsheetml/2006/main> namespace.

End informative Annex.

Annex D. (informative)

Differences Between ECMA-376:2012 and ECMA-376:2006

This annex is informative.

This annex highlights the differences between the versions of the Transitional form of the Office Open XML schemas, as defined in ECMA-376:2012 and the schemas as defined by ECMA-376:2006.

D.1 WordprocessingML

The following changes occurred to the WordprocessingML schema:

- The algorithmName, hashValue, saltValue, and spinCount attributes were added to documentProtection (Part 1, §17.15.1.29) and writeProtection (Part 1, §17.15.1.93)
- The allStyles, customStyles, latentStyles, stylesInUse, headingStyles, numberingStyles, tableStyles, directFormattingOnRuns, directFormattingOnParagraphs, directFormattingOnNumbering, directFormattingOnTables, clearFormatting, top3HeadingStyles, and visibleStyles attributes were added to the stylePaneFormatFilter element (Part 1, §17.15.1.85)
- The bdo element (Part 1, §17.3.2.3) was added
- The characterSet attribute was added to the charset element (Part 1, §17.8.3.2)
- The compatSetting element (Part 1, §17.15.3.4) was added
- The conformance attribute was added to document (Part 1, §17.2.3)
- The content model of ST_HpsMeasure (Part 1, §17.18.42) was modified to allow ST_PositiveUniversalMeasure (Part 1, §22.9.2.12)
- The content model of ST_OnOff (Part 1, §22.9.2.7) was changed to an xsd:boolean, removing the values on and off
- The content model of ST_SignedHpsMeasure (Part 1, §17.18.80) was modified to allow ST_UniversalMeasure (Part 1, §22.9.2.15)
- The content model of ST_SignedTwipsMeasure (Part 1, §17.18.81) was modified to allow ST_UniversalMeasure (Part 1, §22.9.2.15)
- The contentPart element (Part 1, §17.3.3.2) was added
- The dir element (Part 1, §17.3.2.8) was added
- The end element (Part 1, §17.4.10) was added
- The end element (Part 1, §17.4.11) was added
- The end element (Part 1, §17.4.12) was added

- The end element (Part 1, §17.4.13) was added
- The firstRow, lastRow, firstColumn, lastColumn, noHBand, and noVBand attributes were added to the tblLook element (Part 1, §17.4.55; Part 1, §17.4.56)
- The firstRow, lastRow, firstColumn, lastColumn, oddVBand, evenVBand, oddHBand, evenHBand, firstRowFirstColumn, firstRowLastColumn, lastRowFirstColumn, and lastRowLastColumn attributes were added to the cnfStyle element (Part 1, §17.3.1.8; Part 1, §17.4.8; Part 1, §17.4.7)
- The following enumeration values were added to the ST_Border simple type (Part 1, §17.18.2): earth3, triangle1, triangle2, triangleCircle1, triangleCircle2, shapes1, shapes2, custom
- The following enumeration values were added to the ST_CalendarType simple type (Part 1, §22.9.2.1): gregorianArabic, gregorianMeFrench, and gregorianUs.
- The following enumeration values were added to the ST_Jc simple type (Part 1, §17.18.44): start, end
- The following enumeration values were added to the ST_NumberFormat simple type (Part 1, §17.18.59): bahtText, dollarText, custom
- The following enumeration values were added to the ST_TabJc simple type (Part 1, §17.18.84): start, end
- The following enumeration values were added to the ST_TextDirection simple type (Part 1, §17.18.93): tb, r1, lr, tbV, r1V, and lrV.
- The following enumeration values were removed from the ST_Border simple type (Part 1, §17.18.2): tribal1, tribal2, tribal3, tribal4, tribal5, tribal6
- The fontSz attribute on the readModeInkLockDown element (Part 1, §17.15.1.66) was modified to use ST_DecimalNumberOrPercent (Part 1, §17.18.11)
- The format attribute was added to the numFmt element (Part 1, §17.9.18)
- The header element (Part 1, §17.4.18) was added
- The headers element (Part 1, §17.4.19) was added
- The id attribute was added to the left element (Part 1, §17.6.7) and right element (Part 1, §17.6.15)
- The id attribute was added to the tc element (Part 1, §17.4.66)
- The id, bottomLeft, and bottomRight attributes were added to the bottom element (Part 1, §17.6.2)
- The id, topLeft, and topRight attributes were added to the top element (Part 1, §17.6.21)
- The jc element (Part 1, §17.4.29) was modified to use the ST_JcTable simple type (Part 1, §17.18.45)
- The label element (Part 1, §17.5.2.19) was added
- The longDesc element (Part 1, §17.15.2.23) was added
- The objectEmbed element (Part 1, §17.3.3.20) was added
- The objectLink element (Part 1, §17.3.3.21) was added
- The percent attribute on the zoom element (Part 1, §17.15.1.94) was modified to use ST_DecimalNumberOrPercent (Part 1, §17.18.11)
- The ST_ColorSchemeIndex simple type was renamed to ST_WmlColorSchemeIndex (Part 1, §17.18.103)
- The ST_DecimalNumberOrPercent (Part 1, §17.18.11) simple type was added
- The ST_Direction simple type (Part 1, §17.18.12) was added
- The ST_DocType simple type (Part 1, §17.18.19) was modified to allow any xsd:string
- The ST_JcTable simple type (Part 1, §17.18.45) was added

- The ST_LangCode simple type was removed
- The ST_MailMergeDataType simple type (Part 1, §17.18.54) was modified to allow any xsd:string
- The ST_ObjectDrawAspect simple type (Part 1, §17.18.60) was added
- The ST_ObjectUpdateMode simple type (Part 1, §17.18.61) was added
- The ST_StyleSort simple type (Part 1, §17.18.82) was added
- The ST_UnqualifiedPercentage simple type (§14.10.10) was added
- The start element (Part 1, §17.4.34) was added
- The start element (Part 1, §17.4.35) was added
- The start element (Part 1, §17.4.36) was added
- The start element (Part 1, §17.4.37) was added
- The start, startChars, end, endChars attributes were added to the ind element (Part 1, §17.3.1.12)
- The tabIndex element (Part 1, §17.5.2.41) was added
- The target attribute was added to the optimizeForBrowser element (Part 1, §17.15.2.33)
- The tblCaption element (Part 1, §17.4.41) was added
- The tblDescription element (Part 1, §17.4.47) was added
- The title element (Part 1, §17.15.2.43) was added
- The uiCompat97To2003 element was removed
- The vendorID and dllVersions attributes on the activeWritingStyle element (Part 1, §17.15.1.1) was modified to use ST_String (Part 1, §22.9.2.13)

D.2 SpreadsheetML

The following changes occurred to the SpreadsheetML schema:

- The algorithmName, hashValue, saltValue, and spinCount attributes were added to sheetProtection (Part 1, §18.3.1.85; Part 1, §18.3.1.84), protectedRange (Part 1, §18.3.1.71), sheetProtection (Part 1, §18.3.1.85), and fileSharing (Part 1, §18.2.12)
- The anchor element (Part 1, §18.3.1.1) was added
- The characterSet attribute was added to the textPr element (Part 1, §18.13.12) and the webPublishing element (Part 1, §18.2.24)
- The commentPr element (Part 1, §18.7.5) was added
- The conformance attribute was added to the workbook element (Part 1, §18.2.27)
- The controlPr element (Part 1, §18.3.1.20) was added
- The drawingHF element (Part 1, §18.3.1.37) was added
- The end element (Part 1, §18.8.16) was added
- The objectPr element (Part 1, §18.3.1.56) was added
- The paperHeight and paperWidth attributes were added to the pageSetup element (Part 1, §18.3.1.63)
- The paperHeight and paperWidth attributes were added to the pageSetup element (Part 1, §18.3.1.64)
- The refreshedDateIso attribute was added to the pivotCacheDefinition element (Part 1, §18.10.1.67)
- The Schema element (Part 1, §18.16.4) now allows mixed content
- The SchemaLanguage attribute was added to the schema element (Part 1, §18.16.4)
- The securityDescriptor element (Part 1, §18.3.1.77) was added

- The shapeId attribute was added to the comment element (Part 1, §18.7.3)
- The ST_CalendarType simple type (Part 1, §22.9.2.1) now allows an enumeration value of saka
- The ST_CellType simple type (Part 1, §18.18.11) now allows an enumeration value of d
- The ST_FileType simple type (Part 1, §18.18.29) now allows enumeration values of lin and other
- The ST_PivotAreaType simple type (Part 1, §18.18.58) now allows an enumeration value of topEnd
- The ST_TextHAlign simple type (Part 1, §18.18.80) was added
- The ST_TextVAlign simple type (Part 1, §18.18.81) was added
- The ST_XmlDataType simple type (Part 1, §18.18.93) was modified to allow any xsd:string
- The start element (Part 1, §18.8.37) was added
- The startLabels attribute was added to the dataConsolidate element (Part 1, §18.3.1.29)
- The vallso and maxVallso attributes were added to the dynamicFilter element (Part 1, §18.3.2.5)
- The workbookPasswordCharacterSet, revisionsPasswordCharacterSet, revisionsAlgorithmName, revisionsHashValue, revisionsSaltValue, revisionsSpinCount, workbookAlgorithmName, workbookHashValue, workbookSaltValue, and workbookSpinCount attributes were added to the workbookProtection element (Part 1, §18.2.29)

D.3 PresentationML

The following changes occurred to the PresentationML schema:

- The algorithmName, hashValue, saltValue, and spinCount attributes were added to the modifyVerifier element (Part 1, §19.2.1.19)
- The conformance attribute was added to the presentation element (Part 1, §19.2.1.26)
- The contentPart element was added (Part 1, §19.3.1.14)
- The pubBrowser attribute on the htmlPubPr element (§16.2.1.1) was renamed target
- The ST_HtmlPublishWebBrowserSupport simple type was removed and replaced by xsd:string

D.4 DrawingML

D.4.1 DrawingML – Main

The following changes occurred to the DrawingML Main schema:

- The builtIn attribute was removed from the snd element (Part 1, §19.5.68)
- The content model of ST_Coordinate (Part 1, §20.1.10.16) was modified to allow ST_UniversalMeasure (Part 1, §22.9.2.15)
- The content model of ST_Coordinate32 (Part 1, §20.1.10.17) was modified to allow ST_UniversalMeasure (Part 1, §22.9.2.15)
- The content model of ST_FixedPercentage (Part 1, §20.1.10.24) was modified to allow ST_FixedPercentage (Part 1, §22.9.2.3)
- The content model of ST_Percentage (Part 1, §20.1.10.40) was modified to allow ST_Percentage (Part 1, §22.9.2.9)
- The content model of ST_PositiveFixedPercentage (Part 1, §20.1.10.45) was modified to allow ST_PositiveFixedPercentage (Part 1, §22.9.2.10)

- The content model of ST_PositivePercentage (Part 1, §20.1.10.46) was modified to allow ST_PositivePercentage (Part 1, §22.9.2.11)
- The contentType attribute was added to the videoFile (Part 1, §20.1.3.6) and audioFile elements (Part 1, §20.1.3.2)
- The header element (Part 1, §21.1.3.3) was added
- The headers element (Part 1, §21.1.3.4) was added
- The id attribute was added to the tc element (Part 1, §21.1.3.16)
- The rtl element (Part 1, §21.1.2.2.8) was added
- The ST_PresetColorVal simple type (Part 1, §20.1.10.48) now allows enumeration values of: darkBlue, darkCyan, darkGoldenrod, darkGray, darkGrey, darkGreen, darkKhaki, darkMagenta, darkOliveGreen, darkOrange, darkOrchid, darkRed, darkSalmon, darkSeaGreen, darkSlateBlue, darkSlateGray, darkSlateGrey, darkTurquoise, darkViolet, dkGrey, dkSlateGrey, dimGrey, grey, lightBlue, lightCoral, lightCyan, lightGoldenrodYellow, lightGray, lightGrey, lightGreen, lightPink, lightSalmon, lightSeaGreen, lightSkyBlue, lightSlateGray, lightSlateGrey, lightSteelBlue, lightYellow, ltGrey, ltSlateGrey, mediumAquamarine, mediumBlue, mediumOrchid, mediumPurple, mediumSeaGreen, mediumSlateBlue, mediumSpringGreen, mediumTurquoise, mediumVioletRed, slateGrey
- The ST_TextFontScalePercent simple type was renamed to ST_TextFontScalePercentOrPercentString (Part 1, §20.1.10.67) and modified to allow ST_Percentage (Part 1, §22.9.2.9)
- The ST_TextPoint simple type (Part 1, §20.1.10.74) was modified to allow ST_UniversalMeasure (Part 1, §22.9.2.15)
- The ST_TextSpacingPercent simple type was renamed to ST_TextSpacingPercentOrPercentString (Part 1, §20.1.10.77) and modified to allow ST_Percentage (Part 1, §22.9.2.9)
- The title attribute was added to the cNvPr element (Part 1, §20.1.2.2.8)

D.4.2 DrawingML – Chart

The following changes occurred to the Chart schema:

- The paperHeight and paperWidth attributes were added to the pageSetup element (Part 1, §21.2.2.134)

D.4.3 DrawingML – Diagrams

The following changes occurred to the Diagram schema:

- The ST_HorizontalAlignment simple type was renamed to ST_DiagramHorizontalAlignment (Part 1, §21.4.7.24)
- The ST_TextAlignment simple type was renamed to ST_DiagramTextAlignment (Part 1, §21.4.7.25)

D.4.4 DrawingML – Spreadsheet Drawing

The following changes occurred to the Spreadsheet Drawing schema:

- The contentPart element (Part 1, §20.5.2.12) was added

D.5 VML

D.5.1 VML

The following changes occurred to the VML schema:

- The ST_TrueFalseBlank simple type (§20.1.2.6) now allows enumeration values of True and False

D.5.2 VML – Office Drawing

The following changes occurred to the Office Drawing schema:

- The ST_DiagramLayout simple type (§19.2.3.10) was added
- The equationxml element (§19.2.2.10) was added
- The contentType attribute was added to the ink element (§19.2.2.15)
- The ST_AlternateMathContentType simple type (§19.2.3.1) was added
- The ST_OLELinkType simple type (§19.2.3.19) was modified to allow any xsd:string
- The ST_TrueFalseBlank simple type (§20.1.2.6) now allows enumeration values of True and False

D.5.3 VML – Spreadsheet Drawing

The following changes occurred to the Spreadsheet Drawing schema:

- The ST_CF simple type (§19.4.3.1) was modified to allow any xsd:string
- The ST_TrueFalseBlank simple type (§20.1.2.6) now allows enumeration values of true and false

D.6 Shared

D.6.1 Shared – Bibliography

The following changes occurred to the Bibliography schema:

- The ST_String255 simple type was removed and replaced by ST_String (Part 1, §22.9.2.13)

D.6.2 Shared – Custom Properties Variant Types

The following changes occurred to the Custom Properties Variant Types schema:

- The cf element was removed
- The ST_Cf simple type was removed

D.6.3 Shared – Math

The following changes occurred to the Math schema:

- The ST_YAlign simple type (Part 1, §22.9.2.20) now uses an enumeration value of bottom in place of bot, and allows values inside and outside
- The ST_XAlign simple type (Part 1, §22.9.2.18) now allows values inside and outside

- The content model of ST_OnOff (Part 1, §22.9.2.7) was changed to an xsd:boolean, removing the values on and off

D.6.4 Shared Simple Types

The following changes occurred to shared simple types:

- The ST_UniversalMeasure simple type (Part 1, §22.9.2.15) was added
- The ST_Algorithm simple type (§20.1.2.1) now uses an enumeration value of custom in place of invalid
- The ST_Algorithm simple type (§20.1.2.2) now uses an enumeration value of custom in place of invalid
- The ST_CryptProv simple type (§20.1.2.4) now uses an enumeration value of custom in place of invalid
- The content model of ST_OnOff (Part 1, §22.9.2.7) was changed to an xsd:boolean, removing the values on and off
- The content model of ST_TwipsMeasure (Part 1, §22.9.2.14) was modified to allow ST_PositiveUniversalMeasure (Part 1, §22.9.2.12)
- The ST_PositiveUniversalMeasure simple type (Part 1, §22.9.2.12) was added
- The ST_Percentage simple type (Part 1, §22.9.2.9) was added
- The ST_FixedPercentage simple type (Part 1, §22.9.2.3) was added
- The ST_PositivePercentage simple type (Part 1, §22.9.2.11) was added
- The ST_PositiveFixedPercentage simple type (Part 1, §22.9.2.10) was added

D.7 Custom XML Schema References

The following changes occurred to the Custom XML Schema References schema:

- The schemaLanguage attribute was added to the schema element (Part 1, §23.2.1)

End informative annex.

Bibliography

The following documents are useful references for implementers and users of this International Standard, in addition to the Normative References:

Information on elements, attributes, and OPC parts in ECMA-376 (OOXML),
<http://purl.oclc.org/ooxml/onlineInfomativeAnnexes>