

Second Edition, 2008

ECMA-376-4

Information technology —

Document description and processing languages —

Office Open XML File Formats

Part 4:

Transitional Migration Features

Table of Contents

Foreword	ix
Introduction	x
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	10
6. Acronyms and Abbreviations	11
7. General Description	12
8. Additional Shared Parts	13
8.1 VML Drawing Part	13
9. WordprocessingML Reference Material	15
9.1 Table of Contents.....	15
9.2 Paragraphs and Rich Formatting.....	18
9.2.1 Paragraphs.....	18
9.2.2 Run Content.....	19
9.3 Tables.....	22
9.3.1 left (Table Cell Leading Edge Border)	22
9.3.2 left (Table Leading Edge Border)	22
9.3.3 left (Table Cell Leading Margin Exception)	23
9.3.4 left (Table Cell Leading Margin Default)	23
9.3.5 right (Table Cell Trailing Edge Border).....	23
9.3.6 right (Table Trailing Edge Border).....	24
9.3.7 right (Table Cell Trailing Margin Default)	24
9.3.8 right (Table Cell Trailing Margin Exception)	24
9.3.9 Additional attribute for cnfStyle element (Part 1, §17.4.8)	24
9.3.10 Additional attribute for cnfStyle element (Part 1, §17.4.7)	25
9.3.11 Additional attribute for tblLook element (Part 1, §17.4.56)	27
9.3.12 Additional attribute for tblLook element (Part 1, §17.4.55)	27
9.4 Fonts	28
9.4.1 Elements	28
9.5 Numbering	30
9.5.1 pict (Picture Numbering Symbol Properties).....	30
9.6 Annotations.....	31
9.6.1 Revisions	31
9.7 Settings	40
9.7.1 Legacy Password Hash Algorithm.....	40

9.7.2	Document Settings	47
9.7.3	Compatibility Settings.....	62
9.8	Miscellaneous Topics	146
9.8.1	Text Box Content	146
9.9	Fields and Hyperlinks	149
9.9.1	Syntax	149
9.9.2	Legacy language references	150
9.9.3	Use of DOS File Paths	157
9.9.4	Field definitions	158
9.9.5	fldData (Custom Field Data)	166
9.9.6	fldData (Custom Field Data)	167
9.10	Simple Types.....	168
9.10.1	Additional member types for the union in ST_DecimalNumberOrPercent (Part 1, §17.18.11) ...	168
9.10.2	Additional enumeration values for ST_Jc (Part 1, §17.18.44)	168
9.10.3	Additional enumeration values for ST_JcTable (Part 1, §17.18.45)	169
9.10.4	Additional enumeration values for ST_StyleSort (Part 1, §17.18.82)	169
9.10.5	Additional enumeration values for ST_TabJc (Part 1, §17.18.84)	169
9.10.6	Additional enumeration values for ST_TextDirection (Part 1, §17.18.93)	170
9.10.7	ST_Cnf (Conditional Formatting Bitmask)	170
9.10.8	ST_UnqualifiedPercentage (Percentage Value Without Percent Sign)	171
10.	SpreadsheetML Reference Material.....	173
10.1	Table of Contents	173
10.2	Workbook	173
10.2.1	Additional attribute for fileSharing element (Part 1, §18.2.12)	173
10.2.2	Additional attribute for webPublishing element (Part 1, §18.2.24).....	174
10.2.3	Additional attributes for workbookProtection element (Part 1, §18.2.29)	174
10.3	Worksheets.....	180
10.3.1	Worksheets.....	180
10.3.2	AutoFilter Settings	183
10.4	Styles.....	184
10.4.1	left (Leading Edge Border).....	184
10.4.2	right (Trailing Edge Border)	184
10.5	Pivot Tables.....	185
10.5.1	Pivot Tables.....	185
10.6	External Data Connections	185
10.6.1	Additional attribute for textPr element (Part 1, §18.13.12)	185
10.7	Simple Types.....	186
10.7.1	Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)	186
10.7.2	ST_UnsignedShortHex (Unsigned Short Hex).....	186
11.	PresentationML Reference Material	187
11.1	Table of Contents	187
11.2	Presentation	187
11.2.1	Presentation Properties.....	187
11.3	Slides.....	196
11.3.1	Embedded Objects	196
11.4	Simple Types.....	197
11.4.1	ST_WebColorType (HTML Slide Navigation Control Colors)	197

11.4.2	ST_WebEncoding (Web Encoding)	197
11.4.3	ST_WebScreenSize (HTML/Web Screen Size Target)	197
12.	DrawingML - Framework Reference Material.....	199
12.1	DrawingML - Main	199
12.1.1	Table of Contents	199
12.1.2	Simple Types	199
12.2	DrawingML - Legacy Compatibility	202
12.2.1	Table of Contents	202
12.2.2	Basics	202
13.	DrawingML - Components Reference Material.....	204
13.1	DrawingML - Charts	204
13.1.1	Table of Contents	204
13.1.2	Elements	204
14.	VML Reference Material	205
14.1	VML.....	205
14.1.1	Table of Contents	206
14.1.2	Elements	207
14.1.3	Simple Types.....	595
14.2	VML - Office Drawing.....	603
14.2.1	Table of Contents	604
14.2.2	Elements	605
14.2.3	Simple Types.....	731
14.3	VML - WordprocessingML Drawing.....	744
14.3.1	Table of Contents	744
14.3.2	Elements	745
14.3.3	Simple Types.....	754
14.4	VML - SpreadsheetML Drawing.....	760
14.4.1	Table of Contents	761
14.4.2	Elements	763
14.4.3	Simple Types.....	797
14.5	VML - PresentationML Drawing	799
14.5.1	Table of Contents	800
14.5.2	Elements	800
15.	Shared MLs Reference Material.....	802
15.1	Shared Simple Types.....	802
15.1.1	Table of Contents	802
15.1.2	Simple Types.....	802
Annex A. (normative) Schemas – W3C XML Schema.....		811
A.1	WordprocessingML.....	811
A.2	SpreadsheetML.....	879
A.3	PresentationML.....	963
A.4	DrawingML - Framework	996
A.4.1	DrawingML - Main	996
A.4.2	DrawingML - Picture	1054
A.4.3	DrawingML - Legacy Compatibility	1054

A.4.4	DrawingML - Locked Canvas	1054
A.4.5	DrawingML - WordprocessingML Drawing.....	1055
A.4.6	DrawingML - SpreadsheetML Drawing.....	1058
A.5	DrawingML - Components	1062
A.5.1	DrawingML - Charts	1062
A.5.2	DrawingML - Chart Drawings.....	1089
A.5.3	DrawingML - Diagrams	1092
A.6	VML.....	1113
A.6.1	VML.....	1113
A.6.2	VML - Office Drawing.....	1123
A.6.3	VML - WordprocessingML Drawing.....	1133
A.6.4	VML - SpreadsheetML Drawing.....	1135
A.6.5	VML - PresentationML Drawing	1137
A.7	Shared MLs.....	1137
A.7.1	Math	1137
A.7.2	Extended Properties	1148
A.7.3	Custom Properties	1149
A.7.4	Variant Types	1151
A.7.5	Custom XML Data Properties	1154
A.7.6	Bibliography.....	1155
A.7.7	Additional Characteristics.....	1158
A.7.8	Office Document Relationships	1158
A.7.9	Shared Simple Types.....	1159
A.8	Custom XML Schema References	1162
Annex B. (informative) Schemas – RELAX NG		1163
B.1	WordprocessingML.....	1163
B.1.1	Part Schemas	1207
B.2	SpreadsheetML.....	1213
B.2.1	Part Schemas	1301
B.3	PresentationML.....	1308
B.3.1	Part Schemas	1331
B.4	DrawingML - Framework	1334
B.4.1	DrawingML - Main	1334
B.4.2	DrawingML – Compatibility	1379
B.4.3	DrawingML - Picture	1379
B.4.4	DrawingML - Locked Canvas.....	1379
B.4.5	DrawingML - Wordprocessing Drawing.....	1380
B.4.6	DrawingML - Spreadsheet Drawing.....	1382
B.5	DrawingML - Components	1384
B.5.1	DrawingML - Chart.....	1384
B.5.2	DrawingML - Chart Drawing	1402
B.5.3	DrawingML - Diagrams	1404
B.6	VML.....	1420
B.6.1	VML - Main	1420
B.6.2	VML - Office Drawing.....	1427
B.6.3	VML - Wordprocessing Drawing.....	1435
B.6.4	VML - Spreadsheet Drawing.....	1436

B.6.5	VML - Presentation Drawing.....	1438
B.7	Shared MLs.....	1439
B.7.1	Math	1439
B.7.2	Extended Properties	1444
B.7.3	Custom Properties	1445
B.7.4	Variant Types	1446
B.7.5	Custom XML Data Properties	1450
B.7.6	Bibliography.....	1450
B.7.7	Additional Characteristics.....	1452
B.7.8	Office Document Relationships	1453
B.7.9	Shared Simple Types.....	1453
B.8	Custom XML Schema References	1455
B.9	Additional Resources	1456
B.9.1	Any.....	1456
B.9.2	XML.....	1456
Annex C. (informative)	Namespace Prefix Mapping in Examples.....	1457
Annex D. (informative)	Differences Between ECMA-376:2008 and ECMA-376:2006	1459

Foreword

Changes from the 1st edition were made to align this 2nd edition Standard with ISO/IEC 29500:2008. Both this 2nd edition and ISO/IEC 29500:2008 refer to the 1st edition. As such, this 2nd edition does not cancel or replace the 1st edition.

Some important differences between ECMA-376:2008 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.

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 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 shall only be used by documents of conformance class WML Transitional, SML Transitional, or PML Transitional.

2. Conformance

2.1 Document Conformance

Document conformance is purely syntactic.

- A conforming document shall conform to the transitional W3C XML Schema, and any additional syntax constraints.
- The document shall be of category Wordprocessing, Spreadsheet, or Presentation (see Part 1, §4).
- The document character set shall conform to the Unicode Standard and ISO/IEC 10646:2003, with either the UTF-8 or UTF-16 encoding form, as required by the XML 1.0 standard.
- Any XML element or attribute not explicitly included in ECMA-376 shall use the extensibility mechanisms described by ECMA-376-1 and ECMA-376-3.

Each Part of this multi-part standard has its own conformance clause. The term *conformance class* is used to disambiguate conformance within different Parts of this multi-part standard. This Part of ECMA-376 defines the following document conformance classes:

- *WML Transitional*, if the document is a conforming document of category Wordprocessing that conforms to the transitional schema.
- *SML Transitional*, if the document is a conforming document of category Spreadsheet that conforms to the transitional schema.
- *PML Transitional*, if the document is a conforming document of category Presentation that conforms to the transitional schema.

[Note: Other document conformance classes could be defined in the future. *end note*]

[Note: A document cannot be of more than one of the above conformance classes. *end note*]

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 10646:2003, *Information technology — Universal Multiple-Octet Coded Character Set (UCS)*.

ISO/IEC 14496-22:2007, *Information technology — Coding of audio-visual objects — Part 22: Open Font Format*

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)*. Retrieved from KDE Developer's Corner:
<http://developer.kde.org/documentation/tutorials/kparts/>

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://scaleplus.law.gov.au/html/pastereg/3/1519/pdf/NatMeasurement1999.pdf>

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/documentation/QuickTime/QTFF/qtff.pdf>

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*, Version 5.0, defined by: The Unicode Standard, Version 5.0 (Reading, MA, Addison-Wesley, 2006. ISBN 0-321-48091-0) , <http://www.unicode.org/unicode/standard>.

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, Eve Maler, Jean Paoli, C. M. Sperberg-McQueen, John Cowan, and François Yergeau (editors). *Extensible Markup Language (XML) 1.1*, Third Edition. World Wide Web Consortium. 2004. <http://www.w3.org/TR/2004/REC-xml11-20040204/>

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

XML Namespaces, Bray, Tim, Dave Hollander, Andrew Layman, and Richard Tobin (editors). *Namespaces in XML 1.1*. World Wide Web Consortium. 2004. <http://www.w3.org/TR/2004/REC-xml-names11-20040204/>

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.txt

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*]

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.

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, as in "*normative*".
1. In each definition of a term in §4 (Terms and Definitions), the term is written in bold, as in "**behavior**".
2. The tag name of an XML element is written using an Element style, as in "document".
3. The name of an XML attribute is written using an Attribute style, as in "id".
4. The value of an XML attribute is written using a constant-width style, as in "CommentReference".
5. The qualified or unqualified name of a simple type, complex type, or base datatype is written using a Type style, as in "xsd:anyURI".

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 is intended for use by implementers, academics, and application programmers. As such, it contains a considerable amount of explanatory material that, strictly speaking, is not necessary in a formal specification.

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

1. Front matter (clauses 1–7);
2. Main body (clauses 8–15);
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 8–15
- 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*]

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 Reference Material

9.1 Table of Contents

This subclause is informative.

9.2	Paragraphs and Rich Formatting.....	18
9.2.1	Paragraphs.....	18
9.2.1.1	Additional attribute for cnfStyle element (Part 1, §17.3.1.8)	18
9.2.1.2	Additional attributes for ind element (Part 1, §17.3.1.12).....	19
9.2.2	Run Content.....	19
9.2.2.1	control (Floating Embedded Control).....	19
9.2.2.2	pict (VML Object).....	21
9.3	Tables.....	22
9.3.1	left (Table Cell Leading Edge Border)	22
9.3.2	left (Table Leading Edge Border)	22
9.3.3	left (Table Cell Leading Margin Exception)	23
9.3.4	left (Table Cell Leading Margin Default)	23
9.3.5	right (Table Cell Trailing Edge Border).....	23
9.3.6	right (Table Trailing Edge Border).....	24
9.3.7	right (Table Cell Trailing Margin Default)	24
9.3.8	right (Table Cell Trailing Margin Exception)	24
9.3.9	Additional attribute for cnfStyle element (Part 1, §17.4.8)	24
9.3.10	Additional attribute for cnfStyle element (Part 1, §17.4.7)	25
9.3.11	Additional attribute for tblLook element (Part 1, §17.4.56)	27
9.3.12	Additional attribute for tblLook element (Part 1, §17.4.55)	27
9.4	Fonts	28
9.4.1	Elements	28
9.4.1.1	Additional attribute for charset element (Part 1, §17.8.3.2)	28
9.5	Numbering	30
9.5.1	pict (Picture Numbering Symbol Properties).....	30
9.6	Annotations.....	31
9.6.1	Revisions	31
9.6.1.1	numberingChange (Previous Numbering Field Properties).....	31
9.6.1.2	numberingChange (Previous Paragraph Numbering Properties).....	34
9.7	Settings	40
9.7.1	Legacy Password Hash Algorithm.....	40
9.7.2	Document Settings	47
9.7.2.1	hdrShapeDefaults (Default Properties for VML Objects in Header and Footer)	47
9.7.2.2	shapeDefaults (Default Properties for VML Objects in Main Document)	48
9.7.2.3	Additional attributes for documentProtection element (Part 1, §17.15.1.29)	48

9.7.2.4	Additional attribute for stylePaneFormatFilter element (Part 1, §17.15.1.85)	54
9.7.2.5	Additional attributes for writeProtection element (Part 1, §17.15.1.93)	56
9.7.3	Compatibility Settings.....	62
9.7.3.1	alignTablesRowByRow (Align Table Rows Independently).....	62
9.7.3.2	allowSpaceOfSameStyleInTable (Allow Contextual Spacing of Paragraphs in Tables).....	64
9.7.3.3	autofitToFirstFixedWidthCell (Allow Table Columns To Exceed Preferred Widths of Constituent Cells)	66
9.7.3.4	autoSpaceLikeWord95 (Incorrectly Adjust Text Spacing for Specific Unicode Ranges)	68
9.7.3.5	cachedColBalance (Use Cached Paragraph Information for Column Balancing)	70
9.7.3.6	convMailMergeEsc (Treat Backslash Quotation Delimiter as Two Quotation Marks)	71
9.7.3.7	displayHangulFixedWidth (Always Use Fixed Width for Hangul Characters).....	71
9.7.3.8	doNotAutofitConstrainedTables (Do Not AutoFit Tables To Fit Next To Wrapped Objects)	73
9.7.3.9	doNotBreakConstrainedForcedTable (Don't Break Table Rows Around Floating Tables)	74
9.7.3.10	doNotBreakWrappedTables (Do Not Allow Floating Tables To Break Across Pages)	76
9.7.3.11	doNotSnapToGridInCell (Do Not Snap to Document Grid in Table Cells with Objects)	77
9.7.3.12	doNotSuppressIndentation (Do Not Ignore Floating Objects When Calculating Paragraph Indentation)	79
9.7.3.13	doNotSuppressParagraphBorders (Do Not Suppress Paragraph Borders Next To Frames)	81
9.7.3.14	doNotUseEastAsianBreakRules (Do Not Compress Compressible Characters When Using Document Grid).....	82
9.7.3.15	doNotUseHTMLParagraphAutoSpacing (Use Fixed Paragraph Spacing for HTML Auto Setting)	83
9.7.3.16	doNotUseIndentAsNumberingTabStop (Ignore Hanging Indent When Creating Tab Stop After Numbering)	85
9.7.3.17	doNotVertAlignCellWithSp (Don't Vertically Align Cells Containing Floating Objects)	86
9.7.3.18	doNotVertAlignInTxbx (Ignore Vertical Alignment in Textboxes)	88
9.7.3.19	doNotWrapTextWithPunct (Do Not Allow Hanging Punctuation With Character Grid)	90
9.7.3.20	footnoteLayoutLikeWW8 (Ignore Page Break from Continuous Section Break).....	91
9.7.3.21	forgetLastTabAlignment (Ignore Width of Last Tab Stop When Aligning Paragraph If It Is Not Left Aligned).....	94
9.7.3.22	growAutofit (Allow Tables to AutoFit Into Page Margins)	96
9.7.3.23	layoutRawTableWidth (Ignore Space Before Table When Deciding If Table Should Wrap Floating Object)	97
9.7.3.24	layoutTableRowsApart (Allow Table Rows to Wrap Inline Objects Independently)	99
9.7.3.25	lineWrapLikeWord6 (Ignore Compression of Full-Width Punctuation Ending a Line)	101
9.7.3.26	mwSmallCaps (Use Specific Small Caps Algorithm).....	102
9.7.3.27	noColumnBalance (Do Not Balance Text Columns within a Section).....	103
9.7.3.28	noExtraLineSpacing (Do Not Center Content on Lines With Exact Line Height)	105
9.7.3.29	noLeading (Do Not Add Leading Between Lines of Text)	106
9.7.3.30	noSpaceRaiseLower (Do Not Increase Line Height for Raised/Lowered Text).....	108
9.7.3.31	noTabHangInd (Do Not Create Custom Tab Stop for Hanging Indent)	109
9.7.3.32	printBodyTextBeforeHeader (Print Body Text before Header/Footer Contents)	110
9.7.3.33	printColBlack (Print Colors as Black And White without Dithering)	111
9.7.3.34	selectFldWithFirstOrLastChar (Select Field When First or Last Character Is Selected)	112
9.7.3.35	shapeLayoutLikeWW8 (Ignore Text Wrapping around Objects at Bottom of Page).....	113
9.7.3.36	showBreaksInFrames (Display Page/Column Breaks Present in Frames)	115
9.7.3.37	spacingInWholePoints (Only Expand/Condense Text By Whole Points).....	117
9.7.3.38	splitPgBreakAndParaMark (Always Move Paragraph Mark to Page after a Page Break)	119
9.7.3.39	subFontBySize (Require Exact Size During Font Substitution)	121

9.7.3.40	suppressBottomSpacing (Ignore Exact Line Height for Last Line on Page)	121
9.7.3.41	suppressSpacingAtTopOfPage (Ignore Minimum Line Height for First Line on Page)	123
9.7.3.42	suppressSpBfAfterPgBrk (Do Not Use Space Before On First Line After a Page Break)	125
9.7.3.43	suppressTopSpacing (Ignore Minimum and Exact Line Height for First Line on Page)	127
9.7.3.44	suppressTopSpacingWP (Use Static Text Leading)	128
9.7.3.45	swapBordersFacingPages (Swap Paragraph Borders on Odd Numbered Pages)	129
9.7.3.46	truncateFontHeightsLikeWP6 (Use Truncated Integer Division For Font Calculation)	131
9.7.3.47	underlineTabInNumList (Underline Following Character Following Numbering)	132
9.7.3.48	useAltKinsokuLineBreakRules (Use Alternate Set of East Asian Line Breaking Rules)	133
9.7.3.49	useAnsiKerningPairs (Use ANSI Kerning Pairs from Fonts)	134
9.7.3.50	useFELayout (Do Not Bypass East Asian/Complex Script Layout Code)	135
9.7.3.51	useNormalStyleForList (Do Not Automatically Apply List Paragraph Style To Bulleted/Numbered Text)	136
9.7.3.52	usePrinterMetrics (Use Printer Metrics To Display Documents)	137
9.7.3.53	useSingleBorderforContiguousCells (Use Simplified Rules For Table Border Conflicts)	138
9.7.3.54	useWord2002TableStyleRules (Incorrectly Display Top Border of Conditional Columns)	139
9.7.3.55	useWord97LineBreakRules (Use Incorrect Inter-Character Spacing Rules)	141
9.7.3.56	wpJustification (Fit To Expanded Width When Performing Full Justification)	143
9.7.3.57	wpSpaceWidth (Use Specific Space Width)	144
9.7.3.58	wrapTrailSpaces (Line Wrap Trailing Spaces)	145
9.8	Miscellaneous Topics	146
9.8.1	Text Box Content	146
9.8.1.1	txbxContent (Rich Text Box Content Container)	146
9.9	Fields and Hyperlinks	149
9.9.1	Syntax	149
9.9.2	Legacy language references	150
9.9.3	Use of DOS File Paths	157
9.9.4	Field definitions	158
9.9.4.1	AUTONUM	158
9.9.4.2	AUTONUMLGL	159
9.9.4.3	AUTONUMOUT	160
9.9.4.4	BARCODE	161
9.9.4.5	BIDIOUTLINE	162
9.9.4.6	EQ	162
9.9.4.7	INFO	166
9.9.5	fldData (Custom Field Data)	166
9.9.6	fldData (Custom Field Data)	167
9.10	Simple Types	168
9.10.1	Additional member types for the union in ST_DecimalNumberOrPercent (Part 1, §17.18.11)	168
9.10.2	Additional enumeration values for ST_Jc (Part 1, §17.18.44)	168
9.10.3	Additional enumeration values for ST_JcTable (Part 1, §17.18.45)	169
9.10.4	Additional enumeration values for ST_StyleSort (Part 1, §17.18.82)	169
9.10.5	Additional enumeration values for ST_TabJc (Part 1, §17.18.84)	169
9.10.6	Additional enumeration values for ST_TextDirection (Part 1, §17.18.93)	170
9.10.7	ST_Cnf (Conditional Formatting Bitmask)	170
9.10.8	ST_UnqualifiedPercentage (Percentage Value Without Percent Sign)	171

End of informative text.

9.2 Paragraphs and Rich Formatting

9.2.1 Paragraphs

9.2.1.1 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? • 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></pre>

Attributes	Description
	<pre> <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 (§9.10.7).</p>

9.2.1.2 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>

9.2.2 Run Content

9.2.2.1 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]*

Parent Elements
pict (§9.2.2.2); pict (§9.5.1)

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 http://schemas.openxmlformats.org/officeDocument/2006/relationships/control 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>
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:</i> Consider the following WordprocessingML markup for an embedded control in a document:</p>

Attributes	Description
	<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</i>: This positioning data is sufficient to display the control in any case where:</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</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="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*]

9.2.2.2 [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 (§14.1).

[*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*]

Parent Elements
r (Part 1, §22.1.2.87); r (Part 1, §17.3.2.25)

Child Elements	Subclause
control (Floating Embedded Control)	§9.2.2.1
movie (Embedded Video)	Part 1, §17.3.3.17
Any element in the urn:schemas-microsoft-com:vml namespace	§14.1
Any element in the urn:schemas-microsoft-com:office:office namespace	§14.2

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

9.3 Tables

9.3.1 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.

Parent Elements
tcBorders (Part 1, §17.4.67)

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

9.3.2 left (Table Leading Edge Border)

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

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

Parent Elements
<code>tblBorders</code> (Part 1, §17.4.39); <code>tblBorders</code> (Part 1, §17.4.40)

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

9.3.3 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 `bidirectional` property (Part 1, §17.4.1) applied, this cell margin is applied to the right edge of the cell.

Parent Elements
<code>tcMar</code> (Part 1, §17.4.69)

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

9.3.4 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 `bidirectional` property (Part 1, §17.4.1) applied, this cell margin is applied to the right edge of the cell.

Parent Elements
<code>tblCellMar</code> (Part 1, §17.4.42); <code>tblCellMar</code> (Part 1, §17.4.43)

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

9.3.5 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 `bidirectional` property (Part 1, §17.4.1) applied, this border is applied to the left edge of the cell.

Parent Elements
<code>tcBorders</code> (Part 1, §17.4.67)

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

9.3.6 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.

Parent Elements
tblBorders (Part 1, §17.4.39); tblBorders (Part 1, §17.4.40)

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

9.3.7 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.

Parent Elements
tblCellMar (Part 1, §17.4.42); tblCellMar (Part 1, §17.4.43)

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

9.3.8 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.

Parent Elements
tcMar (Part 1, §17.4.69)

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

9.3.9 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)	Specifies the set of conditional formatting properties that have been applied to this object.

Attributes	Description
	<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 (§9.10.7).</p>

9.3.10 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 (§9.10.7).</p>

9.3.11 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>

9.3.12 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

Attributes	Description
	<ul style="list-style-type: none">• 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>

9.4 Fonts

9.4.1 Elements

9.4.1.1 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>The value of this attribute shall be interpreted as follows:</p> <table><tr><th>Value</th><th>Description</th></tr></table>	Value	Description
Value	Description		

Attributes	Description	
	0x00	Specifies a Latin character set. (IANA name <code>iso-8859-1</code>)
	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 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 data-bbox="415 1503 1446 1535">[Example: Consider the following value for an attribute of type <code>ST_UCharHexNumber</code>:</p> <pre data-bbox="451 1575 695 1606"><... w:val="BE"/></pre> <p data-bbox="415 1646 1461 1709">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 data-bbox="415 1749 1433 1812">The possible values for this attribute are defined by the <code>ST_UCharHexNumber</code> simple type (Part 1, §17.18.98).</p>		

9.5 Numbering

9.5.1 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]

Parent Elements
numPicBullet (Part 1, §17.9.21)

Child Elements	Subclause
control (Floating Embedded Control)	§9.2.2.1
movie (Embedded Video)	Part 1, §17.3.3.17
Any element in the urn:schemas-microsoft-com:vml namespace	§14.1
Any element in the urn:schemas-microsoft-com:office:office namespace	§14.2

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

9.6 Annotations

9.6.1 Revisions

9.6.1.1 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*]

Parent Elements
fldChar (Part 1, §17.16.18)

Attributes	Description
author (Annotation Author)	Specifies the author for an annotation within a WordprocessingML document. If this attribute is omitted, then no author shall be associated with the parent annotation

Attributes	Description
	<p>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> <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>

Attributes	Description
	The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).
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="padding-left: 40px;">Some 1<ins>2</ins>. text</p> <p>This revision to the field result would be stored as follows in the WordprocessingML:</p> <pre style="padding-left: 40px;"><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*]

9.6.1.2 [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
13	Iroha
14	decimalFullWidth
15	decimalHalfWidth
16	japaneseLegal
17	japaneseDigitalTenThousand
18	decimalEnclosedCircle
19	decimalFullWidth2

nfc Value	ST_NumberFormat enumeration equivalent
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
51	hindiNumbers
52	hindiCounting
53	thaiLetters
54	thaiNumbers
55	thaiCounting
56	vietnameseCounting
57	numberInDash

nfc Value	ST_NumberFormat enumeration equivalent
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*]

Parent Elements
numPr (Part 1, §17.3.1.19)

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>

Attributes	Description
<p>date (Annotation Date)</p>	<p>The possible values for this attribute are defined by the ST_String simple type (Part 1, §22.9.2.13).</p> <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>
<p>id (Annotation Identifier)</p>	<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> <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>
<p>original (Previous Numbering Value)</p>	<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</p>

Attributes	Description
	<p>revision, as follows:</p> <p>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*]

9.7 Settings

9.7.1 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 $0x5585 \text{ XOR } 0x0D42 = 0x58C7$.
 - 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 = $((\text{low-order word SHR } 14) \text{ AND } 0x0001) \text{ OR } ((\text{low-order word SHL } 1) \text{ AND } 0x7FFF) \text{ XOR 'e'}$ = $((0 \text{ SHR } 14) \text{ AND } 0x0001) \text{ OR } ((0 \text{ SHL } 1) \text{ AND } 0x7FFF) \text{ XOR } 0x65 = 0x0065$.
 - The next to last character of the password is 'l' (0x6C). Again, by the formula, $((0x0065 \text{ SHR } 14) \text{ AND } 0x0001) \text{ OR } ((0x0065 \text{ SHL } 1) \text{ AND } 0x7FFF) \text{ XOR } 0x6C = (0x0000 \text{ OR } 0x00CA) \text{ XOR } 0x6C = 0x00CA \text{ XOR } 0x6C = 0x00A6$.
 - 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 = $((0x1199 \text{ SHR } 14) \text{ AND } 0x0001) \text{ OR } ((0x1199 \text{ SHL } 1) \text{ AND } 0x7FFF) \text{ XOR } 0x0007 \text{ XOR } 0xCE4B = 0x2332 \text{ XOR } 0x0007 \text{ XOR } 0xCE4B = 0x2335 \text{ XOR } 0xCE4B = 0xED7E$.
- The end result for the key is 0x64CEED7E.

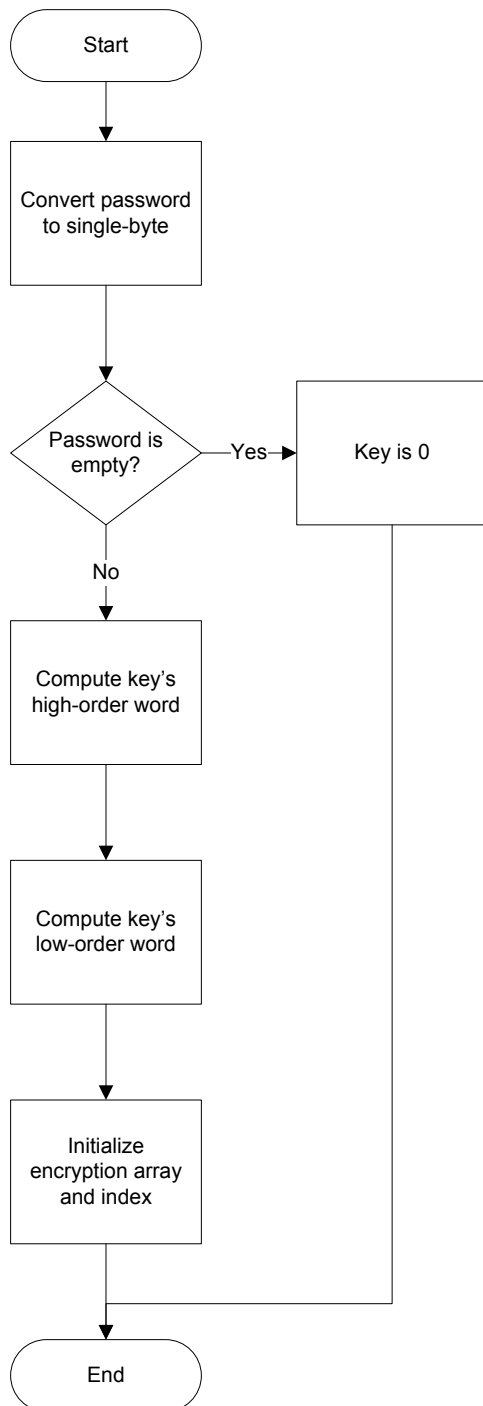
end example]

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

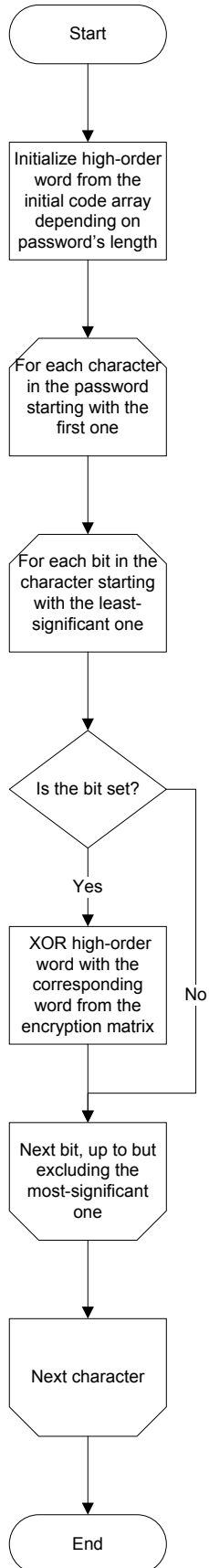
Second, the byte order of the result shall be reversed [*Example:* 0x64CEED7E becomes 7EEDCE64. *end example]*, and that value shall be hashed as defined by the attribute values.

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

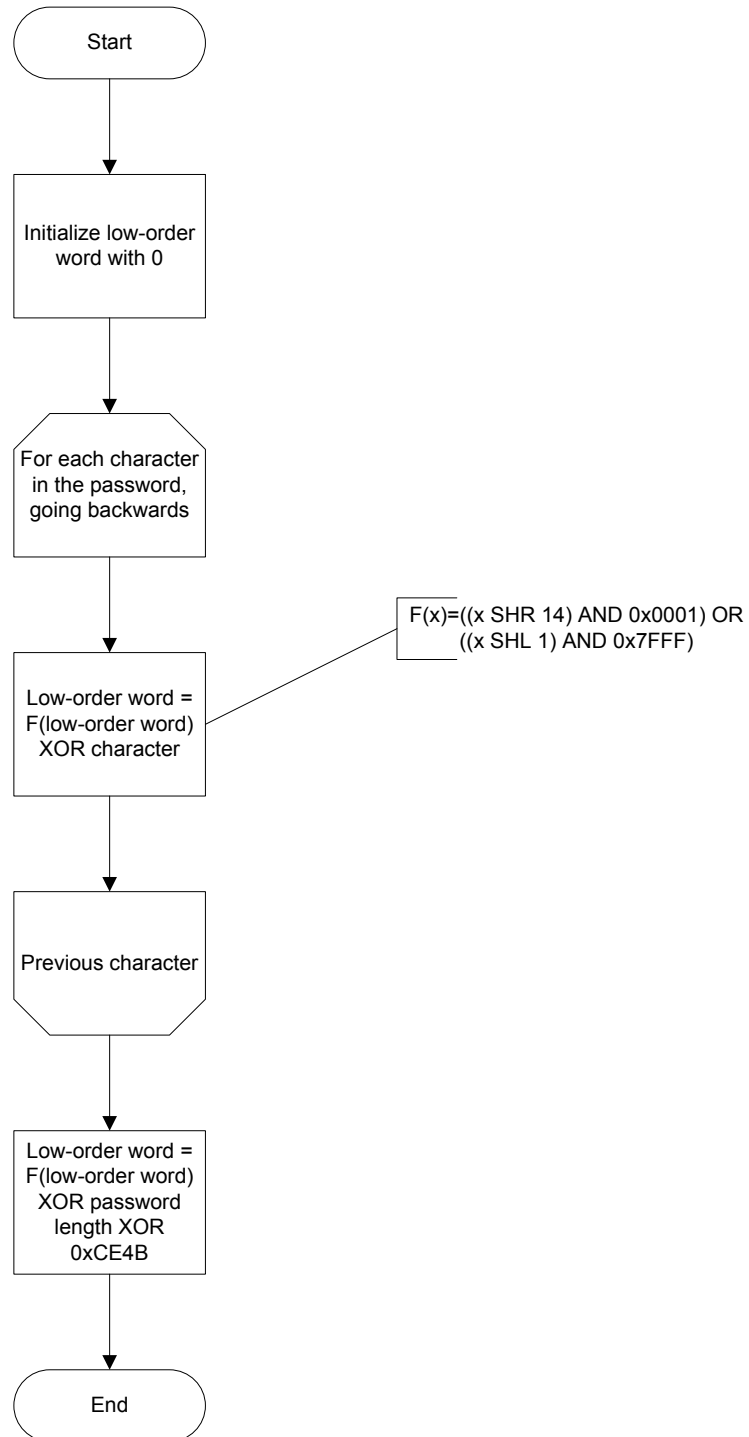
Calculate Key



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*]

9.7.2 Document Settings

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

This element specifies the default parameters for object using the VML syntax (§14.1) 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 (§14.2) 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*]

Parent Elements
settings (Part 1, §17.15.1.78)

Child Elements	Subclause
Any element in the urn:schemas-microsoft-com:office:office namespace	§14.2

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

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

This element specifies the default parameters for object using the VML syntax (§14.1) 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 (§14.2) 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*]

Parent Elements
settings (Part 1, §17.15.1.78)

Child Elements	Subclause
Any element in the urn:schemas-microsoft-com:office:office namespace	§14.2

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

9.7.2.3 [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. <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>

Attributes	Description																																
	<pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" 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 (§15.1.2.1).</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="415 821 1349 1631"> <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 WordprocessingML document with the following information stored in one of its protection elements:</p> <pre><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="4"</pre>	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><code>w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></code></p> <p>The <code>cryptAlgorithmSid</code> 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>The possible values for this attribute are defined by the <code>ST_DecimalNumber</code> simple type (Part 1, §17.18.10).</p>
<p><code>cryptAlgorithmType</code> (Cryptographic Algorithm Type)</p>	<p>Specifies the type of cryptographic algorithm used by this protection. [Note: The initial version of ECMA-376 only supports a single algorithm type - <code>typeAny</code> - but future versions can expand this as necessary. <i>end note</i>]</p> <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="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p>The <code>cryptAlgorithmType</code> attribute value of <code>typeAny</code> 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 <code>ST_AlgType</code> simple type (§15.1.2.2).</p>
<p><code>cryptProvider</code> (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>[Example: 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 <code>cryptProvider</code> attribute value of <code>Krista'sProvider</code> 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 <code>ST_String</code> simple type (Part 1, §22.9.2.13).</p>
<p><code>cryptProviderType</code> (Cryptographic</p>	<p>Specifies the type of cryptographic provider to be used.</p>

Attributes	Description
Provider Type)	<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 (§15.1.2.4).</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> <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>

Attributes	Description
	<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> <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</p>

Attributes	Description
	<p>element (if any) followed by the SHA-1 algorithm (specified via the <code>cryptAlgorithmSid</code> 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 <code>base64Binary</code> 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> <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 <code>base64Binary</code> datatype.</p>

9.7.2.4 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 <code>customStyle</code> 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> </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 <code>customStyle</code> 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.
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 <code>customStyle</code> 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.								

Attributes	Description	
	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 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 data-bbox="418 1675 1385 1707"><i>[Example: Consider a document with the following value in its document settings:</i></p> <pre data-bbox="456 1745 1094 1776"><w:stylePaneFormatFilter w:val="2002" /></pre> <p data-bbox="418 1814 1433 1845">The val attribute specifies two suggested filter options for the list of document styles:</p> <ul data-bbox="464 1852 1024 1883" style="list-style-type: none"> • Only custom styles should be shown (0002) 		

Attributes	Description
	<ul style="list-style-type: none"> 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>

9.7.2.5 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 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. <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"</pre>

Attributes	Description																								
	<p><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>																								
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 - <code>hash</code> - 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 <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 (§15.1.2.1).</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 border="1" data-bbox="415 1297 1351 1877"> <thead> <tr> <th>Value</th><th>Algorithm</th></tr> </thead> <tbody> <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> </tbody> </table>	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.
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.																								

Attributes	Description								
	<table border="1" data-bbox="418 245 1349 478"> <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 data-bbox="418 516 1398 583">[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre data-bbox="456 621 1127 758"><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="4" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p data-bbox="418 795 1442 863">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="418 900 1471 968">The possible values for this attribute are defined by the ST_DecimalNumber simple type (Part 1, §17.18.10).</p>	12	SHA-256	13	SHA-384	14	SHA-512	Any other value	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="418 984 1442 1089">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 data-bbox="418 1127 1398 1194">[<i>Example:</i> Consider a WordprocessingML document with the following information stored in one of its protection elements:</p> <pre data-bbox="456 1232 1127 1369"><... w:cryptAlgorithmClass="hash" w:cryptAlgorithmType="typeAny" w:cryptAlgorithmSid="1" w:hash="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p data-bbox="418 1407 1471 1474">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 data-bbox="418 1512 1365 1579">The possible values for this attribute are defined by the ST_AlgType simple type (§15.1.2.2).</p>								
cryptProvider (Cryptographic Provider)	<p data-bbox="418 1593 1471 1740">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 data-bbox="418 1778 1471 1845">If this attribute is omitted, then the built-in cryptographic provider on the system shall be used.</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: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 (§15.1.2.4).</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, 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</p>

Attributes	Description
	<p>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>
<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 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>
<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>

Attributes	Description
	<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> <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>

9.7.3 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 (Part 1, §17.15.1.21) 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 (§9.7.3.31). *end example*]

9.7.3.1 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.2 `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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.3 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.4 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>𐤁𐤅</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 (𐤁𐤅). 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 𐤁𐤅 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:

ab⁷⁷cd
ab 2 2 cd

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.5 **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]*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.6 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]

Parent Elements
compat (Part 1, §17.15.1.21)

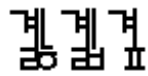
This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

9.7.3.7 displayHangulFixedWidth (Always Use Fixed Width for Hangul Characters)

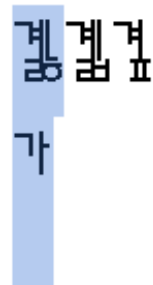
This element specifies whether applications should assume that all characters in the Hangul Syllables Unicode sub range (character values between 0xAC00 and 0xD7FF) are of a single fixed width or shall use the characters widths defined by the font in use (typical for a proportional width font).

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:



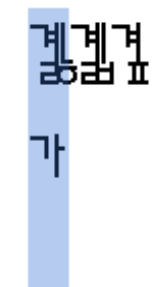
The default presentation would have each of those characters using the widths defined by the font (the highlighting indicates that each character has its own width):



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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.8 `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 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.9 **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:

[illegible]

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

Parent Elements
compat (Part 1, §17.15.1.21)

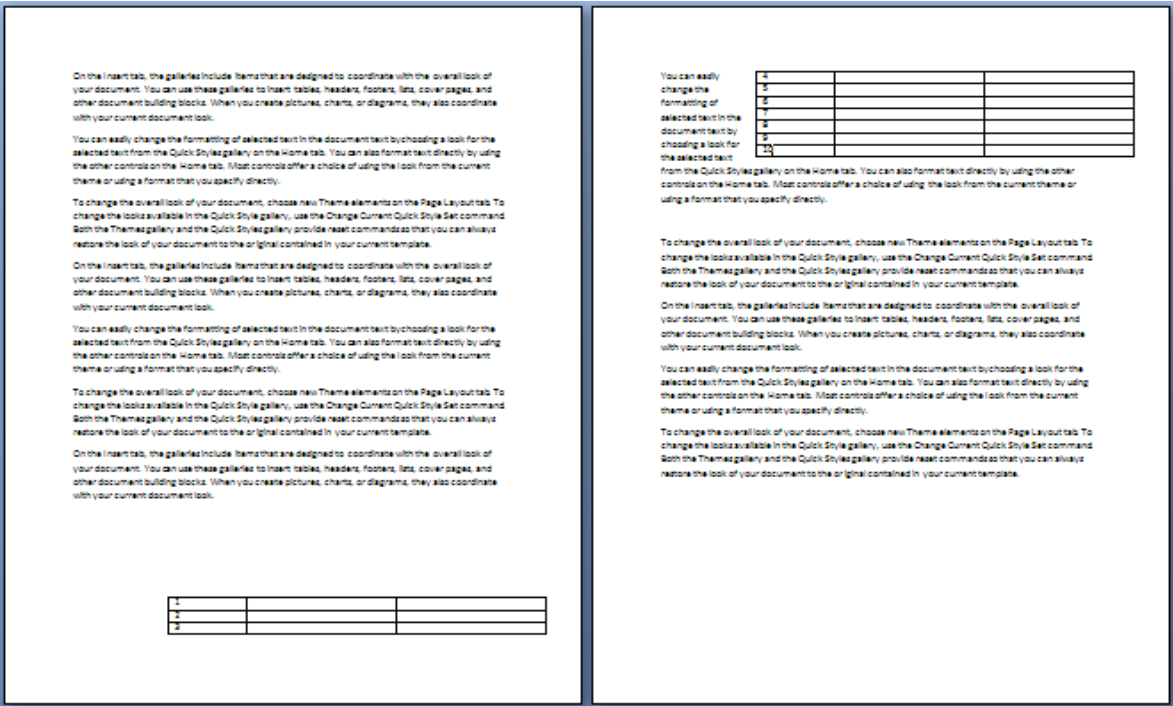
This element’s content model is defined by the common boolean property definition in Part 1, §17.17.4.

9.7.3.10 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 tblpPr 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:

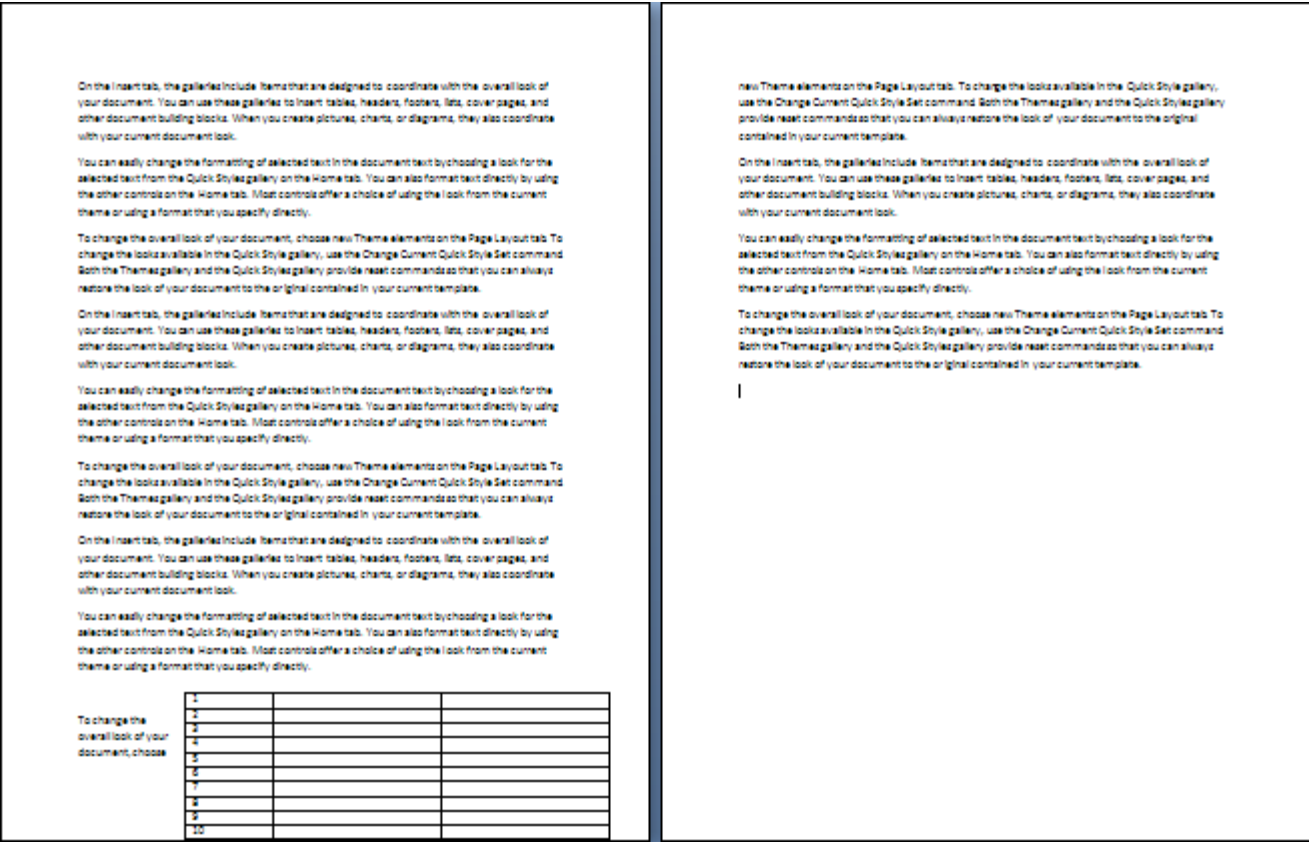


The default presentation of this document results in that table being broken across two pages of content.

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:



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

Parent Elements
compat (Part 1, §17.15.1.21)

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

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

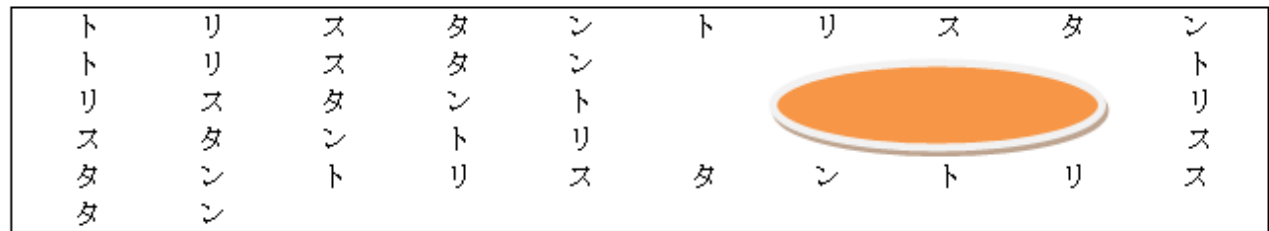
This element specifies whether a document grid defined using the docGrid element (Part 1, §17.6.5) shall be applied to the contents of table cells in that section which also contain floating objects. Note that the floating object shall be part of the cell, and simply not displayed over the cell due to 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="snapT0Cars" w:charSpace="146636" />  
</w:sectPr>
```

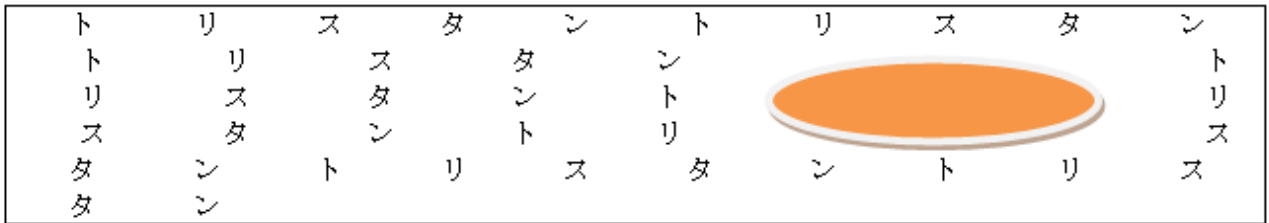
If this document contains a table with a single cell, containing some text and a single floating shape, the contents of the cell are still snapped to the 10 characters per line character grid, as follows:



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:



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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.12 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.

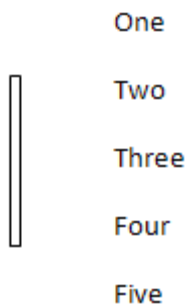
Typically, the presence of a floating object on the same line or lines as a paragraph shall only affect the text when the floating object occurs where that text would normally be presented. [*Example*: Text at a 1" indentation would only be displaced by a floating object that appears at that position 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:



-
1. One

2. Two

3. Three

4. Four

5. Five

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:

- One

Two

Three

Four

Five
1. One

2. Two

3. Three

4. Four

5. Five

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

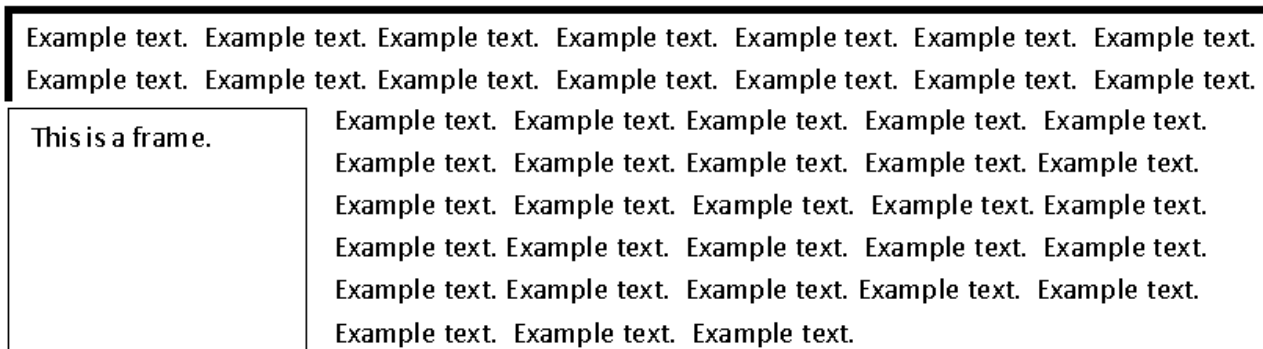
9.7.3.13 `doNotSuppressParagraphBorders` (Do Not Suppress Paragraph Borders Next To Frames)

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:

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.	
This is a frame.	Example text. Example text.

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.14 [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="snapTOcars" 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.15 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

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

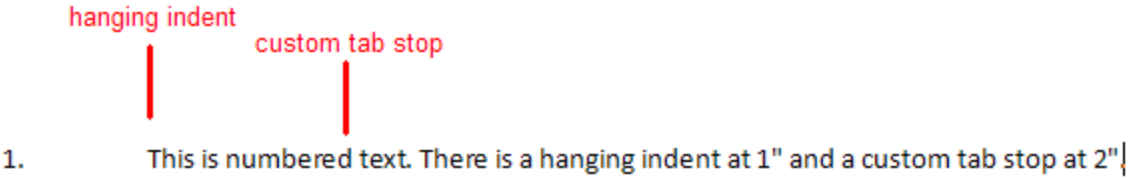
This element specifies whether applications shall use the custom tab stop generated by the hanging indent (if any) when advancing the text after the numbering for a numbered paragraph.

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:lvl1 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:lvl1>
</w:abstractNum>
```

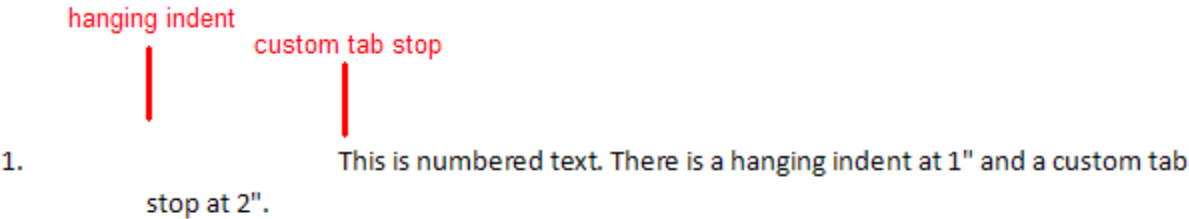
The default presentation of this document results in the tab stop generated by the numbering advancing to the virtual tab stop generated by the hanging indent at 1", as follows:



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:



end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

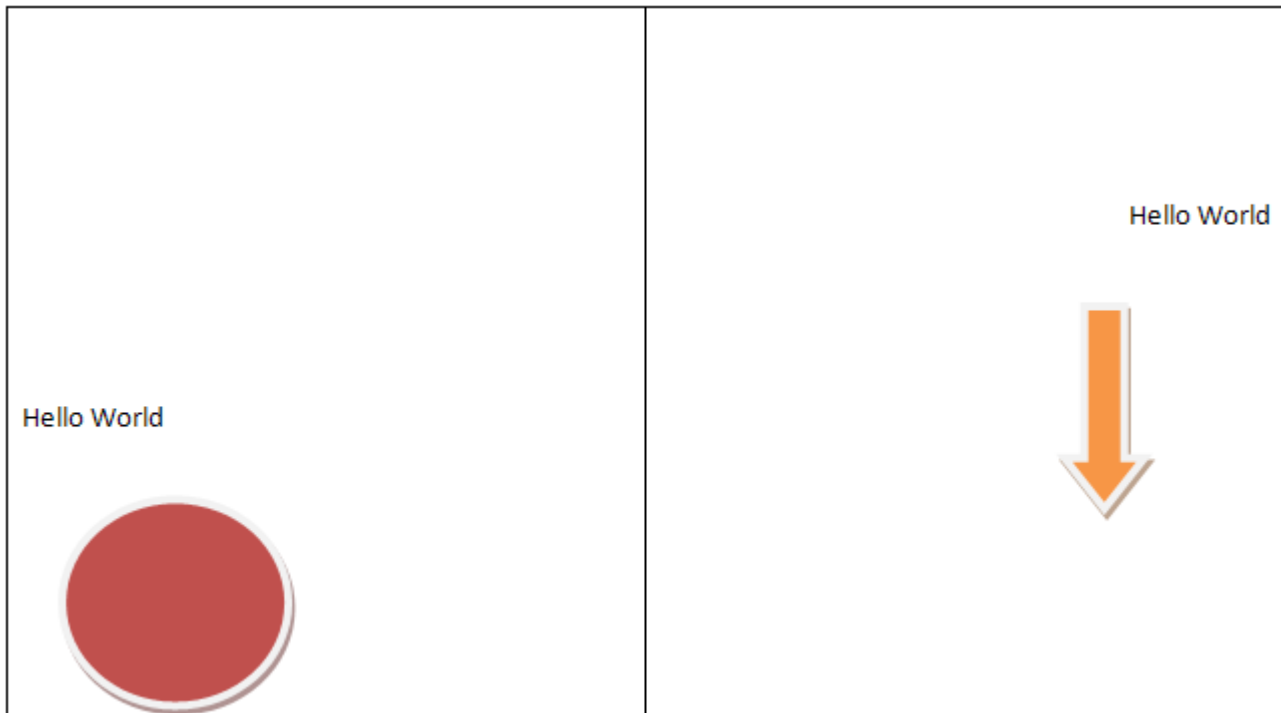
9.7.3.17 doNotVertAlignCellWithSp (Don't Vertically Align Cells Containing Floating Objects)

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.

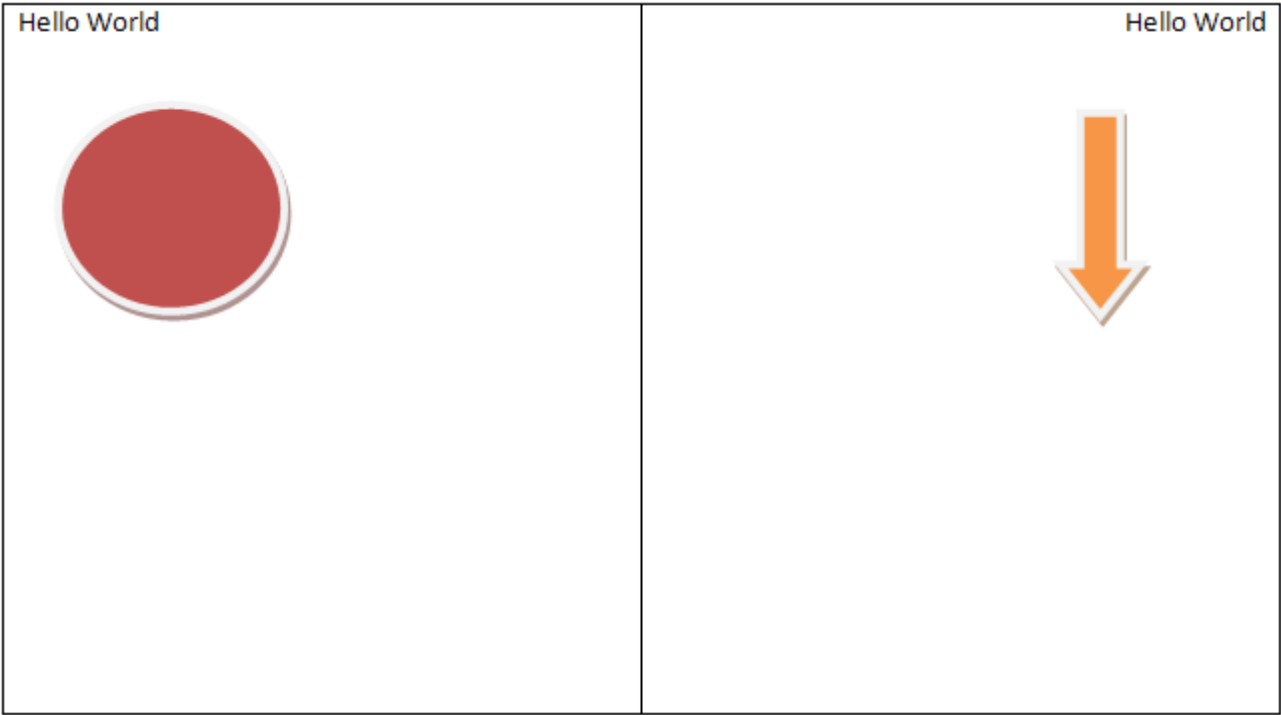
The default presentation of this document results in each cell (including the extents of the floating objects) being vertically aligned as specified, as follows:



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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.18 **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.

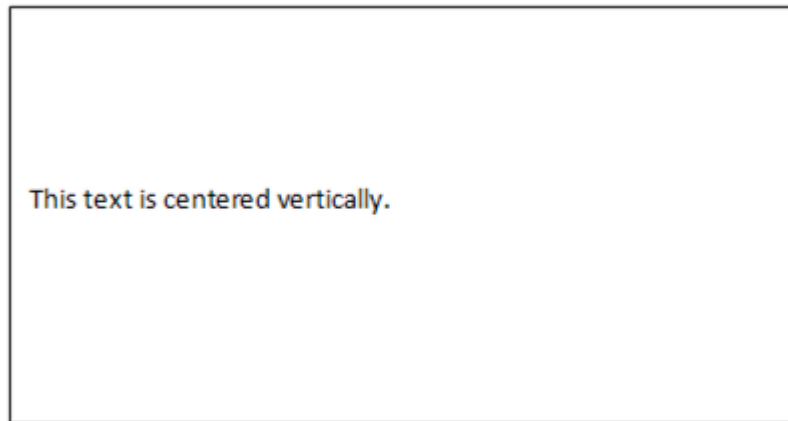
[*Example:* Consider a WordprocessingML table with a single center-aligned text box:

```

<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>

```

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



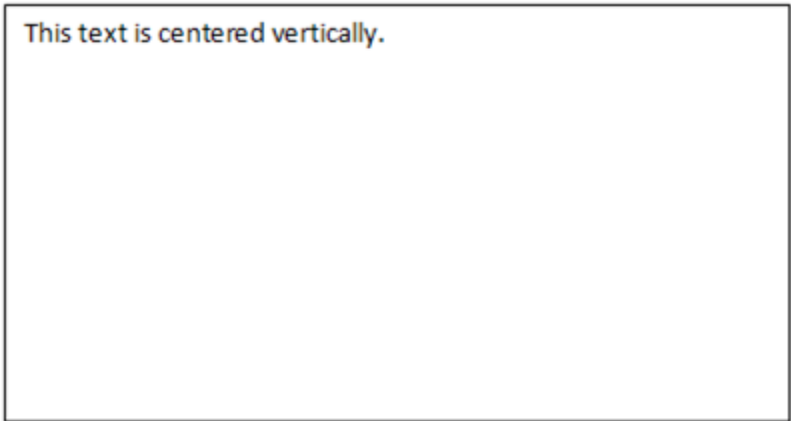
However, if this compatibility setting is turned on:

```

<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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.19 **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="snapTOCars" 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.20 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>

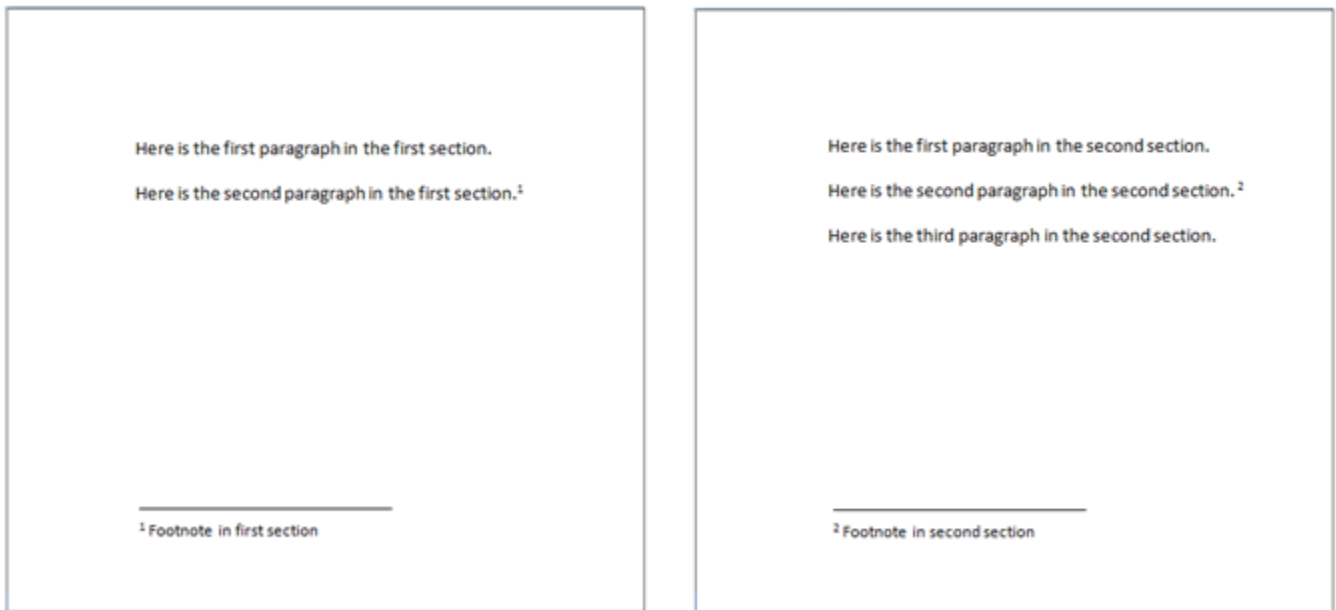
```

```

    <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:

<p>Here is the first paragraph in the first section.</p> <p>Here is the second paragraph in the first section.¹</p> <p>.....Section Break (Continuous)</p> <p>Here is the first paragraph in the second section.</p> <hr/> <p>¹ Footnote in first section</p>	<p>Here is the second paragraph in the second section.²</p> <p>Here is the third paragraph in the second section.</p> <hr/> <p>² Footnote in second section</p>
---	---

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.21 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

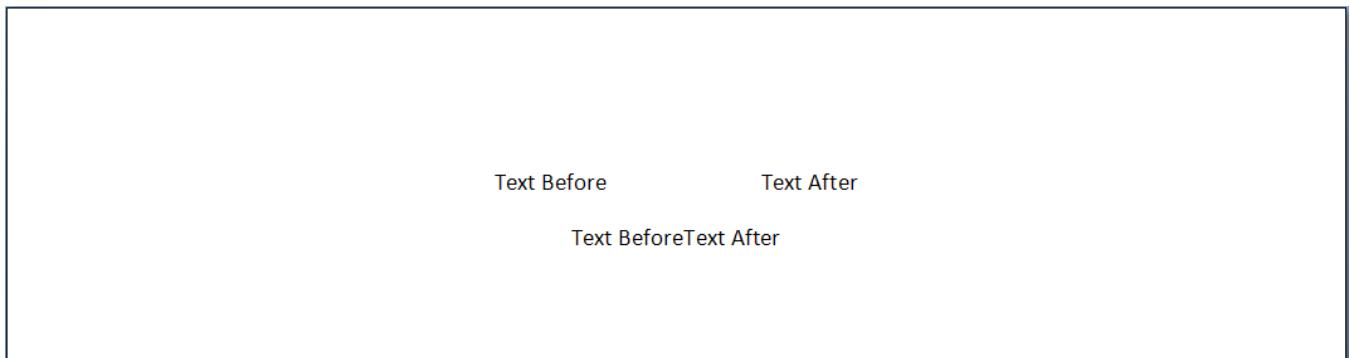
This is done to ensure that tab stops on a line do not change when the contents of the paragraph are aligned (i.e. the tab stops should not have to take into account the paragraph alignment).

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>
```

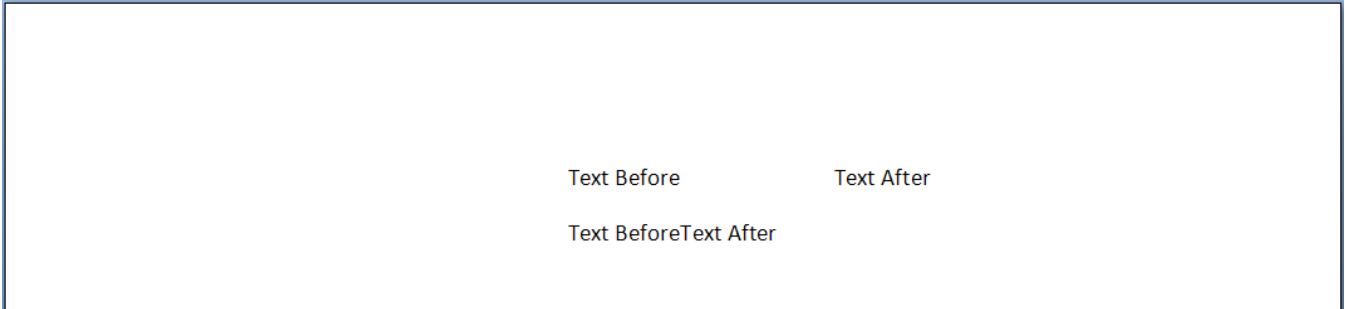
The default presentation would determine the full width of each line including the tab stops, finally aligning the resulting text to the center position as requested by the jc element:



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:



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*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.22 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:

veryverylongnonbreakingstringinthis table	veryverylongnonbreakingstringinthis table	veryverylongnonbreakingstringin this table

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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.23 [layoutRawTableWidth \(Ignore Space Before Table When Deciding If Table Should Wrap Floating Object\)](#)

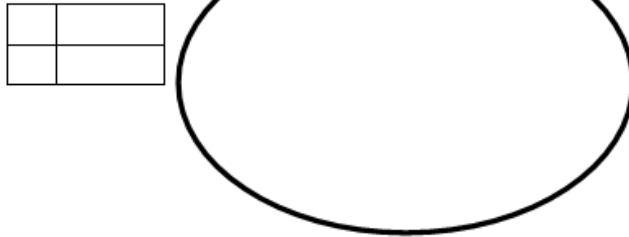
This element specifies how tables which have been indented from the margin using the `tblInd` element (Part 1, §17.4.51) shall be wrapped around floating objects.

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.

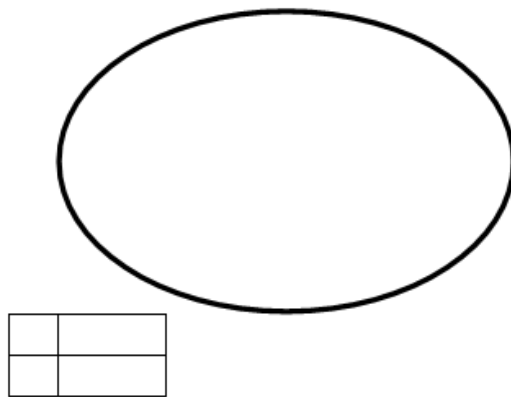
[*Example:* Consider a WordprocessingML document with a floating shape using square wrapping, next to a table which has been indented one inch from the left margin:

```
<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:



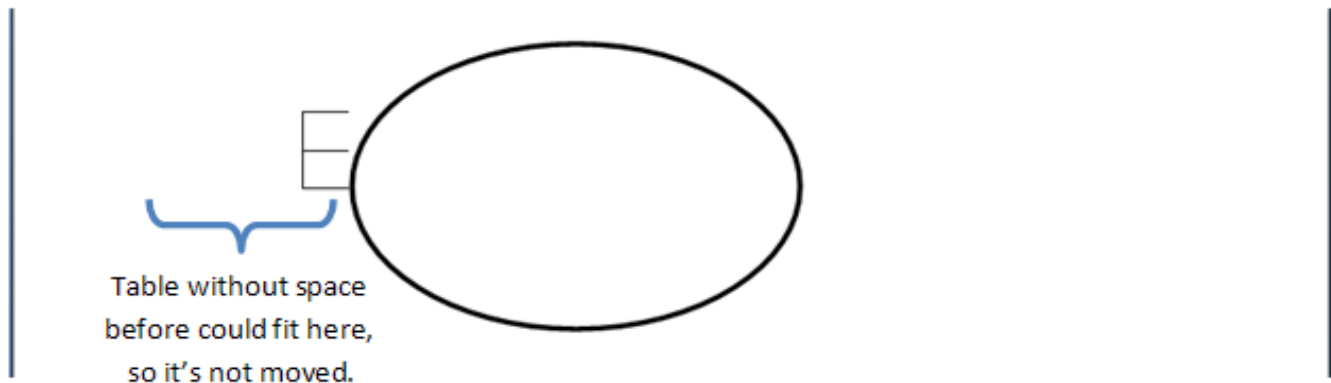
If this object is then moved to the left, such that it would clip the table, the default presentation would have the entire table moved below the shape, since it does not fit in the remaining space on the line:



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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

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

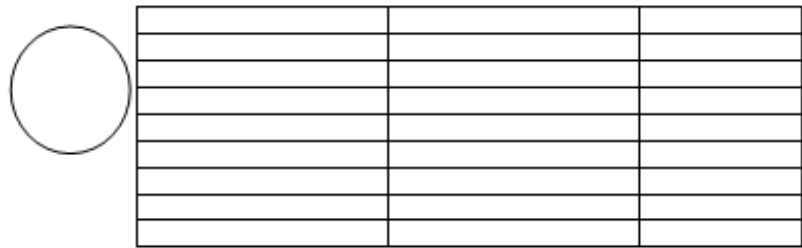
This element specifies whether tables which are wrapping around floating objects shall wrap around the object as a whole, or if each table row shall individually wrap the object as needed (causing a more stuttered, yet tighter, wrapping of the object).

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.

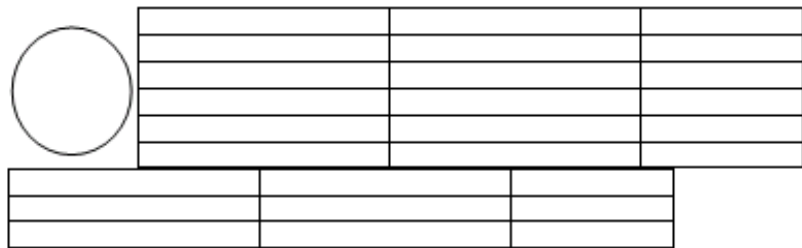


However, if this compatibility setting is turned on:

```
<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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.25 `lineWrapLikeWord6` (Ignore Compression of Full-Width Punctuation Ending a Line)

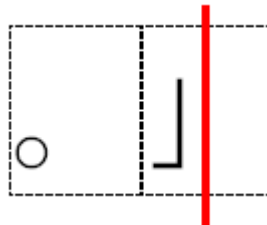
This element specifies that applications should ignore the character compression settings specified by the `characterSpacingControl` element (Part 1, §17.15.1.18) when determining if one more character fits within the text margins on each line of the document. This setting typically results in a character 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.26 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.27 `noColumnBalance` (Do Not Balance Text Columns within a Section)

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 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.28 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.29 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.30 **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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

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

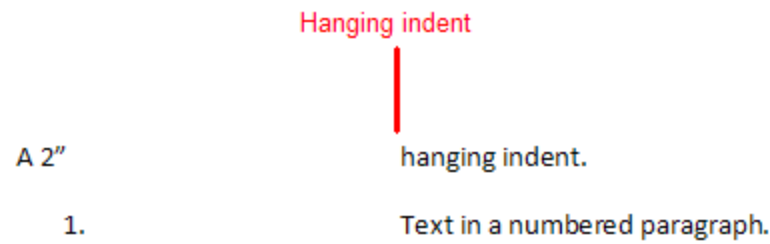
This element specifies whether applications should always create a hanging indent as a custom tab stop when handling tabs within the contents of a WordprocessingML paragraph. The dontUseIndentAsNumberingTabStop element (§9.7.3.16) specifies if this tab stop shall be used in the case of a tab added as 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>
```

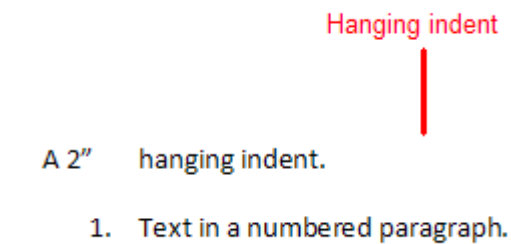
The default presentation would have both the numbering and the tab in the regular paragraph advancing to the 2" custom tab stop generated by the hanging indent:



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:



end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.32 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]*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.33 `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]*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.34 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.35 `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:

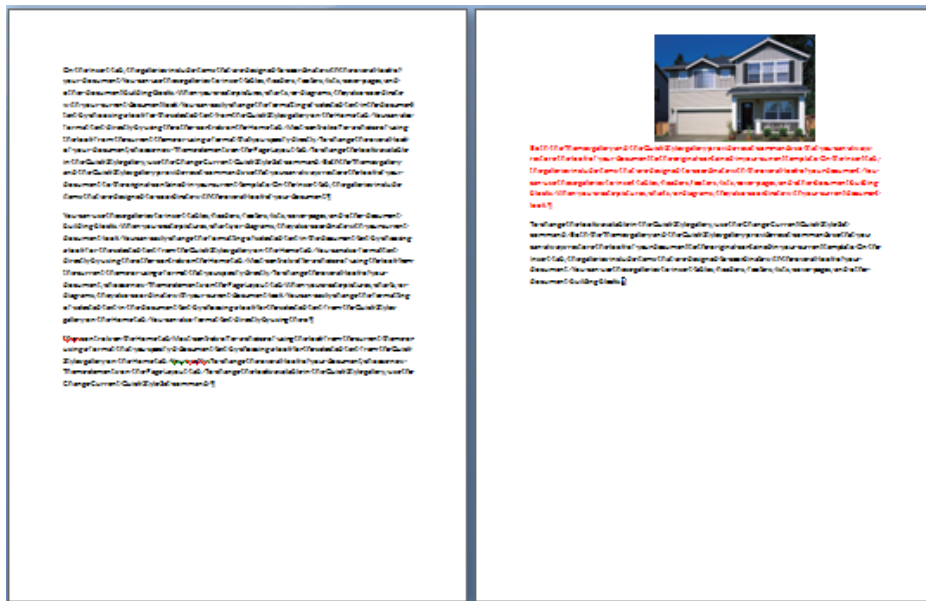
```
<w:p>
  <w:r>
    <w:t>Sample text. Sample text. Sample text. Sample text. Sample text. Sample
text.</w:t>
  </w:r>
  <w:r>
    <w:drawing>
      <wp:anchor ... >
        <wp:positionV relativeFrom="line">
          <wp:posOffset>-428914</wp:posOffset>
        </wp:positionV>
```

```

<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):



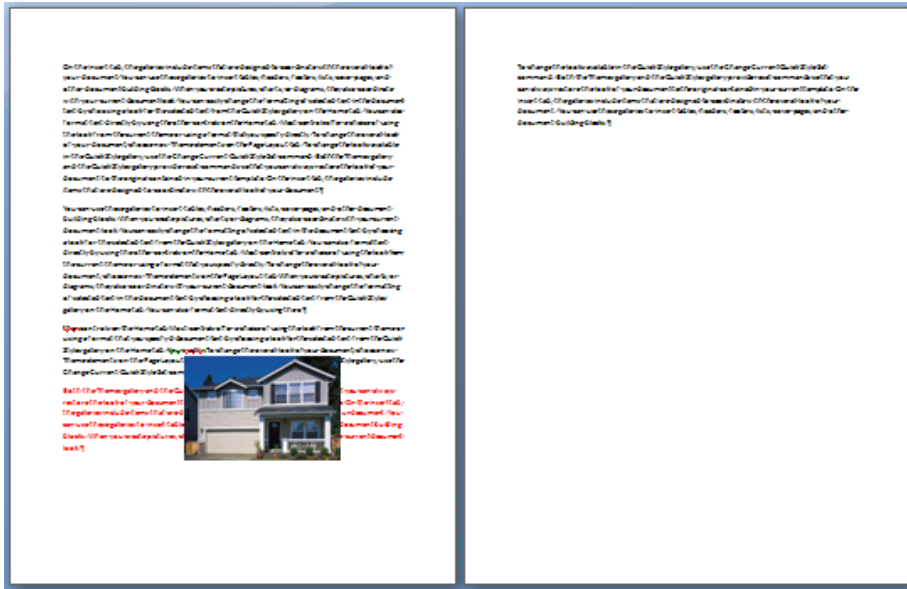
If this compatibility setting is turned on:

```

<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]

Parent Elements
compat (Part 1, §17.15.1.21)

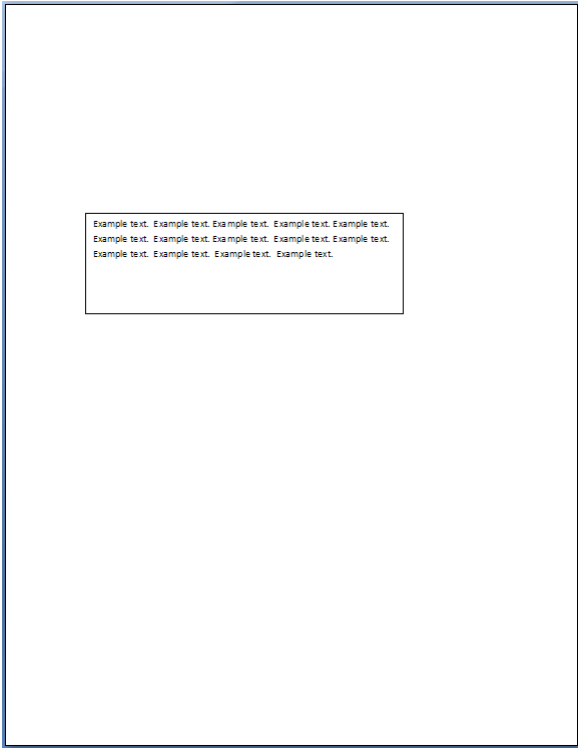
This element's content model is defined by the common boolean property definition in Part 1, §17.17.4.

9.7.3.36 showBreaksInFrames (Display Page/Column Breaks Present in Frames)

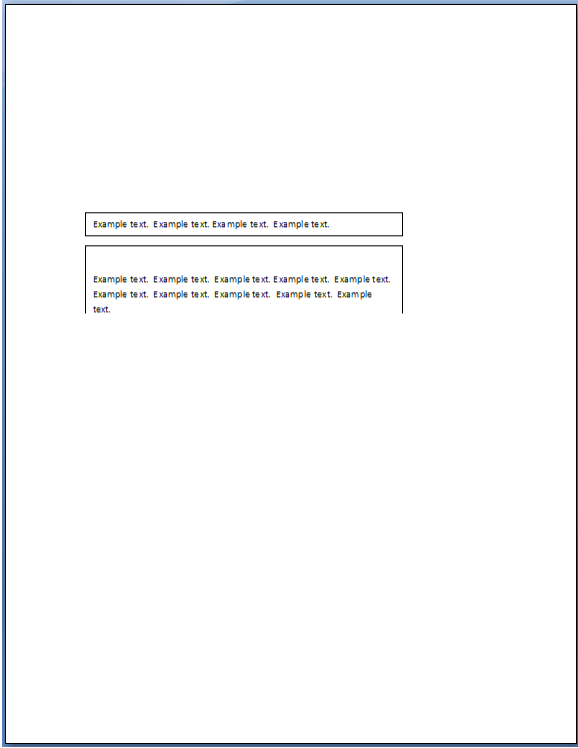
This element specifies whether applications should honor the presence of page and/or column breaks which are present within the contents of paragraphs which have been defined as frames using the framePr element (Part 1, §17.3.1.11).

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:



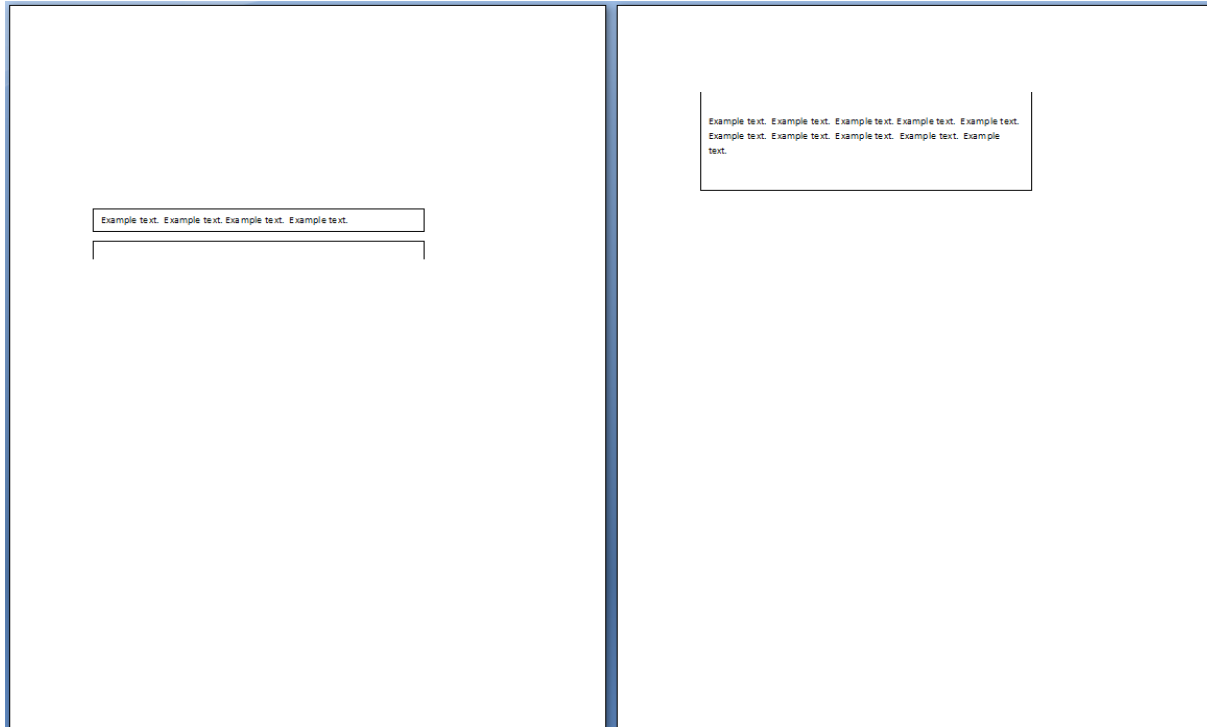
The default presentation would display the page break inline in the frame (breaking the frame into two) but would not actually break the page:



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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.37 [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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.38 [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.

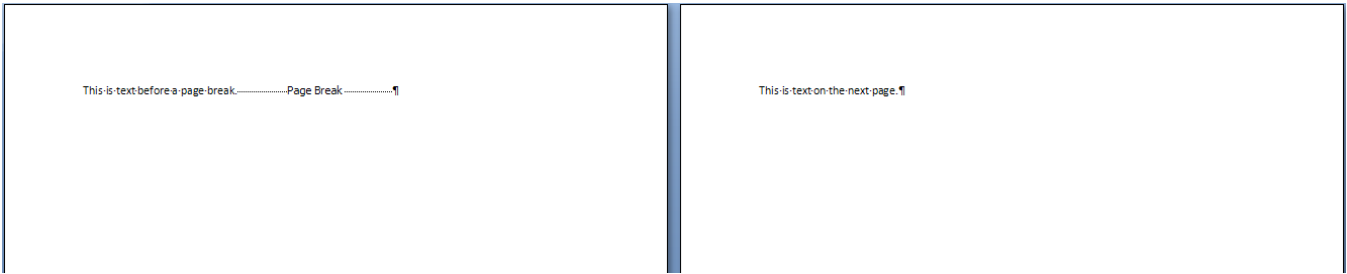
Typically, a page break defined using the `br` element (Part 1, §17.3.3.1) is treated as run-level content, which means that although it delimits the end of the page, if there is no content after it within the current paragraph, that the paragraph shall also end on that page. This element, when present 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>
```

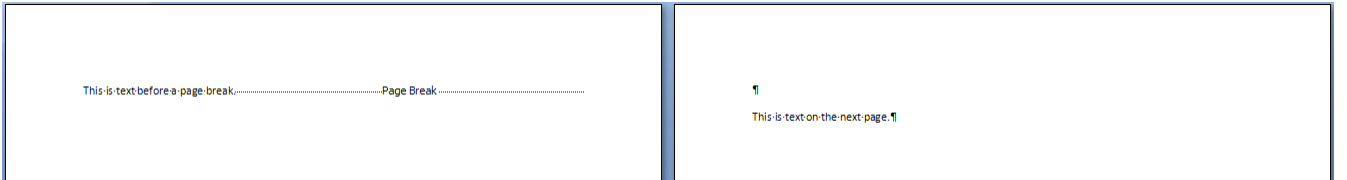
The default presentation would have the text content `This is text on the next page.` as the first line of the second page, as there is no run content after the page break in paragraph one, and therefore no need for a new line on page two (in this image, a graphical illustration of the pilcrow and the 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.39 **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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.40 **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:

This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example text.	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is example	This is example text. This is example text. This is example text. This is example text.
This is example text. This is example text. This is example text. This is	This is example text. This is example text. This is example text. This is example text.
Example text.	

However, if this compatibility setting is turned on:

```

<w:compat>
  <w:suppressBottomSpacing />
</w:compat>

```

Then that constraint must be lifted for the last line on the page (although all other lines are unaffected), resulting in the following output:

[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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.41 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.

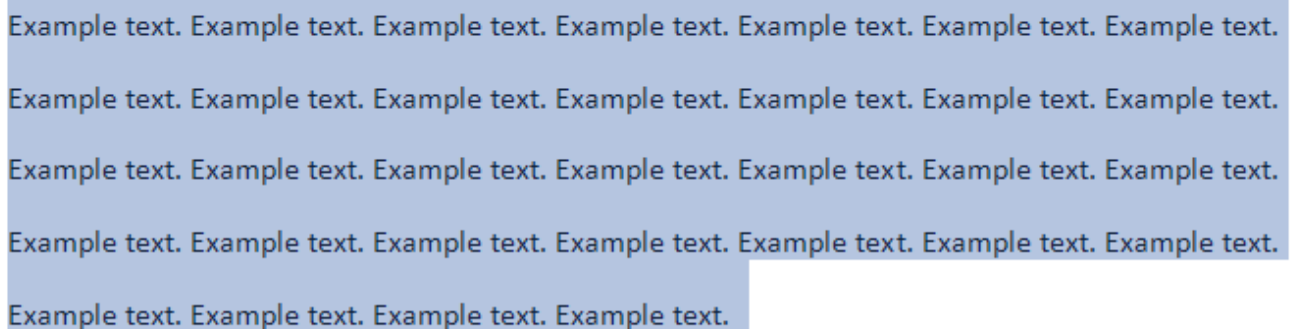
[*Example:* Consider a WordprocessingML document whose first paragraph has a line spacing setting requiring at least 25 points of space per line:

```

<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.

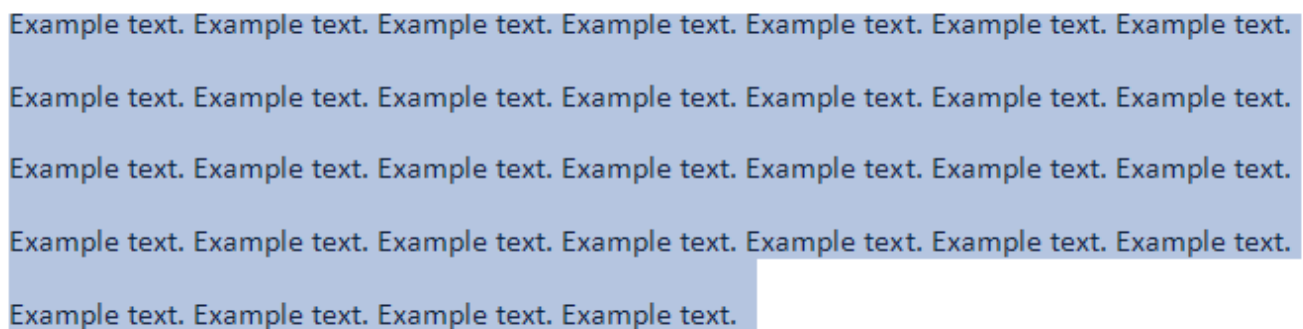
However, if this compatibility setting is turned on:

```

<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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.42 `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.

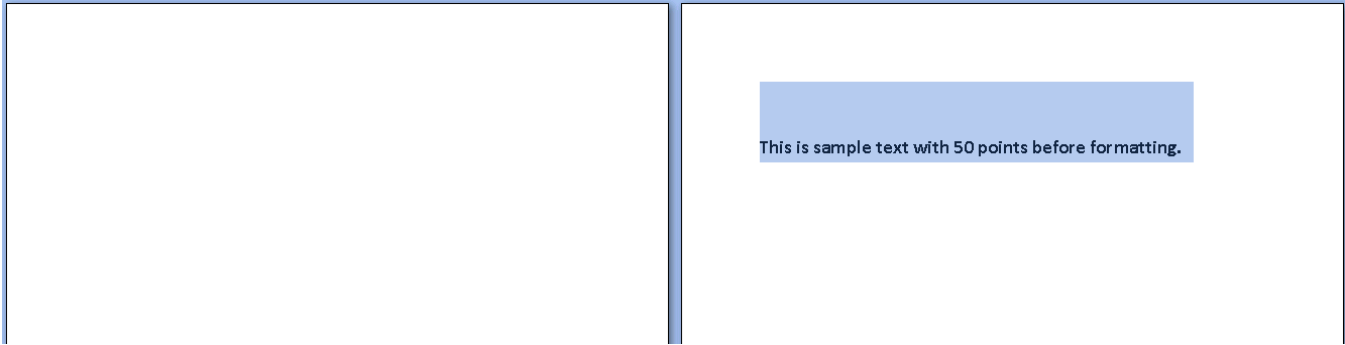
This leads to a situation where the only run content on the page with the page break is the break itself, with all subsequent content on the following page. In this case, applications shall apply the value specified by the spacing element's `before` attribute to the first line on the new page (since 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>
```

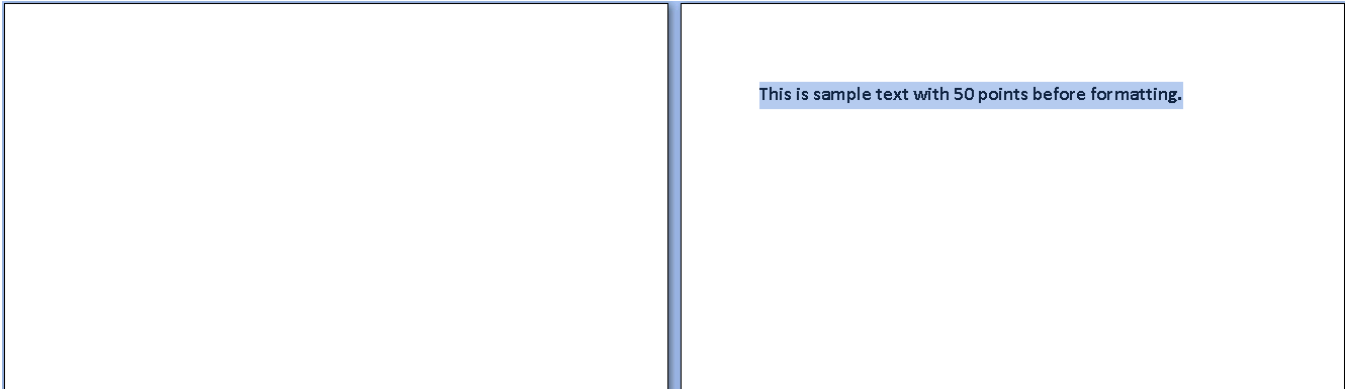
The default presentation would have the necessary amount of space added to the first line on the second page, as the page break was not preceded by any run content (highlighting has been added to the image below in order to illustrate the additional spacing above the first line):



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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.43 `suppressTopSpacing` (Ignore Minimum and Exact Line Height for First Line on Page)

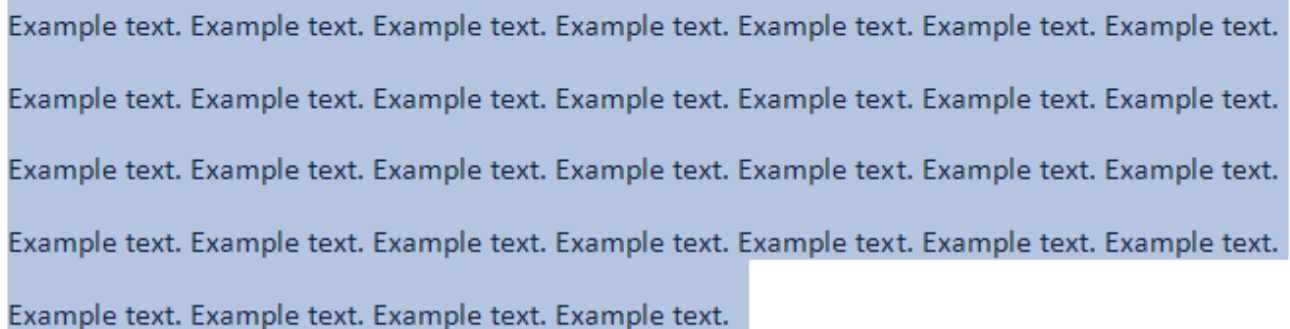
This element specifies whether the minimum line height specified using the spacing element (Part 1, §17.3.1.33) with a `lineRule` attribute value of `atLeast` or `exact` shall be ignored for the first line on each page.

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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.44 suppressTopSpacingWP (Use Static Text Leading)

(The terms *baseline to baseline distance* and *unitsPerEm*, used below, are defined in ISO/IEC 14496-22:2007.)

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 14496-22:2007. This element, when present with a val attribute value of true (or equivalent), specifies that applications should calculate this as follows:

$$BTBD = unitsPerEm + 2pt$$

[*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]*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.45 `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:

<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>
---	---

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.46 `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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.47 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 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.48 [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:
!,.,:;?]}‘’~–||”...:、。 “ 々>》」』】〕! ” ’) , . : ; ?] ` | } ~ ¢
- 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.49 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.50 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]*

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.51 `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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.52 `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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

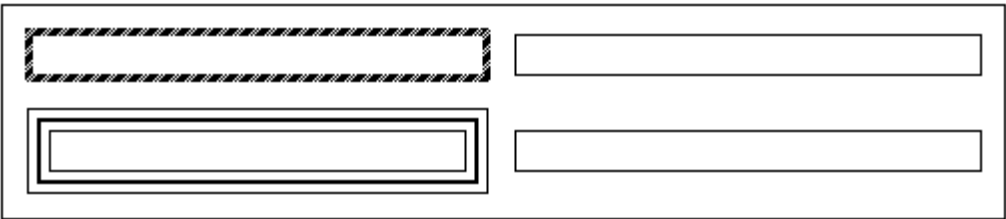
9.7.3.53 useSingleBorderforContiguousCells (Use Simplified Rules For Table Border Conflicts)

This element specifies whether applications should use an alternate simplified algorithm when handling conflicts between adjacent table borders within a table.

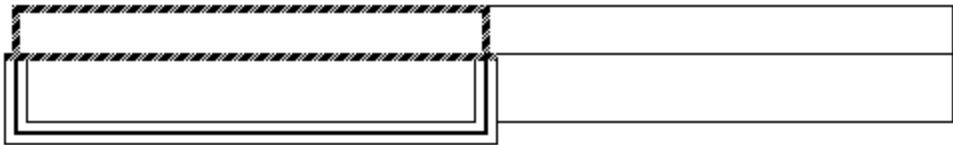
Typically, the conflicts between two adjacent table borders are handled using the conflict resolution algorithm defined in Part 1, §17.4.39 of ECMA-376. This element, when present with a val attribute value of true (or equivalent), specifies that rather than using that algorithm to determine the outcome 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.54 [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 as 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]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.55 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):

平成	19	年	12	月	20	日	1
1.	(

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:

平成	19	年	12	月	20	日	1
1.	(

end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.56 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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.57 **wpSpaceWidth (Use Specific Space Width)**

(The terms *ascent* and *descent* are used as defined in ISO/IEC 14496-22:2007.)

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:

$$w_{space} = \left(\frac{ascent + descent}{3} \right)$$

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*]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.7.3.58 `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.

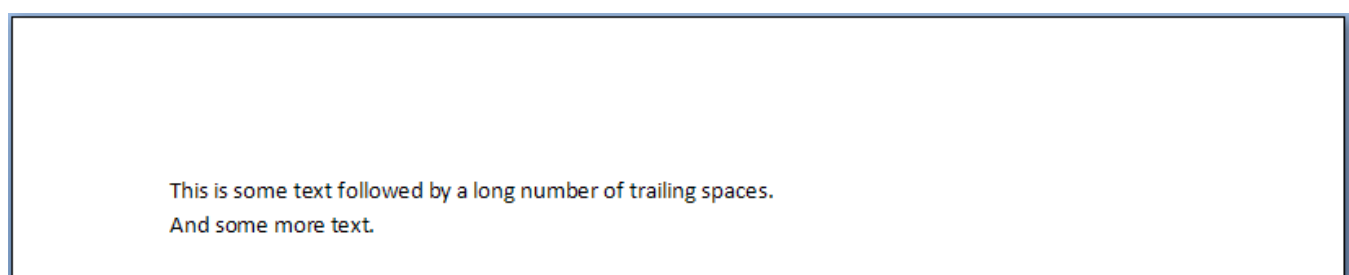
Typically, applications do not line wrap trailing spaces, instead allowing an unbounded number of trailing spaces on a line, with the next non-space character starting at the first character position on the next line. This element, when present with a `val` attribute value of `true` (or equivalent), 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:

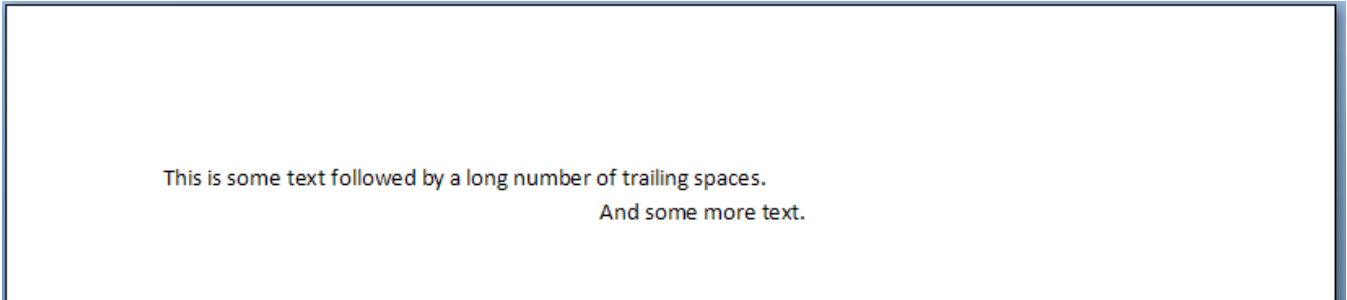


This is some text followed by a long number of trailing spaces.
And some more text.

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:



end example]

Parent Elements
compat (Part 1, §17.15.1.21)

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

9.8 Miscellaneous Topics

9.8.1 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 (§14.1.2.22), which contains within it a single txbxContent element that contains all of the desired WordprocessingML content.

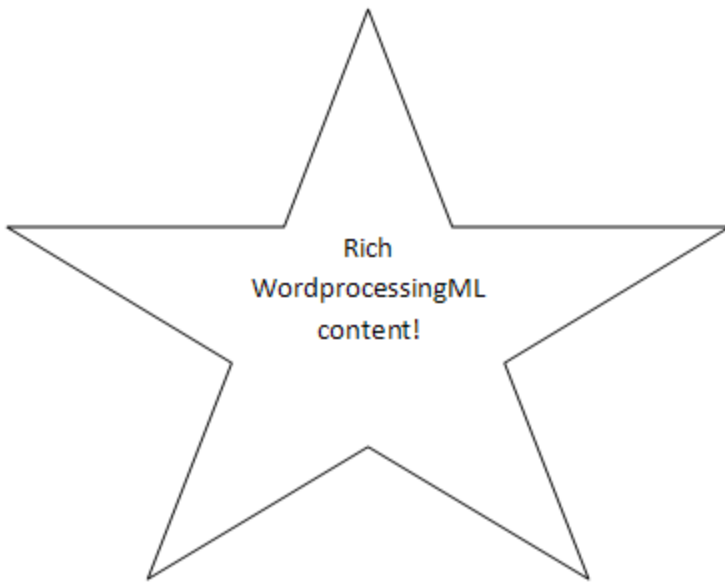
9.8.1.1 txbxContent (Rich Text Box Content Container)

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 (§14.1).

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 (§14.1.2.19) (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>
```

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. *end example*

Parent Elements
textbox (§14.1.2.22)

Child Elements	Subclause
altChunk (Anchor for Imported External Content)	Part 1, §17.17.2.1
bookmarkEnd (Bookmark End)	Part 1, §17.13.6.1
bookmarkStart (Bookmark Start)	Part 1, §17.13.6.2
commentRangeEnd (Comment Anchor Range End)	Part 1, §17.13.4.3
commentRangeStart (Comment Anchor Range Start)	Part 1, §17.13.4.4
customXml (Block-Level Custom XML Element)	Part 1, §17.5.1.6
customXmlDelRangeEnd (Custom XML Markup Deletion End)	Part 1, §17.13.5.4
customXmlDelRangeStart (Custom XML Markup Deletion Start)	Part 1, §17.13.5.5
customXmlInsRangeEnd (Custom XML Markup Insertion End)	Part 1, §17.13.5.6
customXmlInsRangeStart (Custom XML Markup Insertion Start)	Part 1, §17.13.5.7
customXmlMoveFromRangeEnd (Custom XML Markup Move Source End)	Part 1, §17.13.5.8
customXmlMoveFromRangeStart (Custom XML Markup Move Source Start)	Part 1, §17.13.5.9
customXmlMoveToRangeEnd (Custom XML Markup Move Destination Location End)	Part 1, §17.13.5.10
customXmlMoveToRangeStart (Custom XML Markup Move Destination Location Start)	Part 1, §17.13.5.11
del (Deleted Run Content)	Part 1, §17.13.5.14
ins (Inserted Run Content)	Part 1, §17.13.5.18
moveFrom (Move Source Run Content)	Part 1, §17.13.5.22
moveFromRangeEnd (Move Source Location Container - End)	Part 1, §17.13.5.23
moveFromRangeStart (Move Source Location Container - Start)	Part 1, §17.13.5.24
moveTo (Move Destination Run Content)	Part 1,

Child Elements	Subclause
	§17.13.5.25
moveToRangeEnd (Move Destination Location Container - End)	Part 1, §17.13.5.27
moveToRangeStart (Move Destination Location Container - Start)	Part 1, §17.13.5.28
oMath (Office Math)	Part 1, §22.1.2.77
oMathPara (Office Math Paragraph)	Part 1, §22.1.2.78
p (Paragraph)	Part 1, §17.3.1.22
permEnd (Range Permission End)	Part 1, §17.13.7.1
permStart (Range Permission Start)	Part 1, §17.13.7.2
proofErr (Proofing Error Anchor)	Part 1, §17.13.8.1
sdt (Block-Level Structured Document Tag)	Part 1, §17.5.2.29
tbl (Table)	Part 1, §17.4.38

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

9.9 Fields and Hyperlinks

9.9.1 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 §9.9.4.6 *) |
    "INFO", info-category, [field-argument] ;
eq-argument-list=
    expression, {eq-list-separator, expression} ;
eq-list-separator=
    comma | semicolon ;          (* depending on the rules in §9.9.4.6 *)
info-category:
    "AUTHOR" | "COMMENTS" | "CREATEDATE" | "EDITTIME" |
    "FILENAME" | "FILESIZE" | "KEYWORDS" | "LASTSAVEDBY" |
    "NUMCHARS" | "NUMPAGES" | "NUMWORDS" | "PRINTDATE" |
    "REVNUM" | "SAVEDATE" | "SUBJECT" | "TEMPLATE" | "TITLE" ;

```

9.9.2 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
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

Language Code	Description (informative)	BCP 47 Code
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
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

Language Code	Description (informative)	BCP 47 Code
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
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

Language Code	Description (informative)	BCP 47 Code
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
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

Language Code	Description (informative)	BCP 47 Code
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
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

Language Code	Description (informative)	BCP 47 Code
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
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

Language Code	Description (informative)	BCP 47 Code
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
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.	

9.9.3 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*]

9.9.4 Field definitions

9.9.4.1 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]

9.9.4.2 AUTONUMGL

Syntax:

AUTONUMGL [*switches*]

Description: For legal and technical publications, use the nine built-in heading styles to format headings in the document, and then insert an AUTONUMGL field at the beginning of each heading paragraph. The numbers reflect the heading levels that correspond to the heading styles. If an AUTONUMGL 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*.

\e	Removes the trailing separator (period).
\s <i>field-argument</i>	<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:

```
AUTONUMLGL
AUTONUMLGL \* Arabic \s :
AUTONUMLGL \* alphabetic \s " "xxx
AUTONUMLGL \* ROMAN
AUTONUMLGL \e xxx
```

The results are:

```
1.
2:
c xxx
IV.
5xxx
```

end example]

9.9.4.3 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]

9.9.4.4 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:

|||||.....|||||.....|||||.....|||||.....

|||||.....|||||.....

|| | ||

||| ||| |||

Bookmark not defined!

end example]

9.9.4.5 **BIDIOUTLINE**

Syntax:

BIDIOUTLINE

Description: This field is identical to the AUTONUMLGL field (§9.9.4.3), 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 §9.9.4.3.

Switches: None.

9.9.4.6 **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.

$\backslash a$ produces an array using the argument values in *eq-argument-list* (which are in row-major order) and the *field-specific-switches* below:

$\backslash ac$	Alignment is centered in each array column.
$\backslash al$	Alignment is left in each array column.
$\backslash ar$	Alignment is right in each array column.
$\backslash co$ <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.
$\backslash hs$ <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.
$\backslash vs$ <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.

$\backslash b$ 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:

$\backslash bc$ $\backslash char$	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.
$\backslash lc$ $\backslash char$	Uses the character designated by <i>char</i> as the left bracket character.
$\backslash rc$ $\backslash char$	Uses the character designated by <i>char</i> as the right bracket character.

$\backslash d$ 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:

$\backslash ba$ <i>field-argument</i>	Draws to the left (backward) the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .
$\backslash fo$ <i>field-argument</i>	Draws to the right (forward) the integral number of points specified by <i>text</i> in this switch's <i>field-argument</i> .
$\backslash li$	Underlines the space up to the next character.

$\backslash f$ 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.

$\backslash i$ 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:

$\backslash fc$ $\backslash char$	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 (0, 10))`
`EQ \o (0, 0, 0) EQ \o (0, +) EQ \o \ar (0, |, _)`
`EQ \r (2) EQ \r (2, x)`
`a EQ \s \up (2) + b EQ \s \up (2)`
`a EQ \x (+) b a EQ \x \to \le (+) b a EQ \x \bo \ri (+) b`

The results are:

1000 20
 A Sunday
 $\left| -100 \right| \left| \sqrt[3]{a+b} \right|$
 xx xx_____ xx
 $\frac{1}{32} \frac{7}{64}$
 $\int_0^{\infty} x \sum_0^{10x} \prod_0^{5x}$
 $\left\{ \frac{x}{0.34} \right\}_0^5 \left\{ \frac{x}{0.34} \right\}_0^5$
 0, 10
 [0, 10)

0 0 0

$\sqrt{2}$ $\sqrt[2]{x}$

$a^2 + b^2$

$a \boxed{+} b$ $a \overline{+} b$ $a \underline{+} b$

end example]

9.9.4.7 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.

9.9.5 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*

Parent Elements
fldChar (Part 1, §17.16.18)

Attributes	Description
<p>xml:space (Content Contains Significant Whitespace)</p> <p>Namespace: http://www.w3.org/XML/1998/namespace</p>	<p>Specifies how white space should be handled for the contents of this element using the W3C space preservation rules.</p> <p>[<i>Example:</i> Consider the following run contained within a WordprocessingML document:</p> <pre><w:r> <w:t> significant whitespace </w:t> </w:r></pre> <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*]

9.9.6 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*

Parent Elements
fldSimple (Part 1, §17.16.19)

Attributes	Description
xml:space (Content Contains Significant Whitespace) Namespace: http://www.w3.org/XML/1998/namespace	<p>Specifies how white space should be handled for the contents of this element using the W3C space preservation rules.</p> <p>[<i>Example:</i> Consider the following run contained within a WordprocessingML document:</p> <pre><w:r> <w:t> significant whitespace </w:t> </w:r></pre> <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*]

9.10 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.

9.10.1 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 (§9.10.8).

9.10.2 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.

9.10.3 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)
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)

9.10.4 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 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).

9.10.5 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.

9.10.6 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.

9.10.7 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]*`.

Referenced By
cnfStyle@val (§9.3.10); cnfStyle@val (§9.2.1.1); cnfStyle@val (§9.3.9)

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

9.10.8 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.

Referenced By
ST_DecimalNumberOrPercent (Part 1, §17.18.11)

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

10. SpreadsheetML Reference Material

10.1 Table of Contents

This subclause is informative.

10.2	Workbook	173
10.2.1	Additional attribute for fileSharing element (Part 1, §18.2.12)	173
10.2.2	Additional attribute for webPublishing element (Part 1, §18.2.24)	174
10.2.3	Additional attributes for workbookProtection element (Part 1, §18.2.29)	174
10.3	Worksheets	180
10.3.1	Worksheets	180
10.3.1.1	legacyDrawing (Legacy Drawing Reference)	180
10.3.1.2	legacyDrawingHF (Legacy Drawing Reference in Header Footer)	181
10.3.1.3	Additional attribute for dataConsolidate element (Part 1, §18.3.1.29)	181
10.3.1.4	Additional attributes for protectedRange element (Part 1, §18.3.1.71)	182
10.3.1.5	Additional attribute for sheetProtection element (Part 1, §18.3.1.84)	182
10.3.1.6	Additional attribute for sheetProtection element (Part 1, §18.3.1.85)	183
10.3.2	AutoFilter Settings	183
10.3.2.1	Additional attributes for dynamicFilter element (Part 1, §18.3.2.5)	183
10.4	Styles	184
10.4.1	left (Leading Edge Border)	184
10.4.2	right (Trailing Edge Border)	184
10.5	Pivot Tables	185
10.5.1	Pivot Tables	185
10.5.1.1	Additional attribute for pivotCacheDefinition element (Part 1, §18.10.1.67)	185
10.6	External Data Connections	185
10.6.1	Additional attribute for textPr element (Part 1, §18.13.12)	185
10.7	Simple Types	186
10.7.1	Additional enumeration values for ST_PivotAreaType (Part 1, §18.18.58)	186
10.7.2	ST_UnsignedShortHex (Unsigned Short Hex)	186

End of informative text.

10.2 Workbook

10.2.1 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 (§10.7.2).</p>

10.2.2 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>

10.2.3 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-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</p>

Attributes	Description
	<p>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.</p> <p>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; // Load Character Set Tables and determine // double/single byte nature. // This will depend on how the character sets are </pre>

Attributes	Description
	<pre> 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; } byte* AppendDoubleByte(TABLECLASS pTables, wchar_t wcIn, byte* szOutput) { // Look up bytes that we want to append. UINT16 bytesOut = pTables->LookUpDoubleByte(wcIn); </pre>

Attributes	Description
	<pre> // 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 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 </pre>

Attributes	Description
	<pre> 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 points wchar_t LookUpWideChar(UINT16 bytesIn) { // How you access the table depends on your storage mechanism. </pre>

Attributes	Description
	<pre> // First find the DBCSTABLE where the LeadByte matches // the lead (most significant) input byte. if (exists DBCSTABLE{bytesIn >> 8}) { DbcsTable = DBCSTABLE{bytesIn >> 8}; // Look up the line in DbcsTable where the first column // matches the input trail (least significant) byte, // and then return the wchar_t value from the second column. if (exists DbcsTable{bytesIn & 0xFF}) return DbcsTable{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 §9.7.1. 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) { pch = &szPassword[cchPassword]; while (pch-- != szPassword) { wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); </pre>

Attributes	Description
	<pre> wPasswordHash ^= *pch; } wPasswordHash = ((wPasswordHash >> 14) & 0x01) ((wPasswordHash << 1) & 0x7fff); wPasswordHash ^= cchPassword; wPasswordHash ^= (0x8000 ('N' << 8) 'K'); } return(wPasswordHash); } end example] </pre> <p>The possible values for this attribute are defined by the ST_UnsignedShortHex simple type (§10.7.2).</p>
revisionsPassword CharacterSet (Revisions Password Character Set)	<p>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> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
workbookPasswor d (Legacy Workbook Password)	<p>Specifies the legacy hash of the password required for unlocking revisions in this workbook.</p> <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 (§10.7.2).</p>
workbookPasswor dCharacterSet (Workbook Password Character Set)	<p>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> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

10.3 Worksheets

10.3.1 Worksheets

10.3.1.1 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]

Parent Elements
chartsheet (Part 1, §18.3.1.12); dialogsheets (Part 1, §18.3.1.34); worksheet (Part 1, §18.3.1.99)

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]

10.3.1.2 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.

Parent Elements
chartsheet (Part 1, §18.3.1.12); dialogsheets (Part 1, §18.3.1.34); worksheet (Part 1, §18.3.1.99)

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]

10.3.1.3 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.

10.3.1.4 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 (§10.7.2).
securityDescriptor (Security Descriptor)	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. The format of a securityDescriptor is application defined; however, it is recommended that the following format be used for interoperability between implementations: <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>

10.3.1.5 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 (§10.7.2).</p>

10.3.1.6 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)	<p>Specifies the legacy hash of the password required for editing this worksheet.</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 (§10.7.2).</p>

10.3.2 AutoFilter Settings

10.3.2.1 Additional attributes for dynamicFilter element (Part 1, §18.3.2.5)

The following additional attributes can be 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/valIso and shall not use maxVal/maxValIso:</p>

Attributes	Description
	<p>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 value for dynamic filter. See description of maxVal/maxValIso 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>

10.4 Styles

10.4.1 left (Leading Edge Border)

Semantically equivalent to start (Part 1, §18.8.37).

Parent Elements
border (Part 1, §18.8.4)

Child Elements	Subclause
color (Data Bar Color)	Part 1, §18.3.1.15

Attributes	Description
style (Line Style)	<p>The line style for this border.</p> <p>The possible values for this attribute are defined by the ST_BorderStyle simple type (Part 1, §18.18.3).</p>

[*Note*: The W3C XML Schema definition of this element's content model ([CT_BorderPr](#)) is located in §A.2. *end note*]

10.4.2 right (Trailing Edge Border)

Semantically equivalent to end (Part 1, §18.8.16).

Parent Elements
border (Part 1, §18.8.4)

Child Elements	Subclause
color (Data Bar Color)	Part 1, §18.3.1.15

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*]

10.5 Pivot Tables

10.5.1 Pivot Tables

10.5.1.1 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 Refreshed Date)	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 refresh the cache. If refreshedDateIso and refreshedDate are both present, refreshedDateIso shall take precedence. The possible values for this attribute are defined by the W3C XML Schema double datatype.

10.6 External Data Connections

10.6.1 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	Code page associated with the text file. This attribute is used only for backwards

Attributes	Description
Page)	<p>compatibility, and is ignored if the characterSet attribute is present.</p> <p>[<i>Note:</i> There are a number of code page technologies. One example of potential values can be found at: http://www.unicode.org/Public/MAPPINGS <i>end note</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>

10.7 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.

10.7.1 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.

10.7.2 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).

Referenced By
fileSharing@reservationPassword (§10.2.1) ; protectedRange@password (§10.3.1.4) ; sheetProtection@password (§10.3.1.5) ; sheetProtection@password (§10.3.1.6) ; workbookProtection@revisionsPassword (§10.2.3) ; workbookProtection@workbookPassword (§10.2.3)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_UnsignedShortHex](#)) is located in §A.2. *end note*]

11. PresentationML Reference Material

11.1 Table of Contents

This subclause is informative.

11.2	Presentation.....	187
11.2.1	Presentation Properties.....	187
11.2.1.1	htmlPubPr (HTML Publishing Properties).....	187
11.2.1.2	webPr (Web Properties)	189
11.2.1.3	Additional attributes for modifyVerifier element (Part 1, §19.2.1.19)	190
11.3	Slides	196
11.3.1	Embedded Objects	196
11.3.1.1	Additional attribute for control element (Part 1, §19.3.2.1).....	196
11.3.1.2	Additional attribute for oleObj element (Part 1, §19.3.2.4).....	196
11.4	Simple Types	197
11.4.1	ST_WebColorType (HTML Slide Navigation Control Colors)	197
11.4.2	ST_WebEncoding (Web Encoding)	197
11.4.3	ST_WebScreenSize (HTML/Web Screen Size Target)	197

End of informative text.

11.2 Presentation

11.2.1 Presentation Properties

11.2.1.1 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.

Parent Elements
presentationPr (Part 1, §19.2.1.27)

Child Elements	Subclause
custShow (Custom Show)	Part 1, §19.2.1.5
extLst (Extension List)	Part 1, §19.2.1.12
sldAll (All Slides)	Part 1,

Child Elements	Subclause
	§19.2.1.32
sldRg (Slide Range)	Part 1, §19.2.1.38

Attributes	Description										
id (Publish Path) Namespace: .../officeDocument /2006/relationships	Specifies the path that should be used when publishing. 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)	<p>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.</p> <p>The following reserved values and their targets are listed below:</p> <table border="1"> <thead> <tr> <th>Value</th><th>Target</th></tr> </thead> <tbody> <tr> <td>W3C XHTML+CSS1</td><td>W3C XHTML 1.0 + CSS 1</td></tr> <tr> <td>W3C HTML4+CSS1</td><td>W3C HTML 4.01 + CSS 1</td></tr> <tr> <td>W3C XHTML+CSS2</td><td>W3C XHTML 1.0 + CSS 2</td></tr> <tr> <td>W3C HTML4+CSS2</td><td>W3C HTML 4.01 + CSS 2</td></tr> </tbody> </table> <p>[<i>Example:</i> For example, consider the following set of HTML publishing settings:</p> <pre><p:htmlPubPr ... target="W3C HTML4+CSS2"> ... </p:htmlPubPr></pre> <p>The target attribute explicitly declares that any web page generated from this document should target the W3C HTML4+CSS2 profile. <i>end example</i>]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	Value	Target	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
Value	Target										
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										
title (HTML Output)	Specifies a title for the HTML output file.										

Attributes	Description
Title)	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*]

11.2.1.2 webPr (Web Properties)

This element specifies all general output properties that pertain to generating a web format version of the presentation document.

Parent Elements
presentationPr (Part 1, §19.2.1.27)

Child Elements	Subclause
extLst (Extension List)	Part 1, §19.2.1.12

Attributes	Description
allowPng (Allow PNG in HTML output)	Specifies whether to allow the output of PNG format pictures in the HTML document. The possible values for this attribute are defined by the W3C XML Schema boolean datatype.
clr (Slide Navigation Colors for HTML output)	Specifies the color constraints that are to be used when generating HTML output. The possible values for this attribute are defined by the ST_WebColorType simple type (§11.4.1).
encoding (Encoding for HTML output)	Specifies the particular HTML character set encoding that should be used when generating output. The possible values for this attribute are defined by the ST_WebEncoding simple type (§11.4.2).
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 (§11.4.3).
organizeInFolders (Organize HTML output in folders)	Specifies whether the supporting output files should be automatically organized into a folder.

Attributes	Description
	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*]

11.2.1.3 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)	<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. <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 PresentationML document with the following information stored in its protection element:</i></p>

Attributes	Description
	<pre data-bbox="456 285 1192 384"><... p:algIdExt="0000000A" p:algIdExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p data-bbox="415 426 1435 489">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 data-bbox="415 531 1455 594">The possible values for this attribute are defined by the W3C XML Schema unsignedInt datatype.</p>
algIdExtSource (Algorithm Extensibility Source)	<p data-bbox="415 615 1425 678">Specifies the application which defined the algorithm value specified by the algIdExt attribute.</p> <p data-bbox="415 720 1474 783"><i>[Example: Consider a PresentationML document with the following information stored in one its protection element:</i></p> <pre data-bbox="456 831 1192 930"><... p:algIdExt="0000000A" p:algIdExtSource="futureCryptography" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p data-bbox="415 972 1474 1035">The algIdExtSource attribute value of futureCryptography specifies that the algorithm used here was published by the futureCryptography application. <i>end example</i>]</p> <p data-bbox="415 1077 1377 1140">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
cryptAlgorithmClass (Cryptographic Algorithm Class)	<p data-bbox="415 1155 1446 1260">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 data-bbox="415 1302 1474 1365"><i>[Example: Consider a PresentationML document with the following information stored in its protection element:</i></p> <pre data-bbox="456 1413 1192 1539"><... p:cryptAlgorithmClass="hash" p:cryptAlgorithmType="typeAny" p:cryptAlgorithmSid="1" p:hashData="9oN7nWkCAyEZib1RomSJTjmPpCY=" /></pre> <p data-bbox="415 1581 1442 1644">The cryptAlgorithmClass attribute value of hash specifies that the algorithm used for the password is a hashing algorithm. <i>end example</i>]</p> <p data-bbox="415 1686 1370 1749">The possible values for this attribute are defined by the ST_AlgorithmClass simple type (§15.1.2.1).</p>
cryptAlgorithmSid (Cryptographic Hashing Algorithm)	<p data-bbox="415 1770 1471 1873">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>

Attributes	Description																																
	<p>The possible values for this attribute shall be interpreted as follows:</p> <table data-bbox="415 317 1349 1127"> <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="453 1272 1192 1407"><... 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 datatype.</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>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>																																

Attributes	Description
	<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 (§15.1.2.2).</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> <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 (§15.1.2.4).</p>
cryptProviderType Ext (Cryptographic Provider Type)	<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>

Attributes	Description
Extensibility)	<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>
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 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> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
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</p>

Attributes	Description
	<p>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> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary datatype.</p>
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</p>

Attributes	Description
	<p>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>

11.3 Slides

11.3.1 Embedded Objects

11.3.1.1 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>The possible values for this attribute are defined by the ST_ShapeID simple type (Part 1, §20.1.10.55).</p>

11.3.1.2 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>The possible values for this attribute are defined by the ST_ShapeID simple type (Part 1, §20.1.10.55).</p>

11.4 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.

11.4.1 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.

Referenced By
webPr@clr (§11.2.1.2)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WebColorType](#)) is located in §A.3. *end note*]

11.4.2 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.

Referenced By
webPr@encoding (§11.2.1.2)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WebEncoding](#)) is located in §A.3. *end note*]

11.4.3 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 720x515)	Screen size is 720x512 pixels
800x600 (HTML/Web Size Enumeration 800x600)	Screen size is 800x600 pixels

Referenced By
webPr@imgSz (§11.2.1.2)

[Note: The W3C XML Schema definition of this simple type's content model (ST_WebScreenSize) is located in §A.3. *end note*]

12. DrawingML - Framework Reference Material

12.1 DrawingML - Main

12.1.1 Table of Contents

This subclause is informative.

12.1.2 Simple Types	199
12.1.2.1 Additional member types for the union in ST_FixedPercentage (Part 1, §20.1.10.24)	199
12.1.2.2 Additional member types for the union in ST_Percentage (Part 1, §20.1.10.40)	199
12.1.2.3 Additional member types for the union in ST_PositiveFixedPercentage (Part 1, §20.1.10.45)	200
12.1.2.4 Additional member types for the union in ST_PositivePercentage (Part 1, §20.1.10.46)	200
12.1.2.5 Additional member types for the union in ST_TextFontScalePercentOrPercentString (Part 1, §20.1.10.67)	200
12.1.2.6 Additional member types for the union in ST_TextSpacingPercentOrPercentString (Part 1, §20.1.10.77)	200
12.1.2.7 ST_FixedPercentageDecimal (Fixed Percentage)	200
12.1.2.8 ST_PositiveFixedPercentageDecimal (Positive Fixed Percentage)	201
12.1.2.9 ST_PositivePercentageDecimal (Positive Percentage as Decimal Number).....	201
12.1.2.10 ST_TextFontScalePercent (Text Font Scale Percentage)	201
12.1.2.11 ST_TextSpacingPercent (Text Spacing Percent)	202

End of informative text.

12.1.2 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.

12.1.2.1 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 (§12.1.2.7).

12.1.2.2 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 1, §20.1.10.41).

12.1.2.3 **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 (§12.1.2.8).

12.1.2.4 **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 (§12.1.2.9).

12.1.2.5 **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 (§12.1.2.10).

12.1.2.6 **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 (§12.1.2.11).

12.1.2.7 **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.

Referenced By
ST_FixedPercentage (Part 1, §20.1.10.24)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_FixedPercentageDecimal](#)) is located in §A.4.1. *end note*]

12.1.2.8 [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.

Referenced By
ST_PositiveFixedPercentage (Part 1, §20.1.10.45); ST_TLTimeAnimateValueTime (Part 1, §19.7.39)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_PositiveFixedPercentageDecimal](#)) is located in §A.4.1. *end note*]

12.1.2.9 [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.

Referenced By
ST_PositivePercentage (Part 1, §20.1.10.46)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_PositivePercentageDecimal](#)) is located in §A.4.1. *end note*]

12.1.2.10 [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.

Referenced By
ST_TextFontScalePercentOrPercentString (Part 1, §20.1.10.67)

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST_TextFontScalePercent](#)) is located in §A.4.1. *end note*]

12.1.2.11 **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.

Referenced By
ST_TextSpacingPercentOrPercentString (Part 1, §20.1.10.77)

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST_TextSpacingPercent](#)) is located in §A.4.1. *end note*]

12.2 **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*]

12.2.1 **Table of Contents**

This subclause is informative.

12.2.2 Basics	202
12.2.2.1 legacyDrawing (Legacy Drawing Object)	203

End of informative text.

12.2.2 **Basics**

Legacy Compatibility is part of the shape definitions and properties of the DrawingML framework.

12.2.2.1 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><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 §A.4.3. end note]

13. DrawingML - Components Reference Material

13.1 DrawingML - Charts

13.1.1 Table of Contents

This subclause is informative.

13.1.2 Elements	204
13.1.2.1 legacyDrawingHF (Legacy Drawing for Headers and Footers)	204

End of informative text.

13.1.2 Elements

13.1.2.1 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.

Parent Elements
printSettings (Part 1, §21.2.2.148)

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*]

14. VML Reference Material

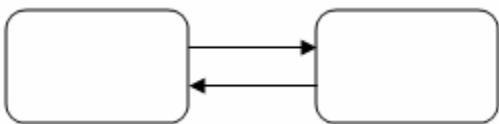
[No documentation has been entered for this section heading.]

14.1 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).

14.1.1 Table of Contents

This subclause is informative.

14.1.2 Elements	207
14.1.2.1 arc (Arc Segment)	207
14.1.2.2 background (Document Background).....	235
14.1.2.3 curve (Bezier Curve)	238
14.1.2.4 f (Single Formula).....	266
14.1.2.5 fill (Shape Fill Properties).....	270
14.1.2.6 formulas (Set of Formulas)	282
14.1.2.7 group (Shape Group)	282
14.1.2.8 h (Shape Handle)	306
14.1.2.9 handles (Set of Handles).....	310
14.1.2.10 image (Image File)	310
14.1.2.11 imagedata (Image Data)	341
14.1.2.12 line (Line)	349
14.1.2.13 oval (Oval).....	377
14.1.2.14 path (Shape Path)	403
14.1.2.15 polyline (Multiple Path Line)	412
14.1.2.16 rect (Rectangle)	439
14.1.2.17 roundrect (Rounded Rectangle)	466
14.1.2.18 shadow (Shadow Effect)	493

14.1.2.19	shape (Shape Definition)	499
14.1.2.20	shapetype (Shape Template).....	529
14.1.2.21	stroke (Line Stroke Settings).....	557
14.1.2.22	textbox (Text Box).....	570
14.1.2.23	textpath (Text Layout Path).....	582
14.1.3	Simple Types	595
14.1.3.1	ST_EditAs (Shape Grouping Types).....	595
14.1.3.2	ST_Ext (VML Extension Handling Behaviors).....	595
14.1.3.3	ST_FillMethod (Gradient Fill Computation Type).....	596
14.1.3.4	ST_FillType (Shape Fill Type)	597
14.1.3.5	ST_ImageAspect (Image Scaling Behavior).....	598
14.1.3.6	ST_ShadowType (Shadow Type).....	598
14.1.3.7	ST_StrokeArrowLength (Stroke Arrowhead Length)	599
14.1.3.8	ST_StrokeArrowType (Stroke Arrowhead Type)	599
14.1.3.9	ST_StrokeArrowWidth (Stroke Arrowhead Width)	600
14.1.3.10	ST_StrokeEndCap (Stroke End Cap Type)	601
14.1.3.11	ST_StrokeJoinStyle (Line Join Type).....	601
14.1.3.12	ST_StrokeLineStyle (Stroke Line Style)	602

End of informative text.

14.1.2 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*]

14.1.2.1 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6



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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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>Specifies the bottom 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: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>
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:	<p>Specifies whether the shape is a graphical bullet. Default is false.</p> <p>[Example:</p>

Attributes	Description
urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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. <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 (§14.2.3.3).</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>[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>


Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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="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 (§14.2.3.3).</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>[<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 (§15.1.2.3).</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></pre> <pre><v:shape ... style="top:1;left:1; width:50;height:100"> </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>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 (§14.2.2.3) 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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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>

Attributes	Description
	<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>  <p><i>end example]</i></p>

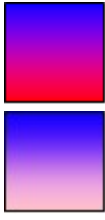
Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	Specifies the kind of automatic layout to apply to the diagram node. This is only meaningful 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. <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 (§14.2.3.10).
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 (§14.2.3.10).
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:	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" ... ></pre>

Attributes	Description
urn:schemas-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 ST_TrueFalse simple type (§15.1.2.5).</p>
endAngle (Ending Angle)	<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema decimal datatype.</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 (§15.1.2.3).</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" ...></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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§15.1.2.5).</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>



Attributes	Description
	The possible values for this attribute are defined by the ST_HrAlign simple type (§14.2.3.16).
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 (§15.1.2.5).
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>(§15.1.2.5).</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 (§14.2.3.17).</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 (§15.1.2.5).</p>
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-</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>

Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§15.1.2.6).</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 ST_TrueFalse simple type (§15.1.2.5).</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 ST_TrueFalse simple type (§15.1.2.5).</p>
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. [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> <div data-bbox="451 1661 779 1879">  <p>opacity="1"</p> <p>opacity=".25"</p> </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>
<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 (§15.1.2.5).</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 (§15.1.2.5).</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:</p>	<p>Specifies an optional string that an application can use to identify the particular shape. Default is no value.</p>

Attributes	Description
urn:schemas-microsoft-com:office:office	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.
startAngle (Starting Angle)	Specifies an angle that defines the starting point of the arc. The angle is measured in degrees clockwise from the vertical. Default is 0. <i>[Example: This arc begins in the upper-right quadrant:</i> <pre><v:arc ... startangle="45" ... > </v:arc></pre> <i>end example]</i> The possible values for this attribute are defined by the W3C XML Schema decimal datatype.
strokecolor (Shape Stroke Color)	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 (§14.1.2.21) 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. <i>[Example:</i> <pre><v:shape ... strokecolor="red" ...> </v:shape></pre>  <i>end example]</i> The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).
stroked (Shape Stroke Toggle)	Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) defines other strokes. The on attribute of the stroke element

Attributes	Description
	<p>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 (§15.1.2.5).</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 (§14.1.2.21) 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>
<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>

Attributes	Description									
	<i>end example]</i>									
	<table> <tr> <th data-bbox="415 317 662 365">Property</th><th data-bbox="662 317 1482 365">Description</th></tr> <tr> <td data-bbox="415 365 662 632">flip</td><td data-bbox="662 365 1482 632"> <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="415 632 662 1041">height</td><td data-bbox="662 632 1482 1041"> <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 1041 662 1524">left</td><td data-bbox="662 1041 1482 1524"> <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 1524 662 1890">margin-bottom</td><td data-bbox="662 1524 1482 1890"> <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. </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> <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.
	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

Attributes	Description	
		<ul style="list-style-type: none"> • 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 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</p>

Attributes	Description	
		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.
	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:

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.
	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 (§14.1.2.22):		
	Property	Description
	direction	Specifies the direction of the text in the textbox. Default is ltr. This property is superseded by the mso-direction-alt property

Attributes	Description
	<p>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. • <code>vertical-ideographic</code> - Ideographic text is displayed vertically. • <code>horizontal-ideographic</code> - Ideographic text is displayed horizontally.
<code>mso-direction-alt</code>	<p>Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code>.</p>
<code>mso-fit-shape-to-text</code>	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code>.</p>
<code>mso-fit-text-to-shape</code>	<p>Specifies whether the text stretches to fit the textbox. Default is <code>false</code>.</p>
<code>mso-layout-flow-alt</code>	<p>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>.</p>
<code>mso-next-textbox</code>	<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>
<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>	<p>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>.</p>
<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</p>

Attributes	Description	
		<p>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 (§14.1.2.23):	

Attributes	Description																		
		<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="6">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	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:																		

Attributes	Description	
		<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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top • position • rotation • top 		

Attributes	Description																
	<ul style="list-style-type: none"> • 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="415 583 1482 1220"> <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>																

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 (§15.1.2.5).</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 (§15.1.2.5).</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_Arc](#)) is located in §A.6.1. *end note*]

14.1.2.2 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 (§14.1.2.5), or images.

[*Example:* The following shades the page background a pale red:

```
<v:background fillcolor="#c0504d">
</v:background>
```

This uses the fill element (§14.1.2.5) to create a gradient background fill:

```
<v:background>
  <v:fill type="gradient" color="#c0504d" color2="#f0504d" angle="45"/>
</v:background>
```


end example]

Parent Elements
background (Part 1, §17.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
fill (Shape Fill Properties)	§14.1.2.5

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 auto, which uses o:bwnormal for normal black-and-white rendering and o:bwpure for pure black-and-white rendering</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 (§14.2.3.3).</p>
bwnormal (Normal)	Specifies the black-and-white mode for normal black-and-white output devices. Default

Attributes	Description
<p>Black-and-White Mode)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>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 (§14.2.3.3).</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 (§14.2.3.3).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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" ... > </v:shape></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p>

Attributes	Description
<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 data-bbox="451 426 792 527"><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 (§15.1.2.5).</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 data-bbox="451 1066 889 1136"><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>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 data-bbox="459 1434 630 1661" style="list-style-type: none"> • 544,376 • 640,480 • 720,512 • 800,600 • 1024,768 • 1152,862 <p>The possible values for this attribute are defined by the ST_ScreenSize simple type (§14.2.3.23).</p>

[*Note*: The W3C XML Schema definition of this element's content model ([CT_Background](#)) is located in §A.6.1.
end note]

14.1.2.3 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3

Child Elements	Subclause
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>


Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
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)	<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)	<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 <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 (§15.1.2.5).</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 (§15.1.2.5).</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 <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>


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 (§14.2.3.3).</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 (§14.2.3.3).</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 (§14.2.3.3).</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 (§15.1.2.3).</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> <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 (§14.2.2.3) 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 (§15.1.2.5).</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>[<i>Example:</i></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 (§15.1.2.5).</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 (§14.2.3.7).</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>
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><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
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"> </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>
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 (§14.2.3.10).</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 (§14.2.3.10).</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</p>

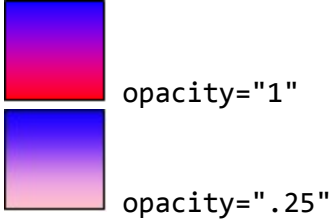
Attributes	Description
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>datatype.</p> <p>Specifies that an event message is sent when a shape is double-clicked. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:doubleclicknotify="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 (§15.1.2.3).</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 (§15.1.2.5).</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 (§15.1.2.5).</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><i>end example]</i></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 <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 (§15.1.2.5).</p>
<p>hralign (Horizontal Rule Alignment)</p>	<p>Specifies the alignment of a horizontal rule. Default is <code>left</code>.</p>

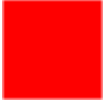

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<p>[Example:</p> <pre><v:shape ... o:hralign="center" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§14.2.3.16).</p>
href (Hyperlink Target)	<p>Specifies 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 (§15.1.2.5).</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)	<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
Namespace: 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 (§15.1.2.5).</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 (§14.2.3.17).</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</p>

Attributes	Description
<p>ole (Embedded Object Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>(§15.1.2.5).</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 (§15.1.2.6).</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 (§15.1.2.5).</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 (§15.1.2.5).</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>

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>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 (§15.1.2.5).</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 (§15.1.2.5).</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>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema integer datatype.
spid (Optional String) Namespace: urn:schemas-microsoft-com:office:office	Specifies an optional string that an application can use to identify the particular shape. 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 (§14.1.2.21) 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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) 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><tr><td>flip</td><td>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-	Specifies relative horizontal position data for objects in

Attributes	Description	
	horizontal-relative	<p>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 the origin.</p>

Attributes	Description	
	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 (§14.1.2.22):</p> <table> <tr> <th data-bbox="415 352 662 401">Property</th><th data-bbox="662 352 1484 401">Description</th></tr> <tr> <td data-bbox="415 401 662 667">direction</td><td data-bbox="662 401 1484 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="415 667 662 1041">layout-flow</td><td data-bbox="662 667 1484 1041"> <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="415 1041 662 1152">mso-direction-alt</td><td data-bbox="662 1041 1484 1152"> <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="415 1152 662 1264">mso-fit-shape-to-text</td><td data-bbox="662 1152 1484 1264"> <p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p> </td></tr> <tr> <td data-bbox="415 1264 662 1375">mso-fit-text-to-shape</td><td data-bbox="662 1264 1484 1375"> <p>Specifies whether the text stretches to fit the textbox. Default is false.</p> </td></tr> <tr> <td data-bbox="415 1375 662 1528">mso-layout-flow-alt</td><td data-bbox="662 1375 1484 1528"> <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="415 1528 662 1640">mso-next-textbox</td><td data-bbox="662 1528 1484 1640"> <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="415 1640 662 1864">mso-rotate</td><td data-bbox="662 1640 1484 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> </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">topmiddlebottomtop-centermiddle-centerbottom-centertop-baselinebottom-baselinetop-center-baselinebottom-center-baseline											
	The following properties are only used by the textpath element (§14.1.2.23):												
	<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">normalitalicoblique - 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:</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">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:
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">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:												

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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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 (§15.1.2.5).</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 (§15.1.2.5).</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]*

14.1.2.4 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 (§14.1.2.19)), 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]

Parent Elements
formulas (§14.1.2.6)

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> </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.
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.																		

Attributes	Description	
	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 $\times 2^{16}$. 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</p>		

Attributes	Description																								
	<p>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 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="418 743 1208 1877"> <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 (§14.1.2.14)).</td></tr> <tr> <td>ylimo</td><td>The y value of the limo attribute (see also the path element (§14.1.2.14)).</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 (§14.1.2.21).</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 (§14.1.2.5).</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</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 (§14.1.2.14)).	ylimo	The y value of the limo attribute (see also the path element (§14.1.2.14)).	hasstroke	1 if the shape has a stroke operation, 0 if it does not, as determined by the on attribute of the stroke element (§14.1.2.21).	hasfill	1 if the shape has a fill operation, 0 if it does not, as determined by the on attribute of the fill element (§14.1.2.5).	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
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 (§14.1.2.14)).																								
ylimo	The y value of the limo attribute (see also the path element (§14.1.2.14)).																								
hasstroke	1 if the shape has a stroke operation, 0 if it does not, as determined by the on attribute of the stroke element (§14.1.2.21).																								
hasfill	1 if the shape has a fill operation, 0 if it does not, as determined by the on attribute of the fill element (§14.1.2.5).																								
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																								

Attributes	Description	
		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).
	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*]



14.1.2.5 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.

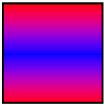
Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); background (§14.1.2.2); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)

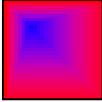
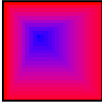
Child Elements	Subclause
fill (Shape Fill Extended Properties)	§14.2.2.13


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>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
althref (Alternate Image Reference Location) Namespace: urn:schemas- microsoft- com:office:office	<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>
angle (Gradient Angle)	<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"</pre>



Attributes	Description
	<pre>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>
aspect (Image Aspect Ratio)	<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. • 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>[<i>Example:</i> 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 (§14.1.3.5).</p>
color (Primary Color)	<p>Specifies the main fill color; functions the same as the fillcolor attribute of the shape element (§14.1.2.19). 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>[<i>Example:</i> The shape is blue:</p> <pre><v:shape ... fillcolor="red" ... > <v:fill color="blue"/> </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 (§15.1.2.3).</p>
<p>color2 (Secondary Color)</p>	<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> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</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 (§14.1.2.19) or the color attribute of the fill element (§14.1.2.5), is used at the 0% endpoint. The secondary color, specified by the color2 attribute of the fill element (§14.1.2.5), is used at the 100% endpoint. The numeric values 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. <i>end example]</i></p> <p>[Example: The shape is filled with a horizontal gradient colored, from bottom to top, red, yellow, green, blue:</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>


Attributes	Description
<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 (§15.1.2.5).</p>
<p>focus (Gradient Center)</p>	<p>Specifies the center starting position of a gradient. Values range from 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> <p>[Example: 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>[Example: 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>

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>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>[Example: 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. 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</p>










Attributes	Description
<p>id (Relationship to Part)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>datatype.</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 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>method (Gradient Fill Method)</p>	<p>Specifies the method used to generate the transition from color to color2 in a gradient fill. Default is sigma.</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 (§14.1.3.3).</p>


Attributes	Description
on (Fill Toggle)	<p>Specifies whether to fill the shape. Default is true. This attribute overrides the shape's fill attribute.</p> <p>[<i>Example:</i> 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 (§15.1.2.5).</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" 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> <div data-bbox="451 1100 779 1318">  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>
opacity2 (Secondary Color Opacity) Namespace: urn:schemas- microsoft- com:office:office	<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>

Attributes	Description
	<div data-bbox="451 243 553 348" data-label="Image"> </div> <div data-bbox="570 321 764 352" data-label="Text"> <p>opacity2="1"</p> </div> <div data-bbox="451 354 553 459" data-label="Image"> </div> <div data-bbox="570 430 795 462" data-label="Text"> <p>opacity2=".25"</p> </div> <p data-bbox="414 501 576 533"><i>end example]</i></p> <p data-bbox="414 573 1375 636">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
origin (Fill Image Origin)	<p data-bbox="414 657 1479 827">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: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p data-bbox="414 869 1463 932"><i>[Example: The origin of the image is 25% to the right and 25% above the image's top left corner:</i></p> <pre data-bbox="451 974 1045 1073"> <v:fill type="tile" src="myimage.gif" origin="0.25,-0.25"> </v:fill> </pre> <div data-bbox="414 1108 516 1213" data-label="Image"> </div> <p data-bbox="414 1253 576 1285"><i>end example]</i></p> <p data-bbox="414 1325 1375 1388">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
position (Fill Image Position)	<p data-bbox="414 1409 1455 1545">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 data-bbox="414 1585 1435 1650"><i>[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:</i></p> <pre data-bbox="451 1692 1045 1791"> <v:fill type="tile" src="myimage.gif" position="0.25,0.25"> </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>
recolored (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> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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: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>

Attributes	Description
	<div data-bbox="451 247 592 388" data-label="Image"> </div> <div data-bbox="609 359 816 388" data-label="Text"> <p>rotate="true"</p> </div> <div data-bbox="451 394 592 535" data-label="Image"> </div> <div data-bbox="609 506 833 535" data-label="Text"> <p>rotate="false"</p> </div> <p data-bbox="415 579 574 609"><i>end example]</i></p> <p data-bbox="415 648 1390 714">The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
size (Fill Image Size)	<p data-bbox="415 732 1266 762">Specifies the size of the fill image. Default is the native image pixel size.</p> <p data-bbox="415 804 1105 833"><i>[Example: The image is reduced in size disproportionately:</i></p> <pre data-bbox="451 875 1045 972"><v:fill type="tile" src="myimage.gif" size="25pt,15pt"> </v:fill></pre> <div data-bbox="415 1010 516 1110" data-label="Image"> </div> <p data-bbox="415 1152 574 1182"><i>end example]</i></p> <p data-bbox="415 1222 1375 1287">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
src (Fill Image Source)	<p data-bbox="415 1304 992 1333">Specifies the URL specifying the fill image to use.</p> <p data-bbox="415 1375 535 1404"><i>[Example:</i></p> <pre data-bbox="451 1446 951 1509"><v:fill ... src="myimage.gif" ... > </v:fill></pre> <p data-bbox="415 1551 574 1581"><i>end example]</i></p> <p data-bbox="415 1621 1375 1686">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
title (Title) Namespace: urn:schemas- microsoft-	<p data-bbox="415 1707 1375 1772">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 data-bbox="415 1814 535 1843"><i>[Example:</i></p>

Attributes	Description				
com:office:office	<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> <pre><v:fill color="red" color2="blue" type="solid"> </v:fill></pre> <div style="display: flex; align-items: center;">  <div style="margin-left: 10px;">type="solid"</div> </div> <div style="display: flex; align-items: center;">  <div style="margin-left: 10px;">type="gradient"</div> </div> <div style="display: flex; align-items: center;">  <div style="margin-left: 10px;">type="gradientradial"</div> </div> <p>Applied to a simple square using the following fill elements, the three image types look like this:</p> <table border="1" data-bbox="415 1577 1203 1879"> <tr> <td data-bbox="415 1577 573 1730">  </td><td data-bbox="573 1577 1203 1730"> <pre><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></pre> </td></tr> <tr> <td data-bbox="415 1730 573 1879">  </td><td data-bbox="573 1730 1203 1879"> <pre><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></pre> </td></tr> </table>		<pre><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></pre>		<pre><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></pre>
	<pre><v:fill src="myimage.gif" type="tile" size="50%,50%"> </v:fill></pre>				
	<pre><v:fill src="myimage.gif" type="frame" size="50%,50%"> </v:fill></pre>				

Attributes	Description
	<div></div> <div><pre><v:fill src="myimage.gif" color="red" color2="blue" type="pattern"> </v:fill></pre></div> <div>end example]</div> <div>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</div>

[Note: The W3C XML Schema definition of this element’s content model ([CT_Fill](#)) is located in §A.6.1. end note]

14.1.2.6 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 (§14.1.2.4).

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

Child Elements	Subclause
f (Single Formula)	§14.1.2.4

[Note: The W3C XML Schema definition of this element’s content model ([CT_Formulas](#)) is located in §A.6.1. end note]

14.1.2.7 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)


Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
arc (Arc Segment)	§14.1.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
curve (Bezier Curve)	§14.1.2.3
diagram (VML Diagram)	§14.2.2.8
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
group (Shape Group)	§14.1.2.7
handles (Set of Handles)	§14.1.2.9
image (Image File)	§14.1.2.10
imagedata (Image Data)	§14.1.2.11
line (Line)	§14.1.2.12
lock (Shape Protections)	§14.2.2.18
oval (Oval)	§14.1.2.13
path (Shape Path)	§14.1.2.14
polyline (Multiple Path Line)	§14.1.2.15
rect (Rectangle)	§14.1.2.16
roundrect (Rounded Rectangle)	§14.1.2.17
shadow (Shadow Effect)	§14.1.2.18
shape (Shape Definition)	§14.1.2.19
shapetype (Shape Template)	§14.1.2.20
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21


Child Elements	Subclause
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§15.1.2.5).</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> <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</p>

Attributes	Description
	datatype.
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"> </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>
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 (§14.2.3.10).</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 (§14.2.3.10).</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</p>

Attributes	Description
	datatype.
<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 (§15.1.2.5).</p>
<p>editas (Group Diagram Type)</p>	<p>Specifies which diagram type the contained shapes represent. This is used in conjunction with the diagram element (§14.2.2.8). 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>end example]</p> <p>The possible values for this attribute are defined by the ST_EditAs simple type (§14.1.3.1).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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>

Attributes	Description
	The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).
filled (Shape Fill Toggle) Namespace: urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.16).</p>
href (Hyperlink Target)	<p>Specifies a hyperlink URL target for the shape. Default is no value.</p> <p>[Example:</p>

Attributes	Description
	<p><code><v:shape ... href="http://www.openxmlformats.org" ... ></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>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 <code>false</code>.</p> <p>[Example:</p> <p><code><v:shape ... o:hrnoshade="true" ... ></code> <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 (§15.1.2.5).</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> <p><code><v:shape ... o:hrpct="85" ... ></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 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 <code>hr</code> is <code>true</code>. Default is <code>false</code>.</p> <p>[Example:</p> <p><code><v:shape ... o:hrstd="true" ... ></code> <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 (§15.1.2.5).</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>

Attributes	Description
	<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	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. [Example: <pre><v:shape ... o:insetmode="auto" ... > </v:shape></pre> <i>end example]</i> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).</p>
oned (Shape Handle Toggle) Namespace: urn:schemas-microsoft-com:office:office	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. [Example: <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <i>end example]</i> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
print (Print Toggle)	Specifies whether the shape is printed. Default is true. [Example: <pre><v:shape ... print="false" ... > </v:shape></pre> <i>end example]</i> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
regroupid (Regroup)	Specifies a previous group for a shape. An ID number is used to identify groups of shapes

Attributes	Description						
<p>ID)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>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>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>end example]</p> <table border="1" data-bbox="415 1434 1482 1864"> <thead> <tr> <th data-bbox="415 1434 664 1482">Property</th><th data-bbox="664 1434 1482 1482">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1482 664 1751">flip</td><td data-bbox="664 1482 1482 1751"> <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 1751 664 1864">height</td><td data-bbox="664 1751 1482 1864"> <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">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 (§14.1.2.22):</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</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.vertical - Text is displayed vertically.vertical-ideographic - Ideographic text is displayed vertically.horizontal-ideographic - 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		
	The following properties are only used by the textpath element (§14.1.2.23):		
	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:	
		Value	Description
		normal lighter 100 200 300 400	Treated as non-bold.
bold		Treated as bold.	

Attributes	Description	
		<div> bolder 500 600 700 800 900 </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.

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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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.	
tablelimits (Table Row Height Limits) Namespace: urn:schemas-microsoft-com:office:office	Specifies a list of minimum height values for each row in a table. Default is no value. 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. [Example:	

Attributes	Description												
	<pre><v:shape ... o:tablelimits="30pt 20pt" ... > </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>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 data-bbox="415 751 1260 947"> <tr> <th>Bit</th><th>Description</th></tr> <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> </table> <p><i>[Example: Decimal 3 means that bits 1 and 2 are set.</i></p> <pre><v:shape ... o:tableproperties="3" ... > </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>	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 data-bbox="415 1419 1479 1890"> <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> <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> </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.	_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.
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.												

Attributes	Description				
	<table border="1" data-bbox="415 245 1481 415"> <tr> <td data-bbox="415 245 630 331"><code>_self</code></td><td data-bbox="630 245 1481 331">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="415 331 630 415"><code>_top</code></td><td data-bbox="630 331 1481 415">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 457 537 485">[Example:</p> <pre data-bbox="456 527 1062 659"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 701 574 728">end example]</p> <p data-bbox="415 770 1377 833">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<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>_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="415 848 1481 911">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 953 537 980">[Example:</p> <pre data-bbox="456 1022 935 1092"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1134 574 1161">end example]</p> <p data-bbox="415 1203 1377 1266">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 data-bbox="415 1283 1481 1346">Specifies whether the user has added the shape to a master slide. Default is <code>false</code>. Used by PresentationML.</p> <p data-bbox="415 1388 537 1415">[Example:</p> <pre data-bbox="456 1457 984 1526"> <v:shape ... o:userdrawn="true" ... > </v:shape> </pre> <p data-bbox="415 1568 574 1596">end example]</p> <p data-bbox="415 1638 1390 1701">The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>				
userhidden (Hide Script Anchors) Namespace: urn:schemas-	<p data-bbox="415 1717 1481 1822">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 data-bbox="415 1864 537 1892">[Example:</p>				

Attributes	Description
microsoft-com:office:office	<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 (§15.1.2.5).</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 <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_Group](#)) is located in §A.6.1. *end note]*

14.1.2.8 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]

Parent Elements
handles (§14.1.2.9)

Attributes	Description
invx (Invert Handle's X Position)	<p>Specifies whether the x position of the handle should be inverted according to: $x_{\text{new}} = \text{coordorigin}_x + \text{coordsize}_x - x_{\text{old}}$</p> <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 (§15.1.2.5).</p>
invy (Invert Handle's Y Position)	<p>Specifies whether the y position of the handle should be inverted according to: $y_{\text{new}} = \text{coordorigin}_y + \text{coordsize}_y - y_{\text{old}}$</p> <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 (§15.1.2.5).</p>
map (Handle Coordinate Mapping)	<p>Specifies how the x and y positions of the handle are mapped from the coordsize range into the specified range. Default is "0,1000".</p> <p>[Example:</p>

Attributes	Description
	<pre><v:handles> <v:h ... map="-1000,1000" ... /> </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>
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><i>end example]</i></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" ... /> </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.
radiusrange (Handle Polar Radius Range)	<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>
switch (Handle Inversion Toggle)	<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 (§15.1.2.6).</p>
xrange (Handle X Position Range)	<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> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
yrange (Handle Y	Specifies a range of minimum and maximum values that constrain the y position of a

Attributes	Description
Position Range)	<p>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]*

14.1.2.9 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.

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

Child Elements	Subclause
h (Shape Handle)	§14.1.2.8

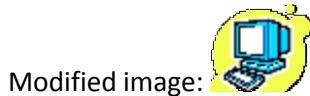
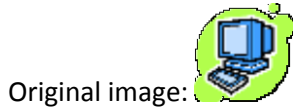
[*Note:* The W3C XML Schema definition of this element's content model ([CT_Handles](#)) is located in §A.6.1. *end note]*

14.1.2.10 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 (§14.1.2.11), 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22

Child Elements	Subclause
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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>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 <i>false</i>.</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 (§15.1.2.5).</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> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>




Attributes	Description
	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 <code>false</code> . <i>[Example:</i> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).


Attributes	Description
<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 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 (§15.1.2.5).</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 grayscale 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 (§14.2.3.3).</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 (§14.2.3.3).</p>
<p>bwpure (Pure Black-and-White)</p>	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p>

Attributes	Description
Mode) Namespace: urn:schemas- microsoft- com:office:office	<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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 (§15.1.2.3).</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)	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§14.2.2.3) element to create a clipping region.</p>



Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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).</p>



Attributes	Description
	<p>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> <p>[<i>Example:</i></p>

Attributes	Description
	<pre><v:image ... cropbottom="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>
cropleft (Image Left Crop)	<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
cropright (Image Right Crop)	<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. <i>end example]</i></p> <p>[Example:</p> <pre><v:image ... cropright="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</p>

Attributes	Description
	datatype.
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><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>
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><i>[Example:</i></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 (§14.2.3.10).</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><i>[Example:</i></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 (§14.2.3.10).</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 (§15.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 (§15.1.2.3).</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 (§15.1.2.5).</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 (§15.1.2.5).</p>
<p>gain (Image Intensity)</p>	<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> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>


Attributes	Description
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>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 (§15.1.2.5).</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 (§15.1.2.5).</p>
hralign (Horizontal	<p>Specifies the alignment of a horizontal rule. Default is left.</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 (§14.2.3.16).</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.17).</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 (§15.1.2.5).
<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 (§15.1.2.6).</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 (§15.1.2.5).</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 (§15.1.2.5).</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><code>opacity="1"</code></p> <p><code>opacity=".25"</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 (§15.1.2.5).</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 (§15.1.2.5).</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>src (Image Source)</p>	<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>
<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 (§14.1.2.21) 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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre data-bbox="451 751 1063 856"><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 (§15.1.2.5).</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 (§14.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre data-bbox="451 1392 982 1465"><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 border="1"> <thead> <tr> <th data-bbox="415 703 664 751">Property</th><th data-bbox="664 703 1479 751">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 751 664 1018">flip</td><td data-bbox="664 751 1479 1018"> <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 1018 664 1430">height</td><td data-bbox="664 1018 1479 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 1479 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> </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. • <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">• <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 (§14.1.2.22):	
	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 horizontally.
	mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code>.</p>
	mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code>.</p>
	mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is <code>false</code>.</p>
	mso-layout-flow-alt	<p>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>.</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">090180-90									
	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">topmiddlebottomtop-centermiddle-centerbottom-centertop-baselinebottom-baselinetop-center-baselinebottom-center-baseline									
	The following properties are only used by the textpath element (§14.1.2.23):										
	<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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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 1484 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 (§15.1.2.5).</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 (§15.1.2.5).</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 data-bbox="451 428 1159 527"><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]*

14.1.2.11 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 (§14.1.2.10), 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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

Attributes	Description
althref (Alternate Image Reference) Namespace: urn:schemas-microsoft-com:office:office	<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> <p>end example]</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 <i>false</i>.</p> <p>[Example:</p> <pre><v:image ... bilevel="true" ...> </v:image></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
blacklevel (Image Brightness)	<p>Specifies the image brightness. Default is 0.</p> <p>[Example:</p> <pre><v:image ... blacklevel="0.1" ...> </v:image></pre>  <p>end example]</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>



Attributes	Description
	<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 (§15.1.2.3).</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> <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>
cropleft (Image Left Crop)	<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>
cropright (Image Right Crop)	<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.</p>


Attributes	Description
	<p><i>end example]</i></p> <p>[<i>Example:</i></p> <pre><v:image ... cropright="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>
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:</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: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 (§15.1.2.5).</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 (§15.1.2.3).</p>

Attributes	Description
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> <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>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> 

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
<p>href (Explicit Relationship to Hyperlink Target)</p> <p>Namespace: .../officeDocument/2006/relationships</p>	<p>Specifies the relationship ID of the relationship to the hyperlink 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: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</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>
movie (Movie Reference) Namespace: urn:schemas-microsoft-com:office:office	<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> <pre><v:imagedata ... o:movie="1434" ...> </v:imagedata></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema float datatype.</p>
oleid (Image Embedded Object ID) Namespace: urn:schemas-microsoft-com:office:office	<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>
pict (Explicit Relationship to Alternate Image Data) Namespace: .../officeDocument/2006/relationships	<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>

Attributes	Description
	<p>< ... r:id="rId10" r:pict="rId7"/></p> <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>recolortarget (Black Recoloring Color)</p>	<p>Specifies the color to which black should be recolored.</p> <p>[Example:</p> <p><v:imagedata r:id="rId4" recolortarget="red"> </v:imagedata></p> <div data-bbox="451 716 553 821">  </div> <p>no recolor</p> <div data-bbox="451 831 553 936">  </div> <p>recolortarget="red"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p>
<p>relid (Relationship to Part)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<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> <p><v:imagedata ... o:relid="rId10" ...> </v:imagedata></p> <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>src (Image Source)</p>	<p>Specifies the URL of the image to use.</p> <p>[Example:</p> <p><v:image ... src="myimage.gif" ...> </v:image></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>title (Image Data Title)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies the title of an embedded 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>

[Note: The W3C XML Schema definition of this element's content model ([CT_ImageData](#)) is located in §A.6.1.
end note]

14.1.2.12 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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><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 (§15.1.2.5).</p>



Attributes	Description
<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 (§15.1.2.5).</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 (Border Left Color)</p>	<p>Specifies the left 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: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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.3).</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 (§14.2.3.3).</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	<p><code><v:shape ... o:bwmode="auto" o:bwpure="highcontrast" ... ></code> <code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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> <p><code><v:image ... chromakey="white" ...></code> <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 (§15.1.2.3).</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> <p>...</p> <p><code>.narrowstyle {width:50;height:100}</code></p> <p>...</p> <p><code><v:shape ... class="narrowstyle"</code> <code>style="top:1;left:1"></code> <code></v:shape></code></p> <p><code><v:shape ... style="top:1;left:1;</code> <code>width:50;height:100"></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>
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 (§14.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <p><code><v:shape ... o:clip="true"></code> <code></v:shape></code></p>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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>

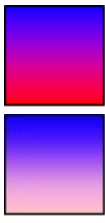
Attributes	Description
	The possible values for this attribute are defined by the ST_DiagramLayout simple type (§14.2.3.10).
dgmlayoutmru (Diagram Node Recent Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies the kind of automatic layout most recently used on the child 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>end example]</p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§14.2.3.10).</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>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 (§15.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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>

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 (§15.1.2.3).</p>
filled (Shape Fill Toggle)	<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 (§15.1.2.5).</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 (§15.1.2.5).</p>



Attributes	Description
from (Line Start) Namespace: urn:schemas-microsoft-com:office:office	<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>
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 (§15.1.2.5).</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 (§14.2.3.16).</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>


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>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 (§15.1.2.5).</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>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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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" ... ></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>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 (§14.2.3.17).</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 (§15.1.2.5).</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 (§15.1.2.6).</p>
<p>oleicon (Embedded Object Icon Toggle)</p>	<p>Specifies whether an embedded object is displayed as an icon. Default is false.</p> <p>[Example:</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
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. [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> <div data-bbox="451 1360 779 1581">  <div data-bbox="568 1438 779 1581"> <p>opacity="1"</p> <p>opacity=".25"</p> </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>
preferrelative (Relative Resize Toggle)	<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>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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 (§15.1.2.5).</p>
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>end example]</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-	<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>

Attributes	Description
microsoft-com:office:office	
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 (§14.1.2.21) 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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... ></pre>

Attributes	Description						
	<p data-bbox="451 247 613 279"></v:shape></p>  <p data-bbox="414 464 574 495"><i>end example]</i></p> <p data-bbox="414 535 1377 598">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>						
<p data-bbox="142 619 386 682">style (Shape Styling Properties)</p>	<p data-bbox="414 619 1481 720">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="414 760 1453 898">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="414 938 535 970"><i>[Example:</i></p> <pre data-bbox="451 974 1453 1073"> <v:shape ... style='position:absolute;width:100pt;height:50pt' ... > </v:shape> </pre> <p data-bbox="414 1077 574 1108"><i>end example]</i></p> <table data-bbox="414 1144 1477 1858"> <thead> <tr> <th data-bbox="418 1150 662 1192">Property</th><th data-bbox="662 1150 1472 1192">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="418 1192 662 1459">flip</td><td data-bbox="662 1192 1472 1459"> <p data-bbox="678 1203 1442 1266">Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul data-bbox="727 1312 1393 1449" 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="418 1459 662 1852">height</td><td data-bbox="662 1459 1472 1852"> <p data-bbox="678 1470 1453 1575">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 data-bbox="727 1617 1458 1852" 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> </tbody> </table>	Property	Description	flip	<p data-bbox="678 1203 1442 1266">Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul data-bbox="727 1312 1393 1449" 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 data-bbox="678 1470 1453 1575">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 data-bbox="727 1617 1458 1852" 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.
Property	Description						
flip	<p data-bbox="678 1203 1442 1266">Specifies that the orientation of a shape is flipped. Default is no value. Allowed values are:</p> <ul data-bbox="727 1312 1393 1449" 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 data-bbox="678 1470 1453 1575">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 data-bbox="727 1617 1458 1852" 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	
	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> <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 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> <ul style="list-style-type: none"> • absolute • top

Attributes	Description	
		<ul style="list-style-type: none"> • 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 values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square.

Attributes	Description	
		<ul style="list-style-type: none"> • 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.
	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>


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.
	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 (§14.1.2.22):		
	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-	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p>

Attributes	Description					
	to-text					
	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	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 (§14.1.2.23):						
<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</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
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					

Attributes	Description														
		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	<div>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:</div> <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="4">Treated as bold.</td></tr><tr><td>bolder</td></tr><tr><td>500</td></tr><tr><td>600</td></tr></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.	bolder	500
Value	Description														
normal	Treated as non-bold.														
lighter															
100															
200															
300															
400															
bold	Treated as bold.														
bolder															
500															
600															

Attributes	Description	
		<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-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:

Attributes	Description								
	<ul style="list-style-type: none"> • tightening • tracking 								
	<table> <tr> <td data-bbox="418 373 662 407">v-text-spacing</td><td data-bbox="662 373 1479 441">Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.</td></tr> </table>	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.						
v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.								
	<p>The line (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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> <tr> <th data-bbox="418 1528 630 1577">Value</th><th data-bbox="630 1528 1479 1577">Description</th></tr> <tr> <td data-bbox="418 1577 630 1661"><targetname></td><td data-bbox="630 1577 1479 1661">String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td data-bbox="418 1661 630 1745">_blank</td><td data-bbox="630 1661 1479 1745">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="418 1745 630 1822">_media</td><td data-bbox="630 1745 1479 1822">Specifies that the linked document is loaded into the browser's multimedia pane.</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.	_media	Specifies that the linked document is loaded into the browser's multimedia pane.
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.								

Attributes	Description								
	<table border="1" data-bbox="415 247 1481 583"> <tr> <td data-bbox="415 247 626 331"><code>_parent</code></td><td data-bbox="626 247 1481 331">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="415 331 626 415"><code>_search</code></td><td data-bbox="626 331 1481 415">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="415 415 626 499"><code>_self</code></td><td data-bbox="626 415 1481 499">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="415 499 626 583"><code>_top</code></td><td data-bbox="626 499 1481 583">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 625 535 657"><i>[Example:</i></p> <pre data-bbox="454 693 1063 829"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 865 576 896"><i>end example]</i></p> <p data-bbox="415 934 1377 1003">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<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>_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 1018 1481 1087">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 1129 535 1161"><i>[Example:</i></p> <pre data-bbox="454 1197 933 1266"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1302 576 1333"><i>end example]</i></p> <p data-bbox="415 1371 1377 1440">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>								
to (Line End Point)	<p data-bbox="415 1455 1481 1560">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 1602 535 1633"><i>[Example:</i></p> <pre data-bbox="454 1669 1096 1738"> <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 (§15.1.2.5).</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 (§15.1.2.5).</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_Line](#)) is located in §A.6.1. *end note*]

14.1.2.13 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30

Child Elements	Subclause
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6


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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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>


Attributes	Description
	<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	Specifies the bottom border color of an inline shape. Default is no value. [Example: <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. [Example: <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. [Example: <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:	Specifies the top border color of an inline shape. Default is no value. [Example:

Attributes	Description
urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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 (§15.1.2.5).</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 "auto" 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 (§14.2.3.3).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</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>

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 (§14.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre> <v:shape ... o:clip="true"> </v:shape> end example] </pre> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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> end example] </pre> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
<p>connectortype (Shape Connector Type)</p> <p>Namespace:</p>	<p>Specifies the kind of connector used for joining shapes. Default is straight.</p> <p>[Example:</p> <pre> <v:shape ... o:connectortype="elbow" ... > </pre>

Attributes	Description
urn:schemas-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 ST_ConnectorType simple type (§14.2.3.7).</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"</pre>



Attributes	Description
	<pre>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 (§14.2.3.10).</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 (§14.2.3.10).</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 (§15.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 (§15.1.2.3).</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.16).</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 (§15.1.2.5).</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 (§15.1.2.5).</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>end example]</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>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).</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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>

Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§15.1.2.6).</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 (§15.1.2.5).</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 (§15.1.2.5).</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">  <div data-bbox="571 1667 747 1701">opacity="1"</div>  <div data-bbox="571 1776 779 1808">opacity=".25"</div> </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>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 (§15.1.2.5).</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 (§15.1.2.5).</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-</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>

Attributes	Description
com:office:office	
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 (§14.1.2.21) 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>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p><i>[Example:</i></p> <pre><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 (§15.1.2.5).</p>

Attributes	Description						
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 (§14.1.2.21) 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> <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 1392 1477 1850"> <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> </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 parent object's width.

Attributes	Description	
	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 • char

Attributes	Description	
	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-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.</p>

Attributes	Description	
		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	<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"> • 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.

Attributes	Description	
		<ul style="list-style-type: none"> • <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 (§14.1.2.22):		
	Property	Description
	<code>direction</code>	<p>Specifies the direction of the text in the textbox. Default is <code>ltr</code>. This property is 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.
	<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.

Attributes	Description	
	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												
	The following properties are only used by the textpath element (§14.1.2.23):												
	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="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>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> bolder 500 600 700 800 900 </div>
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 • underline • overline • line-through • blink
v-rotate-letters		<p>Specifies whether the letters of the text are rotated counterclockwise by 90 degrees. Default is false.</p>
v-same-letter-heights		<p>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.</p>
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		<p>Specifies whether kerning is turned on. Default is false.</p>
v-text-reverse		<p>Specifies whether the layout order of rows is reversed. Default is false. This is used for vertical text layout.</p>

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">tighteningtracking
	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.
	<p>The line (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) elements ignore the following properties:</p> <ul style="list-style-type: none">topleftwidthheight <p>The following properties are not inherited by an element that references a shapetype element (§14.1.2.20) via the id attribute:</p> <ul style="list-style-type: none">flipheightleftmargin-leftmargin-toppositionrotationtopvisibilitywidthz-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:
	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="412 245 626 327"><code>_blank</code></td><td data-bbox="626 245 1479 327">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="412 327 626 409"><code>_media</code></td><td data-bbox="626 327 1479 409">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="412 409 626 491"><code>_parent</code></td><td data-bbox="626 409 1479 491">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="412 491 626 573"><code>_search</code></td><td data-bbox="626 491 1479 573">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="412 573 626 655"><code>_self</code></td><td data-bbox="626 573 1479 655">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="412 655 626 737"><code>_top</code></td><td data-bbox="626 655 1479 737">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 (§15.1.2.5).</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 (§15.1.2.5).</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_Oval](#)) is located in §A.6.1. *end note]*


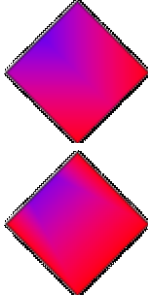
14.1.2.14 path (Shape Path)

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 (§14.1.2.6) 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]*


Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

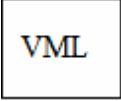

Attributes	Description
arrowok (Arrowhead Display Toggle)	<p>Specifies whether arrowheads are allowed to be displayed. This attribute overrides all other arrowhead attributes in the parent or the stroke element (§14.1.2.21). Default is false.</p> <p>[Example:</p> <pre><v:shape style="width:50;height:50"> <v:stroke endarrow="block"/> <v:path arrowok="true" v="m 0,0 l 1000,0 1000,1000 e"/> </v:shape></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
connectangles (Connection Point Connect Angles)	<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>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
connectlocs (Connection Points)	<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 connecttype is custom. Default is no value.</p> <p>[Example: Connection points exist at the midpoints of the sides of the square:</p>

Attributes	Description
com:office:office	<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>
connecttype (Connection Point Type) Namespace: urn:schemas- microsoft- com:office:office	<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 (§14.2.3.8).</p>
extrusionok (Extrusion Toggle) Namespace: urn:schemas- microsoft- com:office:office	<p>Specifies whether an extrusion is allowed to be displayed. This attribute overrides all other extrusion attributes in the parent or the extrusion element (§14.2.2.11). 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 1428 553 1530" data-label="Image"> </div> <pre><v:path o:extrusionok="false"/></pre> <div data-bbox="451 1537 587 1673" data-label="Image"> </div> <pre><v:path o:extrusionok="true"/></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>

Attributes	Description
fillok (Shape Fill Toggle)	<p>Specifies whether a fill is allowed to be displayed. This attribute overrides all other fill attributes in the parent or fill element (§14.1.2.5). Default is true.</p> <p>[Example:</p> <pre> <v:shape style="width:50;height:50" fillcolor="red"> <v:path filllok="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 (§15.1.2.5).</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 (§14.1.2.5).</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> <p>gradientshapeok="true"</p> <p>end example]</p>

Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>insetpenok (Inset Stroke From Path Flag)</p>	<p>Specifies whether the stroke can be inset from the path. If this is false, it overrides the insetpen 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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
<p>limo (Limo Stretch Point)</p>	<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>end example]</p> <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 (§14.1.2.18). Default is true.</p>

Attributes	Description
	<p>[<i>Example</i>: 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><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
strokeok (Stroke Toggle)	<p>Specifies whether a stroke is displayed. This attribute overrides all other stroke attributes in the parent or the stroke element (§14.1.2.21). Default is true.</p> <p>[<i>Example</i>: 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><i>end example</i>]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
textboxrect (Text Box Bounding Box)	<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 shape. The textboxrect dimensions clip any text that extends beyond its region.</p> <p>[<i>Example</i>: 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"/></pre>

Attributes	Description
	<pre><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 <i>false</i>.</p> <p>If <i>true</i>, this indicates that the path is an appropriate warping path for the textpath element (§14.1.2.23). Otherwise, the textpath element shall be ignored.</p> <p>[<i>Example:</i> 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 (§15.1.2.5).</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> <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 (§14.1.2.6) with a list of formulas that are substituted into the path using the @ symbol followed by the number of the formula. The adj

Attributes	Description																																								
	<p>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> <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, cpy+y).</td></tr><tr><td>v</td><td>rcurveto</td><td>6*</td><td>Cubic bézier curve using the given coordinate relative to the current point.</td></tr><tr><td>nf</td><td>nofill</td><td>0</td><td>The current set of sub-paths (delimited by e) is not filled.</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, 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.
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.																																						

Attributes	Description			
	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
	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).

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.			

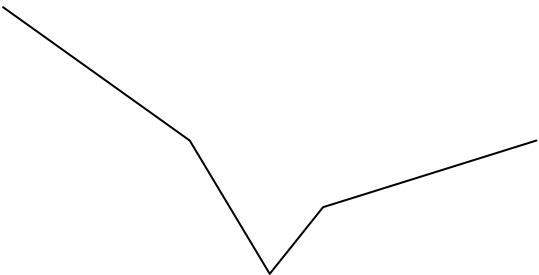
[Note: The W3C XML Schema definition of this element’s content model ([CT_Path](#)) is located in §A.6.1. *end note*]

14.1.2.15 [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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
ink (Ink)	§14.2.2.15
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type</p>



Attributes	Description
<p>allowoverlap (Allow Shape Overlap)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>(§15.1.2.5).</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 (§15.1.2.5).</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 (Border Left Color)</p>	<p>Specifies the left border color of an inline shape. Default is no value.</p>

Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<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>
bodderrightcolor (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:bodderrightcolor="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>
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 (§15.1.2.5).</p>
button (Button Behavior Toggle)	<p>Specifies whether a shape exhibits button press behavior on click. Default is false.</p> <p>[Example:</p>


Attributes	Description
Namespace: 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 (§15.1.2.5).</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 (§14.2.3.3).</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 (§14.2.3.3).</p>
bwpure (Pure Black-and-White Mode) Namespace:	<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
urn:schemas-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 (§14.2.3.3).</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 (§15.1.2.3).</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-	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§14.2.2.3) element to create a clipping region.</p> <p>[Example:</p> <pre><v:shape ... o:clip="true"></pre>

Attributes	Description
com:office:office	<p></v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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"</pre>

Attributes	Description
	<pre>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" 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
	<p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§14.2.3.10).</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>end example]</p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§14.2.3.10).</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 (§15.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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>

Attributes	Description
	<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 (§15.1.2.3).</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>[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 (§15.1.2.5).</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</p>


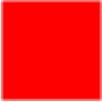

Attributes	Description
<p>hr (Horizontal Rule Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>(§15.1.2.5).</p> <p>Specifies that a shape is a horizontal rule. Default is false.</p> <p>[Example:</p> <pre><v:shape ... o:hr="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§14.2.3.16).</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 (§15.1.2.5).</p>

Attributes	Description
<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 (§15.1.2.5).</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>

Attributes	Description
	The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).
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 (§15.1.2.5).</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 (§15.1.2.6).</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 (§15.1.2.5).</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 (§15.1.2.5).</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. [<i>Example:</i> A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>[<i>Example:</i> The red color is 25% opaque:</p> <pre data-bbox="451 653 1015 751"><v:fill type="gradient" color="red" color2="blue" opacity=".25"> </v:fill></pre> <div data-bbox="451 787 779 1003">  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>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>[<i>Example:</i></p> <pre data-bbox="451 1818 1063 1881"><v:shape ... o:preferrelative="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 (§15.1.2.5).</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 (§15.1.2.5).</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>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 (§14.1.2.21) overrides this. Colors are typically specified as either a named color, such as red, or six hexadecimal digits</p>

Attributes	Description
	<p>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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </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>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: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></p> <table border="1" data-bbox="415 966 1487 1877"> <thead> <tr> <th data-bbox="415 966 664 1012">Property</th><th data-bbox="664 966 1487 1012">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1012 664 1283">flip</td><td data-bbox="664 1012 1487 1283"> <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 1283 664 1692">height</td><td data-bbox="664 1283 1487 1692"> <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 1692 664 1877">left</td><td data-bbox="664 1692 1487 1877"> <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> </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>
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>								

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-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.

Attributes	Description	
	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> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside

Attributes	Description
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 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>

Attributes	Description	
		<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 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.

Attributes	Description	
		<ul style="list-style-type: none"> • <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 (§14.1.2.22):		
	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-	<p>Specifies the alternate layout flow for text in textboxes. This</p>

Attributes	Description							
	flow-alt	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	Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are: <ul style="list-style-type: none">090180-90						
	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">topmiddlebottomtop-centermiddle-centerbottom-centertop-baselinebottom-baselinetop-center-baselinebottom-center-baseline						
The following properties are only used by the textpath element (§14.1.2.23):								
<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></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.
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.							

Attributes	Description																		
	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="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></tbody></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-	Specifies whether a shadow is applied to the text on a text path.																		

Attributes	Description	
	shadow	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.

Attributes	Description														
	<p>The line (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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 1285 1482 1829"> <thead> <tr> <th data-bbox="415 1285 626 1331">Value</th><th data-bbox="626 1285 1482 1331">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1331 626 1419"><targetname></td><td data-bbox="626 1331 1482 1419">String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td data-bbox="415 1419 626 1507">_blank</td><td data-bbox="626 1419 1482 1507">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="415 1507 626 1596">_media</td><td data-bbox="626 1507 1482 1596">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="415 1596 626 1684">_parent</td><td data-bbox="626 1596 1482 1684">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="415 1684 626 1772">_search</td><td data-bbox="626 1684 1482 1772">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="415 1772 626 1835">_self</td><td data-bbox="626 1772 1482 1835">Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> </tbody> </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.	_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).
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).														

Attributes	Description		
	<table border="1" data-bbox="415 243 1482 331"> <tr> <td data-bbox="415 243 626 331"><code>_top</code></td><td data-bbox="626 243 1482 331">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 369 537 401"><i>[Example:</i></p> <pre data-bbox="451 438 1062 569"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 611 578 642"><i>end example]</i></p> <p data-bbox="415 680 1377 747">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<code>_top</code>	Specifies that the linked document is loaded into the topmost window.
<code>_top</code>	Specifies that the linked document is loaded into the topmost window.		
title (Shape Title)	<p data-bbox="415 764 1482 831">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 873 537 905"><i>[Example:</i></p> <pre data-bbox="451 942 935 1010"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1052 578 1083"><i>end example]</i></p> <p data-bbox="415 1121 1377 1188">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 data-bbox="415 1205 1482 1272">Specifies whether the user has added the shape to a master slide. Default is <code>false</code>. Used by PresentationML.</p> <p data-bbox="415 1314 537 1346"><i>[Example:</i></p> <pre data-bbox="451 1383 984 1451"> <v:shape ... o:userdrawn="true" ... > </v:shape> </pre> <p data-bbox="415 1493 578 1524"><i>end example]</i></p> <p data-bbox="415 1562 1390 1629">The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>		
userhidden (Hide Script Anchors) Namespace: urn:schemas-microsoft-com:office:office	<p data-bbox="415 1646 1482 1745">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 data-bbox="415 1787 537 1818"><i>[Example:</i></p> <pre data-bbox="451 1856 1000 1887"> <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 (§15.1.2.5).</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_PolyLine](#)) is located in §A.6.1. *end note]*

14.1.2.16 `rect` (Rectangle)

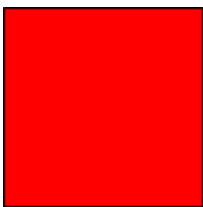
This element is used to draw a simple rectangle. The CSS2 style content width and height define the width and height of the rectangle.

[Example:

```

<v:rect fillcolor="red"
  style="position:relative;top:0;left:0;width:100;height:100">
</v:rect>

```



end example]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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>



Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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><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</p>

Attributes	Description
	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 <code>false</code> . <i>[Example:</i> <pre><v:shape ... o:bullet="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).


Attributes	Description
<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 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 (§15.1.2.5).</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 grayscale 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 (§14.2.3.3).</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 (§14.2.3.3).</p>
<p>bwpure (Pure Black-and-White)</p>	<p>Specifies the black-and-white mode for pure black-and-white output devices. Default is auto.</p>

Attributes	Description
Mode) Namespace: urn:schemas- microsoft- com:office:office	<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 (§14.2.3.3).</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 (§15.1.2.3).</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> <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)	<p>Specifies that the clipping region is active. This is used in conjunction with the clippath (§14.2.2.3) element to create a clipping region.</p>

Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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).</p>

Attributes	Description
	<p>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 Size)</p>	<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>
<p>dgmlayout (Diagram Node Layout Identifier)</p> <p>Namespace:</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>[<i>Example:</i></p>


Attributes	Description
urn:schemas-microsoft-com:office:office	<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 (§14.2.3.10).</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 (§14.2.3.10).</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 (§15.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) is</p>

Attributes	Description
	<p>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 (§15.1.2.3).</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 (§15.1.2.5).</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>




Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§14.2.3.16).</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-</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>

Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-</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" ... ></pre>

Attributes	Description
com:office:office	<p></v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).</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 (§15.1.2.5).</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> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§15.1.2.6).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
oned (Shape Handle Toggle) Namespace:	<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>

Attributes	Description
urn:schemas-microsoft-com:office:office	<p>[<i>Example:</i></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 (§15.1.2.5).</p>
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 1/65536ths if a trailing "f" is supplied. [<i>Example:</i> A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></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> <div data-bbox="451 926 779 1144">  <div data-bbox="570 1003 779 1144"> opacity="1" opacity=".25" </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>
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 (§15.1.2.5).</p>
print (Print Toggle)	<p>Specifies whether the shape is printed. Default is true.</p>

Attributes	Description
	<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 (§15.1.2.5).</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>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 (§14.1.2.21) 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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) 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	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 (§14.1.2.22):	
	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">090180-90									
	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">topmiddlebottomtop-centermiddle-centerbottom-centertop-baselinebottom-baselinetop-center-baselinebottom-center-baseline									
	The following properties are only used by the textpath element (§14.1.2.23):										
	<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="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">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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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><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><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></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 (§15.1.2.5).</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 (§15.1.2.5).</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 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>

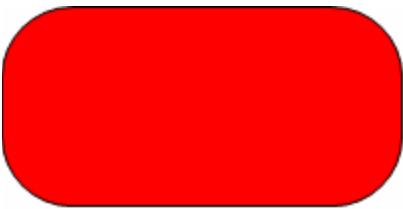
[Note: The W3C XML Schema definition of this element’s content model ([CT_Rect](#)) is located in §A.6.1. end note]

14.1.2.17 **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:200;height:100">  
</v:roundrect>
```



end example]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3

Child Elements	Subclause
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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 (§15.1.2.5).</p>
allowoverlap (Allow Shape Overlap)	<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>[<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 (§15.1.2.5).</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>[<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>
arcsize (Rounded Corner Arc Size)	<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>
borderbottomcolor (Bottom Border)	<p>Specifies the bottom border color of an inline shape. Default is no value.</p>

Attributes	Description
<p>Color)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</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 (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>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>Specifies whether the shape is a graphical bullet. Default is false.</p> <p>[Example:</p>


Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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. <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 (§14.2.3.3).</p>
bwnormal (Normal Black-and-White Mode) Namespace: urn:schemas-microsoft-	<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" ... ></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 ST_BWMode simple type (§14.2.3.3).</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>[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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).</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 (§15.1.2.3).</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> <v:shape ... style="top:1;left:1; width:50;height:100"> </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>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 (§14.2.2.3) 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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</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>

Attributes	Description
	<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>  <p><i>end example]</i></p>



Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
dgmlayout (Diagram Node Layout Identifier) Namespace: urn:schemas-microsoft-com:office:office	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. <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 (§14.2.3.10).
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 (§14.2.3.10).
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:	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" ... ></pre>

Attributes	Description
urn:schemas-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 ST_TrueFalse simple type (§15.1.2.5).</p>
fillcolor (Fill Color)	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 data-bbox="451 758 935 825"><v:shape ... fillcolor="red" ... > </v:shape></pre> <p>This is equivalent to:</p> <pre data-bbox="451 934 1000 1001"><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 (§15.1.2.3).</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 data-bbox="451 1371 789 1472"><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 (§15.1.2.5).</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>



Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.16).</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</p>


Attributes	Description
	datatype.
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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 <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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>end example]</p>

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string datatype.
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>end example]</p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).</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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.6).</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>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</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> <div data-bbox="451 1224 779 1444">  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-</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" ... ></pre>

Attributes	Description
com:office:office	<p></v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
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 (§14.1.2.21) overrides this. Colors are</p>

Attributes	Description
	<p>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 (§15.1.2.3).</p>
stroked (Shape Stroke Toggle)	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </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>								
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: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></p> <table border="1" data-bbox="415 1075 1474 1879"> <thead> <tr> <th data-bbox="415 1075 662 1121">Property</th><th data-bbox="662 1075 1474 1121">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1121 662 1390">flip</td><td data-bbox="662 1121 1474 1390"> <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 1390 662 1801">height</td><td data-bbox="662 1390 1474 1801"> <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 1801 662 1879">left</td><td data-bbox="662 1801 1474 1879"> <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.</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> <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.</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> <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.</p>								

Attributes	Description	
		<p>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> <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 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> <ul style="list-style-type: none"> • absolute • top • center • bottom

Attributes	Description	
		<ul style="list-style-type: none"> • 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 values are:</p> <ul style="list-style-type: none"> • square - Wraps text inside the shape in a square. • none - Text does not wrap.

Attributes	Description	
	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 (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

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 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 (§14.1.2.22):		
	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>

Attributes	Description	
	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 (§14.1.2.23):		
	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.

Attributes	Description																
	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	<div>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:</div> <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="6">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></tbody></table>	Value	Description	normal	Treated as non-bold.	lighter	100	200	300	400	bold	Treated as bold.	bolder	500	600	700
Value	Description																
normal	Treated as non-bold.																
lighter																	
100																	
200																	
300																	
400																	
bold	Treated as bold.																
bolder																	
500																	
600																	
700																	
800																	

Attributes	Description		
		900	
	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.	
	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	

Attributes	Description												
	<table border="1" data-bbox="418 247 1477 331"> <tr> <td data-bbox="418 247 662 331">v-text-spacing</td><td data-bbox="662 247 1477 331">Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.</td></tr> </table> <p>The line (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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>	v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.										
v-text-spacing	Specifies the amount of spacing for text in 100ths of single line spacing. Default is 100.												
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="418 1409 1477 1871"> <thead> <tr> <th data-bbox="418 1409 630 1457">Value</th><th data-bbox="630 1409 1477 1457">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="418 1457 630 1541"><targetname></td><td data-bbox="630 1457 1477 1541">String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td data-bbox="418 1541 630 1625">_blank</td><td data-bbox="630 1541 1477 1625">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="418 1625 630 1709">_media</td><td data-bbox="630 1625 1477 1709">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="418 1709 630 1793">_parent</td><td data-bbox="630 1709 1477 1793">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="418 1793 630 1871">_search</td><td data-bbox="630 1793 1477 1871">Specifies that the linked document is loaded into the browser's search pane.</td></tr> </tbody> </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.	_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.
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.												

Attributes	Description				
	<table border="1" data-bbox="415 245 1481 415"> <tr> <td data-bbox="415 245 630 331"><code>_self</code></td><td data-bbox="630 245 1481 331">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="415 331 630 415"><code>_top</code></td><td data-bbox="630 331 1481 415">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 457 537 485"><i>[Example:</i></p> <pre data-bbox="456 527 1062 659"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 701 578 728"><i>end example]</i></p> <p data-bbox="415 770 1377 833">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<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>_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.				
<p data-bbox="139 852 350 879">title (Shape Title)</p>	<p data-bbox="415 852 1481 915">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 957 537 984"><i>[Example:</i></p> <pre data-bbox="456 1026 935 1094"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1136 578 1163"><i>end example]</i></p> <p data-bbox="415 1205 1377 1268">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
<p data-bbox="139 1287 363 1350">userdrawn (Exists In Master Slide)</p> <p data-bbox="139 1392 350 1528">Namespace: urn:schemas-microsoft-com:office:office</p>	<p data-bbox="415 1287 1481 1350">Specifies whether the user has added the shape to a master slide. Default is <code>false</code>. Used by PresentationML.</p> <p data-bbox="415 1392 537 1419"><i>[Example:</i></p> <pre data-bbox="456 1461 984 1528"> <v:shape ... o:userdrawn="true" ... > </v:shape> </pre> <p data-bbox="415 1570 578 1598"><i>end example]</i></p> <p data-bbox="415 1640 1390 1703">The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>				
<p data-bbox="139 1722 358 1785">userhidden (Hide Script Anchors)</p> <p data-bbox="139 1827 310 1890">Namespace: urn:schemas-</p>	<p data-bbox="415 1722 1481 1827">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 data-bbox="415 1869 537 1896"><i>[Example:</i></p>				

Attributes	Description
microsoft-com:office:office	<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 (§15.1.2.5).</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_RoundRect](#)) is located in §A.6.1.
end note]

14.1.2.18 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>
```



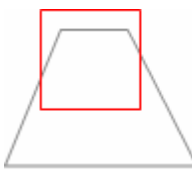



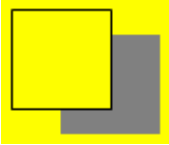

end example]

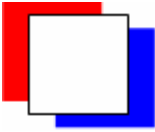

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)

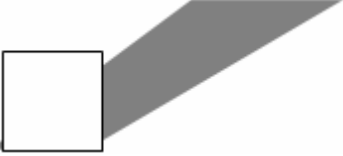
Attributes	Description
color (Shadow Primary Color)	<p>Specifies the color of the primary shadow. Default is gray (RGB 128,128,128).</p> <p>[Example:</p> <pre><v:shadow on="true" color="green"> </v:shadow></pre> <p>Applied to a simple square the shadow looks like this:</p>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p>
color2 (Shadow Secondary Color)	<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>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</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>
matrix (Shadow Perspective Matrix)	<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> <div data-bbox="451 932 800 1050">  <p>matrix=",,,,"</p> </div> <p>s_{xx}, s_{yy} specify scaling factors for the x and y dimensions:</p> <div data-bbox="451 1159 912 1272">  <p>matrix="2,,,,,"</p> </div> <div data-bbox="451 1308 816 1520">  <p>matrix=",,,2,,"</p> </div> <p>s_{xy}, s_{yx} specify skews in the x and y dimensions:</p> <div data-bbox="451 1625 1013 1745">  <p>matrix=",2,,,,,"</p> </div>

Attributes	Description
	<div></div> <div>matrix="",-2,,,"</div> <p>p_x, p_y effectively set the perspective trapezoid skews along the x and y dimensions:</p> <div></div> <div>matrix=",,,,.000001,"</div> <div></div> <div>matrix=",,,,,-.000002"</div> <p><i>end example]</i></p> <p>[Example:</p> <div><pre><v:shadow on="true" type="perspective" matrix="1.25,-2,,1.5,,.000001" offset="38pt,-6pt"> </v:shadow></pre></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>
obscured (Shadow Transparency)	<p>Specifies whether a shadow is transparent. Default is <code>false</code>. If <code>true</code>, the shadow is transparent if there is no fill on the shape.</p> <p>[Example:</p>

Attributes	Description
	<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 (§15.1.2.5).</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> <p>[Example:</p> <pre> <v:shadow on="true" offset="50%,25%"> </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>
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>

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>
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 (§15.1.2.5).</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. <i>[Example: For example, a value of "52429f" represents 52429/65536 or 0.8. end example]</i></p> <p><i>[Example:</i></p> <pre><v:shadow type="double" on="true" opacity=".5" color="blue" offset="10pt,5pt" color2="red" offset2="-10pt,-5pt"> </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>
origin (Shadow Origin)	<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: This example is unchanged from above except for the addition of the origin attribute:</i></p> <pre><v:shadow on="true" type="perspective" matrix="1.25,-2,,1.5,,.000001" offset="38pt,-6pt" origin="10%,-10%"> </v:shadow></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>														
type (Shadow Type)	<p>Specifies the kind of shadow. Default is <code>single</code>. Allowed values are:</p> <table border="1" data-bbox="415 659 1357 1104"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <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> <tr> <td><code>emboss</code></td><td>The shadow has an embossed look.</td></tr> </tbody> </table> <p>The possible values for this attribute are defined by the <code>ST_ShadowType</code> simple type (§14.1.3.6).</p>	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.	<code>emboss</code>	The shadow has an embossed look.
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.														
<code>emboss</code>	The shadow has an embossed look.														

[Note: The W3C XML Schema definition of this element's content model ([CT_Shadow](#)) is located in §A.6.1. *end note*]

14.1.2.19 `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 (§14.1.2.7). If a `shapetype` element (§14.1.2.20) 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
equationxml (Storage for Alternate Math Content)	§14.2.2.10
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
ink (Ink)	§14.2.2.15
iscomment (Ink Annotation Flag)	§14.5.2.1
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30

Child Elements	Subclause
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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. <i>[Example: For example, #2 references the second value in the adj list. end example]</i></p> <p><i>[Example: The following shape uses formulas to define a simple rectangle. The adj values are referenced by the eqn attribute of the f element (§14.1.2.4) and in turn referenced by the path element (§14.1.2.14).</i></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><i>This is the equivalent of:</i></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>


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>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 (§15.1.2.5).</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 (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string</p>


Attributes	Description
	datatype.
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.

Attributes	Description
<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 <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 (§15.1.2.5).</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 <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 (§15.1.2.5).</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 <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><i>bwnormal</i> and <i>bwpure</i> are subordinate to <i>bwmode</i>. If <i>bwmode</i> is "auto" then the value for <i>bwnormal</i> or <i>bwpure</i> 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 grayscale 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 (§14.2.3.3).</p>
<p>bwnormal (Normal Black-and-White Mode)</p>	<p>Specifies the black-and-white mode for normal black-and-white output devices. Default is <i>auto</i>.</p> <p>[Example: This shape renders in a pale grayscale in a normal black-and-white</p>


Attributes	Description
Namespace: urn:schemas-microsoft-com:office:office	environment: <pre><v:shape ... o:bwmode="auto" o:bwnormal="lightgrayscale" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_BWMode simple type (§14.2.3.3).
bwpure (Pure Black-and-White Mode) Namespace: urn:schemas-microsoft-com:office:office	Specifies the black-and-white mode for pure black-and-white output devices. Default is auto. <i>[Example: This shape renders in high contrast when in a pure black-and-white environment:</i> <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 (§14.2.3.3).
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 (§15.1.2.3).
class (CSS Reference)	Specifies a reference to the definition of a CSS style. Default is no value. <i>[Example: The snippets below are equivalent:</i> <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 (§14.2.2.3) 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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§14.2.3.7).</p>

Attributes	Description
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"> </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 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 (§14.2.3.10).</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 (§14.2.3.10).</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</p>



Attributes	Description
<p>doubleclicknotify (Double-click Notification Toggle)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>datatype.</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 (§15.1.2.5).</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 (§14.2.2.10). end note]</p> <p>The XML markup stored in this attribute shall be escaped as needed to contain only those characters legal in an attribute value.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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 (§15.1.2.3).</p>
<p>filled (Shape Fill)</p>	<p>Specifies whether the closed path is filled. Default is true. This attribute is overridden</p>

Attributes	Description
Toggle)	<p>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 (§15.1.2.5).</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 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 (§15.1.2.5).</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. <i>end rationale</i>]</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>




Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema base64Binary 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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_HrAlign simple type (§14.2.3.16).</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-</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>

Attributes	Description
com:office:office	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>
<p>insetmode (Text Inset Mode)</p> <p>Namespace: urn:schemas-microsoft-</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" ... ></pre>

Attributes	Description
com:office:office	<p></v:shape></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_InsetMode simple type (§14.2.3.17).</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 (§15.1.2.5).</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> <p>The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§15.1.2.6).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
oned (Shape Handle Toggle) Namespace:	<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>

Attributes	Description
urn:schemas-microsoft-com:office:office	<p>[<i>Example:</i></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 (§15.1.2.5).</p>
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:</i> A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></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> <div data-bbox="451 926 779 1144">  <div data-bbox="570 1003 747 1035" style="display: inline-block; vertical-align: middle;">opacity="1"</div>  <div data-bbox="570 1113 779 1144" style="display: inline-block; vertical-align: middle;">opacity=".25"</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>
path (Edge Path)	<p>Specifies the line that makes up the edges of a shape. See the v attribute of the path element (§14.1.2.14) 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>

Attributes	Description
	<p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
<p>print (Print Toggle)</p>	<p>Specifies whether the shape is printed. Default is true.</p> <p><i>[Example:</i></p> <pre data-bbox="451 512 902 575"><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 (§15.1.2.5).</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: The shape was part of a group identified by the ID 040754:</i></p> <pre data-bbox="451 947 1013 1010"><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 (§14.1.2.21) 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>

Attributes	Description
	<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 (§15.1.2.3).</p>
<p>stroked (Shape Stroke Toggle)</p>	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) overrides this attribute.</p> <p>[Example:</p> <pre><v:shape ... strokeweight="3pt" ... > </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>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: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></p> <table border="1"> <thead> <tr> <th data-bbox="415 894 664 940">Property</th><th data-bbox="664 894 1479 940">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 940 664 1209">flip</td><td data-bbox="664 940 1479 1209"> <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 1209 664 1619">height</td><td data-bbox="664 1209 1479 1619"> <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 1619 664 1877">left</td><td data-bbox="664 1619 1479 1877"> <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 </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
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 								

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 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</p>

Attributes	Description	
		<p>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> <ul style="list-style-type: none"> • absolute • top • center • bottom • inside • outside
	mso-position-vertical-	<p>Specifies relative vertical position data for objects in WordprocessingML documents. This modifies the mso-</p>

Attributes	Description	
	relative	<p>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 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

Attributes	Description	
		<p>are ignored. If the object is anchored inline, this value is used.</p> <ul style="list-style-type: none"> • 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 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.

Attributes	Description													
	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 (§14.1.2.22):													
		<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><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-alt</td><td><p>Specifies an alternate direction for text in textboxes. Overrides the direction property. The only allowed value is context.</p></td></tr><tr><td>mso-fit-shape-to-text</td><td><p>Specifies whether the shape stretches to fit the text in the textbox. Default is false.</p></td></tr><tr><td>mso-fit-text-to-shape</td><td><p>Specifies whether the text stretches to fit the textbox. Default is false.</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-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>
	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>													

Attributes	Description							
	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	Specifies a specific rotation value for text in a textbox. Default is 0. Allowed values are: <ul style="list-style-type: none">090180-90						
	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">topmiddlebottomtop-centermiddle-centerbottom-centertop-baselinebottom-baselinetop-center-baselinebottom-center-baseline						
The following properties are only used by the textpath element (§14.1.2.23):								
<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></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.
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.							

Attributes	Description																	
	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	<div>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:</div> <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
Value	Description																	
normal	Treated as non-bold.																	
lighter																		
100																		
200																		
300																		
400																		
bold	Treated as bold.																	
bolder																		
500																		
600																		
700																		
800																		
900																		
mso-text-	Specifies whether a shadow is applied to the text on a text path.																	

Attributes	Description	
	shadow	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.

Attributes	Description														
	<p>The line (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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"> <thead> <tr> <th data-bbox="415 1287 626 1335">Value</th><th data-bbox="626 1287 1479 1335">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 1335 626 1419"><targetname></td><td data-bbox="626 1335 1479 1419">String containing the name of the frame or window in which to load the document.</td></tr> <tr> <td data-bbox="415 1419 626 1503">_blank</td><td data-bbox="626 1419 1479 1503">Specifies that the linked document is loaded into a new blank window. This window is not named.</td></tr> <tr> <td data-bbox="415 1503 626 1587">_media</td><td data-bbox="626 1503 1479 1587">Specifies that the linked document is loaded into the browser's multimedia pane.</td></tr> <tr> <td data-bbox="415 1587 626 1671">_parent</td><td data-bbox="626 1587 1479 1671">Specifies that the linked document is loaded into the immediate parent of the document containing the link.</td></tr> <tr> <td data-bbox="415 1671 626 1755">_search</td><td data-bbox="626 1671 1479 1755">Specifies that the linked document is loaded into the browser's search pane.</td></tr> <tr> <td data-bbox="415 1755 626 1835">_self</td><td data-bbox="626 1755 1479 1835">Specifies that the linked document is loaded into the window in which the link was clicked (the active window).</td></tr> </tbody> </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.	_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).
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).														

Attributes	Description		
	<table border="1" data-bbox="415 245 1481 331"> <tr> <td data-bbox="415 245 626 331"><code>_top</code></td><td data-bbox="626 245 1481 331">Specifies that the linked document is loaded into the topmost window.</td></tr> </table> <p data-bbox="415 373 535 401"><i>[Example:</i></p> <pre data-bbox="456 443 1062 573"> <v:shape ... href="http://www.openxmlformats.org" target="_self" ... > </v:shape> </pre> <p data-bbox="415 615 574 642"><i>end example]</i></p> <p data-bbox="415 684 1377 751">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>	<code>_top</code>	Specifies that the linked document is loaded into the topmost window.
<code>_top</code>	Specifies that the linked document is loaded into the topmost window.		
title (Shape Title)	<p data-bbox="415 764 1481 831">Specifies the text displayed when the mouse pointer moves over the shape. Default is no value.</p> <p data-bbox="415 873 535 900"><i>[Example:</i></p> <pre data-bbox="456 942 935 1010"> <v:shape ... title="tooltip" ... > </v:shape> </pre> <p data-bbox="415 1052 574 1079"><i>end example]</i></p> <p data-bbox="415 1121 1377 1188">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		
type (Shape Type Reference)	<p data-bbox="415 1199 1451 1308">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 data-bbox="415 1350 1451 1417"><i>[Example:</i> 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 data-bbox="456 1459 1386 1791"> <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 data-bbox="415 1833 574 1860"><i>end example]</i></p>		

Attributes	Description
	The possible values for this attribute are defined by the W3C XML Schema string 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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>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_Shape](#)) is located in §A.6.1. *end note*]

14.1.2.20 shapetype (Shape Template)

This element defines a shape template that can be used to create other shapes. Shapetype is identical to the shape element (§14.1.2.19) 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]

Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1)

Child Elements	Subclause
anchorlock (Anchor Location Is Locked)	§14.3.2.1
borderbottom (Bottom Border)	§14.3.2.2
borderleft (Left Border)	§14.3.2.3
borderright (Right Border)	§14.3.2.4

Child Elements	Subclause
bordertop (Top Border)	§14.3.2.5
callout (Callout)	§14.2.2.2
ClientData (Attached Object Data)	§14.4.2.12
clippath (Shape Clipping Path)	§14.2.2.3
complex (Complex)	§14.2.2.7
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
formulas (Set of Formulas)	§14.1.2.6
handles (Set of Handles)	§14.1.2.9
imagedata (Image Data)	§14.1.2.11
lock (Shape Protections)	§14.2.2.18
path (Shape Path)	§14.1.2.14
shadow (Shadow Effect)	§14.1.2.18
signatureline (Digital Signature Line)	§14.2.2.30
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22
textdata (VML Diagram Text)	§14.5.2.2
textpath (Text Layout Path)	§14.1.2.23
wrap (Text Wrapping)	§14.3.2.6

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 are referenced by the eqn attribute of the f element (§14.1.2.4) and in turn referenced by the path element (§14.1.2.14).</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> </pre>


Attributes	Description
	<pre> <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 (§15.1.2.5).</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</p>


Attributes	Description
	(§15.1.2.5).
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)	<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)	<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
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>
bordercolor (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:bordercolor="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 false.</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 (§15.1.2.5).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
bwmode (Black-and-White Mode) Namespace: urn:schemas-	<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>


Attributes	Description
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 (§14.2.3.3).</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 (§14.2.3.3).</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 (§14.2.3.3).</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 (§15.1.2.3).</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 (§14.2.2.3) 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 (§15.1.2.5).</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"></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 (§15.1.2.5).</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 (§14.2.3.7).</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</p>	<p>Specifies the size of the shape's coordinate space in coordinate units. Default is</p>

Attributes	Description
<p>(Coordinate Space Size)</p>	<p>"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>
<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 (§14.2.3.10).</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>

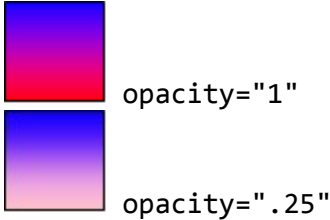
Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_DiagramLayout simple type (§14.2.3.10).</p>
<p>dgmnodekind (Diagram Node Identifier)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<p>Specifies an optional, application-defined 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 (§15.1.2.5).</p>
<p>fillcolor (Fill Color)</p>	<p>Specifies the color to use for the fill. Default is white. If the fill element (§14.1.2.5) 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
	The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).
filled (Shape Fill Toggle) Namespace: urn:schemas-microsoft-com:office:office	<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 (§15.1.2.5).</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 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 (§15.1.2.5).</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 (§15.1.2.5).</p>



Attributes	Description
hralign (Horizontal Rule Alignment) Namespace: urn:schemas-microsoft-com:office:office	<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 <code>ST_HrAlign</code> simple type (§14.2.3.16).</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 <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 <code>ST_TrueFalse</code> simple type (§15.1.2.5).</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)	<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>

Attributes	Description
<p>Display Toggle)</p> <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 (§15.1.2.5).</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 (§14.2.3.17).</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 (§15.1.2.5).
master (Master Element Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether the shapetype is a master element. If true, it is rendered by the rendering engine. Default is false. The possible values for this attribute are defined by the W3C XML Schema string datatype.
ole (Embedded Object Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether the shape is an embedded object. Default is false. <i>[Example:</i> <pre><v:shape ... o:ole="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalseBlank simple type (§15.1.2.6).
oleicon (Embedded Object Icon Toggle) Namespace: urn:schemas-microsoft-com:office:office	Specifies whether an embedded object is displayed as an icon. Default is false. <i>[Example:</i> <pre><v:shape ... o:oleicon="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
oned (Shape Handle Toggle) Namespace: urn:schemas-microsoft-com:office:office	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. <i>[Example:</i> <pre><v:shape ... o:oned="true" ... > </v:shape></pre> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).

Attributes	Description
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>
path (Edge Path)	<p>Specifies the line that makes up the edges of a shape. See the v attribute of the path element (§14.1.2.14) 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 (§15.1.2.5).</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>

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>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 (§14.1.2.21) 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 (§15.1.2.3).</p>
<p>stroked (Shape Stroke Toggle)</p>	<p>Specifies whether the path defining the shape is stroked with a solid line. The stroke element (§14.1.2.21) 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 (§15.1.2.5).</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 (§14.1.2.21) 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</p>

Attributes	Description										
	<p>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. • <percentage>- Value expressed as a percentage of the parent object's width. </td></tr> <tr> <td>margin-bottom</td><td> <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</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> <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</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> <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</p>										

Attributes	Description	
		<p>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. • <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.
	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-	Specifies the distance from the bottom of the shape to the text

Attributes	Description	
	distance-bottom	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 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 (§14.1.2.22):	
	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 (§14.1.2.23):

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.

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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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 (§15.1.2.5).</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 (§15.1.2.5).</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_Shapetype](#)) is located in §A.6.1. end note]

14.1.2.21 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>
```



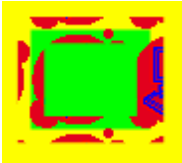

end example]



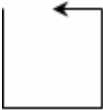
Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)


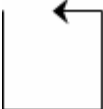
Child Elements	Subclause
bottom (Text Box Bottom Stroke)	§14.2.2.1




Child Elements	Subclause
column (Text Box Interior Stroke)	§14.2.2.6
left (Text Box Left Stroke)	§14.2.2.16
right (Text Box Right Stroke)	§14.2.2.26
top (Text Box Top Stroke)	§14.2.2.32

Attributes	Description
alhref (Alternate Image Reference) Namespace: urn:schemas-microsoft-com:office:office	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p><i>[Example:</i></p> <pre><v:stroke ... alhref="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: The shape stroke is blue:</i></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 (§15.1.2.3).</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><i>[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.</i></p> <pre><v:background fillcolor="yellow"/></pre>



Attributes	Description
	<pre> <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 (§15.1.2.3).</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><i>[Example:</i></p> <pre> <v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke> </pre>


Attributes	Description
	 <pre data-bbox="456 428 1029 527"><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></pre>  <p data-bbox="414 709 574 737"><i>end example]</i></p> <p data-bbox="414 779 1377 842">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
endarrow (Line End Arrowhead)	<p data-bbox="414 863 1401 926">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="461 972 613 1188" style="list-style-type: none"> • none • block • classic • diamond • oval • open <p data-bbox="414 1255 534 1283"><i>[Example:</i></p> <pre data-bbox="456 1325 935 1352"><v:stroke endarrow="classic"/></pre>  <p data-bbox="414 1541 574 1568"><i>end example]</i></p> <p data-bbox="414 1610 1430 1673">The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§14.1.3.8).</p>
endarrowlength (Line End Arrowhead Length)	<p data-bbox="414 1696 1435 1759">Specifies the length of the arrowhead at the end of a line. Default is medium. Allowed values are:</p> <ul data-bbox="461 1806 600 1869" style="list-style-type: none"> • short • medium


Attributes	Description
	<ul style="list-style-type: none"> • 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 (§14.1.3.7).</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> <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 (§14.1.3.9).</p>
endcap (Line End Cap)	<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>



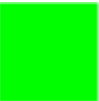
Attributes	Description
	 <p>endcap="flat"</p> <p>endcap="square"</p> <p>endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§14.1.3.10).</p>
filltype (Stroke Image Style)	<p>Specifies the kind of fill used for the background of a stroke. Default is <code>solid</code>. 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>[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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</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>


Attributes	Description
<p>Namespace: urn:schemas-microsoft-com:office:office</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 (§15.1.2.5).</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> <p>Default is no value.</p> <p>[Example:</p> <pre><v:shape ... id="myShape" ... ></pre>

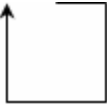
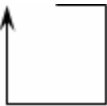
Attributes	Description								
	<p><code></v:shape></code></p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>								
imagealignshape (Stoke Image Alignment)	<p>Specifies the alignment of the stroke image. If true, the image 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><p>imagealignshape="false"</p></div> <div><p>imagealignshape="false"</p></div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table><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>	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>[Example:</p> <pre><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre> <div data-bbox="451 489 604 835">  </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 (§14.1.3.5).</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>end example]</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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>

Attributes	Description
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>  <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 (§14.1.3.11).</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>

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§14.1.3.12).</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 data-bbox="454 745 1112 850"><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 data-bbox="454 1344 1242 1480"><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 (§15.1.2.5).</p>
opacity (Stroke Opacity)	<p>Specifies the amount of transparency of a stroke. Default is 1.0.</p>

Attributes	Description
	<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>relid (Relationship to Part)</p> <p>Namespace: urn:schemas-microsoft-com:office:office</p>	<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>
<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

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 (§14.1.3.8).</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 (§14.1.3.7).</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

Attributes	Description
	<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 (§14.1.3.9).</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]

14.1.2.22 [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 (§14.1.2.14).

[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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)

Child Elements	Subclause
txbxContent (Rich Text Box Content Container)	§9.8.1.1

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 (§14.1.2.14).</p>

Attributes	Description
	<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>  <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 (§14.2.3.17).</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 (§15.1.2.5).</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' ...</pre>

Attributes	Description										
	<div data-bbox="414 254 617 348"> <pre>> </v:shape> end example]</pre> </div> <table border="1" data-bbox="414 386 1479 1890"> <thead> <tr> <th data-bbox="414 386 662 432">Property</th><th data-bbox="662 386 1479 432">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="414 432 662 701">flip</td><td data-bbox="662 432 1479 701"> <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="414 701 662 1108">height</td><td data-bbox="662 701 1479 1108"> <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="414 1108 662 1591">left</td><td data-bbox="662 1108 1479 1591"> <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="414 1591 662 1890">margin-bottom</td><td data-bbox="662 1591 1479 1890"> <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, </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. • <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,
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, 										

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 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-horizontal	<p>Specifies the horizontal positioning data for objects in WordprocessingML documents. Default is absolute. Allowed values are:</p>

Attributes	Description	
		<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 the origin.</p>
	mso-wrap-	<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</p>

Attributes	Description	
	distance-left	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.
	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

Attributes	Description	
		<p>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 top of those with lower numbers. Negative numbers are allowed.
The following properties are only used by the textbox element (§14.1.2.22):		
	Property	Description

Attributes	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 • -90
	mso-text-scale	<p>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.</p>

Attributes	Description	
	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 (§14.1.2.23):	

Attributes	Description																		
	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="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	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																		

Attributes	Description	
		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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top 		

Attributes	Description
	<ul style="list-style-type: none">• 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*]

14.1.2.23 [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 (§14.1.2.14).








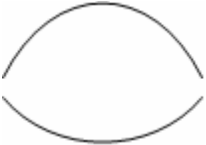
[Example:

```
<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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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" data-bbox="418 562 1205 718"> <tr> <td></td><td>fitpath="true"</td></tr> <tr> <td></td><td>fitpath="false"</td></tr> </table> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>		fitpath="true"		fitpath="false"
	fitpath="true"				
	fitpath="false"				
fitshape (Shape Fit Toggle)	<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>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>				

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>end example]</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 (§14.1.2.14) overrides this.</p> <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> <hr/> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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' ... ></pre>

Attributes	Description										
	<div data-bbox="415 247 617 315"> <code></v:shape></code> <i>end example]</i> </div> <table border="1" data-bbox="415 352 1481 1894"> <thead> <tr> <th data-bbox="415 352 662 399">Property</th><th data-bbox="662 352 1481 399">Description</th></tr> </thead> <tbody> <tr> <td data-bbox="415 399 662 667">flip</td><td data-bbox="662 399 1481 667"> <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="415 667 662 1075">height</td><td data-bbox="662 667 1481 1075"> <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 1075 662 1558">left</td><td data-bbox="662 1075 1481 1558"> <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 1558 662 1894">margin-bottom</td><td data-bbox="662 1558 1481 1894"> <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 </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. • <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
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 										

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.
	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

Attributes	Description	
		<ul style="list-style-type: none"> • 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 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</p>

Attributes	Description	
		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.
	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:



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.
	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 (§14.1.2.22):		
	Property	Description

Attributes	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 • -90
	mso-text-scale	<p>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.</p>

Attributes	Description	
	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 (§14.1.2.23):	

Attributes	Description																		
	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="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	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																		

Attributes	Description	
		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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) via the id attribute:</p> <ul style="list-style-type: none"> • flip • height • left • margin-left • margin-top 	

Attributes	Description
	<ul style="list-style-type: none"> • position • rotation • top • visibility • width • z-index <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
trim (Text Path Trim Toggle)	<p>Specifies whether extra space is removed above and below the text. If true, space reserved for ascenders and descenders is removed. Default is false.</p> <p>[Example: The shape path is duplicated as a second shape and overlaid on the textpath for illustrative purposes:</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="true"/> <v:textpath on="true" fitshape="true" string="vml" trim="true"/> </v:shape> </pre> <div style="display: flex; align-items: center; justify-content: center;">  <div style="margin-left: 10px;">trim="true"</div> </div> <div style="display: flex; align-items: center; justify-content: center; margin-top: 10px;">  <div style="margin-left: 10px;">trim="false"</div> </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 (§15.1.2.5).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_TextPath](#)) is located in §A.6.1. *end note*]

14.1.3 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.

14.1.3.1 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.
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.

Referenced By
group@editas (§14.1.2.7)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_EditAs](#)) is located in §A.6.1. *end note*]

14.1.3.2 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

Enumeration Value	Description
	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.




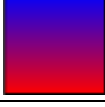
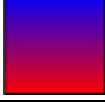
Referenced By
bottom@ext (§14.2.2.1); callout@ext (§14.2.2.2); colormenu@ext (§14.2.2.4); colormru@ext (§14.2.2.5); column@ext (§14.2.2.6); complex@ext (§14.2.2.7); diagram@ext (§14.2.2.8); extrusion@ext (§14.2.2.11); fill@ext (§14.2.2.13); idmap@ext (§14.2.2.14); left@ext (§14.2.2.16); lock@ext (§14.2.2.18); regrouptable@ext (§14.2.2.23); rel@ext (§14.2.2.24); relationtable@ext (§14.2.2.25); right@ext (§14.2.2.26); rules@ext (§14.2.2.27); shapedefaults@ext (§14.2.2.28); shapelayout@ext (§14.2.2.29); signatureline@ext (§14.2.2.30); skew@ext (§14.2.2.31); top@ext (§14.2.2.32)

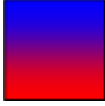
[Note: The W3C XML Schema definition of this simple type's content model ([ST_Ext](#)) is located in §A.6.1. *end note*]

14.1.3.3 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 
linear (Linear Fill)	Linear blend 
linear sigma (Linear Sigma Fill)	Linear sigma blend 
none (No Gradient Fill)	No blend 
sigma (Sigma Fill)	Sigma blend 

Enumeration Value	Description
	





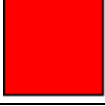

Referenced By
fill@method (§14.1.2.5)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_FillMethod](#)) is located in §A.6.1.
end note]

14.1.3.4 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 colors. 
solid (Solid Fill)	The fill pattern is a solid color. 
tile (Tiled Image)	The fill image is tiled. 




Referenced By
bottom@filltype (§14.2.2.1); column@filltype (§14.2.2.6); fill@type (§14.1.2.5); left@filltype (§14.2.2.16); right@filltype (§14.2.2.26); stroke@filltype (§14.1.2.21); top@filltype (§14.2.2.32)

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST_FillType](#)) is located in §A.6.1. *end note*]

14.1.3.5 [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. 

Referenced By
bottom@imageaspect (§14.2.2.1); column@imageaspect (§14.2.2.6); fill@aspect (§14.1.2.5); left@imageaspect (§14.2.2.16); right@imageaspect (§14.2.2.26); stroke@imageaspect (§14.1.2.21); top@imageaspect (§14.2.2.32)

[*Note*: The W3C XML Schema definition of this simple type’s content model ([ST_ImageAspect](#)) is located in §A.6.1. *end note*]

14.1.3.6 [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.




Referenced By
shadow@type (§14.1.2.18)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ShadowType](#)) is located in §A.6.1. *end note*]

14.1.3.7 [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 







Referenced By
bottom@endarrowlength (§14.2.2.1); bottom@startarrowlength (§14.2.2.1); column@endarrowlength (§14.2.2.6); column@startarrowlength (§14.2.2.6); left@endarrowlength (§14.2.2.16); left@startarrowlength (§14.2.2.16); right@endarrowlength (§14.2.2.26); right@startarrowlength (§14.2.2.26); stroke@endarrowlength (§14.1.2.21); stroke@startarrowlength (§14.1.2.21); top@endarrowlength (§14.2.2.32); top@startarrowlength (§14.2.2.32)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowLength](#)) is located in §A.6.1. *end note*]

14.1.3.8 [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 
none (No Arrowhead)	No arrowhead 
open (Open Arrowhead)	Open arrowhead 
oval (Oval Arrowhead)	Round arrowhead 




Referenced By
bottom@endarrow (§14.2.2.1); bottom@startarrow (§14.2.2.1); column@endarrow (§14.2.2.6); column@startarrow (§14.2.2.6); left@endarrow (§14.2.2.16); left@startarrow (§14.2.2.16); right@endarrow (§14.2.2.26); right@startarrow (§14.2.2.26); stroke@endarrow (§14.1.2.21); stroke@startarrow (§14.1.2.21); top@endarrow (§14.2.2.32); top@startarrow (§14.2.2.32)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowType](#)) is located in §A.6.1. *end note*]

14.1.3.9 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 




Referenced By
bottom@endarrowwidth (§14.2.2.1); bottom@startarrowwidth (§14.2.2.1); column@endarrowwidth (§14.2.2.6); column@startarrowwidth (§14.2.2.6); left@endarrowwidth (§14.2.2.16); left@startarrowwidth (§14.2.2.16); right@endarrowwidth (§14.2.2.26); right@startarrowwidth (§14.2.2.26); stroke@endarrowwidth (§14.1.2.21); stroke@startarrowwidth (§14.1.2.21); top@endarrowwidth (§14.2.2.32); top@startarrowwidth (§14.2.2.32)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeArrowWidth](#)) is located in §A.6.1. *end note*]

14.1.3.10 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 
square (Square End)	Square end 




Referenced By
bottom@endcap (§14.2.2.1); column@endcap (§14.2.2.6); left@endcap (§14.2.2.16); right@endcap (§14.2.2.26); stroke@endcap (§14.1.2.21); top@endcap (§14.2.2.32)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeEndCap](#)) is located in §A.6.1. *end note*]

14.1.3.11 ST_StrokeJoinStyle (Line Join Type)

This simple type specifies the join styles for a polyline (§14.1.2.15).

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 





Referenced By
bottom@joinstyle (§14.2.2.1); column@joinstyle (§14.2.2.6); left@joinstyle (§14.2.2.16); right@joinstyle (§14.2.2.26); stroke@joinstyle (§14.1.2.21); top@joinstyle (§14.2.2.32)



[Note: The W3C XML Schema definition of this simple type's content model ([ST_StrokeJoinStyle](#)) is located in §A.6.1. *end note*]

14.1.3.12 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 

Enumeration Value	Description
	
thinThin (Two Thin Lines)	Two thin lines 

Referenced By
bottom@linestyle (§14.2.2.1); column@linestyle (§14.2.2.6); left@linestyle (§14.2.2.16); right@linestyle (§14.2.2.26); stroke@linestyle (§14.1.2.21); top@linestyle (§14.2.2.32)

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST StrokeLineStyle](#)) is located in §A.6.1. *end note*]

14.2 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 specularity 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).

14.2.1 Table of Contents

This subclause is informative.

14.2.2 Elements 605

14.2.2.1 bottom (Text Box Bottom Stroke) 605

14.2.2.2 callout (Callout) 618

14.2.2.3 clippath (Shape Clipping Path)..... 620

14.2.2.4 colormenu (UI Default Colors)..... 623

14.2.2.5 colormru (Most Recently Used Colors) 624

14.2.2.6 column (Text Box Interior Stroke) 625

14.2.2.7 complex (Complex) 638

14.2.2.8 diagram (VML Diagram)..... 638

14.2.2.9 entry (Regroup Entry) 642

14.2.2.10 equationxml (Storage for Alternate Math Content)..... 643

14.2.2.11 extrusion (3D Extrusion) 644

14.2.2.12 FieldCodes (WordprocessingML Field Switches) 659

14.2.2.13 fill (Shape Fill Extended Properties) 660

14.2.2.14 idmap (Shape ID Map) 661

14.2.2.15 ink (Ink) 662

14.2.2.16 left (Text Box Left Stroke) 663

14.2.2.17 LinkType (Embedded Object Alternate Image Request) 675

14.2.2.18 lock (Shape Protections) 676

14.2.2.19 LockedField (Embedded Object Cannot Be Refreshed)..... 678

14.2.2.20 OLEObject (Embedded OLE Object)..... 678

14.2.2.21 proxy (Shape Reference) 681

14.2.2.22 r (Rule) 682

14.2.2.23 regrouptable (Shape Grouping History) 684

14.2.2.24 rel (Diagram Relationship)..... 685

14.2.2.25 relationtable (Diagram Relationship Table)..... 687

14.2.2.26 right (Text Box Right Stroke)..... 688

14.2.2.27	rules (Rule Set).....	700
14.2.2.28	shapedefaults (New Shape Defaults)	701
14.2.2.29	shapelayout (Shape Layout Properties)	713
14.2.2.30	signatureline (Digital Signature Line)	713
14.2.2.31	skew (Skew Transform)	718
14.2.2.32	top (Text Box Top Stroke)	719
14.2.3	Simple Types	731
14.2.3.1	ST_AlternateMathContentType (Alternate Math Content Type)	731
14.2.3.2	ST_Angle (Callout Angles).....	732
14.2.3.3	ST_BWMode (Black And White Modes)	732
14.2.3.4	ST_CalloutDrop (Callout Drop Location)	733
14.2.3.5	ST_CalloutPlacement (Callout Placement)	733
14.2.3.6	ST_ColorMode (Extrusion Color Types).....	734
14.2.3.7	ST_ConnectorType (Connector Type).....	734
14.2.3.8	ST_ConnectType (Connection Locations Type)	735
14.2.3.9	ST_ContentType (Content Type)	735
14.2.3.10	ST_DiagramLayout (Diagram Layout Type)	736
14.2.3.11	ST_ExtrusionPlane (Extrusion Planes)	736
14.2.3.12	ST_ExtrusionRender (Extrusion Rendering Types)	737
14.2.3.13	ST_ExtrusionType (Extrusion Type)	737
14.2.3.14	ST_FillType (Shape Fill Type)	738
14.2.3.15	ST_How (Alignment Type)	739
14.2.3.16	ST_HrAlign (Alignment Type).....	740
14.2.3.17	ST_InsetMode (Inset Margin Type)	740
14.2.3.18	ST_OLEDrawAspect (Embedded Object Representations).....	741
14.2.3.19	ST_OLELinkType (Embedded Object Alternate Image Request Types)	741
14.2.3.20	ST_OLEType (Embedded Connection Type)	742
14.2.3.21	ST_OLEUpdateMode (Embedded Object Update Method Type).....	742
14.2.3.22	ST_RType (Rule Type)	743
14.2.3.23	ST_ScreenSize (Screen Sizes Type)	743

End of informative text.

14.2.2 Elements

The following elements comprise the contents of the urn:schemas-microsoft-com:office:office namespace:

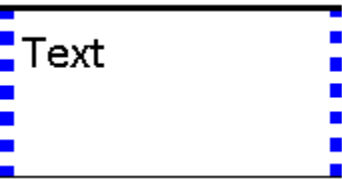
[Note: As the VML format is a format provided for backward compatibility, those VML elements defined in the same urn:schemas-microsoft-com:office:office namespace remain in that namespace as it is already used by millions of documents already using VML. *end note*]

14.2.2.1 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.

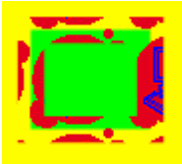

```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```





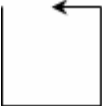

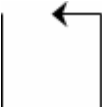
end example]


Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2); stroke (§14.1.2.21)

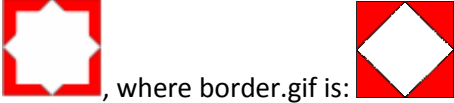
Attributes	Description
alhref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p>[Example:</p> <pre><v:stroke ... alhref="myimage.pcz" ... > </v:stroke></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C 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>




Attributes	Description
<p>color2 (Stroke Alternate Pattern Color)</p>	<p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</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>  <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 (§15.1.2.3).</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 • dash • longdash • dashdot • longdashdot • longdashdotdot

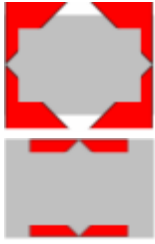
Attributes	Description
	<p>A custom-defined dash pattern can also be specified using a series of numbers. These define the length of the dash (the drawn part of the stroke) and the length of the space between the dashes. The lengths are relative to the line width: a length of 1 is equal to the line width. 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 <p>[Example:</p> <pre><v:stroke endarrow="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 (§14.1.3.8).</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> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</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> <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</p>




Attributes	Description
<p>endcap (Line End Cap)</p>	<p>type (§14.1.3.9).</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> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p style="margin-left: 200px;">endcap="flat"</p> <p style="margin-left: 200px;">endcap="square"</p> <p style="margin-left: 200px;">endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the <code>ST_StrokeEndCap</code> simple type (§14.1.3.10).</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 <code>ST_Ext</code> simple type (§14.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is <code>solid</code>. 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>[Example:</p>

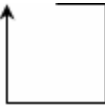
Attributes	Description
	<pre> <v:shape style="width:50;height:50" strokecolor="red" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke filltype="frame" weight="10pt" src="border.gif"/> </v:shape> </pre>  <p>, where border.gif is:</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>


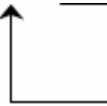
Attributes	Description								
	<p>[<i>Example</i>: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver" style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" 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 (§15.1.2.5).</p>								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table><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>[<i>Example</i>:</p> <pre><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre>  <p>imagealignshape="ignore"</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.								

Attributes	Description
	 <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 (§14.1.3.5).</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 (§15.1.2.5).</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"></pre>

Attributes	Description
	<div data-bbox="453 247 951 315"><pre><v:stroke joinstyle="bevel"/> </v:polyline></pre></div> <div data-bbox="453 348 683 480"></div> <div data-bbox="704 453 972 487"><p>joinstyle="round"</p></div> <div data-bbox="453 485 683 617"></div> <div data-bbox="704 585 972 619"><p>joinstyle="bevel"</p></div> <div data-bbox="453 617 683 749"></div> <div data-bbox="704 726 972 760"><p>joinstyle="miter"</p></div> <div data-bbox="415 800 574 833"><p><i>end example]</i></p></div> <div data-bbox="415 871 1458 938"><p>The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§14.1.3.11).</p></div>
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> <div data-bbox="453 1299 1175 1365"><pre><v:stroke linestyle="thickThin" weight="5pt"> </v:stroke></pre></div> <div data-bbox="415 1398 583 1493"></div> <div data-bbox="415 1533 574 1566"><p><i>end example]</i></p></div> <div data-bbox="415 1604 1463 1671"><p>The possible values for this attribute are defined by the ST_StrokeLineStyle simple type (§14.1.3.12).</p></div>
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>

Attributes	Description
	<pre><v:stroke joinstyle="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 (§15.1.2.5).</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</p>

Attributes	Description
src (Stroke Image Location)	<p>datatype.</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>
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>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§14.1.3.8).</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>

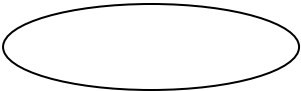
Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start of a line. Default is medium. Allowed values 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 (§14.1.3.9).</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]

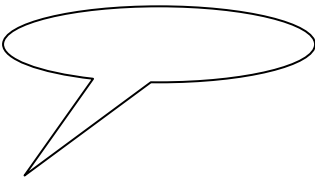
14.2.2.2 **callout (Callout)**

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]

Parent Elements	
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)	

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 (§15.1.2.5).
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 (§14.2.3.2).
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

Attributes	Description
	datatype.
drop (Callout drop position)	<p>Specifies where the drop of a callout is placed.</p> <p>The possible values for this attribute are defined by the ST_CalloutDrop simple type (§14.2.3.4).</p>
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 (§15.1.2.5).</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 (§14.1.3.2).</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 (§15.1.2.5).</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 (§15.1.2.5).</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 (§15.1.2.5).</p>
on (Callout toggle)	Specifies whether a shape is a callout. Default is false.

Attributes	Description
	The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
textborder (Callout text border toggle)	Specifies whether a callout has a text border. Default is true. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
type (Callout type)	Specifies the type of callout. Default is rectangle. Allowed values are: <ul style="list-style-type: none"> rectangle roundedrectangle oval cloud 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_Callout](#)) is located in §A.6.2. *end note*]

14.2.2.3 [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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapetype (§14.1.2.20)

Attributes	Description
v (Path Definition)	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.

Attributes	Description																																
	<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 (§14.1.2.6) 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. <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	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).
	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*]

14.2.2.4 [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]

Parent Elements
shapedefaults (§14.2.2.28)

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 (§14.1.3.2).
extrusioncolor (Default extrusion color)	The default color associated with the 3D extrusion of a VML shape. Default is "#000000". The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).
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 (§15.1.2.3).
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 (§15.1.2.3).
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 (§15.1.2.3).

[Note: The W3C XML Schema definition of this element's content model ([CT_ColorMenu](#)) is located in §A.6.2. end note]

14.2.2.5 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]

Parent Elements
shapedefaults (§14.2.2.28)

Attributes	Description
colors (Recent colors)	<p>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 (§15.1.2.3) for a full description.</p> <p><i>[Example:</i></p> <pre><o:colormru v:ext="edit" colors="#a01aae,#456b69,#06f,#a1ae24,#d57811"/></pre> <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) 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 (§14.1.3.2).</p>

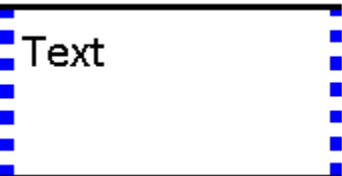
[Note: The W3C XML Schema definition of this element's content model ([CT_ColorMru](#)) is located in §A.6.2. *end note]*

14.2.2.6 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.

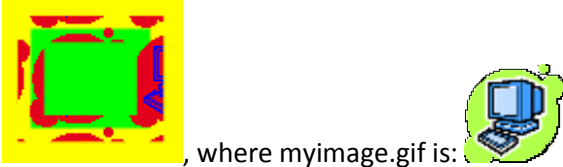
```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```



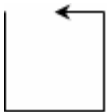



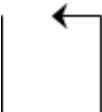
end example]


Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2); stroke (§14.1.2.21)

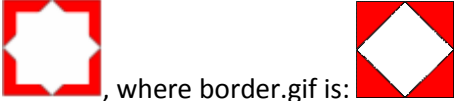
Attributes	Description
althref (Alternate Image Reference)	<p>Specifies an alternate reference for an image in Macintosh PICT format.</p> <p><i>[Example:</i></p> <pre><v:stroke ... althref="myimage.pcz" ... > </v:stroke></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined 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: The shape stroke is blue:</i></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 (§15.1.2.3).</p>





Attributes	Description
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>[<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</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</p>


Attributes	Description
	<p>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 <p>[Example:</p> <pre><v:stroke endarrow="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 (§14.1.3.8).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§14.1.3.9).</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


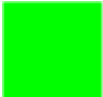

Attributes	Description
	<ul style="list-style-type: none"> • square • round <p>[Example:</p> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p>endcap="flat"</p> <p>endcap="square"</p> <p>endcap="round"</p> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§14.1.3.10).</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>[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. <i>end rationale</i>]</p> <p>The possible values for this attribute are defined by the ST_Ext simple type (§14.1.3.2).</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"</pre>

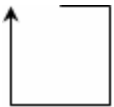
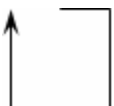
Attributes	Description
	<pre>src="border.gif"/> </v:shape></pre>  <p>, where border.gif is:</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</p>
forcedash (Force Dashed Outline)	<p>Specifies whether a dashed outline is used 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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 <code>true</code>, the image is aligned with the shape. Otherwise, it is aligned with the containing scope. Default is <code>true</code>.</p> <p>[Example: The top position offset shifts the image alignment relative to the containing window:</p> <pre><v:shape fillcolor="silver"</pre>

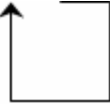
Attributes	Description								
	<pre>style="top:20;width:50;height:50" path="m 0,0 l 0,1000 1000,1000 1000,0 x e"> <v:stroke imagealignshape="false" weight="20pt" filltype="tile" src="myimage.gif"/> </v:shape></pre> <div><p>imagealignshape="false"</p></div> <div><p>imagealignshape="false"</p></div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is ignore. Allowed values are:</p> <table><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><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre> <div><p>imagealignshape="ignore"</p></div> <div><p>imagealignshape="atleast"</p></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><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 (§14.1.3.5).</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 (§15.1.2.5).</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>

Attributes	Description
	 <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 (§14.1.3.11).</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 (§14.1.3.12).</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>

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><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 (§15.1.2.5).</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>

Attributes	Description
	<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> <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 (§14.1.3.8).</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> 

Attributes	Description
	<p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</p>
<p>startarrowwidth (Line Start Arrowhead Width)</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 (§14.1.3.9).</p>
<p>title (Stroke Title)</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><i>end example]</i></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_StrokeChild](#)) is located in §A.6.2.
end note]

14.2.2.7 complex (Complex)

This element specifies that a shapetype contains fragments.

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2); shapetype (§14.1.2.20)

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. [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 (§14.1.3.2).

[Note: The W3C XML Schema definition of this element’s content model (CT_Complex) is located in §A.6.2. end note]

14.2.2.8 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:

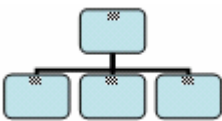
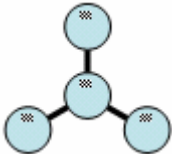
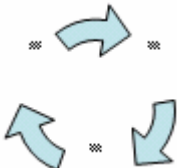


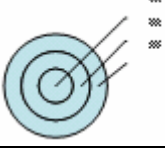
Diagram Type	Example (non-normative)
Organization chart	
Radial	
Cycle	

Diagram Type	Example (non-normative)
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.

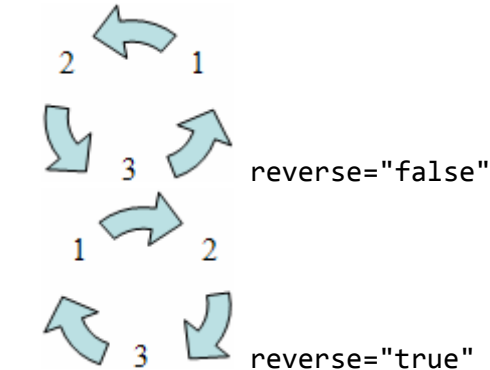
Parent Elements
background (Part 1, §17.2.1); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2)

Child Elements	Subclause
relationtable (Diagram Relationship Table)	§14.2.2.25

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 (§15.1.2.5).</p>
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>

Attributes	Description
	<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 (§15.1.2.5).</p>
<p>constrainbounds (Diagram Layout Extents)</p>	<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>
<p>dgmbasetextscale (Diagram Base Font Size)</p>	<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>
<p>dgmfontsize (Diagram Font Size)</p>	<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> <p>The possible values for this attribute are defined by the W3C XML Schema integer datatype.</p>

Attributes	Description
dgmscalex (Diagram Layout X Scale)	<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>
dgmscaley (Diagram Layout Y Scale)	<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>
dgmstyle (Diagram Style Options)	<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>
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>[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 (§14.1.3.2).</p>

Attributes	Description
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 (§15.1.2.5).</p>

[Note: The W3C XML Schema definition of this element’s content model ([CT_Diagram](#)) is located in §A.6.2. end note]

14.2.2.9 entry (Regroup Entry)

This element specifies a single entry in a regrouptable (§14.2.2.23). 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]

Parent Elements
regrouptable (§14.2.2.23)

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*]

14.2.2.10 `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 §22.1 of 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>
        <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

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shape (§14.1.2.19); shapeDefaults (§9.7.2.2)

Child Elements	Subclause
Any element in any namespace	n/a

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 (§14.2.3.1).

[Note: The W3C XML Schema definition of this element's content model ([CT_EquationXml](#)) is located in §A.6.2. *end note*]

14.2.2.11 [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>
```


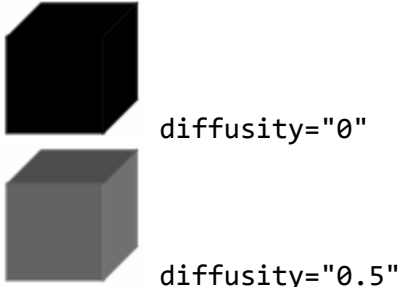


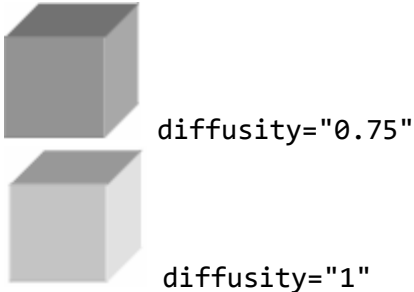

end example]



Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)




Attributes	Description
autorotationcenter (Center of Rotation Toggle)	<p>Specifies whether the center of rotation is the geometric center of the extrusion. Default is false. If true, the geometric center of an extruded shape is (0,0,0). If false, the center of rotation is determined by the rotationcenter attribute.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
backdepth (Backward Extrusion Depth)	<p>Specifies the amount of backward extrusion. Default is 36 pt, default units are points.</p> <p>[Example:</p> <pre><o:extrusion on="true" backdepth="15pt"> </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>
brightness (Brightness)	<p>Specifies the overall brightness of a scene. Default is 0.3. This numeric value can also be specified in 1/65536ths 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 range from 0 to 1 (0f to 65536f), where 0 implies darkness and 1 implies light saturation.</p> <p>[Example:</p>


Attributes	Description
	<div><pre><o:extrusion on="true" brightness="0.4"> </o:extrusion></pre></div> <div><div>brightness="0"</div></div> <div><div>brightness="25000f"</div></div> <div><div>brightness="0.4"</div></div> <div><div>brightness="0.75"</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>
color (Extrusion Color)	<div><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><div><pre><o:extrusion on="true" color="lime" colormode="custom"> </o:extrusion></pre></div><div><div><i>end example]</i></div></div><div><p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p></div></div>


Attributes	Description
colormode (Extrusion Color Mode)	<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><o:extrusion on="true" color="lime" colormode="auto"> </o:extrusion></pre>  <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ColorMode simple type (§14.2.3.6).</p>
diffusivity (Diffuse Reflection)	<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 range from 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. end example]</p> <p>This quantity is not specified using units. The numeric values range from 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 diffusivity 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:</p> <pre><o:extrusion on="true" diffusivity=".75"> </o:extrusion></pre>  <p>diffusivity="0"</p> <p>diffusivity="0.5"</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>
edge (Simulated Bevel)	<p>Specifies the apparent bevel of the extrusion edges. Default is 1 point.</p> <p>[Example:</p> <pre data-bbox="451 871 998 934"><o:extrusion on="true" edge="2pt"> </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>
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vmf	<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 (§14.1.3.2).</p>
facet (Faceting Quality)	<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 range from 1 to 65536, where 1 implies extremely low quality curve approximation and 65536 implies extremely high quality.</p> <p>[Example:</p>


Attributes	Description
	<pre data-bbox="451 285 1047 348"><o:extrusion on="true" facet="65536"> </o:extrusion></pre> <div data-bbox="456 386 686 667">  </div> <p data-bbox="415 716 574 747"><i>end example]</i></p> <p data-bbox="415 787 1375 850">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
foredepth (Forward Extrusion)	<p data-bbox="415 871 1398 903">Specifies the amount of forward extrusion. Default is 0 pt, default units are points.</p> <p data-bbox="415 942 534 974"><i>[Example:</i></p> <pre data-bbox="451 1014 1094 1077"><o:extrusion on="true" foredepth="25pt"> </o:extrusion></pre> <div data-bbox="415 1115 574 1274">  </div> <p data-bbox="415 1314 574 1346"><i>end example]</i></p> <p data-bbox="415 1386 1375 1449">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 1465 1446 1535">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 1575 1479 1638"><i>[Example:</i> The front face is colored as if the shape were not extruded and lit by a 3-D light source:</p> <pre data-bbox="451 1680 1110 1743"><o:extrusion on="true" lightface="false"> </o:extrusion></pre>

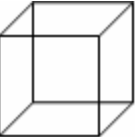

Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
lightharsh (Primary Light Harshness Toggle)	<p>Specifies whether the primary light source is harsh. If <code>false</code>, shadow boundaries are diffused. Default is <code>true</code>.</p> <p>[<i>Example:</i> The secondary light source is turned off so only the primary has an effect:</p> <pre><o:extrusion on="true" lightharsh="false" lightlevel2="0"> </o:extrusion></pre>  <p>lightharsh="false"</p>  <p>lightharsh="true"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
lightharsh2 (Secondary Light Harshness Toggle)	<p>Specifies whether the secondary light source is harsh. If <code>false</code>, shadow boundaries defined by the secondary light source are diffused. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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 1/65536ths if a trailing "f" is supplied (as "f" indicates the value is a fraction). [<i>Example:</i> A value of "52429f" represents 52429/65536 or 0.8. <i>end example]</i></p> <p>This quantity is not specified using units. The numeric values range from 0 to 1 (0f to 65536f), where 0 implies no direct light and 1 implies saturated direct light.</p> <p>[<i>Example:</i> The secondary light source is turned off so only the primary has an effect:</p>



Attributes	Description
	<pre data-bbox="456 285 1062 380"><o:extrusion on="true" lightlevel=".5" lightlevel2="0"> </o:extrusion></pre> <div data-bbox="456 422 867 856">  <div data-bbox="613 533 834 562">lightlevel="1"</div> <div data-bbox="613 680 867 709">lightlevel="0.5"</div> <div data-bbox="613 827 834 856">lightlevel="0"</div> </div> <p data-bbox="415 898 574 928"><i>end example]</i></p> <p data-bbox="415 970 1377 1033">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lightlevel2 (Secondary Light Intensity)	<p data-bbox="415 1052 1471 1188">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 data-bbox="415 1230 1425 1293">This quantity is not specified using units. The numeric values range from 0 to 1 (0f to 65536f), where 0 implies no direct light and 1 implies saturated direct light.</p> <p data-bbox="415 1335 1377 1398">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
lightposition (Primary Light Position)	<p data-bbox="415 1419 1471 1556">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 data-bbox="415 1598 1471 1661">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 data-bbox="415 1703 1414 1732"><i>[Example: The secondary light source is turned off so only the primary has an effect:</i></p> <pre data-bbox="456 1776 1062 1871"><o:extrusion on="true" lightlevel2="0" lightposition="7000,-13000,20000"> </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>
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> <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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
metal (Metallic Surface Toggle)	<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>

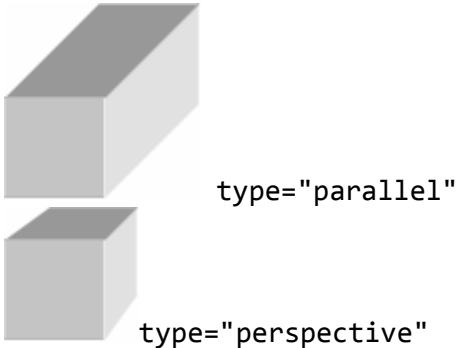

Attributes	Description
	<pre> <o:extrusion on="true" metal="true" lightposition="10000,-10000,10000" lightlevel2="0" specular="1.2" diffusivity="0.6"> </o:extrusion> </pre>  <p>metal="true"</p>  <p>metal="false"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
on (Extrusion Toggle)	<p>Specifies whether an extrusion is displayed. Default is false.</p> <p>[Example:</p> <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 (§15.1.2.5).</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>


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>				
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• yz <p>[Example:</p> <pre><o:extrusion on="true" plane="yz" backdepth="100pt"> </o:extrusion></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_ExtrusionPlane simple type (§14.2.3.11).</p>				
render (Extrusion Render Mode)	<p>Specifies the rendering mode of the extrusion. Default is solid. Allowed values are:</p> <table><tr><th>Value</th><th>Description</th></tr><tr><td>solid</td><td>Rendering displays a solid shape.</td></tr></table>	Value	Description	solid	Rendering displays a solid shape.
Value	Description				
solid	Rendering displays a solid shape.				

Attributes	Description				
	<table border="1" data-bbox="418 245 1258 405"> <tr> <td data-bbox="418 245 626 317">wireframe</td><td data-bbox="626 245 1258 317">Rendering displays a wireframe shape.</td></tr> <tr> <td data-bbox="418 317 626 405">boundingcube</td><td data-bbox="626 317 1258 405">Rendering displays the bounding cube that contains the shape.</td></tr> </table> <p data-bbox="418 447 535 478"><i>[Example:</i></p> <pre data-bbox="451 520 1128 583" style="background-color: #f0f0f0; padding: 5px;"> <o:extrusion on="true" render="wireframe"> </o:extrusion></pre>  <p data-bbox="418 793 576 825"><i>end example]</i></p> <p data-bbox="418 867 1485 930">The possible values for this attribute are defined by the ST_ExtrusionRender simple type (§14.2.3.12).</p>	wireframe	Rendering displays a wireframe shape.	boundingcube	Rendering displays the bounding cube that contains the shape.
wireframe	Rendering displays a wireframe shape.				
boundingcube	Rendering displays the bounding cube that contains the shape.				
rotationangle (X-Y Rotation Angle)	<p data-bbox="418 947 1485 1052">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> <p data-bbox="418 1094 1485 1199">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 data-bbox="418 1241 535 1272"><i>[Example:</i></p> <pre data-bbox="451 1314 1177 1409" style="background-color: #f0f0f0; padding: 5px;"> <o:extrusion on="t" lockrotationcenter="true" rotationangle="10,20"> </o:extrusion></pre>  <p data-bbox="418 1608 576 1640"><i>end example]</i></p> <p data-bbox="418 1682 1372 1745">The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>				
rotationcenter (Rotation Center)	<p data-bbox="418 1759 1485 1833">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 data-bbox="418 1875 1404 1896">The position "0,0,0" is at the center of the shape. Positive numbers are to the right,</p>				

Attributes	Description
	<p>down and toward the viewer, respectively.</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
shininess (Shininess)	<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>
skewamt (Extrusion Skew)	<p>Specifies the amount of skew, or length, of a parallel extrusion. Default is 50%. Applies only if the extrusion type is <code>parallel</code>. This attribute and <code>backdepth</code> interact to create the actual extrusion length. Allowed values range from 0 (0%) to 1 (100%).</p> <p>[Example:</p> <pre data-bbox="451 951 1062 1014"><o:extrusion on="true" skewamt="100%"> </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>
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 <code>parallel</code>.</p> <p>[Example:</p> <pre data-bbox="451 1629 1062 1692"><o:extrusion on="true" skewangle="25"> </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 float datatype.</p>						
<p>specularity (Specularity)</p>	<p>Specifies the specularity of an extruded shape, defined as the ratio of incident light to specularly reflected light. Default is 0. Normal values range from 0 to 1. This numeric value can also be specified in 1/65536ths 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><o:extrusion on="true" specularity="1" lightposition="10000, -10000, 10000" lightlevel2="0"> </o:extrusion></pre> <div style="display: flex; align-items: center; justify-content: center;">  <div style="margin-left: 10px;">specularity="1"</div> </div> <div style="display: flex; align-items: center; justify-content: center; margin-top: 10px;">  <div style="margin-left: 10px;">specularity="0"</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>						
<p>type (Extrusion Type)</p>	<p>Specifies the way that the shape is extruded. Default is parallel. Allowed values are:</p> <table border="1" data-bbox="415 1619 1258 1892"> <thead> <tr> <th>Value</th><th>Description</th></tr> </thead> <tbody> <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,</td></tr> </tbody> </table>	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,
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,						

Attributes	Description		
	<table><tr><td></td><td>which is the same as the vanishing point for unrotated objects.</td></tr></table> <p>[Example:</p> <pre><o:extrusion on="true" type="parallel" backdepth="100pt"> </o:extrusion></pre> <div></div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ExtrusionType simple type (§14.2.3.13).</p>		which is the same as the vanishing point for unrotated objects.
	which is the same as the vanishing point for unrotated objects.		
viewpoint (Extrusion Viewpoint)	<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> <div></div> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>		
viewpointorigin	Specifies the origin of the viewpoint vector for perspective extrusions. This is the origin		

Attributes	Description
(Extrusion Viewpoint Origin)	<p>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 range from 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 data-bbox="451 569 1109 701"><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*]

14.2.2.12 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.

Parent Elements
OLEObject (§14.2.2.20)

14.2.2.13 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 (§14.1.2.5).

Parent Elements
background (Part 1, §17.2.1); fill (§14.1.2.5); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2)

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 (§14.1.3.2).</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> <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 (§14.2.3.14).</p>

[*Note*: The W3C XML Schema definition of this element’s content model (CT_Fill) is located in §A.6.2. *end note*]

14.2.2.14 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.

Parent Elements
shapelayout (§14.2.2.29)

Attributes	Description
data (Shape IDs)	<p>Specifies the data the application uses to generate shape IDs.</p> <p>[<i>Example:</i> 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)	<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>[<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 (§14.1.3.2).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_IdMap](#)) is located in §A.6.2. *end note]*

14.2.2.15 `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+YFASAAaAwAAAAAMA..." annotation="t"
    contentType="application/x-ms-ink"/>
</v:shape>
```

end example]

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); shape (§14.1.2.19); shapeDefaults (§9.7.2.2)

Attributes	Description
annotation (Annotation Flag)	<p>Specifies whether the ink object was created as an annotation rather than through pen input. Default is false. [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. An ink object that represents primary user input through a pen can be left visible. <i>end rationale</i>]</p> <p>[Example:</p> <pre><o:ink ... annotation="true"> </o:ink></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
contentType (Content Type)	<p>Specifies the format of the ink content stored in the i attribute. The syntax is a content type as defined in IETF RFC 2616.</p> <p>If this attribute is omitted, the application should attempt to determine the content type by reading the contents of the i element.</p> <p>The possible values for this attribute are defined by the ST_ContentType simple type</p>

Attributes	Description
	(§14.2.3.9).
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 contentType attribute, but are optional and can be ignored if not recognized.</p> <p>[Example:</p> <pre> <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]

14.2.2.16 left (Text Box Left Stroke)

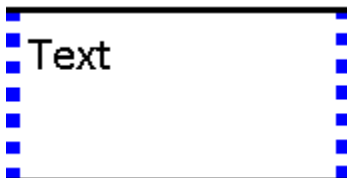
This element specifies the stroke properties for the left border of a text box. It entirely supercedes its parent stroke element if its on attribute is true. Thus the default value of an unspecified attribute overrides a value specified in the parent. If the on attribute is false or not specified, the 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>

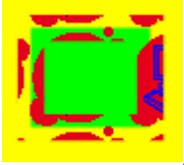


```


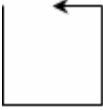



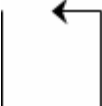
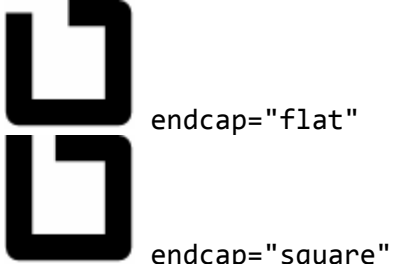
end example]




Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2); stroke (§14.1.2.21)


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: The shape stroke is blue:</i></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 (§15.1.2.3).</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><i>[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.</i></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"></pre>





Attributes	Description
	<pre><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 (§15.1.2.3).</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><v:stroke dashstyle="longdashdotdot" weight="2pt"> </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>
<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§14.1.3.8).</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>



Attributes	Description
	 <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</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> <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 (§14.1.3.9).</p>
endcap (Line End Cap)	<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><i>end example]</i></p>

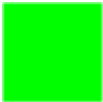
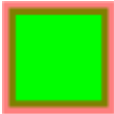
Attributes	Description
	 <p>endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§14.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm1</p>	<p>Specifies an optional value 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 (§14.1.3.2).</p>
<p>filltype (Stroke Image Style)</p>	<p>Specifies the kind of fill used for the background of a stroke. Default is <code>solid</code>. 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 style="display: flex; align-items: center;">  , where border.gif is:  </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</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>


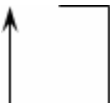
Attributes	Description
	<p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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>

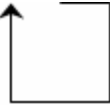
Attributes	Description								
	<div></div> <div>imagealignshape="false"</div> <div>end example]</div> <div>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</div>								
imageaspect (Stroke Image Aspect Ratio)	<div>Specifies how the stroke image aspect 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><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></div> <div><div></div><div>imagealignshape="ignore"</div><div></div><div>imagealignshape="atleast"</div><div></div><div>imagealignshape="atmost"</div></div> <div>end example]</div> <div>The possible values for this attribute are defined by the ST_ImageAspect simple type (§14.1.3.5).</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.								
imagesize (Stroke Image Size)	<div>Specifies the size of the image for the stroke. Default is the size of the image.</div> <div>[Example:</div> <div><v:stroke ... imagesize="10pt,10pt" ... /></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>
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 (§15.1.2.5).</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 data-bbox="451 1360 974 1774">  <div data-bbox="706 1465 974 1774"> <p>joinstyle="round"</p> <p>joinstyle="bevel"</p> <p>joinstyle="miter"</p> </div> </div> <p><i>end example]</i></p>

Attributes	Description
	The possible values for this attribute are defined by the ST_StrokeJoinStyle simple type (§14.1.3.11).
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 (§14.1.3.12).</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 (§15.1.2.5).</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 (§14.1.3.8).</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 (§14.1.3.7).</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 (§14.1.3.9).</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="451 751 971 821"><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]

14.2.2.17 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 (§14.2.3.19).

Parent Elements
OLEObject (§14.2.2.20)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_OLELinkType](#)) is located in §A.6.2.
end note]

14.2.2.18 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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)

Attributes	Description
adjusthandles (Handles Lock)	Specifies whether the handles of a shape are locked from being edited. Default is <code>false</code> . The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§15.1.2.5).
aspectratio (Aspect Ratio Lock)	Specifies whether the aspect ratio of a shape is locked from being edited. Default is <code>false</code> . The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§15.1.2.5).
cropping (Cropping Lock)	Specifies whether cropping of a shape is locked from being edited. Default is <code>false</code> . The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§15.1.2.5).
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:	[<i>Rationale:</i> 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 (§14.1.3.2).
grouping (Grouping Lock)	Specifies whether a shape is locked from being grouped. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
position (Position Lock)	Specifies whether the position of a shape is locked from being edited. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
rotation (Rotation Lock)	Specifies whether the rotation of a shape is locked from being edited. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
selection (Selection Lock)	Specifies whether the shape is locked from being selectable in an editor. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
shapetype (AutoShape Type Lock)	Specifies whether the AutoShape type is locked from being edited. Default is false. If true, the type of an AutoShape cannot be changed in a graphical editor. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
text (Text Lock)	Specifies whether the text attached to a shape is locked from being edited. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
ungrouping (Ungrouping Lock)	Specifies whether a grouped shape is locked from being ungrouped. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).
verticies (Vertices Lock)	Specifies whether the vertices of a path are locked from being edited. Default is false. The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).

[Note: The W3C XML Schema definition of this element's content model ([CT_Lock](#)) is located in §A.6.2. *end note*]

14.2.2.19 **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 (§15.1.2.6).

Parent Elements
OLEObject (§14.2.2.20)

[*Note:* The W3C XML Schema definition of this element's content model (ST_TrueFalseBlank) is located in §A.7.9. *end note*]

14.2.2.20 **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]

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2)

Child Elements	Subclause
FieldCodes (WordprocessingML Field Switches)	§14.2.2.12
LinkType (Embedded Object Alternate Image Request)	§14.2.2.17
LockedField (Embedded Object Cannot Be Refreshed)	§14.2.2.19

Attributes	Description
DrawAspect (Embedded Object Representation)	<p>Specifies how the embedded object is represented visually in the application.</p> <p>[Example:</p> <pre><o:OLEObject ... DrawAspect="Content"> </o:OLEObject></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_OLEDrawAspect simple type (§14.2.3.18).</p>
id (Relationship) Namespace: .../officeDocument /2006/relationships	<p>Specifies the actual OLE object using a standard part relationship lookup.</p> <p>[Example:</p> <pre><o:OLEObject ... r:id="rId5"> </o:OLEObject></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>
ObjectID (Unique ID for Embedded Object)	<p>Specifies a unique ID identifying the embedded object.</p> <p>[Example: 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>

Attributes	Description
	<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>This example is taken from the kde.org web site (http://developer.kde.org/documentation/tutorials/dot/writing-plugins.html), and contains a tutorial on building the plugin referenced by the above markup. <i>end example</i></p> <p>...</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>[Example:</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 Shape)	<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 of the placeholder shape.</p> <p>[Example:</p> <pre><o:OLEObject ... ShapeID="_x0000_i1025"> </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>
Type (Embedded	Specifies the kind of embedded object connection.

Attributes	Description
Object Type)	<p>[Example:</p> <pre><o:OLEObject ... Type="Embed"> </o:OLEObject></pre> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_OLEType simple type (§14.2.3.20).</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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_OLEUpdateMode simple type (§14.2.3.21).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_OLEObject](#)) is located in §A.6.2. *end note]*

14.2.2.21 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]

Parent Elements
r (§14.2.2.22)

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 (§15.1.2.6).</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 (§15.1.2.6).</p>

[Note: The W3C XML Schema definition of this element's content model ([CT_Proxy](#)) is located in §A.6.2. *end note*]

14.2.2.22 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]

Parent Elements
rules (§14.2.2.27)

Child Elements	Subclause
proxy (Shape Reference)	§14.2.2.21

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 (§14.2.3.15).</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

Attributes	Description
	<ul style="list-style-type: none"> • callout • connector • align <p>The possible values for this attribute are defined by the ST_RType simple type (§14.2.3.22).</p>

[Note: The W3C XML Schema definition of this element's content model (CT_R) is located in §A.6.2. *end note*]

14.2.2.23 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]

Parent Elements
shapelayout (§14.2.2.29)

Child Elements	Subclause
entry (Regroup Entry)	§14.2.2.9

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	[<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 (§14.1.3.2).

[*Note*: The W3C XML Schema definition of this element's content model ([CT_RegroupTable](#)) is located in §A.6.2. *end note*]

14.2.2.24 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]

Parent Elements
relationtable (§14.2.2.25)

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 (§14.1.3.2).</p>
idcntr (Diagram Relationship Center Shape)	<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><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>
iddest (Diagram Relationship Destination Shape)	<p>Specifies the identifier of the shape at the destination of the relationship.</p> <p>[<i>Example</i>:</p>

Attributes	Description
	<pre><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>
idsrc (Diagram Relationship Source Shape)	<p>Specifies the identifier of the shape at the source of the relationship.</p> <p>[Example:</p> <pre><o:rel ... idsrc="#s_1036"> </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>

[Note: The W3C XML Schema definition of this element's content model ([CT_Relation](#)) is located in §A.6.2. *end note]*

14.2.2.25 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]

Parent Elements
diagram (§14.2.2.8)

Child Elements	Subclause
rel (Diagram Relationship)	§14.2.2.24

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	[<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 (§14.1.3.2).

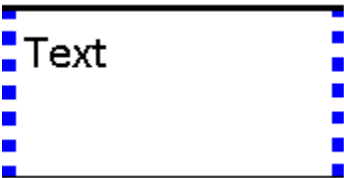
[*Note*: The W3C XML Schema definition of this element’s content model ([CT_RelationTable](#)) is located in §A.6.2. *end note*]

14.2.2.26 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>
```

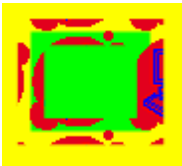





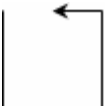
end example]


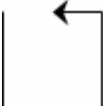
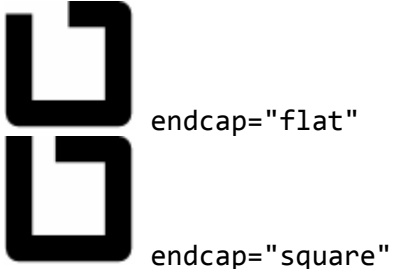
Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict




Parent Elements
(§9.5.1); shapeDefaults (§9.7.2.2); stroke (§14.1.2.21)


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: The shape stroke is blue:</i></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 (§15.1.2.3).</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><i>[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.</i></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"</pre>





Attributes	Description
	<div data-bbox="453 247 1047 315"><pre>src="myimage.gif" color2="blue"/> </v:shape></pre></div> <div data-bbox="414 348 594 512"></div> <div data-bbox="594 485 867 518">, where myimage.gif is:</div> <div data-bbox="867 411 972 512"></div> <div data-bbox="414 556 574 588"><i>end example]</i></div> <div data-bbox="414 625 1395 695"><p>The possible values for this attribute are defined by the ST_ColorType simple type (§15.1.2.3).</p></div>
<div data-bbox="142 709 354 779">dashstyle (Stroke Dash Pattern)</div>	<div data-bbox="414 709 1474 743"><p>Specifies the dot and dash pattern for a stroke. Default is solid. Pre-defined values are:</p></div> <div data-bbox="461 783 730 1146"><ul style="list-style-type: none">• solid• shortdash• shortdot• shortdashdot• shortdashdotdot• dot• dash• longdash• dashdot• longdashdot• longdashdotdot</div> <div data-bbox="414 1184 1455 1465"><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></div> <div data-bbox="414 1505 534 1537"><i>[Example:</i></div> <div data-bbox="453 1575 1060 1675"><pre><v:stroke dashstyle="0 2" weight="3pt" endcap="round"> </v:stroke></pre></div> <div data-bbox="414 1707 524 1822"></div> <div data-bbox="453 1858 1029 1890"><pre><v:stroke dashstyle="longdashdotdot"</pre></div>


Attributes	Description
	<pre>weight="2pt"> </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>
<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><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowType simple type (§14.1.3.8).</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>



Attributes	Description
	 <i>end example]</i> The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).
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>  <i>end example]</i> The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§14.1.3.9).
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> 

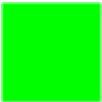
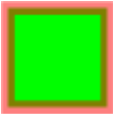
Attributes	Description
	 <p>endcap="round"</p> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§14.1.3.10).</p>
<p>ext (VML Extension Handling Behavior)</p> <p>Namespace: urn:schemas-microsoft-com:vm1</p>	<p>Specifies an optional value 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 (§14.1.3.2).</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> <div style="display: flex; align-items: center;">  , where border.gif is:  </div> <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_FillType simple type (§14.1.3.4).</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 false.</p>

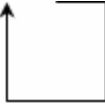

Attributes	Description
	<p>Used by PresentationML placeholders to draw a dashed outline when there is no line and no fill for a shape.</p> <p>[Example:</p> <pre><v:shape ... o:forcedash="true" ... > </v:shape></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</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> <div></div> <p>imagealignshape="false"</p>

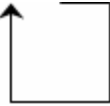
Attributes	Description								
	 <code>imagealignshape="false"</code> <i>end example]</i> The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).								
imageaspect (Stroke Image Aspect Ratio)	<p>Specifies how the stroke image aspect ratio is preserved. Default is <code>ignore</code>. Allowed values are:</p> <table border="1" data-bbox="418 705 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 (§14.1.3.5).	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>
<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 (§15.1.2.5).</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 1360 974 1774">  <div data-bbox="698 1465 974 1501">joinstyle="round"</div> <div data-bbox="698 1600 974 1635">joinstyle="bevel"</div> <div data-bbox="698 1738 974 1774">joinstyle="miter"</div> </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 (§14.1.3.11).</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 (§14.1.3.12).</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 (§15.1.2.5).</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 (§14.1.3.8).</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 (§14.1.3.7).</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 (§14.1.3.9).</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="451 751 971 821"><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]

14.2.2.27 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]

Parent Elements
shapelayout (§14.2.2.29)

Child Elements	Subclause
r (Rule)	§14.2.2.22

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	[<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 (§14.1.3.2).

[*Note*: The W3C XML Schema definition of this element's content model ([CT_Rules](#)) is located in §A.6.2. *end note*]

14.2.2.28 [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:

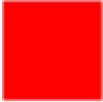

```
<o:shapedefaults v:ext="edit" spidmax="1029"/>
```

end example]

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2)

Child Elements	Subclause
callout (Callout)	§14.2.2.2
colormenu (UI Default Colors)	§14.2.2.4
colormru (Most Recently Used Colors)	§14.2.2.5
extrusion (3D Extrusion)	§14.2.2.11
fill (Shape Fill Properties)	§14.1.2.5
lock (Shape Protections)	§14.2.2.18
shadow (Shadow Effect)	§14.1.2.18
skew (Skew Transform)	§14.2.2.31
stroke (Line Stroke Settings)	§14.1.2.21
textbox (Text Box)	§14.1.2.22

Attributes	Description
allowincell (Allow in Table Cell)	<p>Specifies whether the shape is allowed to be placed in a table cell. Default is <code>false</code>.</p> <p>The possible values for this attribute are defined by the <code>ST_TrueFalse</code> simple type (§15.1.2.5).</p>
ext (VML Extension Handling Behavior) Namespace: urn:schemas-microsoft-com:vm1	<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 <code>ST_Ext</code> simple type (§14.1.3.2).</p>
fill (Shape Fill Toggle)	<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>[<i>Example</i>:</p> <pre><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 <code>ST_TrueFalse</code> simple type (§15.1.2.5).</p>

Attributes	Description
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 (§15.1.2.3).</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 (§14.1.2.21) defines other strokes. The on attribute of the stroke element overrides this attribute. Default is true.</p> <p>[Example:</p> <pre data-bbox="451 915 1045 1016"><v:shape ... fillcolor="red" stroke="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 (§15.1.2.5).</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 (§14.1.2.21) 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 1629 948 1696"><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 (§15.1.2.3).</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: <code><v:shape ... style='position:absolute;width:100pt;height:50pt' ...</code> <code>></code> <code></v:shape></code> <i>end example]</i></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 </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
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 								

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 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</p>

Attributes	Description	
		<p>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> <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> <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">• <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 (§14.1.2.22):																		
	<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 <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.</td></tr><tr><td>layout-flow</td><td><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.</td></tr><tr><td>mso-direction-alt</td><td><p>Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code>.</p></td></tr><tr><td>mso-fit-shape-to-text</td><td><p>Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code>.</p></td></tr><tr><td>mso-fit-text-to-shape</td><td><p>Specifies whether the text stretches to fit the textbox. Default is <code>false</code>.</p></td></tr><tr><td>mso-layout-flow-alt</td><td><p>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>.</p></td></tr><tr><td>mso-next-textbox</td><td><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>mso-rotate</td><td><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</td></tr></table>	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 horizontally.	mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code>.</p>	mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code>.</p>	mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is <code>false</code>.</p>	mso-layout-flow-alt	<p>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>.</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
	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 horizontally.																	
	mso-direction-alt	<p>Specifies an alternate direction for text in textboxes. Overrides the <code>direction</code> property. The only allowed value is <code>context</code>.</p>																	
	mso-fit-shape-to-text	<p>Specifies whether the shape stretches to fit the text in the textbox. Default is <code>false</code>.</p>																	
	mso-fit-text-to-shape	<p>Specifies whether the text stretches to fit the textbox. Default is <code>false</code>.</p>																	
mso-layout-flow-alt	<p>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>.</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																		

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	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 (§14.1.2.23):														
<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</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
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													

Attributes	Description							
		<p>values are:</p> <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 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>	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.						
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-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">leftrightcenterjustify							

Attributes	Description	
		<ul style="list-style-type: none"> • 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 (§14.1.2.12), polyline (§14.1.2.15) and curve (§14.1.2.3) 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 (§14.1.2.20) 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*]

14.2.2.29 shapelayout (Shape Layout Properties)

This element contains child elements that store information used in the editing and layout of shapes.

Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2)

Child Elements	Subclause
idmap (Shape ID Map)	§14.2.2.14
regrouptable (Shape Grouping History)	§14.2.2.23
rules (Rule Set)	§14.2.2.27

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:vmf	[<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 (§14.1.3.2).

[Note: The W3C XML Schema definition of this element's content model ([CT_ShapeLayout](#)) is located in §A.6.2. *end note*]

14.2.2.30 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]

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapetype (§14.1.2.20)

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>[<i>Example:</i></p> <pre><o:signatureline ... o:addlxml="..."> </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>
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>[<i>Example:</i></p> <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 (§15.1.2.5).</p>

Attributes	Description
<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 (§14.1.3.2).</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 (§15.1.2.5).</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 ... provid="{00000000-0000-0000-0000-000000000000}"> </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>

Attributes	Description
showsigndate (Show Signed Date Flag)	<p>Specifies whether the signed signature line image generated should include the date of signing. Default is true.</p> <p>[Example:</p> <pre><o:signatureline ... showsigndate="false"> </o:signatureline></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
signinginstructions (Instructions for Signing)	<p>Specifies text shown to the user at signing time. Default is no value.</p> <p>[Example:</p> <pre><o:signatureline ... o:signinginstructions="Sign here"> </o:signatureline></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>
signinginstructionsset (Use Signing Instructions Flag)	<p>Specifies whether there is data set in the signinginstructions attribute. Default is false.</p> <p>[Example:</p> <pre><o:signatureline ... signinginstructionsset="true"> </o:signatureline></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_TrueFalse simple type (§15.1.2.5).</p>
sigprovurl (Signature Provider Download URL)	<p>Specifies the URL for downloading the signature provider. Default is no value.</p> <p>[Example:</p> <pre><o:signatureline ... o:sigprovurl="http://www.example.com"> </o:signatureline></pre> <p>end example]</p> <p>The possible values for this attribute are defined by the W3C XML Schema string datatype.</p>

Attributes	Description
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><i>[Example:</i></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><i>[Example:</i></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><i>[Example:</i></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]

14.2.2.31 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.

Parent Elements
arc (§14.1.2.1); background (Part 1, §17.2.1); curve (§14.1.2.3); group (§14.1.2.7); hdrShapeDefaults (§9.7.2.1); image (§14.1.2.10); line (§14.1.2.12); object (Part 1, §17.3.3.19); oval (§14.1.2.13); pict (§9.2.2.2); pict (§9.5.1); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapeDefaults (§9.7.2.2); shapedefaults (§14.2.2.28); shapetype (§14.1.2.20)

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	[<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 (§14.1.3.2).
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 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. 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 false. The possible values for this attribute are defined by the ST_TrueFalse simple type

Attributes	Description
	(§15.1.2.5).
origin (Skew Origin)	<p>Specifies the origin of the skew. Default is "0,0".</p> <p>Values are typically a percentage of the shape's size and range from -0.5 to +0.5. Larger 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>

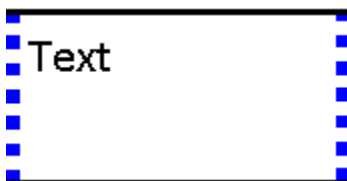
[Note: The W3C XML Schema definition of this element's content model ([CT_Skew](#)) is located in §A.6.2. *end note*]

14.2.2.32 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.

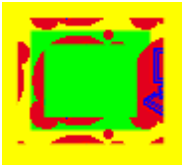


```
<v:stroke weight="2.25pt">
  <o:left v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:top v:ext="view" color="black" weight="2.25pt" on="t"/>
  <o:right v:ext="view" dashstyle="1 1" color="blue" weight="5pt" on="t"/>
  <o:bottom v:ext="view" color="black" on="t"/>
  <o:column v:ext="view" color="#f60" on="t"/>
</v:stroke>
```


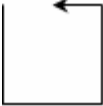



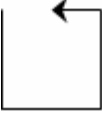
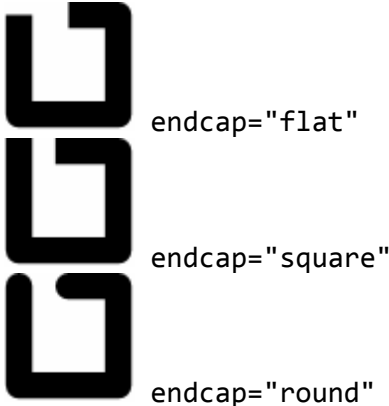
end example]

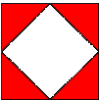
Parent Elements
background (Part 1, §17.2.1); hdrShapeDefaults (§9.7.2.1); object (Part 1, §17.3.3.19); pict (§9.2.2.2); pict (§9.5.1); shapeDefaults (§9.7.2.2); stroke (§14.1.2.21)



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: The shape stroke is blue:</i></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 (§15.1.2.3).</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><i>[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.</i></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
	  <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 (§15.1.2.3).</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>  <pre><v:stroke dashstyle="longdashdotdot" weight="2pt"> </v:stroke></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>
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 (§14.1.3.8).</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>



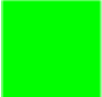
Attributes	Description
<p>endarrowwidth (Line End Arrowhead Width)</p>	<p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</p> <p>Specifies the width of the arrowhead at the end of a line. Default is <code>medium</code>. Allowed values are:</p> <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 (§14.1.3.9).</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> <pre><v:stroke ... endcap="round" weight="10pt" ... /></pre>  <p><i>end example]</i></p>


Attributes	Description
	<p>The possible values for this attribute are defined by the ST_StrokeEndCap simple type (§14.1.3.10).</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 (§14.1.3.2).</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 (§14.1.3.4).</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 (§15.1.2.5).</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 (§15.1.2.5).</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><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><v:stroke filltype="frame" weight="10pt" src="border.gif" imagealignshape="true" imageaspect="atleast"> </v:stroke></pre> <div></div> <p>end example]</p> <p>The possible values for this attribute are defined by the ST_ImageAspect simple type (§14.1.3.5).</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><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 (§15.1.2.5).</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: 10px;">  <div style="margin-left: 10px;"> <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 (§14.1.3.11).</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 (§14.1.3.12).</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 jointstyle="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 (§15.1.2.5).</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 (§14.1.3.8).</p>
startarrowlength (Line Start Arrowhead Length)	<p>Specifies the length of the arrowhead at the start 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 ... startarrowlength="long" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowLength simple type (§14.1.3.7).</p>
startarrowwidth (Line Start Arrowhead Width)	<p>Specifies the width of the arrowhead at the start 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 ... startarrowwidth="wide" ... /></pre>  <p><i>end example]</i></p> <p>The possible values for this attribute are defined by the ST_StrokeArrowWidth simple type (§14.1.3.9).</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]

14.2.3 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.

14.2.3.1 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.

Referenced By
equationxml@contentType (§14.2.2.10)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_AlternateMathContentType](#)) is located in §A.6.2. *end note*]

14.2.3.2 ST_Angle (Callout Angles)

This simple type specifies values for the angle attribute of the callout element (§14.2.2.2).

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.

Referenced By
callout@angle (§14.2.2.2)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_Angle](#)) is located in §A.6.2. *end note*]

14.2.3.3 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.

Enumeration Value	Description
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.

Referenced By
arc@bwmode (§14.1.2.1); arc@bwnormal (§14.1.2.1); arc@bwpure (§14.1.2.1); background@bwmode (§14.1.2.2); background@bwnormal (§14.1.2.2); background@bwpure (§14.1.2.2); curve@bwmode (§14.1.2.3); curve@bwnormal (§14.1.2.3); curve@bwpure (§14.1.2.3); image@bwmode (§14.1.2.10); image@bwnormal (§14.1.2.10); image@bwpure (§14.1.2.10); line@bwmode (§14.1.2.12); line@bwnormal (§14.1.2.12); line@bwpure (§14.1.2.12); oval@bwmode (§14.1.2.13); oval@bwnormal (§14.1.2.13); oval@bwpure (§14.1.2.13); polyline@bwmode (§14.1.2.15); polyline@bwnormal (§14.1.2.15); polyline@bwpure (§14.1.2.15); rect@bwmode (§14.1.2.16); rect@bwnormal (§14.1.2.16); rect@bwpure (§14.1.2.16); roundrect@bwmode (§14.1.2.17); roundrect@bwnormal (§14.1.2.17); roundrect@bwpure (§14.1.2.17); shape@bwmode (§14.1.2.19); shape@bwnormal (§14.1.2.19); shape@bwpure (§14.1.2.19); shapetype@bwmode (§14.1.2.20); shapetype@bwnormal (§14.1.2.20); shapetype@bwpure (§14.1.2.20)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_BWMode](#)) is located in §A.6.2. *end note*]

14.2.3.4 [ST_CalloutDrop \(Callout Drop Location\)](#)

This simple type specifies location values for the drop attribute of the callout element (§14.2.2.2).

This simple type's contents are a restriction of the W3C XML Schema string datatype.

Referenced By
callout@drop (§14.2.2.2)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_CalloutDrop](#)) is located in §A.6.2. *end note*]

14.2.3.5 [ST_CalloutPlacement \(Callout Placement\)](#)

This type defines location values used by the drop attribute of the callout element (§14.2.2.2).

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.

Enumeration Value	Description
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*]

14.2.3.6 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.

Referenced By
extrusion@colormode (§14.2.2.11)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ColorMode](#)) is located in §A.6.2. *end note*]

14.2.3.7 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.

Referenced By
arc@connectortype (§14.1.2.1); curve@connectortype (§14.1.2.3); image@connectortype (§14.1.2.10); line@connectortype (§14.1.2.12); oval@connectortype (§14.1.2.13); polyline@connectortype (§14.1.2.15); rect@connectortype (§14.1.2.16); roundrect@connectortype (§14.1.2.17); shape@connectortype (§14.1.2.19); shapetype@connectortype (§14.1.2.20)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ConnectorType](#)) is located in §A.6.2. *end note*]

14.2.3.8 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.

Referenced By
path@connecttype (§14.1.2.14)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ConnectType](#)) is located in §A.6.2. *end note*]

14.2.3.9 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.

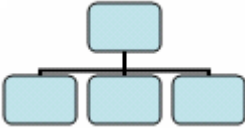
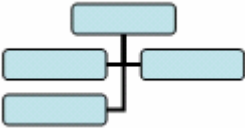
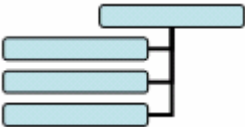
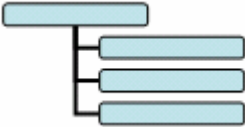
Referenced By
ink@contentType (§14.2.2.15)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ContentType](#)) is located in §A.6.2. *end note*]

14.2.3.10 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. 
3 (Hanging Left Side)	Hanging, left side layout. 

Referenced By
arc@dgmlayout (§14.1.2.1); arc@dgmlayoutmru (§14.1.2.1); curve@dgmlayout (§14.1.2.3); curve@dgmlayoutmru (§14.1.2.3); group@dgmlayout (§14.1.2.7); group@dgmlayoutmru (§14.1.2.7); image@dgmlayout (§14.1.2.10); image@dgmlayoutmru (§14.1.2.10); line@dgmlayout (§14.1.2.12); line@dgmlayoutmru (§14.1.2.12); oval@dgmlayout (§14.1.2.13); oval@dgmlayoutmru (§14.1.2.13); polyline@dgmlayout (§14.1.2.15); polyline@dgmlayoutmru (§14.1.2.15); rect@dgmlayout (§14.1.2.16); rect@dgmlayoutmru (§14.1.2.16); roundrect@dgmlayout (§14.1.2.17); roundrect@dgmlayoutmru (§14.1.2.17); shape@dgmlayout (§14.1.2.19); shape@dgmlayoutmru (§14.1.2.19); shapetype@dgmlayout (§14.1.2.20); shapetype@dgmlayoutmru (§14.1.2.20)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_DiagramLayout](#)) is located in §A.6.2. end note]

14.2.3.11 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.

Referenced By
extrusion@plane (§14.2.2.11)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionPlane](#)) is located in §A.6.2. *end note*]

14.2.3.12 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.

Referenced By
extrusion@render (§14.2.2.11)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionRender](#)) is located in §A.6.2. *end note*]

14.2.3.13 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

Enumeration Value	Description
	is the same as the vanishing point for unrotated objects.




Referenced By
extrusion@type (§14.2.2.11)

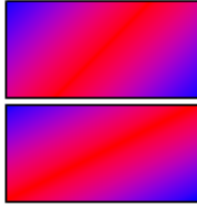



[Note: The W3C XML Schema definition of this simple type's content model ([ST_ExtrusionType](#)) is located in §A.6.2. *end note*]

14.2.3.14 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 (§14.1.2.5) 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. [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:]

Enumeration Value	Description
	 <i>end example]</i>
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> 

Referenced By
fill@type (§14.2.2.13)

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_FillType](#)) is located in §A.6.2.
end note]

14.2.3.15 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.

Referenced By
r@how (§14.2.2.22)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_How](#)) is located in §A.6.2. *end note*]

14.2.3.16 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.

Referenced By
arc@hralign (§14.1.2.1); curve@hralign (§14.1.2.3); group@hralign (§14.1.2.7); image@hralign (§14.1.2.10); line@hralign (§14.1.2.12); oval@hralign (§14.1.2.13); polyline@hralign (§14.1.2.15); rect@hralign (§14.1.2.16); roundrect@hralign (§14.1.2.17); shape@hralign (§14.1.2.19); shapetype@hralign (§14.1.2.20)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_HrAlign](#)) is located in §A.6.2. *end note*]

14.2.3.17 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.

Referenced By
arc@insetmode (§14.1.2.1); curve@insetmode (§14.1.2.3); group@insetmode (§14.1.2.7); image@insetmode (§14.1.2.10); line@insetmode (§14.1.2.12); oval@insetmode (§14.1.2.13); polyline@insetmode (§14.1.2.15); rect@insetmode (§14.1.2.16); roundrect@insetmode (§14.1.2.17); shape@insetmode (§14.1.2.19); shapetype@insetmode (§14.1.2.20); textbox@insetmode (§14.1.2.22)

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_InsetMode](#)) is located in §A.6.2. end note*]

14.2.3.18 [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.

Referenced By
OLEObject@DrawAspect (§14.2.2.20)

[*Note: The W3C XML Schema definition of this simple type's content model ([ST_OLEDrawAspect](#)) is located in §A.6.2. end note*]

14.2.3.19 [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.

Referenced By
LinkType (§14.2.2.17)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_OLELinkType](#)) is located in §A.6.2. *end note*]

14.2.3.20 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.

Referenced By
OLEObject@Type (§14.2.2.20)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_OLEType](#)) is located in §A.6.2. *end note*]

14.2.3.21 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.

Referenced By
OLEObject@UpdateMode (§14.2.2.20)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_OLEUpdateMode](#)) is located in §A.6.2. *end note*]

14.2.3.22 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.

Referenced By
r@type (§14.2.2.22)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_RType](#)) is located in §A.6.2. *end note*]

14.2.3.23 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.

Referenced By
background@targetscreensize (§14.1.2.2)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ScreenSize](#)) is located in §A.6.2. *end note*]

14.3 VML - WordprocessingML Drawing

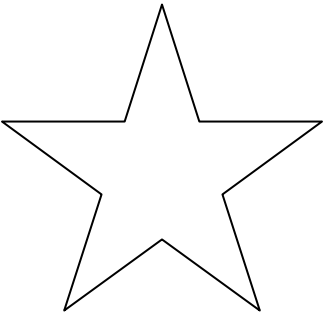
Within a WordprocessingML document, it is possible to include graphical VML objects. When these objects are present in a word processing document, it is necessary to include information about the object which is specific to their presence in a word processing document.

[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]

14.3.1 Table of Contents

This subclause is informative.

14.3.2 Elements 745

14.3.2.1 anchorlock (Anchor Location Is Locked) 745

14.3.2.2	borderbottom (Bottom Border)	746
14.3.2.3	borderleft (Left Border)	747
14.3.2.4	borderright (Right Border)	748
14.3.2.5	bordertop (Top Border)	750
14.3.2.6	wrap (Text Wrapping)	751
14.3.3	Simple Types	754
14.3.3.1	ST_BorderShadow (Border Shadow Type)	754
14.3.3.2	ST_BorderType (Border Type)	754
14.3.3.3	ST_HorizontalAnchor (Horizontal Anchor Type)	757
14.3.3.4	ST_VerticalAnchor (Vertical Anchor Type)	758
14.3.3.5	ST_WrapSide (Text Wrapping Side)	759
14.3.3.6	ST_WrapType (Text Wrapping Type)	759

End of informative text.

14.3.2 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*]

14.3.2.1 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*].

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

[*Note*: The W3C XML Schema definition of this element’s content model ([CT_AnchorLock](#)) is located in §A.6.3.
end note]

14.3.2.2 [borderbottom](#) (Bottom Border)

This element specifies the properties for the bottom border of a VML object.

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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 (§14.3.3.1).</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 (§14.3.3.2).</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</p>

Attributes	Description
	<p>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*]

14.3.2.3 [borderleft \(Left Border\)](#)

This element represents the properties for the left border of a VML object.

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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</p>

Attributes	Description
	<p>applied to the border. <i>end example</i></p> <p>The possible values for this attribute are defined by the ST_BorderShadow simple type (§14.3.3.1).</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 data-bbox="456 617 1016 648"><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 (§14.3.3.2).</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 1163 1273 1299"><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*]

14.3.2.4 `borderright` (Right Border)

This element specifies the properties for the right border of a VML object.

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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 (§14.3.3.1).</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 (§14.3.3.2).</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" .../></pre>

Attributes	Description
	<pre><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*]

14.3.2.5 **bordertop (Top Border)**

This element specifies the properties for the top border of a VML object.

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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 (§14.3.3.1).</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>

Attributes	Description
	<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 single, 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 (§14.3.3.2).</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> <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*]

14.3.2.6 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*].

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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 (§14.3.3.3).</p>
anchory (Vertical Positioning Base)	<p>Specifies the base object from which the vertical positioning of the object should be 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>

Attributes	Description
	<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 (§14.3.3.4).</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 (§14.3.3.5).</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> <p>The possible values for this attribute are defined by the ST_WrapType simple type (§14.3.3.6).</p>

[*Note*: The W3C XML Schema definition of this element's content model ([CT_Wrap](#)) is located in §A.6.3. *end note*]

14.3.3 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.

14.3.3.1 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.

Referenced By
borderbottom@shadow (§14.3.2.2); borderleft@shadow (§14.3.2.3); borderright@shadow (§14.3.2.4); bordertop@shadow (§14.3.2.5)



[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_BorderShadow](#)) is located in §A.6.3. *end note*]



14.3.3.2 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 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

Enumeration Value	Description
	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 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.

Enumeration Value	Description
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 the parent object.
wave (Wavy Border)	<p>Specifies a line border consisting of a wavy line around the parent object.</p> <p>[Example:</p>

Enumeration Value	Description
	 <i>end example]</i>

Referenced By
borderbottom@type (§14.3.2.2); borderleft@type (§14.3.2.3); borderright@type (§14.3.2.4); bordertop@type (§14.3.2.5)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_BorderType](#)) is located in §A.6.3.
end note]

14.3.3.3 [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 paragraph (including text indentations on that</p>

Enumeration Value	Description
	paragraph within the text margins).

Referenced By
wrap@anchorx (§14.3.2.6)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_HorizontalAnchor](#)) is located in §A.6.3. *end note*]

14.3.3.4 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</p>

Enumeration Value	Description
	paragraph.

Referenced By
wrap@anchory (§14.3.2.6)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_VerticalAnchor](#)) is located in §A.6.3. *end note*]

14.3.3.5 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.

Referenced By
wrap@side (§14.3.2.6)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_WrapSide](#)) is located in §A.6.3. *end note*]

14.3.3.6 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

Enumeration Value	Description
	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)	<p>Specifies that text shall not be allowed to wrap around the remaining space on each lines around the VML object.</p> <p>Any text content shall therefore be placed on the next line following the object which does not intersect with the object's extents.</p>

Referenced By
wrap@type (§14.3.2.6)

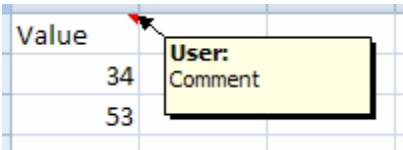
[*Note:* The W3C XML Schema definition of this simple type’s content model ([ST_WrapType](#)) is located in §A.6.3. *end note*]

14.4 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]

14.4.1 Table of Contents

This subclause is informative.

14.4.2	Elements	763
14.4.2.1	Accel (Primary Keyboard Accelerator)	763
14.4.2.2	Accel2 (Secondary Keyboard Accelerator)	764
14.4.2.3	Anchor (Anchor)	764
14.4.2.4	AutoFill (AutoFill)	765
14.4.2.5	AutoLine (AutoLine)	766
14.4.2.6	AutoPict (Automatically Size)	766
14.4.2.7	AutoScale (Font AutoScale)	767
14.4.2.8	Camera (Camera Tool)	767
14.4.2.9	Cancel (Cancel Button)	768
14.4.2.10	CF (Clipboard Format)	768

14.4.2.11	Checked (Checked)	769
14.4.2.12	ClientData (Attached Object Data)	769
14.4.2.13	ColHidden (Comment's Column is Hidden)	773
14.4.2.14	Colored (Dropdown Color Toggle)	773
14.4.2.15	Column (Comment Column Target)	774
14.4.2.16	DDE (Dynamic Data Exchange)	774
14.4.2.17	Default (Default Button)	774
14.4.2.18	DefaultSize (Default Size Toggle)	775
14.4.2.19	Disabled (Macro Disable Toggle)	775
14.4.2.20	Dismiss (Dismiss Button)	776
14.4.2.21	DropLines (Dropdown Maximum Lines)	776
14.4.2.22	DropStyle (Dropdown Style)	777
14.4.2.23	Dx (Scroll Bar Width)	777
14.4.2.24	FirstButton (First Radio Button)	778
14.4.2.25	FmlaGroup (Linked Formula - Group Box)	778
14.4.2.26	FmlaLink (Linked Formula)	779
14.4.2.27	FmlaMacro (Reference to Custom Function)	779
14.4.2.28	FmlaPict (Camera Source Range)	779
14.4.2.29	FmlaRange (List Items Source Range)	780
14.4.2.30	FmlaTxbx (Text Formula)	780
14.4.2.31	Help (Help Button)	780
14.4.2.32	Horiz (Scroll Bar Orientation)	781
14.4.2.33	Inc (Scroll Bar Increment)	781
14.4.2.34	JustLastX (Far East Alignment Toggle)	782
14.4.2.35	LCT (Callback Type)	782
14.4.2.36	ListItem (Non-linked List Item)	783
14.4.2.37	Locked (Lock Toggle)	783
14.4.2.38	LockText (Text Lock)	783
14.4.2.39	MapOCX (Embedded Control)	784
14.4.2.40	Max (Scroll Bar Maximum)	784
14.4.2.41	Min (Scroll Bar Minimum)	785
14.4.2.42	MoveWithCells (Move with Cells)	785
14.4.2.43	MultiLine (Multi-line)	786
14.4.2.44	MultiSel (Multiple Selections)	786
14.4.2.45	NoThreeD (Disable 3D)	786
14.4.2.46	NoThreeD2 (Disable 3D)	787
14.4.2.47	Page (Scroll Bar Page Increment)	787
14.4.2.48	PrintObject (Print Toggle)	788
14.4.2.49	RecalcAlways (Recalculation Toggle)	788
14.4.2.50	Row (Comment Row Target)	789
14.4.2.51	RowHidden (Comment's Row is Hidden)	789
14.4.2.52	ScriptExtended (HTML Script Attributes)	790
14.4.2.53	ScriptLanguage (HTML Script Language)	790
14.4.2.54	ScriptLocation (HTML Script Location)	791
14.4.2.55	ScriptText (HTML Script Text)	791
14.4.2.56	SecretEdit (Password Edit)	791
14.4.2.57	Sel (Selected Entry)	792
14.4.2.58	SelType (Selection Type)	792

14.4.2.59	SizeWithCells (Resize with Cells)	793
14.4.2.60	TextHAlign (Horizontal Text Alignment)	793
14.4.2.61	TextVAlign (Vertical Text Alignment)	794
14.4.2.62	UIObj (UI Object Toggle)	794
14.4.2.63	Val (Scroll bar position)	795
14.4.2.64	ValidIds (Valid ID)	795
14.4.2.65	Visible (Comment Visibility Toggle)	795
14.4.2.66	VScroll (Vertical Scroll)	796
14.4.2.67	VTEdit (Validation Type)	796
14.4.2.68	WidthMin (Minimum Width)	797
14.4.3	Simple Types	797
14.4.3.1	ST_CF (Clipboard Format Type)	797
14.4.3.2	ST_ObjectType (Object Type)	798

End of informative text.

14.4.2 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*]

14.4.2.1 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.2 **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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.3 **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> 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>

Value	Description
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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.4 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.5 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note*: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.6 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note*: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.7 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

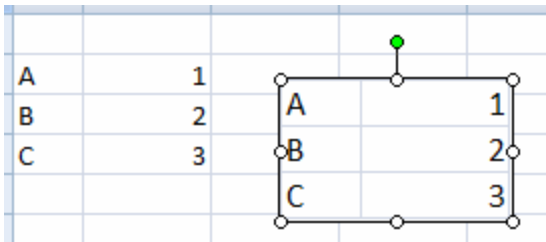
14.4.2.8 Camera (Camera Tool)

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 (§14.4.2.28), 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.9 [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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.10 [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 (§14.4.3.1).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_CF](#)) is located in §A.6.4. *end note*]

14.4.2.11 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
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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.12 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]

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

Child Elements	Subclause
Accel (Primary Keyboard Accelerator)	§14.4.2.1
Accel2 (Secondary Keyboard Accelerator)	§14.4.2.2
Anchor (Anchor)	§14.4.2.3
AutoFill (AutoFill)	§14.4.2.4
AutoLine (AutoLine)	§14.4.2.5

Child Elements	Subclause
AutoPict (Automatically Size)	§14.4.2.6
AutoScale (Font AutoScale)	§14.4.2.7
Camera (Camera Tool)	§14.4.2.8
Cancel (Cancel Button)	§14.4.2.9
CF (Clipboard Format)	§14.4.2.10
Checked (Checked)	§14.4.2.11
ColHidden (Comment's Column is Hidden)	§14.4.2.13
Colored (Dropdown Color Toggle)	§14.4.2.14
Column (Comment Column Target)	§14.4.2.15
DDE (Dynamic Data Exchange)	§14.4.2.16
Default (Default Button)	§14.4.2.17
DefaultSize (Default Size Toggle)	§14.4.2.18
Disabled (Macro Disable Toggle)	§14.4.2.19
Dismiss (Dismiss Button)	§14.4.2.20
DropLines (Dropdown Maximum Lines)	§14.4.2.21
DropStyle (Dropdown Style)	§14.4.2.22
Dx (Scroll Bar Width)	§14.4.2.23
FirstButton (First Radio Button)	§14.4.2.24
FmlaGroup (Linked Formula - Group Box)	§14.4.2.25
FmlaLink (Linked Formula)	§14.4.2.26
FmlaMacro (Reference to Custom Function)	§14.4.2.27
FmlaPict (Camera Source Range)	§14.4.2.28
FmlaRange (List Items Source Range)	§14.4.2.29
FmlaTxbx (Text Formula)	§14.4.2.30
Help (Help Button)	§14.4.2.31
Horiz (Scroll Bar Orientation)	§14.4.2.32
Inc (Scroll Bar Increment)	§14.4.2.33
JustLastX (Far East Alignment Toggle)	§14.4.2.34
LCT (Callback Type)	§14.4.2.35
ListItem (Non-linked List Item)	§14.4.2.36
Locked (Lock Toggle)	§14.4.2.37
LockText (Text Lock)	§14.4.2.38
MapOCX (Embedded Control)	§14.4.2.39
Max (Scroll Bar Maximum)	§14.4.2.40

Child Elements	Subclause
Min (Scroll Bar Minimum)	§14.4.2.41
MoveWithCells (Move with Cells)	§14.4.2.42
MultiLine (Multi-line)	§14.4.2.43
MultiSel (Multiple Selections)	§14.4.2.44
NoThreeD (Disable 3D)	§14.4.2.45
NoThreeD2 (Disable 3D)	§14.4.2.46
Page (Scroll Bar Page Increment)	§14.4.2.47
PrintObject (Print Toggle)	§14.4.2.48
RecalcAlways (Recalculation Toggle)	§14.4.2.49
Row (Comment Row Target)	§14.4.2.50
RowHidden (Comment's Row is Hidden)	§14.4.2.51
ScriptExtended (HTML Script Attributes)	§14.4.2.52
ScriptLanguage (HTML Script Language)	§14.4.2.53
ScriptLocation (HTML Script Location)	§14.4.2.54
ScriptText (HTML Script Text)	§14.4.2.55
SecretEdit (Password Edit)	§14.4.2.56
Sel (Selected Entry)	§14.4.2.57
SelType (Selection Type)	§14.4.2.58
SizeWithCells (Resize with Cells)	§14.4.2.59
TextHAlign (Horizontal Text Alignment)	§14.4.2.60
TextVAlign (Vertical Text Alignment)	§14.4.2.61
UIObj (UI Object Toggle)	§14.4.2.62
Val (Scroll bar position)	§14.4.2.63
ValidIds (Valid ID)	§14.4.2.64
Visible (Comment Visibility Toggle)	§14.4.2.65
VScroll (Vertical Scroll)	§14.4.2.66
VTEdit (Validation Type)	§14.4.2.67
WidthMin (Minimum Width)	§14.4.2.68

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 (§14.4.3.2).</p>

[*Note:* The W3C XML Schema definition of this element's content model ([CT_ClientData](#)) is located in §A.6.4. *end note*]

14.4.2.13 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.14 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.15 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.16 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.17 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.18 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.19 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.20 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.21 **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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.22 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.23 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.24 [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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.25 [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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.26 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.27 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.28 FmlaPict (Camera Source Range)

This element specifies the range of source data cells visible in the camera object (§14.4.2.8). 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.29 [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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.30 [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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.31 [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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.32 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.33 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.34 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model (ST_TrueFalseBlank) is located in §A.7.9.
end note]

14.4.2.35 **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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.36 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.37 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note*: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

14.4.2.38 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.39 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.40 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.41 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.42 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.43 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.44 **MultiSel (Multiple Selections)**

This element specifies a comma-delimited list of selected items. This element overrides the Sel element (§14.4.2.57). This element is used for list boxes that allow multiple selections. See also the SelType element (§14.4.2.58).

[*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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.45 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.46 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.47 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.48 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model (ST_TrueFalseBlank) is located in §A.7.9.
end note]

14.4.2.49 **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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.50 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.51 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.52 **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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.53 **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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.54 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.55 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.56 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.57 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.58 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.59 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.60 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.61 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.62 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[Note: The W3C XML Schema definition of this element’s content model (ST_TrueFalseBlank) is located in §A.7.9.
end note]

14.4.2.63 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.64 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.65 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.66 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 (§15.1.2.6).

Parent Elements
ClientData (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this element’s content model ([ST_TrueFalseBlank](#)) is located in §A.7.9.
end note]

14.4.2.67 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

Value	Description
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.

Parent Elements
ClientData (§14.4.2.12)

14.4.2.68 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.

Parent Elements
ClientData (§14.4.2.12)

14.4.3 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.

14.4.3.1 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. [<i>Example: SVG or JPEG. 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.
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.

Referenced By
CF (§14.4.2.10)

[*Note: The W3C XML Schema definition of this simple type's content model (ST_CF) is located in §A.6.4. end note*]

14.4.3.2 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.

Enumeration Value	Description
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.

Referenced By
ClientData@ObjectType (§14.4.2.12)

[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_ObjectType](#)) is located in §A.6.4. *end note*]

14.5 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]

14.5.1 Table of Contents

This subclause is informative.

14.5.2 Elements 800

14.5.2.1 iscomment (Ink Annotation Flag) 800

14.5.2.2 textdata (VML Diagram Text) 801

End of informative text.

14.5.2 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*]

14.5.2.1 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]

Parent Elements
shape (§14.1.2.19)

[*Note:* The W3C XML Schema definition of this element's content model ([CT_Empty](#)) is located in §A.6.5. *end note]*

14.5.2.2 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]*

Parent Elements
arc (§14.1.2.1); curve (§14.1.2.3); group (§14.1.2.7); image (§14.1.2.10); line (§14.1.2.12); oval (§14.1.2.13); polyline (§14.1.2.15); rect (§14.1.2.16); roundrect (§14.1.2.17); shape (§14.1.2.19); shapetype (§14.1.2.20)

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"/> </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_Rel](#)) is located in §A.6.5. *end note]*

15. Shared MLs Reference Material

15.1 Shared Simple Types

15.1.1 Table of Contents

This subclause is informative.

15.1.2 Simple Types	802
15.1.2.1 ST_AlgClass (Cryptographic Algorithm Classes)	802
15.1.2.2 ST_AlgType (Cryptographic Algorithm Types)	803
15.1.2.3 ST_ColorType (Color Type)	804
15.1.2.4 ST_CryptProv (Cryptographic Provider Types)	805
15.1.2.5 ST_TrueFalse (Boolean Value)	806
15.1.2.6 ST_TrueFalseBlank (Boolean Value with Blank [False] State)	809

End of informative text.

15.1.2 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.

15.1.2.1 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 algIdExt 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.

Referenced By
documentProtection@cryptAlgorithmClass (§9.7.2.3); modifyVerifier@cryptAlgorithmClass (§11.2.1.3); writeProtection@cryptAlgorithmClass (§9.7.2.5)

[*Note*: The W3C XML Schema definition of this simple type's content model ([ST_AlgClass](#)) is located in §A.7.9. *end note*]

15.1.2.2 ST_AlgType (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 algIdExt attribute, generated the hash value.
typeAny (Any Predefined Type)	Specifies that one of the predefined cryptographic algorithms, specified by the parent element's cryptAlgorithmSid attribute, generated the hash

Enumeration Value	Description
	value.

Referenced By
documentProtection@cryptAlgorithmType (§9.7.2.3); modifyVerifier@cryptAlgorithmType (§11.2.1.3); writeProtection@cryptAlgorithmType (§9.7.2.5)














[*Note:* The W3C XML Schema definition of this simple type's content model ([ST_AlgType](#)) is located in §A.7.9. *end note*]

15.1.2.3 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.

Referenced By
arc@chromakey (§14.1.2.1); arc@fillcolor (§14.1.2.1); arc@strokecolor (§14.1.2.1); background@fillcolor (§14.1.2.2); bottom@color (§14.2.2.1); bottom@color2 (§14.2.2.1); colormenu@extrusioncolor (§14.2.2.4); colormenu@fillcolor (§14.2.2.4); colormenu@shadowcolor (§14.2.2.4); colormenu@strokecolor (§14.2.2.4); column@color (§14.2.2.6); column@color2 (§14.2.2.6); curve@chromakey (§14.1.2.3); curve@fillcolor (§14.1.2.3); curve@strokecolor (§14.1.2.3); extrusion@color (§14.2.2.11); fill@color (§14.1.2.5); fill@color2 (§14.1.2.5); group@fillcolor (§14.1.2.7); image@chromakey (§14.1.2.10); image@fillcolor (§14.1.2.10); image@strokecolor (§14.1.2.10); imagedata@chromakey (§14.1.2.11); imagedata@embosscolor (§14.1.2.11); imagedata@recolortarget (§14.1.2.11); left@color (§14.2.2.16); left@color2 (§14.2.2.16); line@chromakey (§14.1.2.12); line@fillcolor (§14.1.2.12); line@strokecolor (§14.1.2.12); oval@chromakey (§14.1.2.13); oval@fillcolor (§14.1.2.13); oval@strokecolor (§14.1.2.13); polyline@chromakey (§14.1.2.15); polyline@fillcolor (§14.1.2.15); polyline@strokecolor (§14.1.2.15); rect@chromakey (§14.1.2.16); rect@fillcolor (§14.1.2.16); rect@strokecolor (§14.1.2.16); right@color (§14.2.2.26); right@color2 (§14.2.2.26); roundrect@chromakey (§14.1.2.17); roundrect@fillcolor (§14.1.2.17); roundrect@strokecolor (§14.1.2.17); shadow@color (§14.1.2.18); shadow@color2 (§14.1.2.18); shape@chromakey (§14.1.2.19); shape@fillcolor (§14.1.2.19); shape@strokecolor (§14.1.2.19); shapedefaults@fillcolor (§14.2.2.28); shapedefaults@strokecolor (§14.2.2.28); shapetype@chromakey (§14.1.2.20); shapetype@fillcolor (§14.1.2.20); shapetype@strokecolor (§14.1.2.20); stroke@color (§14.1.2.21); stroke@color2 (§14.1.2.21); top@color (§14.2.2.32); top@color2 (§14.2.2.32)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_ColorType](#)) is located in §A.7.9. end note]

15.1.2.4 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.

Referenced By
documentProtection@cryptProviderType (§9.7.2.3); modifyVerifier@cryptProviderType (§11.2.1.3); writeProtection@cryptProviderType (§9.7.2.5)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_CryptProv](#)) is located in §A.7.9. *end note*]

15.1.2.5 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.
true (True)	Logical true.

Referenced By
arc@allowincell (§14.1.2.1); arc@allowoverlap (§14.1.2.1); arc@bullet (§14.1.2.1); arc@button (§14.1.2.1); arc@clip (§14.1.2.1); arc@cliptowrap (§14.1.2.1); arc@doubleclicknotify (§14.1.2.1); arc@filled (§14.1.2.1); arc@forcedash (§14.1.2.1); arc@hr (§14.1.2.1); arc@hrnoshade (§14.1.2.1); arc@hrstd (§14.1.2.1); arc@insetpen (§14.1.2.1); arc@oleicon (§14.1.2.1); arc@oned (§14.1.2.1); arc@preferrelative (§14.1.2.1); arc@print (§14.1.2.1); arc@stroked (§14.1.2.1); arc@userdrawn (§14.1.2.1); arc@userhidden (§14.1.2.1);

Referenced By
background@filled (§14.1.2.2); bottom@forcedash (§14.2.2.1); bottom@imagealignshape (§14.2.2.1); bottom@insetpen (§14.2.2.1); bottom@on (§14.2.2.1); callout@accentbar (§14.2.2.2); callout@dropauto (§14.2.2.2); callout@lengthspecified (§14.2.2.2); callout@minusx (§14.2.2.2); callout@minusy (§14.2.2.2); callout@on (§14.2.2.2); callout@textborder (§14.2.2.2); column@forcedash (§14.2.2.6); column@imagealignshape (§14.2.2.6); column@insetpen (§14.2.2.6); column@on (§14.2.2.6); curve@allowincell (§14.1.2.3); curve@allowoverlap (§14.1.2.3); curve@bullet (§14.1.2.3); curve@button (§14.1.2.3); curve@clip (§14.1.2.3); curve@cliptowrap (§14.1.2.3); curve@doubleclicknotify (§14.1.2.3); curve@filled (§14.1.2.3); curve@forcedash (§14.1.2.3); curve@hr (§14.1.2.3); curve@hrnoshade (§14.1.2.3); curve@hrstd (§14.1.2.3); curve@insetpen (§14.1.2.3); curve@oleicon (§14.1.2.3); curve@oned (§14.1.2.3); curve@preferrelative (§14.1.2.3); curve@print (§14.1.2.3); curve@stroked (§14.1.2.3); curve@userdrawn (§14.1.2.3); curve@userhidden (§14.1.2.3); diagram@autoformat (§14.2.2.8); diagram@autolayout (§14.2.2.8); diagram@reverse (§14.2.2.8); extrusion@autorotationcenter (§14.2.2.11); extrusion@lightface (§14.2.2.11); extrusion@lightharsh (§14.2.2.11); extrusion@lightharsh2 (§14.2.2.11); extrusion@lockrotationcenter (§14.2.2.11); extrusion@metal (§14.2.2.11); extrusion@on (§14.2.2.11); fill@alignshape (§14.1.2.5); fill@detectmouseclick (§14.1.2.5); fill@on (§14.1.2.5); fill@recolor (§14.1.2.5); fill@rotate (§14.1.2.5); group@allowincell (§14.1.2.7); group@allowoverlap (§14.1.2.7); group@bullet (§14.1.2.7); group@button (§14.1.2.7); group@doubleclicknotify (§14.1.2.7); group@filled (§14.1.2.7); group@hr (§14.1.2.7); group@hrnoshade (§14.1.2.7); group@hrstd (§14.1.2.7); group@oned (§14.1.2.7); group@print (§14.1.2.7); group@userdrawn (§14.1.2.7); group@userhidden (§14.1.2.7); h@invx (§14.1.2.8); h@invy (§14.1.2.8); image@allowincell (§14.1.2.10); image@allowoverlap (§14.1.2.10); image@bilevel (§14.1.2.10); image@bullet (§14.1.2.10); image@button (§14.1.2.10); image@clip (§14.1.2.10); image@cliptowrap (§14.1.2.10); image@doubleclicknotify (§14.1.2.10); image@filled (§14.1.2.10); image@forcedash (§14.1.2.10); image@grayscale (§14.1.2.10); image@hr (§14.1.2.10); image@hrnoshade (§14.1.2.10); image@hrstd (§14.1.2.10); image@insetpen (§14.1.2.10); image@oleicon (§14.1.2.10); image@oned (§14.1.2.10); image@preferrelative (§14.1.2.10); image@print (§14.1.2.10); image@stroked (§14.1.2.10); image@userdrawn (§14.1.2.10); image@userhidden (§14.1.2.10); imagedata@bilevel (§14.1.2.11); imagedata@detectmouseclick (§14.1.2.11); imagedata@grayscale (§14.1.2.11); ink@annotation (§14.2.2.15); left@forcedash (§14.2.2.16); left@imagealignshape (§14.2.2.16); left@insetpen (§14.2.2.16); left@on (§14.2.2.16); line@allowincell (§14.1.2.12); line@allowoverlap (§14.1.2.12); line@bullet (§14.1.2.12); line@button (§14.1.2.12); line@clip (§14.1.2.12); line@cliptowrap (§14.1.2.12); line@doubleclicknotify (§14.1.2.12); line@filled (§14.1.2.12); line@forcedash (§14.1.2.12); line@hr (§14.1.2.12); line@hrnoshade (§14.1.2.12); line@hrstd (§14.1.2.12); line@insetpen (§14.1.2.12); line@oleicon (§14.1.2.12); line@oned (§14.1.2.12); line@preferrelative (§14.1.2.12); line@print (§14.1.2.12); line@stroked (§14.1.2.12); line@userdrawn (§14.1.2.12); line@userhidden (§14.1.2.12); lock@adjusthandles (§14.2.2.18); lock@aspectratio (§14.2.2.18); lock@cropping (§14.2.2.18); lock@grouping (§14.2.2.18); lock@position (§14.2.2.18); lock@rotation (§14.2.2.18); lock@selection (§14.2.2.18); lock@shapetype (§14.2.2.18); lock@text (§14.2.2.18); lock@ungrouping (§14.2.2.18); lock@verticies (§14.2.2.18); oval@allowincell (§14.1.2.13); oval@allowoverlap (§14.1.2.13); oval@bullet (§14.1.2.13); oval@button (§14.1.2.13); oval@clip (§14.1.2.13); oval@cliptowrap (§14.1.2.13); oval@doubleclicknotify (§14.1.2.13); oval@filled (§14.1.2.13); oval@forcedash (§14.1.2.13); oval@hr (§14.1.2.13); oval@hrnoshade (§14.1.2.13); oval@hrstd (§14.1.2.13); oval@insetpen (§14.1.2.13); oval@oleicon (§14.1.2.13); oval@oned (§14.1.2.13); oval@preferrelative (§14.1.2.13); oval@print (§14.1.2.13); oval@stroked (§14.1.2.13); oval@userdrawn (§14.1.2.13); oval@userhidden (§14.1.2.13); path@arrowok (§14.1.2.14); path@extrusionok (§14.1.2.14); path@fillok (§14.1.2.14); path@gradientshapeok (§14.1.2.14); path@insetpenok (§14.1.2.14); path@shadowok (§14.1.2.14); path@strokeok (§14.1.2.14); path@textpathok (§14.1.2.14); polyline@allowincell (§14.1.2.15); polyline@allowoverlap (§14.1.2.15); polyline@bullet (§14.1.2.15); polyline@button (§14.1.2.15);

Referenced By
<p> polyline@clip (§14.1.2.15); polyline@cliptowrap (§14.1.2.15); polyline@doubleclicknotify (§14.1.2.15); polyline@filled (§14.1.2.15); polyline@forcedash (§14.1.2.15); polyline@hr (§14.1.2.15); polyline@hrnoshade (§14.1.2.15); polyline@hrstd (§14.1.2.15); polyline@insetpen (§14.1.2.15); polyline@oleicon (§14.1.2.15); polyline@oned (§14.1.2.15); polyline@preferrelative (§14.1.2.15); polyline@print (§14.1.2.15); polyline@stroked (§14.1.2.15); polyline@userdrawn (§14.1.2.15); polyline@userhidden (§14.1.2.15); rect@allowincell (§14.1.2.16); rect@allowoverlap (§14.1.2.16); rect@bullet (§14.1.2.16); rect@button (§14.1.2.16); rect@clip (§14.1.2.16); rect@cliptowrap (§14.1.2.16); rect@doubleclicknotify (§14.1.2.16); rect@filled (§14.1.2.16); rect@forcedash (§14.1.2.16); rect@hr (§14.1.2.16); rect@hrnoshade (§14.1.2.16); rect@hrstd (§14.1.2.16); rect@insetpen (§14.1.2.16); rect@oleicon (§14.1.2.16); rect@oned (§14.1.2.16); rect@preferrelative (§14.1.2.16); rect@print (§14.1.2.16); rect@stroked (§14.1.2.16); rect@userdrawn (§14.1.2.16); rect@userhidden (§14.1.2.16); right@forcedash (§14.2.2.26); right@imagealignshape (§14.2.2.26); right@insetpen (§14.2.2.26); right@on (§14.2.2.26); roundrect@allowincell (§14.1.2.17); roundrect@allowoverlap (§14.1.2.17); roundrect@bullet (§14.1.2.17); roundrect@button (§14.1.2.17); roundrect@clip (§14.1.2.17); roundrect@cliptowrap (§14.1.2.17); roundrect@doubleclicknotify (§14.1.2.17); roundrect@filled (§14.1.2.17); roundrect@forcedash (§14.1.2.17); roundrect@hr (§14.1.2.17); roundrect@hrnoshade (§14.1.2.17); roundrect@hrstd (§14.1.2.17); roundrect@insetpen (§14.1.2.17); roundrect@oleicon (§14.1.2.17); roundrect@oned (§14.1.2.17); roundrect@preferrelative (§14.1.2.17); roundrect@print (§14.1.2.17); roundrect@stroked (§14.1.2.17); roundrect@userdrawn (§14.1.2.17); roundrect@userhidden (§14.1.2.17); shadow@obscured (§14.1.2.18); shadow@on (§14.1.2.18); shape@allowincell (§14.1.2.19); shape@allowoverlap (§14.1.2.19); shape@bullet (§14.1.2.19); shape@button (§14.1.2.19); shape@clip (§14.1.2.19); shape@cliptowrap (§14.1.2.19); shape@doubleclicknotify (§14.1.2.19); shape@filled (§14.1.2.19); shape@forcedash (§14.1.2.19); shape@hr (§14.1.2.19); shape@hrnoshade (§14.1.2.19); shape@hrstd (§14.1.2.19); shape@insetpen (§14.1.2.19); shape@oleicon (§14.1.2.19); shape@oned (§14.1.2.19); shape@preferrelative (§14.1.2.19); shape@print (§14.1.2.19); shape@stroked (§14.1.2.19); shape@userdrawn (§14.1.2.19); shape@userhidden (§14.1.2.19); shapedefaults@allowincell (§14.2.2.28); shapedefaults@fill (§14.2.2.28); shapedefaults@stroke (§14.2.2.28); shapetype@allowincell (§14.1.2.20); shapetype@allowoverlap (§14.1.2.20); shapetype@bullet (§14.1.2.20); shapetype@button (§14.1.2.20); shapetype@clip (§14.1.2.20); shapetype@cliptowrap (§14.1.2.20); shapetype@doubleclicknotify (§14.1.2.20); shapetype@filled (§14.1.2.20); shapetype@forcedash (§14.1.2.20); shapetype@hr (§14.1.2.20); shapetype@hrnoshade (§14.1.2.20); shapetype@hrstd (§14.1.2.20); shapetype@insetpen (§14.1.2.20); shapetype@oleicon (§14.1.2.20); shapetype@oned (§14.1.2.20); shapetype@preferrelative (§14.1.2.20); shapetype@print (§14.1.2.20); shapetype@stroked (§14.1.2.20); shapetype@userdrawn (§14.1.2.20); shapetype@userhidden (§14.1.2.20); signatureline@allowcomments (§14.2.2.30); signatureline@issignatureline (§14.2.2.30); signatureline@showsigndate (§14.2.2.30); signatureline@signinginstructionsset (§14.2.2.30); skew@on (§14.2.2.31); stroke@forcedash (§14.1.2.21); stroke@imagealignshape (§14.1.2.21); stroke@insetpen (§14.1.2.21); stroke@on (§14.1.2.21); textbox@singleclick (§14.1.2.22); textpath@fitpath (§14.1.2.23); textpath@fitshape (§14.1.2.23); textpath@on (§14.1.2.23); textpath@trim (§14.1.2.23); textpath@xscale (§14.1.2.23); top@forcedash (§14.2.2.32); top@imagealignshape (§14.2.2.32); top@insetpen (§14.2.2.32); top@on (§14.2.2.32) </p>

[Note: The W3C XML Schema definition of this simple type's content model ([ST TrueFalse](#)) is located in §A.7.9. end note]

15.1.2.6 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.

Referenced By
arc@ole (§14.1.2.1); AutoFill (§14.4.2.4); AutoLine (§14.4.2.5); AutoPict (§14.4.2.6); AutoScale (§14.4.2.7); Camera (§14.4.2.8); Cancel (§14.4.2.9); ColHidden (§14.4.2.13); Colored (§14.4.2.14); curve@ole (§14.1.2.3); DDE (§14.4.2.16); Default (§14.4.2.17); DefaultSize (§14.4.2.18); Disabled (§14.4.2.19); Dismiss (§14.4.2.20); FirstButton (§14.4.2.24); h@switch (§14.1.2.8); Help (§14.4.2.31); Horiz (§14.4.2.32); image@ole (§14.1.2.10); JustLastX (§14.4.2.34); line@ole (§14.1.2.12); Locked (§14.4.2.37); LockedField (§14.2.2.19); LockText (§14.4.2.38); MapOCX (§14.4.2.39); MoveWithCells (§14.4.2.42); MultiLine (§14.4.2.43); NoThreeD (§14.4.2.45); NoThreeD2 (§14.4.2.46); oval@ole (§14.1.2.13); polyline@ole (§14.1.2.15); PrintObject (§14.4.2.48); proxy@end (§14.2.2.21); proxy@start (§14.2.2.21); RecalcAlways (§14.4.2.49); rect@ole (§14.1.2.16); roundrect@ole (§14.1.2.17); RowHidden (§14.4.2.51); SecretEdit (§14.4.2.56); shape@ole (§14.1.2.19); shapetype@ole (§14.1.2.20); SizeWithCells (§14.4.2.59); UIObj (§14.4.2.62); ValidIds (§14.4.2.64); Visible (§14.4.2.65); VScroll (§14.4.2.66)

[Note: The W3C XML Schema definition of this simple type's content model ([ST_TrueFalseBlank](#)) is located in §A.7.9. *end note*]

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

```

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="required"/>
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_DecimalNumberOrPercent">
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:restriction base="xsd:integer">
114         <xsd:minInclusive value="0"/>
115         <xsd:maxInclusive value="600"/>
116     </xsd:restriction>
117 </xsd:simpleType>
118 <xsd:complexType name="CT_TextScale">
119     <xsd:attribute name="val" type="ST_TextScale"/>
120 </xsd:complexType>
121 <xsd:simpleType name="ST_HighlightColor">
122     <xsd:restriction base="xsd:string">
123         <xsd:enumeration value="black"/>
124         <xsd:enumeration value="blue"/>
125         <xsd:enumeration value="cyan"/>
126         <xsd:enumeration value="green"/>
127         <xsd:enumeration value="magenta"/>
128         <xsd:enumeration value="red"/>
129         <xsd:enumeration value="yellow"/>
130         <xsd:enumeration value="white"/>
131         <xsd:enumeration value="darkBlue"/>
132         <xsd:enumeration value="darkCyan"/>
133         <xsd:enumeration value="darkGreen"/>
134         <xsd:enumeration value="darkMagenta"/>
135         <xsd:enumeration value="darkRed"/>
136         <xsd:enumeration value="darkYellow"/>
137         <xsd:enumeration value="darkGray"/>
138         <xsd:enumeration value="lightGray"/>
139         <xsd:enumeration value="none"/>
140     </xsd:restriction>
141 </xsd:simpleType>
142 <xsd:complexType name="CT_Highlight">

```

```

143     <xsd:attribute name="val" type="ST_HighlightColor" use="required"/>
144 </xsd:complexType>
145 <xsd:simpleType name="ST_HexColorAuto">
146     <xsd:restriction base="xsd:string">
147         <xsd:enumeration value="auto"/>
148     </xsd:restriction>
149 </xsd:simpleType>
150 <xsd:simpleType name="ST_HexColor">
151     <xsd:union memberTypes="ST_HexColorAuto s:ST_HexColorRGB"/>
152 </xsd:simpleType>
153 <xsd:complexType name="CT_Color">
154     <xsd:attribute name="val" type="ST_HexColor" use="required"/>
155     <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
156     <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
157     <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
158 </xsd:complexType>
159 <xsd:complexType name="CT_Lang">
160     <xsd:attribute name="val" type="s:ST_Lang" use="required"/>
161 </xsd:complexType>
162 <xsd:complexType name="CT_Guid">
163     <xsd:attribute name="val" type="s:ST_Guid"/>
164 </xsd:complexType>
165 <xsd:simpleType name="ST_Underline">
166     <xsd:restriction base="xsd:string">
167         <xsd:enumeration value="single"/>
168         <xsd:enumeration value="words"/>
169         <xsd:enumeration value="double"/>
170         <xsd:enumeration value="thick"/>
171         <xsd:enumeration value="dotted"/>
172         <xsd:enumeration value="dottedHeavy"/>
173         <xsd:enumeration value="dash"/>
174         <xsd:enumeration value="dashedHeavy"/>
175         <xsd:enumeration value="dashLong"/>
176         <xsd:enumeration value="dashLongHeavy"/>
177         <xsd:enumeration value="dotDash"/>
178         <xsd:enumeration value="dashDotHeavy"/>
179         <xsd:enumeration value="dotDotDash"/>
180         <xsd:enumeration value="dashDotDotHeavy"/>
181         <xsd:enumeration value="wave"/>
182         <xsd:enumeration value="wavyHeavy"/>
183         <xsd:enumeration value="wavyDouble"/>
184         <xsd:enumeration value="none"/>
185     </xsd:restriction>
186 </xsd:simpleType>
187 <xsd:complexType name="CT_Underline">
188     <xsd:attribute name="val" type="ST_Underline" use="optional"/>
189     <xsd:attribute name="color" type="ST_HexColor" use="optional"/>
190     <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
191     <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
192     <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
193 </xsd:complexType>
194 <xsd:simpleType name="ST_TextEffect">
195     <xsd:restriction base="xsd:string">

```

```

196         <xsd:enumeration value="blinkBackground"/>
197         <xsd:enumeration value="lights"/>
198         <xsd:enumeration value="antsBlack"/>
199         <xsd:enumeration value="antsRed"/>
200         <xsd:enumeration value="shimmer"/>
201         <xsd:enumeration value="sparkle"/>
202         <xsd:enumeration value="none"/>
203     </xsd:restriction>
204 </xsd:simpleType>
205 <xsd:complexType name="CT_TextEffect">
206     <xsd:attribute name="val" type="ST_TextEffect" use="required"/>
207 </xsd:complexType>
208 <xsd:simpleType name="ST_Border">
209     <xsd:restriction base="xsd:string">
210         <xsd:enumeration value="nil"/>
211         <xsd:enumeration value="none"/>
212         <xsd:enumeration value="single"/>
213         <xsd:enumeration value="thick"/>
214         <xsd:enumeration value="double"/>
215         <xsd:enumeration value="dotted"/>
216         <xsd:enumeration value="dashed"/>
217         <xsd:enumeration value="dotDash"/>
218         <xsd:enumeration value="dotDotDash"/>
219         <xsd:enumeration value="triple"/>
220         <xsd:enumeration value="thinThickSmallGap"/>
221         <xsd:enumeration value="thickThinSmallGap"/>
222         <xsd:enumeration value="thinThickThinSmallGap"/>
223         <xsd:enumeration value="thinThickMediumGap"/>
224         <xsd:enumeration value="thickThinMediumGap"/>
225         <xsd:enumeration value="thinThickThinMediumGap"/>
226         <xsd:enumeration value="thinThickLargeGap"/>
227         <xsd:enumeration value="thickThinLargeGap"/>
228         <xsd:enumeration value="thinThickThinLargeGap"/>
229         <xsd:enumeration value="wave"/>
230         <xsd:enumeration value="doubleWave"/>
231         <xsd:enumeration value="dashSmallGap"/>
232         <xsd:enumeration value="dashDotStroked"/>
233         <xsd:enumeration value="threeDEmboss"/>
234         <xsd:enumeration value="threeDEngrave"/>
235         <xsd:enumeration value="outset"/>
236         <xsd:enumeration value="inset"/>
237         <xsd:enumeration value="apples"/>
238         <xsd:enumeration value="archedScallops"/>
239         <xsd:enumeration value="babyPacifier"/>
240         <xsd:enumeration value="babyRattle"/>
241         <xsd:enumeration value="balloons3Colors"/>
242         <xsd:enumeration value="balloonsHotAir"/>
243         <xsd:enumeration value="basicBlackDashes"/>
244         <xsd:enumeration value="basicBlackDots"/>
245         <xsd:enumeration value="basicBlackSquares"/>
246         <xsd:enumeration value="basicThinLines"/>
247         <xsd:enumeration value="basicWhiteDashes"/>
248         <xsd:enumeration value="basicWhiteDots"/>

```

```

249     <xsd:enumeration value="basicWhiteSquares"/>
250     <xsd:enumeration value="basicWideInline"/>
251     <xsd:enumeration value="basicWideMidline"/>
252     <xsd:enumeration value="basicWideOutline"/>
253     <xsd:enumeration value="bats"/>
254     <xsd:enumeration value="birds"/>
255     <xsd:enumeration value="birdsFlight"/>
256     <xsd:enumeration value="cabins"/>
257     <xsd:enumeration value="cakeSlice"/>
258     <xsd:enumeration value="candyCorn"/>
259     <xsd:enumeration value="celticKnotwork"/>
260     <xsd:enumeration value="certificateBanner"/>
261     <xsd:enumeration value="chainLink"/>
262     <xsd:enumeration value="champagneBottle"/>
263     <xsd:enumeration value="checkedBarBlack"/>
264     <xsd:enumeration value="checkedBarColor"/>
265     <xsd:enumeration value="checkered"/>
266     <xsd:enumeration value="christmasTree"/>
267     <xsd:enumeration value="circlesLines"/>
268     <xsd:enumeration value="circlesRectangles"/>
269     <xsd:enumeration value="classicalWave"/>
270     <xsd:enumeration value="clocks"/>
271     <xsd:enumeration value="compass"/>
272     <xsd:enumeration value="confetti"/>
273     <xsd:enumeration value="confettiGrays"/>
274     <xsd:enumeration value="confettiOutline"/>
275     <xsd:enumeration value="confettiStreamers"/>
276     <xsd:enumeration value="confettiWhite"/>
277     <xsd:enumeration value="cornerTriangles"/>
278     <xsd:enumeration value="couponCutoutDashes"/>
279     <xsd:enumeration value="couponCutoutDots"/>
280     <xsd:enumeration value="crazyMaze"/>
281     <xsd:enumeration value="creaturesButterfly"/>
282     <xsd:enumeration value="creaturesFish"/>
283     <xsd:enumeration value="creaturesInsects"/>
284     <xsd:enumeration value="creaturesLadyBug"/>
285     <xsd:enumeration value="crossStitch"/>
286     <xsd:enumeration value="cup"/>
287     <xsd:enumeration value="decoArch"/>
288     <xsd:enumeration value="decoArchColor"/>
289     <xsd:enumeration value="decoBlocks"/>
290     <xsd:enumeration value="diamondsGray"/>
291     <xsd:enumeration value="doubled"/>
292     <xsd:enumeration value="doubleDiamonds"/>
293     <xsd:enumeration value="earth1"/>
294     <xsd:enumeration value="earth2"/>
295     <xsd:enumeration value="earth3"/>
296     <xsd:enumeration value="eclipsingSquares1"/>
297     <xsd:enumeration value="eclipsingSquares2"/>
298     <xsd:enumeration value="eggsBlack"/>
299     <xsd:enumeration value="fans"/>
300     <xsd:enumeration value="film"/>
301     <xsd:enumeration value="firecrackers"/>

```

```

302     <xsd:enumeration value="flowersBlockPrint"/>
303     <xsd:enumeration value="flowersDaisies"/>
304     <xsd:enumeration value="flowersModern1"/>
305     <xsd:enumeration value="flowersModern2"/>
306     <xsd:enumeration value="flowersPansy"/>
307     <xsd:enumeration value="flowersRedRose"/>
308     <xsd:enumeration value="flowersRoses"/>
309     <xsd:enumeration value="flowersTeacup"/>
310     <xsd:enumeration value="flowersTiny"/>
311     <xsd:enumeration value="gems"/>
312     <xsd:enumeration value="gingerbreadMan"/>
313     <xsd:enumeration value="gradient"/>
314     <xsd:enumeration value="handmade1"/>
315     <xsd:enumeration value="handmade2"/>
316     <xsd:enumeration value="heartBalloon"/>
317     <xsd:enumeration value="heartGray"/>
318     <xsd:enumeration value="hearts"/>
319     <xsd:enumeration value="heebieJeebies"/>
320     <xsd:enumeration value="holly"/>
321     <xsd:enumeration value="houseFunky"/>
322     <xsd:enumeration value="hypnotic"/>
323     <xsd:enumeration value="iceCreamCones"/>
324     <xsd:enumeration value="lightBulb"/>
325     <xsd:enumeration value="lightning1"/>
326     <xsd:enumeration value="lightning2"/>
327     <xsd:enumeration value="mapPins"/>
328     <xsd:enumeration value="mapleLeaf"/>
329     <xsd:enumeration value="mapleMuffins"/>
330     <xsd:enumeration value="marquee"/>
331     <xsd:enumeration value="marqueeToothed"/>
332     <xsd:enumeration value="moons"/>
333     <xsd:enumeration value="mosaic"/>
334     <xsd:enumeration value="musicNotes"/>
335     <xsd:enumeration value="northwest"/>
336     <xsd:enumeration value="ovals"/>
337     <xsd:enumeration value="packages"/>
338     <xsd:enumeration value="palmsBlack"/>
339     <xsd:enumeration value="palmsColor"/>
340     <xsd:enumeration value="paperClips"/>
341     <xsd:enumeration value="papyrus"/>
342     <xsd:enumeration value="partyFavor"/>
343     <xsd:enumeration value="partyGlass"/>
344     <xsd:enumeration value="pencils"/>
345     <xsd:enumeration value="people"/>
346     <xsd:enumeration value="peopleWaving"/>
347     <xsd:enumeration value="peopleHats"/>
348     <xsd:enumeration value="poinsettias"/>
349     <xsd:enumeration value="postageStamp"/>
350     <xsd:enumeration value="pumpkin1"/>
351     <xsd:enumeration value="pushPinNote2"/>
352     <xsd:enumeration value="pushPinNote1"/>
353     <xsd:enumeration value="pyramids"/>
354     <xsd:enumeration value="pyramidsAbove"/>

```

```

355     <xsd:enumeration value="quadrants"/>
356     <xsd:enumeration value="rings"/>
357     <xsd:enumeration value="safari"/>
358     <xsd:enumeration value="sawtooth"/>
359     <xsd:enumeration value="sawtoothGray"/>
360     <xsd:enumeration value="scaredCat"/>
361     <xsd:enumeration value="seattle"/>
362     <xsd:enumeration value="shadowedSquares"/>
363     <xsd:enumeration value="sharksTeeth"/>
364     <xsd:enumeration value="shorebirdTracks"/>
365     <xsd:enumeration value="skyrocket"/>
366     <xsd:enumeration value="snowflakeFancy"/>
367     <xsd:enumeration value="snowflakes"/>
368     <xsd:enumeration value="sombbrero"/>
369     <xsd:enumeration value="southwest"/>
370     <xsd:enumeration value="stars"/>
371     <xsd:enumeration value="starsTop"/>
372     <xsd:enumeration value="stars3d"/>
373     <xsd:enumeration value="starsBlack"/>
374     <xsd:enumeration value="starsShadowed"/>
375     <xsd:enumeration value="sun"/>
376     <xsd:enumeration value="swirligig"/>
377     <xsd:enumeration value="tornPaper"/>
378     <xsd:enumeration value="tornPaperBlack"/>
379     <xsd:enumeration value="trees"/>
380     <xsd:enumeration value="triangleParty"/>
381     <xsd:enumeration value="triangles"/>
382     <xsd:enumeration value="triangle1"/>
383     <xsd:enumeration value="triangle2"/>
384     <xsd:enumeration value="triangleCircle1"/>
385     <xsd:enumeration value="triangleCircle2"/>
386     <xsd:enumeration value="shapes1"/>
387     <xsd:enumeration value="shapes2"/>
388     <xsd:enumeration value="twistedLines1"/>
389     <xsd:enumeration value="twistedLines2"/>
390     <xsd:enumeration value="vine"/>
391     <xsd:enumeration value="waveline"/>
392     <xsd:enumeration value="weavingAngles"/>
393     <xsd:enumeration value="weavingBraid"/>
394     <xsd:enumeration value="weavingRibbon"/>
395     <xsd:enumeration value="weavingStrips"/>
396     <xsd:enumeration value="whiteFlowers"/>
397     <xsd:enumeration value="woodwork"/>
398     <xsd:enumeration value="xIllusions"/>
399     <xsd:enumeration value="zanyTriangles"/>
400     <xsd:enumeration value="zigZag"/>
401     <xsd:enumeration value="zigZagStitch"/>
402     <xsd:enumeration value="custom"/>
403   </xsd:restriction>
404 </xsd:simpleType>
405 <xsd:complexType name="CT_Border">
406   <xsd:attribute name="val" type="ST_Border" use="required"/>
407   <xsd:attribute name="color" type="ST_HexColor" use="optional"/>

```



```

408     <xsd:attribute name="themeColor" type="ST ThemeColor" use="optional"/>
409     <xsd:attribute name="themeTint" type="ST UcharHexNumber" use="optional"/>
410     <xsd:attribute name="themeShade" type="ST UcharHexNumber" use="optional"/>
411     <xsd:attribute name="sz" type="ST EighthPointMeasure" use="optional"/>
412     <xsd:attribute name="space" type="ST PointMeasure" use="optional"/>
413     <xsd:attribute name="shadow" type="s:ST OnOff" use="optional"/>
414     <xsd:attribute name="frame" type="s:ST OnOff" use="optional"/>
415 </xsd:complexType>
416 <xsd:simpleType name="ST_Shd">
417     <xsd:restriction base="xsd:string">
418         <xsd:enumeration value="nil"/>
419         <xsd:enumeration value="clear"/>
420         <xsd:enumeration value="solid"/>
421         <xsd:enumeration value="horzStripe"/>
422         <xsd:enumeration value="vertStripe"/>
423         <xsd:enumeration value="reverseDiagStripe"/>
424         <xsd:enumeration value="diagStripe"/>
425         <xsd:enumeration value="horzCross"/>
426         <xsd:enumeration value="diagCross"/>
427         <xsd:enumeration value="thinHorzStripe"/>
428         <xsd:enumeration value="thinVertStripe"/>
429         <xsd:enumeration value="thinReverseDiagStripe"/>
430         <xsd:enumeration value="thinDiagStripe"/>
431         <xsd:enumeration value="thinHorzCross"/>
432         <xsd:enumeration value="thinDiagCross"/>
433         <xsd:enumeration value="pct5"/>
434         <xsd:enumeration value="pct10"/>
435         <xsd:enumeration value="pct12"/>
436         <xsd:enumeration value="pct15"/>
437         <xsd:enumeration value="pct20"/>
438         <xsd:enumeration value="pct25"/>
439         <xsd:enumeration value="pct30"/>
440         <xsd:enumeration value="pct35"/>
441         <xsd:enumeration value="pct37"/>
442         <xsd:enumeration value="pct40"/>
443         <xsd:enumeration value="pct45"/>
444         <xsd:enumeration value="pct50"/>
445         <xsd:enumeration value="pct55"/>
446         <xsd:enumeration value="pct60"/>
447         <xsd:enumeration value="pct62"/>
448         <xsd:enumeration value="pct65"/>
449         <xsd:enumeration value="pct70"/>
450         <xsd:enumeration value="pct75"/>
451         <xsd:enumeration value="pct80"/>
452         <xsd:enumeration value="pct85"/>
453         <xsd:enumeration value="pct87"/>
454         <xsd:enumeration value="pct90"/>
455         <xsd:enumeration value="pct95"/>
456     </xsd:restriction>
457 </xsd:simpleType>
458 <xsd:complexType name="CT_Shd">
459     <xsd:attribute name="val" type="ST_Shd" use="required"/>
460     <xsd:attribute name="color" type="ST HexColor" use="optional"/>

```

```

461     <xsd:attribute name="themeColor" type="ST ThemeColor" use="optional"/>
462     <xsd:attribute name="themeTint" type="ST UcharHexNumber" use="optional"/>
463     <xsd:attribute name="themeShade" type="ST UcharHexNumber" use="optional"/>
464     <xsd:attribute name="fill" type="ST HexColor" use="optional"/>
465     <xsd:attribute name="themeFill" type="ST ThemeColor" use="optional"/>
466     <xsd:attribute name="themeFillTint" type="ST UcharHexNumber" use="optional"/>
467     <xsd:attribute name="themeFillShade" type="ST UcharHexNumber" use="optional"/>
468 </xsd:complexType>
469 <xsd:complexType name="CT_VerticalAlignRun">
470     <xsd:attribute name="val" type="s:ST VerticalAlignRun" use="required"/>
471 </xsd:complexType>
472 <xsd:complexType name="CT_FitText">
473     <xsd:attribute name="val" type="s:ST TwipsMeasure" use="required"/>
474     <xsd:attribute name="id" type="ST DecimalNumber" use="optional"/>
475 </xsd:complexType>
476 <xsd:simpleType name="ST_Em">
477     <xsd:restriction base="xsd:string">
478         <xsd:enumeration value="none"/>
479         <xsd:enumeration value="dot"/>
480         <xsd:enumeration value="comma"/>
481         <xsd:enumeration value="circle"/>
482         <xsd:enumeration value="underDot"/>
483     </xsd:restriction>
484 </xsd:simpleType>
485 <xsd:complexType name="CT_Em">
486     <xsd:attribute name="val" type="ST_Em" use="required"/>
487 </xsd:complexType>
488 <xsd:complexType name="CT_Language">
489     <xsd:attribute name="val" type="s:ST Lang" use="optional"/>
490     <xsd:attribute name="eastAsia" type="s:ST Lang" use="optional"/>
491     <xsd:attribute name="bidi" type="s:ST Lang" use="optional"/>
492 </xsd:complexType>
493 <xsd:simpleType name="ST_CombineBrackets">
494     <xsd:restriction base="xsd:string">
495         <xsd:enumeration value="none"/>
496         <xsd:enumeration value="round"/>
497         <xsd:enumeration value="square"/>
498         <xsd:enumeration value="angle"/>
499         <xsd:enumeration value="curly"/>
500     </xsd:restriction>
501 </xsd:simpleType>
502 <xsd:complexType name="CT_EastAsianLayout">
503     <xsd:attribute name="id" type="ST DecimalNumber" use="optional"/>
504     <xsd:attribute name="combine" type="s:ST OnOff" use="optional"/>
505     <xsd:attribute name="combineBrackets" type="ST_CombineBrackets" use="optional"/>
506     <xsd:attribute name="vert" type="s:ST OnOff" use="optional"/>
507     <xsd:attribute name="vertCompress" type="s:ST OnOff" use="optional"/>
508 </xsd:complexType>
509 <xsd:simpleType name="ST_HeightRule">
510     <xsd:restriction base="xsd:string">
511         <xsd:enumeration value="auto"/>
512         <xsd:enumeration value="exact"/>
513         <xsd:enumeration value="atLeast"/>

```

```

514     </xsd:restriction>
515 </xsd:simpleType>
516 <xsd:simpleType name="ST_Wrap">
517     <xsd:restriction base="xsd:string">
518         <xsd:enumeration value="auto"/>
519         <xsd:enumeration value="notBeside"/>
520         <xsd:enumeration value="around"/>
521         <xsd:enumeration value="tight"/>
522         <xsd:enumeration value="through"/>
523         <xsd:enumeration value="none"/>
524     </xsd:restriction>
525 </xsd:simpleType>
526 <xsd:simpleType name="ST_VAnchor">
527     <xsd:restriction base="xsd:string">
528         <xsd:enumeration value="text"/>
529         <xsd:enumeration value="margin"/>
530         <xsd:enumeration value="page"/>
531     </xsd:restriction>
532 </xsd:simpleType>
533 <xsd:simpleType name="ST_HAnchor">
534     <xsd:restriction base="xsd:string">
535         <xsd:enumeration value="text"/>
536         <xsd:enumeration value="margin"/>
537         <xsd:enumeration value="page"/>
538     </xsd:restriction>
539 </xsd:simpleType>
540 <xsd:simpleType name="ST_DropCap">
541     <xsd:restriction base="xsd:string">
542         <xsd:enumeration value="none"/>
543         <xsd:enumeration value="drop"/>
544         <xsd:enumeration value="margin"/>
545     </xsd:restriction>
546 </xsd:simpleType>
547 <xsd:complexType name="CT_FramePr">
548     <xsd:attribute name="dropCap" type="ST_DropCap" use="optional"/>
549     <xsd:attribute name="lines" type="ST_DecimalNumber" use="optional"/>
550     <xsd:attribute name="w" type="s:ST_TwipsMeasure" use="optional"/>
551     <xsd:attribute name="h" type="s:ST_TwipsMeasure" use="optional"/>
552     <xsd:attribute name="vSpace" type="s:ST_TwipsMeasure" use="optional"/>
553     <xsd:attribute name="hSpace" type="s:ST_TwipsMeasure" use="optional"/>
554     <xsd:attribute name="wrap" type="ST_Wrap" use="optional"/>
555     <xsd:attribute name="hAnchor" type="ST_HAnchor" use="optional"/>
556     <xsd:attribute name="vAnchor" type="ST_VAnchor" use="optional"/>
557     <xsd:attribute name="x" type="ST_SignedTwipsMeasure" use="optional"/>
558     <xsd:attribute name="xAlign" type="s:ST_XAlign" use="optional"/>
559     <xsd:attribute name="y" type="ST_SignedTwipsMeasure" use="optional"/>
560     <xsd:attribute name="yAlign" type="s:ST_YAlign" use="optional"/>
561     <xsd:attribute name="hRule" type="ST_HeightRule" use="optional"/>
562     <xsd:attribute name="anchorLock" type="s:ST_OnOff" use="optional"/>
563 </xsd:complexType>
564 <xsd:simpleType name="ST_TabJc">
565     <xsd:restriction base="xsd:string">
566         <xsd:enumeration value="clear"/>

```

```

567         <xsd:enumeration value="start"/>
568         <xsd:enumeration value="center"/>
569         <xsd:enumeration value="end"/>
570         <xsd:enumeration value="decimal"/>
571         <xsd:enumeration value="bar"/>
572         <xsd:enumeration value="num"/>
573         <xsd:enumeration value="left"/>
574         <xsd:enumeration value="right"/>
575     </xsd:restriction>
576 </xsd:simpleType>
577 <xsd:simpleType name="ST_TabTlc">
578     <xsd:restriction base="xsd:string">
579         <xsd:enumeration value="none"/>
580         <xsd:enumeration value="dot"/>
581         <xsd:enumeration value="hyphen"/>
582         <xsd:enumeration value="underscore"/>
583         <xsd:enumeration value="heavy"/>
584         <xsd:enumeration value="middleDot"/>
585     </xsd:restriction>
586 </xsd:simpleType>
587 <xsd:complexType name="CT_TabStop">
588     <xsd:attribute name="val" type="ST_TabJc" use="required"/>
589     <xsd:attribute name="leader" type="ST_TabTlc" use="optional"/>
590     <xsd:attribute name="pos" type="ST_SignedTwipsMeasure" use="required"/>
591 </xsd:complexType>
592 <xsd:simpleType name="ST_LineSpacingRule">
593     <xsd:restriction base="xsd:string">
594         <xsd:enumeration value="auto"/>
595         <xsd:enumeration value="exact"/>
596         <xsd:enumeration value="atLeast"/>
597     </xsd:restriction>
598 </xsd:simpleType>
599 <xsd:complexType name="CT_Spacing">
600     <xsd:attribute name="before" type="s:ST_TwipsMeasure" use="optional"/>
601     <xsd:attribute name="beforeLines" type="ST_DecimalNumber" use="optional"/>
602     <xsd:attribute name="beforeAutospacing" type="s:ST_OnOff" use="optional"/>
603     <xsd:attribute name="after" type="s:ST_TwipsMeasure" use="optional"/>
604     <xsd:attribute name="afterLines" type="ST_DecimalNumber" use="optional"/>
605     <xsd:attribute name="afterAutospacing" type="s:ST_OnOff" use="optional"/>
606     <xsd:attribute name="line" type="ST_SignedTwipsMeasure" use="optional"/>
607     <xsd:attribute name="lineRule" type="ST_LineSpacingRule" use="optional"/>
608 </xsd:complexType>
609 <xsd:complexType name="CT_Ind">
610     <xsd:attribute name="start" type="ST_SignedTwipsMeasure" use="optional"/>
611     <xsd:attribute name="startChars" type="ST_DecimalNumber" use="optional"/>
612     <xsd:attribute name="end" type="ST_SignedTwipsMeasure" use="optional"/>
613     <xsd:attribute name="endChars" type="ST_DecimalNumber" use="optional"/>
614     <xsd:attribute name="left" type="ST_SignedTwipsMeasure" use="optional"/>
615     <xsd:attribute name="leftChars" type="ST_DecimalNumber" use="optional"/>
616     <xsd:attribute name="right" type="ST_SignedTwipsMeasure" use="optional"/>
617     <xsd:attribute name="rightChars" type="ST_DecimalNumber" use="optional"/>
618     <xsd:attribute name="hanging" type="s:ST_TwipsMeasure" use="optional"/>
619     <xsd:attribute name="hangingChars" type="ST_DecimalNumber" use="optional"/>

```

```

620     <xsd:attribute name="firstLine" type="s:ST TwipsMeasure" use="optional"/>
621     <xsd:attribute name="firstLineChars" type="ST DecimalNumber" use="optional"/>
622 </xsd:complexType>
623 <xsd:simpleType name="ST_Jc">
624     <xsd:restriction base="xsd:string">
625         <xsd:enumeration value="start"/>
626         <xsd:enumeration value="center"/>
627         <xsd:enumeration value="end"/>
628         <xsd:enumeration value="both"/>
629         <xsd:enumeration value="mediumKashida"/>
630         <xsd:enumeration value="distribute"/>
631         <xsd:enumeration value="numTab"/>
632         <xsd:enumeration value="highKashida"/>
633         <xsd:enumeration value="lowKashida"/>
634         <xsd:enumeration value="thaiDistribute"/>
635         <xsd:enumeration value="left"/>
636         <xsd:enumeration value="right"/>
637     </xsd:restriction>
638 </xsd:simpleType>
639 <xsd:simpleType name="ST_JcTable">
640     <xsd:restriction base="xsd:string">
641         <xsd:enumeration value="center"/>
642         <xsd:enumeration value="end"/>
643         <xsd:enumeration value="left"/>
644         <xsd:enumeration value="right"/>
645         <xsd:enumeration value="start"/>
646     </xsd:restriction>
647 </xsd:simpleType>
648 <xsd:complexType name="CT_Jc">
649     <xsd:attribute name="val" type="ST_Jc" use="required"/>
650 </xsd:complexType>
651 <xsd:complexType name="CT_JcTable">
652     <xsd:attribute name="val" type="ST_JcTable" use="required"/>
653 </xsd:complexType>
654 <xsd:simpleType name="ST_View">
655     <xsd:restriction base="xsd:string">
656         <xsd:enumeration value="none"/>
657         <xsd:enumeration value="print"/>
658         <xsd:enumeration value="outline"/>
659         <xsd:enumeration value="masterPages"/>
660         <xsd:enumeration value="normal"/>
661         <xsd:enumeration value="web"/>
662     </xsd:restriction>
663 </xsd:simpleType>
664 <xsd:complexType name="CT_View">
665     <xsd:attribute name="val" type="ST_View" use="required"/>
666 </xsd:complexType>
667 <xsd:simpleType name="ST_Zoom">
668     <xsd:restriction base="xsd:string">
669         <xsd:enumeration value="none"/>
670         <xsd:enumeration value="fullPage"/>
671         <xsd:enumeration value="bestFit"/>
672         <xsd:enumeration value="textFit"/>

```

```

673     </xsd:restriction>
674 </xsd:simpleType>
675 <xsd:complexType name="CT_Zoom">
676     <xsd:attribute name="val" type="ST_Zoom" use="optional"/>
677     <xsd:attribute name="percent" type="ST_DecimalNumberOrPercent" use="required"/>
678 </xsd:complexType>
679 <xsd:complexType name="CT_WritingStyle">
680     <xsd:attribute name="lang" type="s:ST_Lang" use="required"/>
681     <xsd:attribute name="vendorID" type="s:ST_String" use="required"/>
682     <xsd:attribute name="dllVersion" type="s:ST_String" use="required"/>
683     <xsd:attribute name="nlCheck" type="s:ST_OnOff" use="optional"/>
684     <xsd:attribute name="checkStyle" type="s:ST_OnOff" use="required"/>
685     <xsd:attribute name="appName" type="s:ST_String" use="required"/>
686 </xsd:complexType>
687 <xsd:simpleType name="ST_Proof">
688     <xsd:restriction base="xsd:string">
689         <xsd:enumeration value="clean"/>
690         <xsd:enumeration value="dirty"/>
691     </xsd:restriction>
692 </xsd:simpleType>
693 <xsd:complexType name="CT_Proof">
694     <xsd:attribute name="spelling" type="ST_Proof" use="optional"/>
695     <xsd:attribute name="grammar" type="ST_Proof" use="optional"/>
696 </xsd:complexType>
697 <xsd:simpleType name="ST_DocType">
698     <xsd:restriction base="xsd:string"/>
699 </xsd:simpleType>
700 <xsd:complexType name="CT_DocType">
701     <xsd:attribute name="val" type="ST_DocType" use="required"/>
702 </xsd:complexType>
703 <xsd:simpleType name="ST_DocProtect">
704     <xsd:restriction base="xsd:string">
705         <xsd:enumeration value="none"/>
706         <xsd:enumeration value="readOnly"/>
707         <xsd:enumeration value="comments"/>
708         <xsd:enumeration value="trackedChanges"/>
709         <xsd:enumeration value="forms"/>
710     </xsd:restriction>
711 </xsd:simpleType>
712 <xsd:attributeGroup name="AG_Password">
713     <xsd:attribute name="algorithmName" type="s:ST_String" use="optional"/>
714     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
715     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
716     <xsd:attribute name="spinCount" type="ST_DecimalNumber" use="optional"/>
717 </xsd:attributeGroup>
718 <xsd:attributeGroup name="AG_TransitionalPassword">
719     <xsd:attribute name="cryptProviderType" type="s:ST_CryptProv"/>
720     <xsd:attribute name="cryptAlgorithmClass" type="s:ST_AlgorithmClass"/>
721     <xsd:attribute name="cryptAlgorithmType" type="s:ST_AlgorithmType"/>
722     <xsd:attribute name="cryptAlgorithmSid" type="ST_DecimalNumber"/>
723     <xsd:attribute name="cryptSpinCount" type="ST_DecimalNumber"/>
724     <xsd:attribute name="cryptProvider" type="s:ST_String"/>
725     <xsd:attribute name="algIdExt" type="ST_LongHexNumber"/>

```

```

726     <xsd:attribute name="algIdExtSource" type="s:ST_String"/>
727     <xsd:attribute name="cryptProviderTypeExt" type="ST_LongHexNumber"/>
728     <xsd:attribute name="cryptProviderTypeExtSource" type="s:ST_String"/>
729     <xsd:attribute name="hash" type="xsd:base64Binary"/>
730     <xsd:attribute name="salt" type="xsd:base64Binary"/>
731 </xsd:attributeGroup>
732 <xsd:complexType name="CT_DocProtect">
733     <xsd:attribute name="edit" type="ST_DocProtect" use="optional"/>
734     <xsd:attribute name="formatting" type="s:ST_OnOff" use="optional"/>
735     <xsd:attribute name="enforcement" type="s:ST_OnOff"/>
736     <xsd:attributeGroup ref="AG_Password"/>
737     <xsd:attributeGroup ref="AG_TransitionalPassword"/>
738 </xsd:complexType>
739 <xsd:simpleType name="ST_MailMergeDocType">
740     <xsd:restriction base="xsd:string">
741         <xsd:enumeration value="catalog"/>
742         <xsd:enumeration value="envelopes"/>
743         <xsd:enumeration value="mailingLabels"/>
744         <xsd:enumeration value="formLetters"/>
745         <xsd:enumeration value="email"/>
746         <xsd:enumeration value="fax"/>
747     </xsd:restriction>
748 </xsd:simpleType>
749 <xsd:complexType name="CT_MailMergeDocType">
750     <xsd:attribute name="val" type="ST_MailMergeDocType" use="required"/>
751 </xsd:complexType>
752 <xsd:simpleType name="ST_MailMergeDataType">
753     <xsd:restriction base="xsd:string"/>
754 </xsd:simpleType>
755 <xsd:complexType name="CT_MailMergeDataType">
756     <xsd:attribute name="val" type="ST_MailMergeDataType" use="required"/>
757 </xsd:complexType>
758 <xsd:simpleType name="ST_MailMergeDest">
759     <xsd:restriction base="xsd:string">
760         <xsd:enumeration value="newDocument"/>
761         <xsd:enumeration value="printer"/>
762         <xsd:enumeration value="email"/>
763         <xsd:enumeration value="fax"/>
764     </xsd:restriction>
765 </xsd:simpleType>
766 <xsd:complexType name="CT_MailMergeDest">
767     <xsd:attribute name="val" type="ST_MailMergeDest" use="required"/>
768 </xsd:complexType>
769 <xsd:simpleType name="ST_MailMergeOdsoFMDFieldType">
770     <xsd:restriction base="xsd:string">
771         <xsd:enumeration value="null"/>
772         <xsd:enumeration value="dbColumn"/>
773     </xsd:restriction>
774 </xsd:simpleType>
775 <xsd:complexType name="CT_MailMergeOdsoFMDFieldType">
776     <xsd:attribute name="val" type="ST_MailMergeOdsoFMDFieldType" use="required"/>
777 </xsd:complexType>
778 <xsd:complexType name="CT_TrackChangesView">

```

```

779     <xsd:attribute name="markup" type="s:ST_OnOff" use="optional"/>
780     <xsd:attribute name="comments" type="s:ST_OnOff" use="optional"/>
781     <xsd:attribute name="insDel" type="s:ST_OnOff" use="optional"/>
782     <xsd:attribute name="formatting" type="s:ST_OnOff" use="optional"/>
783     <xsd:attribute name="inkAnnotations" type="s:ST_OnOff" use="optional"/>
784 </xsd:complexType>
785 <xsd:complexType name="CT_Kinsoku">
786     <xsd:attribute name="lang" type="s:ST_Lang" use="required"/>
787     <xsd:attribute name="val" type="s:ST_String" use="required"/>
788 </xsd:complexType>
789 <xsd:simpleType name="ST_TextDirection">
790     <xsd:restriction base="xsd:string">
791         <xsd:enumeration value="tb"/>
792         <xsd:enumeration value="rl"/>
793         <xsd:enumeration value="lr"/>
794         <xsd:enumeration value="tbV"/>
795         <xsd:enumeration value="rlV"/>
796         <xsd:enumeration value="lrV"/>
797         <xsd:enumeration value="btLr"/>
798         <xsd:enumeration value="lrTb"/>
799         <xsd:enumeration value="lrTbV"/>
800         <xsd:enumeration value="tbLrV"/>
801         <xsd:enumeration value="tbRl"/>
802         <xsd:enumeration value="tbRlV"/>
803     </xsd:restriction>
804 </xsd:simpleType>
805 <xsd:complexType name="CT_TextDirection">
806     <xsd:attribute name="val" type="ST_TextDirection" use="required"/>
807 </xsd:complexType>
808 <xsd:simpleType name="ST_TextAlignment">
809     <xsd:restriction base="xsd:string">
810         <xsd:enumeration value="top"/>
811         <xsd:enumeration value="center"/>
812         <xsd:enumeration value="baseline"/>
813         <xsd:enumeration value="bottom"/>
814         <xsd:enumeration value="auto"/>
815     </xsd:restriction>
816 </xsd:simpleType>
817 <xsd:complexType name="CT_TextAlignment">
818     <xsd:attribute name="val" type="ST_TextAlignment" use="required"/>
819 </xsd:complexType>
820 <xsd:simpleType name="ST_DisplacedByCustomXml">
821     <xsd:restriction base="xsd:string">
822         <xsd:enumeration value="next"/>
823         <xsd:enumeration value="prev"/>
824     </xsd:restriction>
825 </xsd:simpleType>
826 <xsd:simpleType name="ST_AnnotationVMerge">
827     <xsd:restriction base="xsd:string">
828         <xsd:enumeration value="cont"/>
829         <xsd:enumeration value="rest"/>
830     </xsd:restriction>
831 </xsd:simpleType>

```



```

832 <xsd:complexType name="CT_Markup">
833   <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
834 </xsd:complexType>
835 <xsd:complexType name="CT_TrackChange">
836   <xsd:complexContent>
837     <xsd:extension base="CT_Markup">
838       <xsd:attribute name="author" type="s:ST_String" use="required"/>
839       <xsd:attribute name="date" type="ST_DateTime" use="optional"/>
840     </xsd:extension>
841   </xsd:complexContent>
842 </xsd:complexType>
843 <xsd:complexType name="CT_CellMergeTrackChange">
844   <xsd:complexContent>
845     <xsd:extension base="CT_TrackChange">
846       <xsd:attribute name="vMerge" type="ST_AnnotationVMerge" use="optional"/>
847       <xsd:attribute name="vMergeOrig" type="ST_AnnotationVMerge" use="optional"/>
848     </xsd:extension>
849   </xsd:complexContent>
850 </xsd:complexType>
851 <xsd:complexType name="CT_TrackChangeRange">
852   <xsd:complexContent>
853     <xsd:extension base="CT_TrackChange">
854       <xsd:attribute name="displacedByCustomXml" type="ST_DisplacedByCustomXml"
855         use="optional"/>
856     </xsd:extension>
857   </xsd:complexContent>
858 </xsd:complexType>
859 <xsd:complexType name="CT_MarkupRange">
860   <xsd:complexContent>
861     <xsd:extension base="CT_Markup">
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_BookmarkRange">
868   <xsd:complexContent>
869     <xsd:extension base="CT_MarkupRange">
870       <xsd:attribute name="colFirst" type="ST_DecimalNumber" use="optional"/>
871       <xsd:attribute name="colLast" type="ST_DecimalNumber" use="optional"/>
872     </xsd:extension>
873   </xsd:complexContent>
874 </xsd:complexType>
875 <xsd:complexType name="CT_Bookmark">
876   <xsd:complexContent>
877     <xsd:extension base="CT_BookmarkRange">
878       <xsd:attribute name="name" type="s:ST_String" use="required"/>
879     </xsd:extension>
880   </xsd:complexContent>
881 </xsd:complexType>
882 <xsd:complexType name="CT_MoveBookmark">
883   <xsd:complexContent>
884     <xsd:extension base="CT_Bookmark">

```

```

885         <xsd:attribute name="author" type="s:ST_String" use="required"/>
886         <xsd:attribute name="date" type="ST_DateTime" use="required"/>
887     </xsd:extension>
888 </xsd:complexContent>
889 </xsd:complexType>
890 <xsd:complexType name="CT_Comment">
891     <xsd:complexContent>
892         <xsd:extension base="CT_TrackChange">
893             <xsd:sequence>
894                 <xsd:group ref="EG_BlockLevelElts" minOccurs="0" maxOccurs="unbounded"/>
895             </xsd:sequence>
896             <xsd:attribute name="initials" type="s:ST_String" use="optional"/>
897         </xsd:extension>
898     </xsd:complexContent>
899 </xsd:complexType>
900 <xsd:complexType name="CT_TrackChangeNumbering">
901     <xsd:complexContent>
902         <xsd:extension base="CT_TrackChange">
903             <xsd:attribute name="original" type="s:ST_String" use="optional"/>
904         </xsd:extension>
905     </xsd:complexContent>
906 </xsd:complexType>
907 <xsd:complexType name="CT_TblPrExChange">
908     <xsd:complexContent>
909         <xsd:extension base="CT_TrackChange">
910             <xsd:sequence>
911                 <xsd:element name="tblPrEx" type="CT_TblPrExBase" minOccurs="1"/>
912             </xsd:sequence>
913         </xsd:extension>
914     </xsd:complexContent>
915 </xsd:complexType>
916 <xsd:complexType name="CT_TcPrChange">
917     <xsd:complexContent>
918         <xsd:extension base="CT_TrackChange">
919             <xsd:sequence>
920                 <xsd:element name="tcPr" type="CT_TcPrInner" minOccurs="1"/>
921             </xsd:sequence>
922         </xsd:extension>
923     </xsd:complexContent>
924 </xsd:complexType>
925 <xsd:complexType name="CT_TrPrChange">
926     <xsd:complexContent>
927         <xsd:extension base="CT_TrackChange">
928             <xsd:sequence>
929                 <xsd:element name="trPr" type="CT_TrPrBase" minOccurs="1"/>
930             </xsd:sequence>
931         </xsd:extension>
932     </xsd:complexContent>
933 </xsd:complexType>
934 <xsd:complexType name="CT_TblGridChange">
935     <xsd:complexContent>
936         <xsd:extension base="CT_Markup">
937             <xsd:sequence>

```

```

938         <xsd:element name="tblGrid" type="CT_TblGridBase"/>
939     </xsd:sequence>
940 </xsd:extension>
941 </xsd:complexContent>
942 </xsd:complexType>
943 <xsd:complexType name="CT_TblPrChange">
944     <xsd:complexContent>
945         <xsd:extension base="CT_TrackChange">
946             <xsd:sequence>
947                 <xsd:element name="tblPr" type="CT_TblPrBase"/>
948             </xsd:sequence>
949         </xsd:extension>
950     </xsd:complexContent>
951 </xsd:complexType>
952 <xsd:complexType name="CT_SectPrChange">
953     <xsd:complexContent>
954         <xsd:extension base="CT_TrackChange">
955             <xsd:sequence>
956                 <xsd:element name="sectPr" type="CT_SectPrBase" minOccurs="0"/>
957             </xsd:sequence>
958         </xsd:extension>
959     </xsd:complexContent>
960 </xsd:complexType>
961 <xsd:complexType name="CT_PPrChange">
962     <xsd:complexContent>
963         <xsd:extension base="CT_TrackChange">
964             <xsd:sequence>
965                 <xsd:element name="pPr" type="CT_PPrBase" minOccurs="1"/>
966             </xsd:sequence>
967         </xsd:extension>
968     </xsd:complexContent>
969 </xsd:complexType>
970 <xsd:complexType name="CT_RPrChange">
971     <xsd:complexContent>
972         <xsd:extension base="CT_TrackChange">
973             <xsd:sequence>
974                 <xsd:element name="rPr" type="CT_RPrOriginal" minOccurs="1"/>
975             </xsd:sequence>
976         </xsd:extension>
977     </xsd:complexContent>
978 </xsd:complexType>
979 <xsd:complexType name="CT_ParaRPrChange">
980     <xsd:complexContent>
981         <xsd:extension base="CT_TrackChange">
982             <xsd:sequence>
983                 <xsd:element name="rPr" type="CT_ParaRPrOriginal" minOccurs="1"/>
984             </xsd:sequence>
985         </xsd:extension>
986     </xsd:complexContent>
987 </xsd:complexType>
988 <xsd:complexType name="CT_RunTrackChange">
989     <xsd:complexContent>
990         <xsd:extension base="CT_TrackChange">

```

```

991         <xsd:choice minOccurs="0" maxOccurs="unbounded">
992             <xsd:group ref="EG_ContentRunContent"/>
993             <xsd:group ref="m:EG_OMathMathElements"/>
994         </xsd:choice>
995     </xsd:extension>
996 </xsd:complexContent>
997 </xsd:complexType>
998 <xsd:group name="EG_CellMarkupElements">
999     <xsd:choice>
1000         <xsd:element name="cellIns" type="CT_TrackChange" minOccurs="0"/>
1001         <xsd:element name="cellDel" type="CT_TrackChange" minOccurs="0"/>
1002         <xsd:element name="cellMerge" type="CT_CellMergeTrackChange" minOccurs="0"/>
1003     </xsd:choice>
1004 </xsd:group>
1005 <xsd:group name="EG_RangeMarkupElements">
1006     <xsd:choice>
1007         <xsd:element name="bookmarkStart" type="CT_Bookmark"/>
1008         <xsd:element name="bookmarkEnd" type="CT_MarkupRange"/>
1009         <xsd:element name="moveFromRangeStart" type="CT_MoveBookmark"/>
1010         <xsd:element name="moveFromRangeEnd" type="CT_MarkupRange"/>
1011         <xsd:element name="moveToRangeStart" type="CT_MoveBookmark"/>
1012         <xsd:element name="moveToRangeEnd" type="CT_MarkupRange"/>
1013         <xsd:element name="commentRangeStart" type="CT_MarkupRange"/>
1014         <xsd:element name="commentRangeEnd" type="CT_MarkupRange"/>
1015         <xsd:element name="customXmlInsRangeStart" type="CT_TrackChange"/>
1016         <xsd:element name="customXmlInsRangeEnd" type="CT_Markup"/>
1017         <xsd:element name="customXmlDelRangeStart" type="CT_TrackChange"/>
1018         <xsd:element name="customXmlDelRangeEnd" type="CT_Markup"/>
1019         <xsd:element name="customXmlMoveFromRangeStart" type="CT_TrackChange"/>
1020         <xsd:element name="customXmlMoveFromRangeEnd" type="CT_Markup"/>
1021         <xsd:element name="customXmlMoveToRangeStart" type="CT_TrackChange"/>
1022         <xsd:element name="customXmlMoveToRangeEnd" type="CT_Markup"/>
1023     </xsd:choice>
1024 </xsd:group>
1025 <xsd:complexType name="CT_NumPr">
1026     <xsd:sequence>
1027         <xsd:element name="ilvl" type="CT_DecimalNumber" minOccurs="0"/>
1028         <xsd:element name="numId" type="CT_DecimalNumber" minOccurs="0"/>
1029         <xsd:element name="numberingChange" type="CT_TrackChangeNumbering" minOccurs="0"/>
1030         <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
1031     </xsd:sequence>
1032 </xsd:complexType>
1033 <xsd:complexType name="CT_PBdr">
1034     <xsd:sequence>
1035         <xsd:element name="top" type="CT_Border" minOccurs="0"/>
1036         <xsd:element name="left" type="CT_Border" minOccurs="0"/>
1037         <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
1038         <xsd:element name="right" type="CT_Border" minOccurs="0"/>
1039         <xsd:element name="between" type="CT_Border" minOccurs="0"/>
1040         <xsd:element name="bar" type="CT_Border" minOccurs="0"/>
1041     </xsd:sequence>
1042 </xsd:complexType>
1043 <xsd:complexType name="CT_Tabs">

```

```

1044     <xsd:sequence>
1045         <xsd:element name="tab" type="CT_TabStop" minOccurs="1" maxOccurs="unbounded"/>
1046     </xsd:sequence>
1047 </xsd:complexType>
1048 <xsd:simpleType name="ST_TextboxTightWrap">
1049     <xsd:restriction base="xsd:string">
1050         <xsd:enumeration value="none"/>
1051         <xsd:enumeration value="allLines"/>
1052         <xsd:enumeration value="firstAndLastLine"/>
1053         <xsd:enumeration value="firstLineOnly"/>
1054         <xsd:enumeration value="lastLineOnly"/>
1055     </xsd:restriction>
1056 </xsd:simpleType>
1057 <xsd:complexType name="CT_TextboxTightWrap">
1058     <xsd:attribute name="val" type="ST_TextboxTightWrap" use="required"/>
1059 </xsd:complexType>
1060 <xsd:complexType name="CT_PPr">
1061     <xsd:complexContent>
1062         <xsd:extension base="CT_PPrBase">
1063             <xsd:sequence>
1064                 <xsd:element name="rPr" type="CT_ParaRPr" minOccurs="0"/>
1065                 <xsd:element name="sectPr" type="CT_SectPr" minOccurs="0"/>
1066                 <xsd:element name="pPrChange" type="CT_PPrChange" minOccurs="0"/>
1067             </xsd:sequence>
1068         </xsd:extension>
1069     </xsd:complexContent>
1070 </xsd:complexType>
1071 <xsd:complexType name="CT_PPrBase">
1072     <xsd:sequence>
1073         <xsd:element name="pStyle" type="CT_String" minOccurs="0"/>
1074         <xsd:element name="keepNext" type="CT_OnOff" minOccurs="0"/>
1075         <xsd:element name="keepLines" type="CT_OnOff" minOccurs="0"/>
1076         <xsd:element name="pageBreakBefore" type="CT_OnOff" minOccurs="0"/>
1077         <xsd:element name="framePr" type="CT_FramePr" minOccurs="0"/>
1078         <xsd:element name="widowControl" type="CT_OnOff" minOccurs="0"/>
1079         <xsd:element name="numPr" type="CT_NumPr" minOccurs="0"/>
1080         <xsd:element name="suppressLineNumbers" type="CT_OnOff" minOccurs="0"/>
1081         <xsd:element name="pBdr" type="CT_PBdr" minOccurs="0"/>
1082         <xsd:element name="shd" type="CT_Shdt" minOccurs="0"/>
1083         <xsd:element name="tabs" type="CT_Tabs" minOccurs="0"/>
1084         <xsd:element name="suppressAutoHyphens" type="CT_OnOff" minOccurs="0"/>
1085         <xsd:element name="kinsoku" type="CT_OnOff" minOccurs="0"/>
1086         <xsd:element name="wordWrap" type="CT_OnOff" minOccurs="0"/>
1087         <xsd:element name="overflowPunct" type="CT_OnOff" minOccurs="0"/>
1088         <xsd:element name="topLinePunct" type="CT_OnOff" minOccurs="0"/>
1089         <xsd:element name="autoSpaceDE" type="CT_OnOff" minOccurs="0"/>
1090         <xsd:element name="autoSpaceDN" type="CT_OnOff" minOccurs="0"/>
1091         <xsd:element name="bidi" type="CT_OnOff" minOccurs="0"/>
1092         <xsd:element name="adjustRightInd" type="CT_OnOff" minOccurs="0"/>
1093         <xsd:element name="snapToGrid" type="CT_OnOff" minOccurs="0"/>
1094         <xsd:element name="spacing" type="CT_Spacing" minOccurs="0"/>
1095         <xsd:element name="ind" type="CT_Ind" minOccurs="0"/>
1096         <xsd:element name="contextualSpacing" type="CT_OnOff" minOccurs="0"/>

```

```

1097     <xsd:element name="mirrorIndents" type="CT_OnOff" minOccurs="0"/>
1098     <xsd:element name="suppressOverlap" type="CT_OnOff" minOccurs="0"/>
1099     <xsd:element name="jc" type="CT_Jc" minOccurs="0"/>
1100     <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0"/>
1101     <xsd:element name="textAlignment" type="CT_TextAlignment" minOccurs="0"/>
1102     <xsd:element name="textboxTightWrap" type="CT_TextboxTightWrap" minOccurs="0"/>
1103     <xsd:element name="outlineLvl" type="CT_DecimalNumber" minOccurs="0"/>
1104     <xsd:element name="divId" type="CT_DecimalNumber" minOccurs="0"/>
1105     <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
1106   </xsd:sequence>
1107 </xsd:complexType>
1108 <xsd:complexType name="CT_PPrGeneral">
1109   <xsd:complexContent>
1110     <xsd:extension base="CT_PPrBase">
1111       <xsd:sequence>
1112         <xsd:element name="pPrChange" type="CT_PPrChange" minOccurs="0"/>
1113       </xsd:sequence>
1114     </xsd:extension>
1115   </xsd:complexContent>
1116 </xsd:complexType>
1117 <xsd:complexType name="CT_Control">
1118   <xsd:attribute name="name" type="s:ST_String" use="optional"/>
1119   <xsd:attribute name="shapeid" type="s:ST_String" use="optional"/>
1120   <xsd:attribute ref="r:id" use="optional"/>
1121 </xsd:complexType>
1122 <xsd:complexType name="CT_Background">
1123   <xsd:sequence>
1124     <xsd:sequence maxOccurs="unbounded">
1125       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1126         maxOccurs="unbounded"/>
1127       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1128         minOccurs="0" maxOccurs="unbounded"/>
1129     </xsd:sequence>
1130     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0"/>
1131   </xsd:sequence>
1132   <xsd:attribute name="color" type="ST_HexColor" use="optional"/>
1133   <xsd:attribute name="themeColor" type="ST_ThemeColor" use="optional"/>
1134   <xsd:attribute name="themeTint" type="ST_UcharHexNumber" use="optional"/>
1135   <xsd:attribute name="themeShade" type="ST_UcharHexNumber" use="optional"/>
1136 </xsd:complexType>
1137 <xsd:complexType name="CT_Rel">
1138   <xsd:attribute ref="r:id" use="required"/>
1139 </xsd:complexType>
1140 <xsd:complexType name="CT_Object">
1141   <xsd:sequence>
1142     <xsd:sequence maxOccurs="unbounded">
1143       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1144         maxOccurs="unbounded"/>
1145       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1146         minOccurs="0" maxOccurs="unbounded"/>
1147     </xsd:sequence>
1148     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0"/>
1149     <xsd:choice minOccurs="0">

```

```

1150     <xsd:element name="control" type="CT_Control"/>
1151     <xsd:element name="objectLink" type="CT_ObjectLink"/>
1152     <xsd:element name="objectEmbed" type="CT_ObjectEmbed"/>
1153     <xsd:element name="movie" type="CT_Rel"/>
1154   </xsd:choice>
1155 </xsd:sequence>
1156 <xsd:attribute name="dxaOrig" type="s:ST_TwipsMeasure" use="optional"/>
1157 <xsd:attribute name="dyaOrig" type="s:ST_TwipsMeasure" use="optional"/>
1158 </xsd:complexType>
1159 <xsd:complexType name="CT_Picture">
1160   <xsd:sequence>
1161     <xsd:sequence maxOccurs="unbounded">
1162       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:vml" minOccurs="0"
1163         maxOccurs="unbounded"/>
1164       <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
1165         minOccurs="0" maxOccurs="unbounded"/>
1166     </xsd:sequence>
1167     <xsd:element name="movie" type="CT_Rel" minOccurs="0"/>
1168     <xsd:element name="control" type="CT_Control" minOccurs="0"/>
1169   </xsd:sequence>
1170 </xsd:complexType>
1171 <xsd:complexType name="CT_ObjectEmbed">
1172   <xsd:attribute name="drawAspect" type="ST_ObjectDrawAspect" use="optional"/>
1173   <xsd:attribute ref="r:id" use="required"/>
1174   <xsd:attribute name="progId" type="s:ST_String" use="optional"/>
1175   <xsd:attribute name="shapeId" type="s:ST_String" use="optional"/>
1176   <xsd:attribute name="fieldCodes" type="s:ST_String" use="optional"/>
1177 </xsd:complexType>
1178 <xsd:simpleType name="ST_ObjectDrawAspect">
1179   <xsd:restriction base="xsd:string">
1180     <xsd:enumeration value="content"/>
1181     <xsd:enumeration value="icon"/>
1182   </xsd:restriction>
1183 </xsd:simpleType>
1184 <xsd:complexType name="CT_ObjectLink">
1185   <xsd:complexContent>
1186     <xsd:extension base="CT_ObjectEmbed">
1187       <xsd:attribute name="updateMode" type="ST_ObjectUpdateMode" use="required"/>
1188       <xsd:attribute name="lockedField" type="s:ST_OnOff" use="optional"/>
1189     </xsd:extension>
1190   </xsd:complexContent>
1191 </xsd:complexType>
1192 <xsd:simpleType name="ST_ObjectUpdateMode">
1193   <xsd:restriction base="xsd:string">
1194     <xsd:enumeration value="always"/>
1195     <xsd:enumeration value="onCall"/>
1196   </xsd:restriction>
1197 </xsd:simpleType>
1198 <xsd:complexType name="CT_Drawing">
1199   <xsd:choice minOccurs="1" maxOccurs="unbounded">
1200     <xsd:element ref="wp:anchor" minOccurs="0"/>
1201     <xsd:element ref="wp:inline" minOccurs="0"/>
1202   </xsd:choice>

```

```

1203 </xsd:complexType>
1204 <xsd:complexType name="CT_SimpleField">
1205   <xsd:sequence>
1206     <xsd:element name="fldData" type="CT_Text" minOccurs="0" maxOccurs="1"/>
1207     <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
1208   </xsd:sequence>
1209   <xsd:attribute name="instr" type="s:ST_String" use="required"/>
1210   <xsd:attribute name="fldLock" type="s:ST_OnOff"/>
1211   <xsd:attribute name="dirty" type="s:ST_OnOff"/>
1212 </xsd:complexType>
1213 <xsd:simpleType name="ST_FldCharType">
1214   <xsd:restriction base="xsd:string">
1215     <xsd:enumeration value="begin"/>
1216     <xsd:enumeration value="separate"/>
1217     <xsd:enumeration value="end"/>
1218   </xsd:restriction>
1219 </xsd:simpleType>
1220 <xsd:simpleType name="ST_InfoTextType">
1221   <xsd:restriction base="xsd:string">
1222     <xsd:enumeration value="text"/>
1223     <xsd:enumeration value="autoText"/>
1224   </xsd:restriction>
1225 </xsd:simpleType>
1226 <xsd:simpleType name="ST_FFHelpTextVal">
1227   <xsd:restriction base="xsd:string">
1228     <xsd:maxLength value="256"/>
1229   </xsd:restriction>
1230 </xsd:simpleType>
1231 <xsd:simpleType name="ST_FFStatusTextVal">
1232   <xsd:restriction base="xsd:string">
1233     <xsd:maxLength value="140"/>
1234   </xsd:restriction>
1235 </xsd:simpleType>
1236 <xsd:simpleType name="ST_FFName">
1237   <xsd:restriction base="xsd:string">
1238     <xsd:maxLength value="65"/>
1239   </xsd:restriction>
1240 </xsd:simpleType>
1241 <xsd:simpleType name="ST_FFTextType">
1242   <xsd:restriction base="xsd:string">
1243     <xsd:enumeration value="regular"/>
1244     <xsd:enumeration value="number"/>
1245     <xsd:enumeration value="date"/>
1246     <xsd:enumeration value="currentTime"/>
1247     <xsd:enumeration value="currentDate"/>
1248     <xsd:enumeration value="calculated"/>
1249   </xsd:restriction>
1250 </xsd:simpleType>
1251 <xsd:complexType name="CT_FFTextType">
1252   <xsd:attribute name="val" type="ST_FFTextType" use="required"/>
1253 </xsd:complexType>
1254 <xsd:complexType name="CT_FFName">
1255   <xsd:attribute name="val" type="ST_FFName"/>

```



```

1256 </xsd:complexType>
1257 <xsd:complexType name="CT_FldChar">
1258   <xsd:choice>
1259     <xsd:element name="fldData" type="CT_Text" minOccurs="0" maxOccurs="1"/>
1260     <xsd:element name="ffData" type="CT_FFData" minOccurs="0" maxOccurs="1"/>
1261     <xsd:element name="numberingChange" type="CT_TrackChangeNumbering" minOccurs="0"/>
1262   </xsd:choice>
1263   <xsd:attribute name="fldCharType" type="ST_FldCharType" use="required"/>
1264   <xsd:attribute name="fldLock" type="s:ST_OnOff"/>
1265   <xsd:attribute name="dirty" type="s:ST_OnOff"/>
1266 </xsd:complexType>
1267 <xsd:complexType name="CT_Hyperlink">
1268   <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
1269   <xsd:attribute name="tgtFrame" type="s:ST_String" use="optional"/>
1270   <xsd:attribute name="tooltip" type="s:ST_String" use="optional"/>
1271   <xsd:attribute name="docLocation" type="s:ST_String" use="optional"/>
1272   <xsd:attribute name="history" type="s:ST_OnOff" use="optional"/>
1273   <xsd:attribute name="anchor" type="s:ST_String" use="optional"/>
1274   <xsd:attribute ref="r:id"/>
1275 </xsd:complexType>
1276 <xsd:complexType name="CT_FFData">
1277   <xsd:choice maxOccurs="unbounded">
1278     <xsd:element name="name" type="CT_FFName"/>
1279     <xsd:element name="label" type="CT_DecimalNumber" minOccurs="0"/>
1280     <xsd:element name="tabIndex" type="CT_UnsignedDecimalNumber" minOccurs="0"/>
1281     <xsd:element name="enabled" type="CT_OnOff"/>
1282     <xsd:element name="calcOnExit" type="CT_OnOff"/>
1283     <xsd:element name="entryMacro" type="CT_MacroName" minOccurs="0" maxOccurs="1"/>
1284     <xsd:element name="exitMacro" type="CT_MacroName" minOccurs="0" maxOccurs="1"/>
1285     <xsd:element name="helpText" type="CT_FFHelpText" minOccurs="0" maxOccurs="1"/>
1286     <xsd:element name="statusText" type="CT_FFStatusText" minOccurs="0" maxOccurs="1"/>
1287   <xsd:choice>
1288     <xsd:element name="checkBox" type="CT_FFCheckBox"/>
1289     <xsd:element name="ddList" type="CT_FFDDLList"/>
1290     <xsd:element name="textInput" type="CT_FFTextInput"/>
1291   </xsd:choice>
1292 </xsd:choice>
1293 </xsd:complexType>
1294 <xsd:complexType name="CT_FFHelpText">
1295   <xsd:attribute name="type" type="ST_InfoTextType"/>
1296   <xsd:attribute name="val" type="ST_FFHelpTextVal"/>
1297 </xsd:complexType>
1298 <xsd:complexType name="CT_FFStatusText">
1299   <xsd:attribute name="type" type="ST_InfoTextType"/>
1300   <xsd:attribute name="val" type="ST_FFStatusTextVal"/>
1301 </xsd:complexType>
1302 <xsd:complexType name="CT_FFCheckBox">
1303   <xsd:sequence>
1304     <xsd:choice>
1305       <xsd:element name="size" type="CT_HpsMeasure"/>
1306       <xsd:element name="sizeAuto" type="CT_OnOff"/>
1307     </xsd:choice>
1308     <xsd:element name="default" type="CT_OnOff" minOccurs="0"/>

```

```

1309     <xsd:element name="checked" type="CT_OnOff" minOccurs="0"/>
1310   </xsd:sequence>
1311 </xsd:complexType>
1312 <xsd:complexType name="CT_FFDDList">
1313   <xsd:sequence>
1314     <xsd:element name="result" type="CT_DecimalNumber" minOccurs="0"/>
1315     <xsd:element name="default" type="CT_DecimalNumber" minOccurs="0"/>
1316     <xsd:element name="listEntry" type="CT_String" minOccurs="0" maxOccurs="unbounded"/>
1317   </xsd:sequence>
1318 </xsd:complexType>
1319 <xsd:complexType name="CT_FFTextInput">
1320   <xsd:sequence>
1321     <xsd:element name="type" type="CT_FFTextType" minOccurs="0"/>
1322     <xsd:element name="default" type="CT_String" minOccurs="0"/>
1323     <xsd:element name="maxLength" type="CT_DecimalNumber" minOccurs="0"/>
1324     <xsd:element name="format" type="CT_String" minOccurs="0"/>
1325   </xsd:sequence>
1326 </xsd:complexType>
1327 <xsd:simpleType name="ST_SectionMark">
1328   <xsd:restriction base="xsd:string">
1329     <xsd:enumeration value="nextPage"/>
1330     <xsd:enumeration value="nextColumn"/>
1331     <xsd:enumeration value="continuous"/>
1332     <xsd:enumeration value="evenPage"/>
1333     <xsd:enumeration value="oddPage"/>
1334   </xsd:restriction>
1335 </xsd:simpleType>
1336 <xsd:complexType name="CT_SectType">
1337   <xsd:attribute name="val" type="ST_SectionMark"/>
1338 </xsd:complexType>
1339 <xsd:complexType name="CT_PaperSource">
1340   <xsd:attribute name="first" type="ST_DecimalNumber"/>
1341   <xsd:attribute name="other" type="ST_DecimalNumber"/>
1342 </xsd:complexType>
1343 <xsd:simpleType name="ST_NumberFormat">
1344   <xsd:restriction base="xsd:string">
1345     <xsd:enumeration value="decimal"/>
1346     <xsd:enumeration value="upperRoman"/>
1347     <xsd:enumeration value="lowerRoman"/>
1348     <xsd:enumeration value="upperLetter"/>
1349     <xsd:enumeration value="lowerLetter"/>
1350     <xsd:enumeration value="ordinal"/>
1351     <xsd:enumeration value="cardinalText"/>
1352     <xsd:enumeration value="ordinalText"/>
1353     <xsd:enumeration value="hex"/>
1354     <xsd:enumeration value="chicago"/>
1355     <xsd:enumeration value="ideographDigital"/>
1356     <xsd:enumeration value="japaneseCounting"/>
1357     <xsd:enumeration value="aiueo"/>
1358     <xsd:enumeration value="iroha"/>
1359     <xsd:enumeration value="decimalFullWidth"/>
1360     <xsd:enumeration value="decimalHalfWidth"/>
1361     <xsd:enumeration value="japaneseLegal"/>

```

```

1362     <xsd:enumeration value="japaneseDigitalTenThousand"/>
1363     <xsd:enumeration value="decimalEnclosedCircle"/>
1364     <xsd:enumeration value="decimalFullWidth2"/>
1365     <xsd:enumeration value="aiueoFullWidth"/>
1366     <xsd:enumeration value="irohaFullWidth"/>
1367     <xsd:enumeration value="decimalZero"/>
1368     <xsd:enumeration value="bullet"/>
1369     <xsd:enumeration value="ganada"/>
1370     <xsd:enumeration value="chosung"/>
1371     <xsd:enumeration value="decimalEnclosedFullstop"/>
1372     <xsd:enumeration value="decimalEnclosedParen"/>
1373     <xsd:enumeration value="decimalEnclosedCircleChinese"/>
1374     <xsd:enumeration value="ideographEnclosedCircle"/>
1375     <xsd:enumeration value="ideographTraditional"/>
1376     <xsd:enumeration value="ideographZodiac"/>
1377     <xsd:enumeration value="ideographZodiacTraditional"/>
1378     <xsd:enumeration value="taiwaneseCounting"/>
1379     <xsd:enumeration value="ideographLegalTraditional"/>
1380     <xsd:enumeration value="taiwaneseCountingThousand"/>
1381     <xsd:enumeration value="taiwaneseDigital"/>
1382     <xsd:enumeration value="chineseCounting"/>
1383     <xsd:enumeration value="chineseLegalSimplified"/>
1384     <xsd:enumeration value="chineseCountingThousand"/>
1385     <xsd:enumeration value="koreanDigital"/>
1386     <xsd:enumeration value="koreanCounting"/>
1387     <xsd:enumeration value="koreanLegal"/>
1388     <xsd:enumeration value="koreanDigital2"/>
1389     <xsd:enumeration value="vietnameseCounting"/>
1390     <xsd:enumeration value="russianLower"/>
1391     <xsd:enumeration value="russianUpper"/>
1392     <xsd:enumeration value="none"/>
1393     <xsd:enumeration value="numberInDash"/>
1394     <xsd:enumeration value="hebrew1"/>
1395     <xsd:enumeration value="hebrew2"/>
1396     <xsd:enumeration value="arabicAlpha"/>
1397     <xsd:enumeration value="arabicAbjad"/>
1398     <xsd:enumeration value="hindiVowels"/>
1399     <xsd:enumeration value="hindiConsonants"/>
1400     <xsd:enumeration value="hindiNumbers"/>
1401     <xsd:enumeration value="hindiCounting"/>
1402     <xsd:enumeration value="thaiLetters"/>
1403     <xsd:enumeration value="thaiNumbers"/>
1404     <xsd:enumeration value="thaiCounting"/>
1405     <xsd:enumeration value="bahtText"/>
1406     <xsd:enumeration value="dollarText"/>
1407     <xsd:enumeration value="custom"/>
1408   </xsd:restriction>
1409 </xsd:simpleType>
1410 <xsd:simpleType name="ST_PageOrientation">
1411   <xsd:restriction base="xsd:string">
1412     <xsd:enumeration value="portrait"/>
1413     <xsd:enumeration value="landscape"/>
1414   </xsd:restriction>

```

```

1415 </xsd:simpleType>
1416 <xsd:complexType name="CT_PageSz">
1417   <xsd:attribute name="w" type="s:ST_TwipsMeasure"/>
1418   <xsd:attribute name="h" type="s:ST_TwipsMeasure"/>
1419   <xsd:attribute name="orient" type="ST_PageOrientation" use="optional"/>
1420   <xsd:attribute name="code" type="ST_DecimalNumber" use="optional"/>
1421 </xsd:complexType>
1422 <xsd:complexType name="CT_PageMar">
1423   <xsd:attribute name="top" type="ST_SignedTwipsMeasure" use="required"/>
1424   <xsd:attribute name="right" type="s:ST_TwipsMeasure" use="required"/>
1425   <xsd:attribute name="bottom" type="ST_SignedTwipsMeasure" use="required"/>
1426   <xsd:attribute name="left" type="s:ST_TwipsMeasure" use="required"/>
1427   <xsd:attribute name="header" type="s:ST_TwipsMeasure" use="required"/>
1428   <xsd:attribute name="footer" type="s:ST_TwipsMeasure" use="required"/>
1429   <xsd:attribute name="gutter" type="s:ST_TwipsMeasure" use="required"/>
1430 </xsd:complexType>
1431 <xsd:simpleType name="ST_PageBorderZOrder">
1432   <xsd:restriction base="xsd:string">
1433     <xsd:enumeration value="front"/>
1434     <xsd:enumeration value="back"/>
1435   </xsd:restriction>
1436 </xsd:simpleType>
1437 <xsd:simpleType name="ST_PageBorderDisplay">
1438   <xsd:restriction base="xsd:string">
1439     <xsd:enumeration value="allPages"/>
1440     <xsd:enumeration value="firstPage"/>
1441     <xsd:enumeration value="notFirstPage"/>
1442   </xsd:restriction>
1443 </xsd:simpleType>
1444 <xsd:simpleType name="ST_PageBorderOffset">
1445   <xsd:restriction base="xsd:string">
1446     <xsd:enumeration value="page"/>
1447     <xsd:enumeration value="text"/>
1448   </xsd:restriction>
1449 </xsd:simpleType>
1450 <xsd:complexType name="CT_PageBorders">
1451   <xsd:sequence>
1452     <xsd:element name="top" type="CT_TopPageBorder" minOccurs="0"/>
1453     <xsd:element name="left" type="CT_PageBorder" minOccurs="0"/>
1454     <xsd:element name="bottom" type="CT_BottomPageBorder" minOccurs="0"/>
1455     <xsd:element name="right" type="CT_PageBorder" minOccurs="0"/>
1456   </xsd:sequence>
1457   <xsd:attribute name="zOrder" type="ST_PageBorderZOrder" use="optional"/>
1458   <xsd:attribute name="display" type="ST_PageBorderDisplay" use="optional"/>
1459   <xsd:attribute name="offsetFrom" type="ST_PageBorderOffset" use="optional"/>
1460 </xsd:complexType>
1461 <xsd:complexType name="CT_PageBorder">
1462   <xsd:complexContent>
1463     <xsd:extension base="CT_Border">
1464       <xsd:attribute ref="r:id" use="optional"/>
1465     </xsd:extension>
1466   </xsd:complexContent>
1467 </xsd:complexType>

```

```

1468 <xsd:complexType name="CT_BottomPageBorder">
1469   <xsd:complexContent>
1470     <xsd:extension base="CT_PageBorder">
1471       <xsd:attribute ref="r:bottomLeft" use="optional"/>
1472       <xsd:attribute ref="r:bottomRight" use="optional"/>
1473     </xsd:extension>
1474   </xsd:complexContent>
1475 </xsd:complexType>
1476 <xsd:complexType name="CT_TopPageBorder">
1477   <xsd:complexContent>
1478     <xsd:extension base="CT_PageBorder">
1479       <xsd:attribute ref="r:topLeft" use="optional"/>
1480       <xsd:attribute ref="r:topRight" use="optional"/>
1481     </xsd:extension>
1482   </xsd:complexContent>
1483 </xsd:complexType>
1484 <xsd:simpleType name="ST_ChapterSep">
1485   <xsd:restriction base="xsd:string">
1486     <xsd:enumeration value="hyphen"/>
1487     <xsd:enumeration value="period"/>
1488     <xsd:enumeration value="colon"/>
1489     <xsd:enumeration value="emDash"/>
1490     <xsd:enumeration value="enDash"/>
1491   </xsd:restriction>
1492 </xsd:simpleType>
1493 <xsd:simpleType name="ST_LineNumberRestart">
1494   <xsd:restriction base="xsd:string">
1495     <xsd:enumeration value="newPage"/>
1496     <xsd:enumeration value="newSection"/>
1497     <xsd:enumeration value="continuous"/>
1498   </xsd:restriction>
1499 </xsd:simpleType>
1500 <xsd:complexType name="CT_LineNumber">
1501   <xsd:attribute name="countBy" type="ST_DecimalNumber" use="optional"/>
1502   <xsd:attribute name="start" type="ST_DecimalNumber" use="optional"/>
1503   <xsd:attribute name="distance" type="s:ST_TwipsMeasure" use="optional"/>
1504   <xsd:attribute name="restart" type="ST_LineNumberRestart" use="optional"/>
1505 </xsd:complexType>
1506 <xsd:complexType name="CT_PageNumber">
1507   <xsd:attribute name="fmt" type="ST_NumberFormat" use="optional"/>
1508   <xsd:attribute name="start" type="ST_DecimalNumber" use="optional"/>
1509   <xsd:attribute name="chapStyle" type="ST_DecimalNumber" use="optional"/>
1510   <xsd:attribute name="chapSep" type="ST_ChapterSep" use="optional"/>
1511 </xsd:complexType>
1512 <xsd:complexType name="CT_Column">
1513   <xsd:attribute name="w" type="s:ST_TwipsMeasure" use="optional"/>
1514   <xsd:attribute name="space" type="s:ST_TwipsMeasure" use="optional"/>
1515 </xsd:complexType>
1516 <xsd:complexType name="CT_Columns">
1517   <xsd:sequence minOccurs="0">
1518     <xsd:element name="col" type="CT_Column" maxOccurs="45"/>
1519   </xsd:sequence>
1520   <xsd:attribute name="equalWidth" type="s:ST_OnOff" use="optional"/>

```

```

1521     <xsd:attribute name="space" type="s:ST TwipsMeasure" use="optional"/>
1522     <xsd:attribute name="num" type="ST DecimalNumber" use="optional"/>
1523     <xsd:attribute name="sep" type="s:ST OnOff" use="optional"/>
1524 </xsd:complexType>
1525 <xsd:simpleType name="ST_VerticalJc">
1526     <xsd:restriction base="xsd:string">
1527         <xsd:enumeration value="top"/>
1528         <xsd:enumeration value="center"/>
1529         <xsd:enumeration value="both"/>
1530         <xsd:enumeration value="bottom"/>
1531     </xsd:restriction>
1532 </xsd:simpleType>
1533 <xsd:complexType name="CT_VerticalJc">
1534     <xsd:attribute name="val" type="ST VerticalJc" use="required"/>
1535 </xsd:complexType>
1536 <xsd:simpleType name="ST_DocGrid">
1537     <xsd:restriction base="xsd:string">
1538         <xsd:enumeration value="default"/>
1539         <xsd:enumeration value="lines"/>
1540         <xsd:enumeration value="linesAndChars"/>
1541         <xsd:enumeration value="snapTOCars"/>
1542     </xsd:restriction>
1543 </xsd:simpleType>
1544 <xsd:complexType name="CT_DocGrid">
1545     <xsd:attribute name="type" type="ST_DocGrid"/>
1546     <xsd:attribute name="linePitch" type="ST DecimalNumber"/>
1547     <xsd:attribute name="charSpace" type="ST DecimalNumber"/>
1548 </xsd:complexType>
1549 <xsd:simpleType name="ST_HdrFtr">
1550     <xsd:restriction base="xsd:string">
1551         <xsd:enumeration value="even"/>
1552         <xsd:enumeration value="default"/>
1553         <xsd:enumeration value="first"/>
1554     </xsd:restriction>
1555 </xsd:simpleType>
1556 <xsd:simpleType name="ST_FtnEdn">
1557     <xsd:restriction base="xsd:string">
1558         <xsd:enumeration value="normal"/>
1559         <xsd:enumeration value="separator"/>
1560         <xsd:enumeration value="continuationSeparator"/>
1561         <xsd:enumeration value="continuationNotice"/>
1562     </xsd:restriction>
1563 </xsd:simpleType>
1564 <xsd:complexType name="CT_HdrFtrRef">
1565     <xsd:complexContent>
1566         <xsd:extension base="CT_Rel">
1567             <xsd:attribute name="type" type="ST_HdrFtr" use="required"/>
1568         </xsd:extension>
1569     </xsd:complexContent>
1570 </xsd:complexType>
1571 <xsd:group name="EG_HdrFtrReferences">
1572     <xsd:choice>
1573         <xsd:element name="headerReference" type="CT_HdrFtrRef" minOccurs="0"/>

```

```

1574     <xsd:element name="footerReference" type="CT_HdrFtrRef" minOccurs="0"/>
1575   </xsd:choice>
1576 </xsd:group>
1577 <xsd:complexType name="CT_HdrFtr">
1578   <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
1579 </xsd:complexType>
1580 <xsd:group name="EG_SectPrContents">
1581   <xsd:sequence>
1582     <xsd:element name="footnotePr" type="CT_FtnProps" minOccurs="0"/>
1583     <xsd:element name="endnotePr" type="CT_EdnProps" minOccurs="0"/>
1584     <xsd:element name="type" type="CT_SectType" minOccurs="0"/>
1585     <xsd:element name="pgSz" type="CT_PageSz" minOccurs="0"/>
1586     <xsd:element name="pgMar" type="CT_PageMar" minOccurs="0"/>
1587     <xsd:element name="paperSrc" type="CT_PaperSource" minOccurs="0"/>
1588     <xsd:element name="pgBorders" type="CT_PageBorders" minOccurs="0"/>
1589     <xsd:element name="lnNumType" type="CT_LineNumber" minOccurs="0"/>
1590     <xsd:element name="pgNumType" type="CT_PageNumber" minOccurs="0"/>
1591     <xsd:element name="cols" type="CT_Columns" minOccurs="0"/>
1592     <xsd:element name="formProt" type="CT_OnOff" minOccurs="0"/>
1593     <xsd:element name="vAlign" type="CT_VerticalJc" minOccurs="0"/>
1594     <xsd:element name="noEndnote" type="CT_OnOff" minOccurs="0"/>
1595     <xsd:element name="titlePg" type="CT_OnOff" minOccurs="0"/>
1596     <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0"/>
1597     <xsd:element name="bidi" type="CT_OnOff" minOccurs="0"/>
1598     <xsd:element name="rtlGutter" type="CT_OnOff" minOccurs="0"/>
1599     <xsd:element name="docGrid" type="CT_DocGrid" minOccurs="0"/>
1600     <xsd:element name="printerSettings" type="CT_Rel" minOccurs="0"/>
1601   </xsd:sequence>
1602 </xsd:group>
1603 <xsd:attributeGroup name="AG_SectPrAttributes">
1604   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
1605   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
1606   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
1607   <xsd:attribute name="rsidSect" type="ST_LongHexNumber"/>
1608 </xsd:attributeGroup>
1609 <xsd:complexType name="CT_SectPrBase">
1610   <xsd:sequence>
1611     <xsd:group ref="EG_SectPrContents" minOccurs="0"/>
1612   </xsd:sequence>
1613   <xsd:attributeGroup ref="AG_SectPrAttributes"/>
1614 </xsd:complexType>
1615 <xsd:complexType name="CT_SectPr">
1616   <xsd:sequence>
1617     <xsd:group ref="EG_HdrFtrReferences" minOccurs="0" maxOccurs="6"/>
1618     <xsd:group ref="EG_SectPrContents" minOccurs="0"/>
1619     <xsd:element name="sectPrChange" type="CT_SectPrChange" minOccurs="0"/>
1620   </xsd:sequence>
1621   <xsd:attributeGroup ref="AG_SectPrAttributes"/>
1622 </xsd:complexType>
1623 <xsd:simpleType name="ST_BrType">
1624   <xsd:restriction base="xsd:string">
1625     <xsd:enumeration value="page"/>
1626     <xsd:enumeration value="column"/>

```

```

1627     <xsd:enumeration value="textWrapping"/>
1628   </xsd:restriction>
1629 </xsd:simpleType>
1630 <xsd:simpleType name="ST_BrClear">
1631   <xsd:restriction base="xsd:string">
1632     <xsd:enumeration value="none"/>
1633     <xsd:enumeration value="left"/>
1634     <xsd:enumeration value="right"/>
1635     <xsd:enumeration value="all"/>
1636   </xsd:restriction>
1637 </xsd:simpleType>
1638 <xsd:complexType name="CT_Br">
1639   <xsd:attribute name="type" type="ST_BrType" use="optional"/>
1640   <xsd:attribute name="clear" type="ST_BrClear" use="optional"/>
1641 </xsd:complexType>
1642 <xsd:simpleType name="ST_PTabAlignment">
1643   <xsd:restriction base="xsd:string">
1644     <xsd:enumeration value="left"/>
1645     <xsd:enumeration value="center"/>
1646     <xsd:enumeration value="right"/>
1647   </xsd:restriction>
1648 </xsd:simpleType>
1649 <xsd:simpleType name="ST_PTabRelativeTo">
1650   <xsd:restriction base="xsd:string">
1651     <xsd:enumeration value="margin"/>
1652     <xsd:enumeration value="indent"/>
1653   </xsd:restriction>
1654 </xsd:simpleType>
1655 <xsd:simpleType name="ST_PTabLeader">
1656   <xsd:restriction base="xsd:string">
1657     <xsd:enumeration value="none"/>
1658     <xsd:enumeration value="dot"/>
1659     <xsd:enumeration value="hyphen"/>
1660     <xsd:enumeration value="underscore"/>
1661     <xsd:enumeration value="middleDot"/>
1662   </xsd:restriction>
1663 </xsd:simpleType>
1664 <xsd:complexType name="CT_PTab">
1665   <xsd:attribute name="alignment" type="ST_PTabAlignment" use="required"/>
1666   <xsd:attribute name="relativeTo" type="ST_PTabRelativeTo" use="required"/>
1667   <xsd:attribute name="leader" type="ST_PTabLeader" use="required"/>
1668 </xsd:complexType>
1669 <xsd:complexType name="CT_Sym">
1670   <xsd:attribute name="font" type="s:ST_String"/>
1671   <xsd:attribute name="char" type="ST_ShortHexNumber"/>
1672 </xsd:complexType>
1673 <xsd:simpleType name="ST_ProofErr">
1674   <xsd:restriction base="xsd:string">
1675     <xsd:enumeration value="spellStart"/>
1676     <xsd:enumeration value="spellEnd"/>
1677     <xsd:enumeration value="gramStart"/>
1678     <xsd:enumeration value="gramEnd"/>
1679   </xsd:restriction>

```



```

1680 </xsd:simpleType>
1681 <xsd:complexType name="CT_ProofErr">
1682   <xsd:attribute name="type" type="ST_ProofErr" use="required"/>
1683 </xsd:complexType>
1684 <xsd:simpleType name="ST_EdGrp">
1685   <xsd:restriction base="xsd:string">
1686     <xsd:enumeration value="none"/>
1687     <xsd:enumeration value="everyone"/>
1688     <xsd:enumeration value="administrators"/>
1689     <xsd:enumeration value="contributors"/>
1690     <xsd:enumeration value="editors"/>
1691     <xsd:enumeration value="owners"/>
1692     <xsd:enumeration value="current"/>
1693   </xsd:restriction>
1694 </xsd:simpleType>
1695 <xsd:complexType name="CT_Perm">
1696   <xsd:attribute name="id" type="s:ST_String" use="required"/>
1697   <xsd:attribute name="displacedByCustomXml" type="ST_DisplacedByCustomXml" use="optional"/>
1698 </xsd:complexType>
1699 <xsd:complexType name="CT_PermStart">
1700   <xsd:complexContent>
1701     <xsd:extension base="CT_Perm">
1702       <xsd:attribute name="edGrp" type="ST_EdGrp" use="optional"/>
1703       <xsd:attribute name="ed" type="s:ST_String" use="optional"/>
1704       <xsd:attribute name="colFirst" type="ST_DecimalNumber" use="optional"/>
1705       <xsd:attribute name="colLast" type="ST_DecimalNumber" use="optional"/>
1706     </xsd:extension>
1707   </xsd:complexContent>
1708 </xsd:complexType>
1709 <xsd:complexType name="CT_Text">
1710   <xsd:simpleContent>
1711     <xsd:extension base="s:ST_String">
1712       <xsd:attribute ref="xml:space" use="optional"/>
1713     </xsd:extension>
1714   </xsd:simpleContent>
1715 </xsd:complexType>
1716 <xsd:group name="EG_RunInnerContent">
1717   <xsd:choice>
1718     <xsd:element name="br" type="CT_Br"/>
1719     <xsd:element name="t" type="CT_Text"/>
1720     <xsd:element name="contentPart" type="CT_Rel"/>
1721     <xsd:element name="delText" type="CT_Text"/>
1722     <xsd:element name="instrText" type="CT_Text"/>
1723     <xsd:element name="delInstrText" type="CT_Text"/>
1724     <xsd:element name="noBreakHyphen" type="CT_Empty"/>
1725     <xsd:element name="softHyphen" type="CT_Empty" minOccurs="0"/>
1726     <xsd:element name="dayShort" type="CT_Empty" minOccurs="0"/>
1727     <xsd:element name="monthShort" type="CT_Empty" minOccurs="0"/>
1728     <xsd:element name="yearShort" type="CT_Empty" minOccurs="0"/>
1729     <xsd:element name="dayLong" type="CT_Empty" minOccurs="0"/>
1730     <xsd:element name="monthLong" type="CT_Empty" minOccurs="0"/>
1731     <xsd:element name="yearLong" type="CT_Empty" minOccurs="0"/>
1732     <xsd:element name="annotationRef" type="CT_Empty" minOccurs="0"/>

```

```

1733     <xsd:element name="footnoteRef" type="CT_Empty" minOccurs="0"/>
1734     <xsd:element name="endnoteRef" type="CT_Empty" minOccurs="0"/>
1735     <xsd:element name="separator" type="CT_Empty" minOccurs="0"/>
1736     <xsd:element name="continuationSeparator" type="CT_Empty" minOccurs="0"/>
1737     <xsd:element name="sym" type="CT_Sym" minOccurs="0"/>
1738     <xsd:element name="pgNum" type="CT_Empty" minOccurs="0"/>
1739     <xsd:element name="cr" type="CT_Empty" minOccurs="0"/>
1740     <xsd:element name="tab" type="CT_Empty" minOccurs="0"/>
1741     <xsd:element name="object" type="CT_Object"/>
1742     <xsd:element name="pict" type="CT_Picture"/>
1743     <xsd:element name="fldChar" type="CT_FldChar"/>
1744     <xsd:element name="ruby" type="CT_Ruby"/>
1745     <xsd:element name="footnoteReference" type="CT_FtnEdnRef"/>
1746     <xsd:element name="endnoteReference" type="CT_FtnEdnRef"/>
1747     <xsd:element name="commentReference" type="CT_Markup"/>
1748     <xsd:element name="drawing" type="CT_Drawing"/>
1749     <xsd:element name="ptab" type="CT_PTab" minOccurs="0"/>
1750     <xsd:element name="lastRenderedPageBreak" type="CT_Empty" minOccurs="0" maxOccurs="1"/>
1751   </xsd:choice>
1752 </xsd:group>
1753 <xsd:complexType name="CT_R">
1754   <xsd:sequence>
1755     <xsd:group ref="EG_RPr" minOccurs="0"/>
1756     <xsd:group ref="EG_RunInnerContent" minOccurs="0" maxOccurs="unbounded"/>
1757   </xsd:sequence>
1758   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
1759   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
1760   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
1761 </xsd:complexType>
1762 <xsd:simpleType name="ST_Hint">
1763   <xsd:restriction base="xsd:string">
1764     <xsd:enumeration value="default"/>
1765     <xsd:enumeration value="eastAsia"/>
1766     <xsd:enumeration value="cs"/>
1767   </xsd:restriction>
1768 </xsd:simpleType>
1769 <xsd:simpleType name="ST_Theme">
1770   <xsd:restriction base="xsd:string">
1771     <xsd:enumeration value="majorEastAsia"/>
1772     <xsd:enumeration value="majorBidi"/>
1773     <xsd:enumeration value="majorAscii"/>
1774     <xsd:enumeration value="majorHAnsi"/>
1775     <xsd:enumeration value="minorEastAsia"/>
1776     <xsd:enumeration value="minorBidi"/>
1777     <xsd:enumeration value="minorAscii"/>
1778     <xsd:enumeration value="minorHAnsi"/>
1779   </xsd:restriction>
1780 </xsd:simpleType>
1781 <xsd:complexType name="CT_Fonts">
1782   <xsd:attribute name="hint" type="ST_Hint"/>
1783   <xsd:attribute name="ascii" type="s:ST_String"/>
1784   <xsd:attribute name="hAnsi" type="s:ST_String"/>
1785   <xsd:attribute name="eastAsia" type="s:ST_String"/>

```

```

1786 <xsd:attribute name="cs" type="s:ST_String"/>
1787 <xsd:attribute name="asciiTheme" type="ST_Theme"/>
1788 <xsd:attribute name="hAnsiTheme" type="ST_Theme"/>
1789 <xsd:attribute name="eastAsiaTheme" type="ST_Theme"/>
1790 <xsd:attribute name="cstheme" type="ST_Theme"/>
1791 </xsd:complexType>
1792 <xsd:group name="EG_RPrBase">
1793   <xsd:sequence>
1794     <xsd:element name="rStyle" type="CT_String" minOccurs="0"/>
1795     <xsd:element name="rFonts" type="CT_Fonts" minOccurs="0"/>
1796     <xsd:element name="b" type="CT_OnOff" minOccurs="0"/>
1797     <xsd:element name="bCs" type="CT_OnOff" minOccurs="0"/>
1798     <xsd:element name="i" type="CT_OnOff" minOccurs="0"/>
1799     <xsd:element name="iCs" type="CT_OnOff" minOccurs="0"/>
1800     <xsd:element name="caps" type="CT_OnOff" minOccurs="0"/>
1801     <xsd:element name="smallCaps" type="CT_OnOff" minOccurs="0"/>
1802     <xsd:element name="strike" type="CT_OnOff" minOccurs="0"/>
1803     <xsd:element name="dstrike" type="CT_OnOff" minOccurs="0"/>
1804     <xsd:element name="outline" type="CT_OnOff" minOccurs="0"/>
1805     <xsd:element name="shadow" type="CT_OnOff" minOccurs="0"/>
1806     <xsd:element name="emboss" type="CT_OnOff" minOccurs="0"/>
1807     <xsd:element name="imprint" type="CT_OnOff" minOccurs="0"/>
1808     <xsd:element name="noProof" type="CT_OnOff" minOccurs="0"/>
1809     <xsd:element name="snapToGrid" type="CT_OnOff" minOccurs="0"/>
1810     <xsd:element name="vanish" type="CT_OnOff" minOccurs="0"/>
1811     <xsd:element name="webHidden" type="CT_OnOff" minOccurs="0"/>
1812     <xsd:element name="color" type="CT_Color" minOccurs="0"/>
1813     <xsd:element name="spacing" type="CT_SignedTwipsMeasure" minOccurs="0"/>
1814     <xsd:element name="w" type="CT_TextScale" minOccurs="0"/>
1815     <xsd:element name="kern" type="CT_HpsMeasure" minOccurs="0"/>
1816     <xsd:element name="position" type="CT_SignedHpsMeasure" minOccurs="0"/>
1817     <xsd:element name="sz" type="CT_HpsMeasure" minOccurs="0"/>
1818     <xsd:element name="szCs" type="CT_HpsMeasure" minOccurs="0"/>
1819     <xsd:element name="highlight" type="CT_Highlight" minOccurs="0"/>
1820     <xsd:element name="u" type="CT_Underline" minOccurs="0"/>
1821     <xsd:element name="effect" type="CT_TextEffect" minOccurs="0"/>
1822     <xsd:element name="bdr" type="CT_Border" minOccurs="0"/>
1823     <xsd:element name="shd" type="CT_Shd" minOccurs="0"/>
1824     <xsd:element name="fitText" type="CT_FitText" minOccurs="0"/>
1825     <xsd:element name="vertAlign" type="CT_VerticalAlignRun" minOccurs="0"/>
1826     <xsd:element name="rtl" type="CT_OnOff" minOccurs="0"/>
1827     <xsd:element name="cs" type="CT_OnOff" minOccurs="0"/>
1828     <xsd:element name="em" type="CT_Em" minOccurs="0"/>
1829     <xsd:element name="lang" type="CT_Language" minOccurs="0"/>
1830     <xsd:element name="eastAsianLayout" type="CT_EastAsianLayout" minOccurs="0"/>
1831     <xsd:element name="specVanish" type="CT_OnOff" minOccurs="0"/>
1832     <xsd:element name="oMath" type="CT_OnOff" minOccurs="0"/>
1833   </xsd:sequence>
1834 </xsd:group>
1835 <xsd:group name="EG_RPrContent">
1836   <xsd:sequence>
1837     <xsd:group ref="EG_RPrBase" minOccurs="0"/>
1838     <xsd:element name="rPrChange" type="CT_RPrChange" minOccurs="0"/>

```

```

1839     </xsd:sequence>
1840 </xsd:group>
1841 <xsd:complexType name="CT_RPr">
1842     <xsd:sequence>
1843         <xsd:group ref="EG_RPrContent" minOccurs="0"/>
1844     </xsd:sequence>
1845 </xsd:complexType>
1846 <xsd:group name="EG_RPr">
1847     <xsd:sequence>
1848         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
1849     </xsd:sequence>
1850 </xsd:group>
1851 <xsd:group name="EG_RPrMath">
1852     <xsd:choice>
1853         <xsd:group ref="EG_RPr"/>
1854         <xsd:element name="ins" type="CT_RPrChange"/>
1855         <xsd:element name="del" type="CT_RPrChange"/>
1856     </xsd:choice>
1857 </xsd:group>
1858 <xsd:complexType name="CT_RPrOriginal">
1859     <xsd:sequence>
1860         <xsd:group ref="EG_RPrBase" minOccurs="0" maxOccurs="unbounded"/>
1861     </xsd:sequence>
1862 </xsd:complexType>
1863 <xsd:complexType name="CT_ParaRPrOriginal">
1864     <xsd:sequence>
1865         <xsd:group ref="EG_ParaRPrTrackChanges" minOccurs="0"/>
1866         <xsd:group ref="EG_RPrBase" minOccurs="0" maxOccurs="unbounded"/>
1867     </xsd:sequence>
1868 </xsd:complexType>
1869 <xsd:complexType name="CT_ParaRPr">
1870     <xsd:sequence>
1871         <xsd:group ref="EG_ParaRPrTrackChanges" minOccurs="0"/>
1872         <xsd:group ref="EG_RPrBase" minOccurs="0"/>
1873         <xsd:element name="rPrChange" type="CT_ParaRPrChange" minOccurs="0"/>
1874     </xsd:sequence>
1875 </xsd:complexType>
1876 <xsd:group name="EG_ParaRPrTrackChanges">
1877     <xsd:sequence>
1878         <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
1879         <xsd:element name="del" type="CT_TrackChange" minOccurs="0"/>
1880         <xsd:element name="moveFrom" type="CT_TrackChange" minOccurs="0"/>
1881         <xsd:element name="moveTo" type="CT_TrackChange" minOccurs="0"/>
1882     </xsd:sequence>
1883 </xsd:group>
1884 <xsd:complexType name="CT_AltChunk">
1885     <xsd:sequence>
1886         <xsd:element name="altChunkPr" type="CT_AltChunkPr" minOccurs="0" maxOccurs="1"/>
1887     </xsd:sequence>
1888     <xsd:attribute ref="r:id" use="optional"/>
1889 </xsd:complexType>
1890 <xsd:complexType name="CT_AltChunkPr">
1891     <xsd:sequence>

```

```

1892     <xsd:element name="matchSrc" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
1893   </xsd:sequence>
1894 </xsd:complexType>
1895 <xsd:simpleType name="ST_RubyAlign">
1896   <xsd:restriction base="xsd:string">
1897     <xsd:enumeration value="center"/>
1898     <xsd:enumeration value="distributeLetter"/>
1899     <xsd:enumeration value="distributeSpace"/>
1900     <xsd:enumeration value="left"/>
1901     <xsd:enumeration value="right"/>
1902     <xsd:enumeration value="rightVertical"/>
1903   </xsd:restriction>
1904 </xsd:simpleType>
1905 <xsd:complexType name="CT_RubyAlign">
1906   <xsd:attribute name="val" type="ST_RubyAlign" use="required"/>
1907 </xsd:complexType>
1908 <xsd:complexType name="CT_RubyPr">
1909   <xsd:sequence>
1910     <xsd:element name="rubyAlign" type="CT_RubyAlign"/>
1911     <xsd:element name="hps" type="CT_HpsMeasure"/>
1912     <xsd:element name="hpsRaise" type="CT_HpsMeasure"/>
1913     <xsd:element name="hpsBaseText" type="CT_HpsMeasure"/>
1914     <xsd:element name="lid" type="CT_Lang"/>
1915     <xsd:element name="dirty" type="CT_OnOff" minOccurs="0"/>
1916   </xsd:sequence>
1917 </xsd:complexType>
1918 <xsd:group name="EG_RubyContent">
1919   <xsd:choice>
1920     <xsd:element name="r" type="CT_R"/>
1921     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
1922   </xsd:choice>
1923 </xsd:group>
1924 <xsd:complexType name="CT_RubyContent">
1925   <xsd:group ref="EG_RubyContent" minOccurs="0" maxOccurs="unbounded"/>
1926 </xsd:complexType>
1927 <xsd:complexType name="CT_Ruby">
1928   <xsd:sequence>
1929     <xsd:element name="rubyPr" type="CT_RubyPr"/>
1930     <xsd:element name="rt" type="CT_RubyContent"/>
1931     <xsd:element name="rubyBase" type="CT_RubyContent"/>
1932   </xsd:sequence>
1933 </xsd:complexType>
1934 <xsd:simpleType name="ST_Lock">
1935   <xsd:restriction base="xsd:string">
1936     <xsd:enumeration value="sdtLocked"/>
1937     <xsd:enumeration value="contentLocked"/>
1938     <xsd:enumeration value="unlocked"/>
1939     <xsd:enumeration value="sdtContentLocked"/>
1940   </xsd:restriction>
1941 </xsd:simpleType>
1942 <xsd:complexType name="CT_Lock">
1943   <xsd:attribute name="val" type="ST_Lock"/>
1944 </xsd:complexType>

```

```

1945 <xsd:complexType name="CT_SdtListItem">
1946   <xsd:attribute name="displayText" type="s:ST String"/>
1947   <xsd:attribute name="value" type="s:ST String"/>
1948 </xsd:complexType>
1949 <xsd:simpleType name="ST_SdtDateMappingType">
1950   <xsd:restriction base="xsd:string">
1951     <xsd:enumeration value="text"/>
1952     <xsd:enumeration value="date"/>
1953     <xsd:enumeration value="dateTime"/>
1954   </xsd:restriction>
1955 </xsd:simpleType>
1956 <xsd:complexType name="CT_SdtDateMappingType">
1957   <xsd:attribute name="val" type="ST_SdtDateMappingType"/>
1958 </xsd:complexType>
1959 <xsd:complexType name="CT_CalendarType">
1960   <xsd:attribute name="val" type="s:ST_CalendarType"/>
1961 </xsd:complexType>
1962 <xsd:complexType name="CT_SdtDate">
1963   <xsd:sequence>
1964     <xsd:element name="dateFormat" type="CT String" minOccurs="0"/>
1965     <xsd:element name="lid" type="CT Lang" minOccurs="0"/>
1966     <xsd:element name="storeMappedDataAs" type="CT_SdtDateMappingType" minOccurs="0"/>
1967     <xsd:element name="calendar" type="CT_CalendarType" minOccurs="0"/>
1968   </xsd:sequence>
1969   <xsd:attribute name="fullDate" type="ST DateTime" use="optional"/>
1970 </xsd:complexType>
1971 <xsd:complexType name="CT_SdtComboBox">
1972   <xsd:sequence>
1973     <xsd:element name="listItem" type="CT_SdtListItem" minOccurs="0" maxOccurs="unbounded"/>
1974   </xsd:sequence>
1975   <xsd:attribute name="lastValue" type="s:ST String" use="optional"/>
1976 </xsd:complexType>
1977 <xsd:complexType name="CT_SdtDocPart">
1978   <xsd:sequence>
1979     <xsd:element name="docPartGallery" type="CT String" minOccurs="0"/>
1980     <xsd:element name="docPartCategory" type="CT String" minOccurs="0"/>
1981     <xsd:element name="docPartUnique" type="CT OnOff" minOccurs="0"/>
1982   </xsd:sequence>
1983 </xsd:complexType>
1984 <xsd:complexType name="CT_SdtDropDownList">
1985   <xsd:sequence>
1986     <xsd:element name="listItem" type="CT_SdtListItem" minOccurs="0" maxOccurs="unbounded"/>
1987   </xsd:sequence>
1988   <xsd:attribute name="lastValue" type="s:ST String" use="optional"/>
1989 </xsd:complexType>
1990 <xsd:complexType name="CT_Placeholder">
1991   <xsd:sequence>
1992     <xsd:element name="docPart" type="CT String"/>
1993   </xsd:sequence>
1994 </xsd:complexType>
1995 <xsd:complexType name="CT_SdtText">
1996   <xsd:attribute name="multiline" type="s:ST OnOff"/>
1997 </xsd:complexType>

```

```

1998 <xsd:complexType name="CT_DataBinding">
1999   <xsd:attribute name="prefixMappings" type="s:ST String"/>
2000   <xsd:attribute name="xpath" type="s:ST String" use="required"/>
2001   <xsd:attribute name="storeItemID" type="s:ST String" use="required"/>
2002 </xsd:complexType>
2003 <xsd:complexType name="CT_SdtPr">
2004   <xsd:choice maxOccurs="unbounded">
2005     <xsd:element name="rPr" type="CT RPr" minOccurs="0"/>
2006     <xsd:element name="alias" type="CT String" minOccurs="0"/>
2007     <xsd:element name="label" type="CT DecimalNumber" minOccurs="0"/>
2008     <xsd:element name="tabIndex" type="CT UnsignedDecimalNumber" minOccurs="0"/>
2009     <xsd:element name="lock" type="CT Lock" minOccurs="0"/>
2010     <xsd:element name="placeholder" type="CT Placeholder" minOccurs="0"/>
2011     <xsd:element name="showingPlcHdr" type="CT OnOff" minOccurs="0"/>
2012     <xsd:element name="dataBinding" type="CT DataBinding" minOccurs="0"/>
2013     <xsd:element name="temporary" type="CT OnOff" minOccurs="0"/>
2014     <xsd:element name="id" type="CT DecimalNumber" minOccurs="0"/>
2015     <xsd:element name="tag" type="CT String" minOccurs="0"/>
2016     <xsd:choice minOccurs="0" maxOccurs="1">
2017       <xsd:element name="equation" type="CT Empty"/>
2018       <xsd:element name="comboBox" type="CT SdtComboBox"/>
2019       <xsd:element name="date" type="CT SdtDate"/>
2020       <xsd:element name="docPartObj" type="CT SdtDocPart"/>
2021       <xsd:element name="docPartList" type="CT SdtDocPart"/>
2022       <xsd:element name="dropDownList" type="CT SdtDropDownList"/>
2023       <xsd:element name="picture" type="CT Empty"/>
2024       <xsd:element name="richText" type="CT Empty"/>
2025       <xsd:element name="text" type="CT SdtText"/>
2026       <xsd:element name="citation" type="CT Empty"/>
2027       <xsd:element name="group" type="CT Empty"/>
2028       <xsd:element name="bibliography" type="CT Empty"/>
2029     </xsd:choice>
2030   </xsd:choice>
2031 </xsd:complexType>
2032 <xsd:complexType name="CT_SdtEndPr">
2033   <xsd:choice maxOccurs="unbounded">
2034     <xsd:element name="rPr" type="CT RPr" minOccurs="0"/>
2035   </xsd:choice>
2036 </xsd:complexType>
2037 <xsd:group name="EG_ContentRunContent">
2038   <xsd:choice>
2039     <xsd:element name="customXml" type="CT CustomXmlRun"/>
2040     <xsd:element name="smartTag" type="CT SmartTagRun"/>
2041     <xsd:element name="sdt" type="CT SdtRun"/>
2042     <xsd:element name="dir" type="CT DirContentRun"/>
2043     <xsd:element name="bdo" type="CT BdoContentRun"/>
2044     <xsd:element name="r" type="CT R"/>
2045     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2046   </xsd:choice>
2047 </xsd:group>
2048 <xsd:complexType name="CT_DirContentRun">
2049   <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2050   <xsd:attribute name="val" type="ST Direction" use="optional"/>

```

```

2051 </xsd:complexType>
2052 <xsd:complexType name="CT_BdoContentRun">
2053   <xsd:group ref="EG PContent" minOccurs="0" maxOccurs="unbounded"/>
2054   <xsd:attribute name="val" type="ST_Direction" use="optional"/>
2055 </xsd:complexType>
2056 <xsd:simpleType name="ST_Direction">
2057   <xsd:restriction base="xsd:string">
2058     <xsd:enumeration value="ltr"/>
2059     <xsd:enumeration value="rtl"/>
2060   </xsd:restriction>
2061 </xsd:simpleType>
2062 <xsd:complexType name="CT_SdtContentRun">
2063   <xsd:group ref="EG PContent" minOccurs="0" maxOccurs="unbounded"/>
2064 </xsd:complexType>
2065 <xsd:group name="EG_ContentBlockContent">
2066   <xsd:choice>
2067     <xsd:element name="customXml" type="CT_CustomXmlBlock"/>
2068     <xsd:element name="sdt" type="CT_SdtBlock"/>
2069     <xsd:element name="p" type="CT_P" minOccurs="0" maxOccurs="unbounded"/>
2070     <xsd:element name="tbl" type="CT_Tbl" minOccurs="0" maxOccurs="unbounded"/>
2071     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2072   </xsd:choice>
2073 </xsd:group>
2074 <xsd:complexType name="CT_SdtContentBlock">
2075   <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
2076 </xsd:complexType>
2077 <xsd:group name="EG_ContentRowContent">
2078   <xsd:choice>
2079     <xsd:element name="tr" type="CT_Row" minOccurs="0" maxOccurs="unbounded"/>
2080     <xsd:element name="customXml" type="CT_CustomXmlRow"/>
2081     <xsd:element name="sdt" type="CT_SdtRow"/>
2082     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2083   </xsd:choice>
2084 </xsd:group>
2085 <xsd:complexType name="CT_SdtContentRow">
2086   <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2087 </xsd:complexType>
2088 <xsd:group name="EG_ContentCellContent">
2089   <xsd:choice>
2090     <xsd:element name="tc" type="CT_Tc" minOccurs="0" maxOccurs="unbounded"/>
2091     <xsd:element name="customXml" type="CT_CustomXmlCell"/>
2092     <xsd:element name="sdt" type="CT_SdtCell"/>
2093     <xsd:group ref="EG_RunLevelElts" minOccurs="0" maxOccurs="unbounded"/>
2094   </xsd:choice>
2095 </xsd:group>
2096 <xsd:complexType name="CT_SdtContentCell">
2097   <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2098 </xsd:complexType>
2099 <xsd:complexType name="CT_SdtBlock">
2100   <xsd:sequence>
2101     <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2102     <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2103     <xsd:element name="sdtContent" type="CT_SdtContentBlock" minOccurs="0" maxOccurs="1"/>

```



```

2104     </xsd:sequence>
2105 </xsd:complexType>
2106 <xsd:complexType name="CT_SdtRun">
2107     <xsd:sequence>
2108         <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2109         <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2110         <xsd:element name="sdtContent" type="CT_SdtContentRun" minOccurs="0" maxOccurs="1"/>
2111     </xsd:sequence>
2112 </xsd:complexType>
2113 <xsd:complexType name="CT_SdtCell">
2114     <xsd:sequence>
2115         <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2116         <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2117         <xsd:element name="sdtContent" type="CT_SdtContentCell" minOccurs="0" maxOccurs="1"/>
2118     </xsd:sequence>
2119 </xsd:complexType>
2120 <xsd:complexType name="CT_SdtRow">
2121     <xsd:sequence>
2122         <xsd:element name="sdtPr" type="CT_SdtPr" minOccurs="0" maxOccurs="1"/>
2123         <xsd:element name="sdtEndPr" type="CT_SdtEndPr" minOccurs="0" maxOccurs="1"/>
2124         <xsd:element name="sdtContent" type="CT_SdtContentRow" minOccurs="0" maxOccurs="1"/>
2125     </xsd:sequence>
2126 </xsd:complexType>
2127 <xsd:complexType name="CT_Attr">
2128     <xsd:attribute name="uri" type="s:ST String"/>
2129     <xsd:attribute name="name" type="s:ST String" use="required"/>
2130     <xsd:attribute name="val" type="s:ST String" use="required"/>
2131 </xsd:complexType>
2132 <xsd:complexType name="CT_CustomXmlRun">
2133     <xsd:sequence>
2134         <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2135         <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2136     </xsd:sequence>
2137     <xsd:attribute name="uri" type="s:ST String"/>
2138     <xsd:attribute name="element" type="s:ST String" use="required"/>
2139 </xsd:complexType>
2140 <xsd:complexType name="CT_SmartTagRun">
2141     <xsd:sequence>
2142         <xsd:element name="smartTagPr" type="CT_SmartTagPr" minOccurs="0" maxOccurs="1"/>
2143         <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2144     </xsd:sequence>
2145     <xsd:attribute name="uri" type="s:ST String"/>
2146     <xsd:attribute name="element" type="s:ST String" use="required"/>
2147 </xsd:complexType>
2148 <xsd:complexType name="CT_CustomXmlBlock">
2149     <xsd:sequence>
2150         <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2151         <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
2152     </xsd:sequence>
2153     <xsd:attribute name="uri" type="s:ST String"/>
2154     <xsd:attribute name="element" type="s:ST String" use="required"/>
2155 </xsd:complexType>
2156 <xsd:complexType name="CT_CustomXmlPr">

```

```

2157     <xsd:sequence>
2158       <xsd:element name="placeholder" type="CT_String" minOccurs="0"/>
2159       <xsd:element name="attr" type="CT_Attr" minOccurs="0" maxOccurs="unbounded"/>
2160     </xsd:sequence>
2161   </xsd:complexType>
2162   <xsd:complexType name="CT_CustomXmlRow">
2163     <xsd:sequence>
2164       <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2165       <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2166     </xsd:sequence>
2167     <xsd:attribute name="uri" type="s:ST_String"/>
2168     <xsd:attribute name="element" type="s:ST_String" use="required"/>
2169   </xsd:complexType>
2170   <xsd:complexType name="CT_CustomXmlCell">
2171     <xsd:sequence>
2172       <xsd:element name="customXmlPr" type="CT_CustomXmlPr" minOccurs="0" maxOccurs="1"/>
2173       <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2174     </xsd:sequence>
2175     <xsd:attribute name="uri" type="s:ST_String"/>
2176     <xsd:attribute name="element" type="s:ST_String" use="required"/>
2177   </xsd:complexType>
2178   <xsd:complexType name="CT_SmartTagPr">
2179     <xsd:sequence>
2180       <xsd:element name="attr" type="CT_Attr" minOccurs="0" maxOccurs="unbounded"/>
2181     </xsd:sequence>
2182   </xsd:complexType>
2183   <xsd:group name="EG_PContent">
2184     <xsd:choice>
2185       <xsd:group ref="EG_ContentRunContent" minOccurs="0" maxOccurs="unbounded"/>
2186       <xsd:element name="fldSimple" type="CT_SimpleField" minOccurs="0" maxOccurs="unbounded"/>
2187       <xsd:element name="hyperlink" type="CT_Hyperlink"/>
2188       <xsd:element name="subDoc" type="CT_Rel"/>
2189     </xsd:choice>
2190   </xsd:group>
2191   <xsd:complexType name="CT_P">
2192     <xsd:sequence>
2193       <xsd:element name="pPr" type="CT_PPr" minOccurs="0"/>
2194       <xsd:group ref="EG_PContent" minOccurs="0" maxOccurs="unbounded"/>
2195     </xsd:sequence>
2196     <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
2197     <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
2198     <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
2199     <xsd:attribute name="rsidP" type="ST_LongHexNumber"/>
2200     <xsd:attribute name="rsidRDefault" type="ST_LongHexNumber"/>
2201   </xsd:complexType>
2202   <xsd:simpleType name="ST_TblWidth">
2203     <xsd:restriction base="xsd:string">
2204       <xsd:enumeration value="nil"/>
2205       <xsd:enumeration value="pct"/>
2206       <xsd:enumeration value="dxa"/>
2207       <xsd:enumeration value="auto"/>
2208     </xsd:restriction>
2209   </xsd:simpleType>

```

```

2210 <xsd:complexType name="CT_Height">
2211   <xsd:attribute name="val" type="s:ST_TwipsMeasure"/>
2212   <xsd:attribute name="hRule" type="ST_HeightRule"/>
2213 </xsd:complexType>
2214 <xsd:complexType name="CT_TblWidth">
2215   <xsd:attribute name="w" type="ST_DecimalNumberOrPercent"/>
2216   <xsd:attribute name="type" type="ST_TblWidth"/>
2217 </xsd:complexType>
2218 <xsd:complexType name="CT_TblGridCol">
2219   <xsd:attribute name="w" type="s:ST_TwipsMeasure"/>
2220 </xsd:complexType>
2221 <xsd:complexType name="CT_TblGridBase">
2222   <xsd:sequence>
2223     <xsd:element name="gridCol" type="CT_TblGridCol" minOccurs="0" maxOccurs="unbounded"/>
2224   </xsd:sequence>
2225 </xsd:complexType>
2226 <xsd:complexType name="CT_TblGrid">
2227   <xsd:complexContent>
2228     <xsd:extension base="CT_TblGridBase">
2229       <xsd:sequence>
2230         <xsd:element name="tblGridChange" type="CT_TblGridChange" minOccurs="0"/>
2231       </xsd:sequence>
2232     </xsd:extension>
2233   </xsd:complexContent>
2234 </xsd:complexType>
2235 <xsd:complexType name="CT_TcBorders">
2236   <xsd:sequence>
2237     <xsd:element name="top" type="CT_Border" minOccurs="0"/>
2238     <xsd:element name="start" type="CT_Border" minOccurs="0"/>
2239     <xsd:element name="left" type="CT_Border" minOccurs="0"/>
2240     <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
2241     <xsd:element name="end" type="CT_Border" minOccurs="0"/>
2242     <xsd:element name="right" type="CT_Border" minOccurs="0"/>
2243     <xsd:element name="insideH" type="CT_Border" minOccurs="0"/>
2244     <xsd:element name="insideV" type="CT_Border" minOccurs="0"/>
2245     <xsd:element name="tl2br" type="CT_Border" minOccurs="0"/>
2246     <xsd:element name="tr2bl" type="CT_Border" minOccurs="0"/>
2247   </xsd:sequence>
2248 </xsd:complexType>
2249 <xsd:complexType name="CT_TcMar">
2250   <xsd:sequence>
2251     <xsd:element name="top" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2252     <xsd:element name="start" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2253     <xsd:element name="left" type="CT_TblWidth" minOccurs="0"/>
2254     <xsd:element name="bottom" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2255     <xsd:element name="end" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2256     <xsd:element name="right" type="CT_TblWidth" minOccurs="0"/>
2257   </xsd:sequence>
2258 </xsd:complexType>
2259 <xsd:simpleType name="ST_Merge">
2260   <xsd:restriction base="xsd:string">
2261     <xsd:enumeration value="continue"/>
2262     <xsd:enumeration value="restart"/>

```

```

2263     </xsd:restriction>
2264 </xsd:simpleType>
2265 <xsd:complexType name="CT_VMerge">
2266     <xsd:attribute name="val" type="ST_Merge"/>
2267 </xsd:complexType>
2268 <xsd:complexType name="CT_HMerge">
2269     <xsd:attribute name="val" type="ST_Merge"/>
2270 </xsd:complexType>
2271 <xsd:complexType name="CT_TcPrBase">
2272     <xsd:sequence>
2273         <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
2274         <xsd:element name="tcW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2275         <xsd:element name="gridSpan" type="CT_DecimalNumber" minOccurs="0"/>
2276         <xsd:element name="hMerge" type="CT_HMerge" minOccurs="0"/>
2277         <xsd:element name="vMerge" type="CT_VMerge" minOccurs="0"/>
2278         <xsd:element name="tcBorders" type="CT_TcBorders" minOccurs="0" maxOccurs="1"/>
2279         <xsd:element name="shd" type="CT_Shdt" minOccurs="0"/>
2280         <xsd:element name="noWrap" type="CT_OnOff" minOccurs="0"/>
2281         <xsd:element name="tcMar" type="CT_TcMar" minOccurs="0" maxOccurs="1"/>
2282         <xsd:element name="textDirection" type="CT_TextDirection" minOccurs="0" maxOccurs="1"/>
2283         <xsd:element name="tcFitText" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
2284         <xsd:element name="vAlign" type="CT_VerticalJc" minOccurs="0"/>
2285         <xsd:element name="hideMark" type="CT_OnOff" minOccurs="0"/>
2286         <xsd:element name="headers" type="CT-Headers" minOccurs="0"/>
2287     </xsd:sequence>
2288 </xsd:complexType>
2289 <xsd:complexType name="CT_TcPr">
2290     <xsd:complexContent>
2291         <xsd:extension base="CT_TcPrInner">
2292             <xsd:sequence>
2293                 <xsd:element name="tcPrChange" type="CT_TcPrChange" minOccurs="0"/>
2294             </xsd:sequence>
2295         </xsd:extension>
2296     </xsd:complexContent>
2297 </xsd:complexType>
2298 <xsd:complexType name="CT_TcPrInner">
2299     <xsd:complexContent>
2300         <xsd:extension base="CT_TcPrBase">
2301             <xsd:sequence>
2302                 <xsd:group ref="EG_CellMarkupElements" minOccurs="0" maxOccurs="1"/>
2303             </xsd:sequence>
2304         </xsd:extension>
2305     </xsd:complexContent>
2306 </xsd:complexType>
2307 <xsd:complexType name="CT_Tc">
2308     <xsd:sequence>
2309         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
2310         <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
2311     </xsd:sequence>
2312     <xsd:attribute name="id" type="s:ST_String" use="optional"/>
2313 </xsd:complexType>
2314 <xsd:simpleType name="ST_Cnf">
2315     <xsd:restriction base="xsd:string">

```

```

2316         <xsd:length value="12"/>
2317         <xsd:pattern value="[01]*"/>
2318     </xsd:restriction>
2319 </xsd:simpleType>
2320 <xsd:complexType name="CT_Cnf">
2321     <xsd:attribute name="val" type="ST_Cnf"/>
2322     <xsd:attribute name="firstRow" type="s:ST_OnOff"/>
2323     <xsd:attribute name="lastRow" type="s:ST_OnOff"/>
2324     <xsd:attribute name="firstColumn" type="s:ST_OnOff"/>
2325     <xsd:attribute name="lastColumn" type="s:ST_OnOff"/>
2326     <xsd:attribute name="oddVBand" type="s:ST_OnOff"/>
2327     <xsd:attribute name="evenVBand" type="s:ST_OnOff"/>
2328     <xsd:attribute name="oddHBand" type="s:ST_OnOff"/>
2329     <xsd:attribute name="evenHBand" type="s:ST_OnOff"/>
2330     <xsd:attribute name="firstRowFirstColumn" type="s:ST_OnOff"/>
2331     <xsd:attribute name="firstRowLastColumn" type="s:ST_OnOff"/>
2332     <xsd:attribute name="lastRowFirstColumn" type="s:ST_OnOff"/>
2333     <xsd:attribute name="lastRowLastColumn" type="s:ST_OnOff"/>
2334 </xsd:complexType>
2335 <xsd:complexType name="CT_Headers">
2336     <xsd:sequence minOccurs="0" maxOccurs="unbounded">
2337         <xsd:element name="header" type="CT_String"/>
2338     </xsd:sequence>
2339 </xsd:complexType>
2340 <xsd:complexType name="CT_TrPrBase">
2341     <xsd:choice maxOccurs="unbounded">
2342         <xsd:element name="cnfStyle" type="CT_Cnf" minOccurs="0" maxOccurs="1"/>
2343         <xsd:element name="divId" type="CT_DecimalNumber" minOccurs="0"/>
2344         <xsd:element name="gridBefore" type="CT_DecimalNumber" minOccurs="0"/>
2345         <xsd:element name="gridAfter" type="CT_DecimalNumber" minOccurs="0"/>
2346         <xsd:element name="wBefore" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2347         <xsd:element name="wAfter" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2348         <xsd:element name="cantSplit" type="CT_OnOff" minOccurs="0"/>
2349         <xsd:element name="trHeight" type="CT_Height" minOccurs="0"/>
2350         <xsd:element name="tblHeader" type="CT_OnOff" minOccurs="0"/>
2351         <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2352         <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2353         <xsd:element name="hidden" type="CT_OnOff" minOccurs="0"/>
2354     </xsd:choice>
2355 </xsd:complexType>
2356 <xsd:complexType name="CT_TrPr">
2357     <xsd:complexContent>
2358         <xsd:extension base="CT_TrPrBase">
2359             <xsd:sequence>
2360                 <xsd:element name="ins" type="CT_TrackChange" minOccurs="0"/>
2361                 <xsd:element name="del" type="CT_TrackChange" minOccurs="0"/>
2362                 <xsd:element name="trPrChange" type="CT_TrPrChange" minOccurs="0"/>
2363             </xsd:sequence>
2364         </xsd:extension>
2365     </xsd:complexContent>
2366 </xsd:complexType>
2367 <xsd:complexType name="CT_Row">
2368     <xsd:sequence>

```

```

2369     <xsd:element name="tblPrEx" type="CT_TblPrEx" minOccurs="0" maxOccurs="1"/>
2370     <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
2371     <xsd:group ref="EG_ContentCellContent" minOccurs="0" maxOccurs="unbounded"/>
2372   </xsd:sequence>
2373   <xsd:attribute name="rsidRPr" type="ST_LongHexNumber"/>
2374   <xsd:attribute name="rsidR" type="ST_LongHexNumber"/>
2375   <xsd:attribute name="rsidDel" type="ST_LongHexNumber"/>
2376   <xsd:attribute name="rsidTr" type="ST_LongHexNumber"/>
2377 </xsd:complexType>
2378 <xsd:simpleType name="ST_TblLayoutType">
2379   <xsd:restriction base="xsd:string">
2380     <xsd:enumeration value="fixed"/>
2381     <xsd:enumeration value="autofit"/>
2382   </xsd:restriction>
2383 </xsd:simpleType>
2384 <xsd:complexType name="CT_TblLayoutType">
2385   <xsd:attribute name="type" type="ST_TblLayoutType"/>
2386 </xsd:complexType>
2387 <xsd:simpleType name="ST_TblOverlap">
2388   <xsd:restriction base="xsd:string">
2389     <xsd:enumeration value="never"/>
2390     <xsd:enumeration value="overlap"/>
2391   </xsd:restriction>
2392 </xsd:simpleType>
2393 <xsd:complexType name="CT_TblOverlap">
2394   <xsd:attribute name="val" type="ST_TblOverlap" use="required"/>
2395 </xsd:complexType>
2396 <xsd:complexType name="CT_TblPPr">
2397   <xsd:attribute name="leftFromText" type="s:ST_TwipsMeasure"/>
2398   <xsd:attribute name="rightFromText" type="s:ST_TwipsMeasure"/>
2399   <xsd:attribute name="topFromText" type="s:ST_TwipsMeasure"/>
2400   <xsd:attribute name="bottomFromText" type="s:ST_TwipsMeasure"/>
2401   <xsd:attribute name="vertAnchor" type="ST_VAnchor"/>
2402   <xsd:attribute name="horzAnchor" type="ST_HAnchor"/>
2403   <xsd:attribute name="tblpXSpec" type="s:ST_XAlign"/>
2404   <xsd:attribute name="tblpX" type="ST_SignedTwipsMeasure"/>
2405   <xsd:attribute name="tblpYSpec" type="s:ST_YAlign"/>
2406   <xsd:attribute name="tblpY" type="ST_SignedTwipsMeasure"/>
2407 </xsd:complexType>
2408 <xsd:complexType name="CT_TblCellMar">
2409   <xsd:sequence>
2410     <xsd:element name="top" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2411     <xsd:element name="start" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2412     <xsd:element name="left" type="CT_TblWidth" minOccurs="0"/>
2413     <xsd:element name="bottom" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2414     <xsd:element name="end" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2415     <xsd:element name="right" type="CT_TblWidth" minOccurs="0"/>
2416   </xsd:sequence>
2417 </xsd:complexType>
2418 <xsd:complexType name="CT_TblBorders">
2419   <xsd:sequence>
2420     <xsd:element name="top" type="CT_Border" minOccurs="0"/>
2421     <xsd:element name="start" type="CT_Border" minOccurs="0"/>

```

```

2422     <xsd:element name="left" type="CT_Border" minOccurs="0"/>
2423     <xsd:element name="bottom" type="CT_Border" minOccurs="0"/>
2424     <xsd:element name="end" type="CT_Border" minOccurs="0"/>
2425     <xsd:element name="right" type="CT_Border" minOccurs="0"/>
2426     <xsd:element name="insideH" type="CT_Border" minOccurs="0"/>
2427     <xsd:element name="insideV" type="CT_Border" minOccurs="0"/>
2428   </xsd:sequence>
2429 </xsd:complexType>
2430 <xsd:complexType name="CT_TblPrBase">
2431   <xsd:sequence>
2432     <xsd:element name="tblStyle" type="CT_String" minOccurs="0"/>
2433     <xsd:element name="tblpPr" type="CT_TblPPr" minOccurs="0" maxOccurs="1"/>
2434     <xsd:element name="tblOverlap" type="CT_TblOverlap" minOccurs="0" maxOccurs="1"/>
2435     <xsd:element name="bidiVisual" type="CT_OnOff" minOccurs="0" maxOccurs="1"/>
2436     <xsd:element name="tblStyleRowBandSize" type="CT_DecimalNumber" minOccurs="0"
2437       maxOccurs="1"/>
2438     <xsd:element name="tblStyleColBandSize" type="CT_DecimalNumber" minOccurs="0"
2439       maxOccurs="1"/>
2440     <xsd:element name="tblW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2441     <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2442     <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2443     <xsd:element name="tblInd" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2444     <xsd:element name="tblBorders" type="CT_TblBorders" minOccurs="0" maxOccurs="1"/>
2445     <xsd:element name="shd" type="CT_Shd" minOccurs="0" maxOccurs="1"/>
2446     <xsd:element name="tblLayout" type="CT_TblLayoutType" minOccurs="0" maxOccurs="1"/>
2447     <xsd:element name="tblCellMar" type="CT_TblCellMar" minOccurs="0" maxOccurs="1"/>
2448     <xsd:element name="tblLook" type="CT_TblLook" minOccurs="0" maxOccurs="1"/>
2449     <xsd:element name="tblCaption" type="CT_String" minOccurs="0" maxOccurs="1"/>
2450     <xsd:element name="tblDescription" type="CT_String" minOccurs="0" maxOccurs="1"/>
2451   </xsd:sequence>
2452 </xsd:complexType>
2453 <xsd:complexType name="CT_TblPr">
2454   <xsd:complexContent>
2455     <xsd:extension base="CT_TblPrBase">
2456       <xsd:sequence>
2457         <xsd:element name="tblPrChange" type="CT_TblPrChange" minOccurs="0"/>
2458       </xsd:sequence>
2459     </xsd:extension>
2460   </xsd:complexContent>
2461 </xsd:complexType>
2462 <xsd:complexType name="CT_TblPrExBase">
2463   <xsd:sequence>
2464     <xsd:element name="tblW" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2465     <xsd:element name="jc" type="CT_JcTable" minOccurs="0" maxOccurs="1"/>
2466     <xsd:element name="tblCellSpacing" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2467     <xsd:element name="tblInd" type="CT_TblWidth" minOccurs="0" maxOccurs="1"/>
2468     <xsd:element name="tblBorders" type="CT_TblBorders" minOccurs="0" maxOccurs="1"/>
2469     <xsd:element name="shd" type="CT_Shd" minOccurs="0" maxOccurs="1"/>
2470     <xsd:element name="tblLayout" type="CT_TblLayoutType" minOccurs="0" maxOccurs="1"/>
2471     <xsd:element name="tblCellMar" type="CT_TblCellMar" minOccurs="0" maxOccurs="1"/>
2472     <xsd:element name="tblLook" type="CT_TblLook" minOccurs="0" maxOccurs="1"/>
2473   </xsd:sequence>
2474 </xsd:complexType>

```

```

2475 <xsd:complexType name="CT_TblPrEx">
2476   <xsd:complexContent>
2477     <xsd:extension base="CT_TblPrExBase">
2478       <xsd:sequence>
2479         <xsd:element name="tblPrExChange" type="CT_TblPrExChange" minOccurs="0"/>
2480       </xsd:sequence>
2481     </xsd:extension>
2482   </xsd:complexContent>
2483 </xsd:complexType>
2484 <xsd:complexType name="CT_Tbl">
2485   <xsd:sequence>
2486     <xsd:group ref="EG_RangeMarkupElements" minOccurs="0" maxOccurs="unbounded"/>
2487     <xsd:element name="tblPr" type="CT_TblPr"/>
2488     <xsd:element name="tblGrid" type="CT_TblGrid"/>
2489     <xsd:group ref="EG_ContentRowContent" minOccurs="0" maxOccurs="unbounded"/>
2490   </xsd:sequence>
2491 </xsd:complexType>
2492 <xsd:complexType name="CT_TblLook">
2493   <xsd:attribute name="firstRow" type="s:ST_OnOff"/>
2494   <xsd:attribute name="lastRow" type="s:ST_OnOff"/>
2495   <xsd:attribute name="firstColumn" type="s:ST_OnOff"/>
2496   <xsd:attribute name="lastColumn" type="s:ST_OnOff"/>
2497   <xsd:attribute name="noHBand" type="s:ST_OnOff"/>
2498   <xsd:attribute name="noVBand" type="s:ST_OnOff"/>
2499   <xsd:attribute name="val" type="ST_ShortHexNumber"/>
2500 </xsd:complexType>
2501 <xsd:simpleType name="ST_FtnPos">
2502   <xsd:restriction base="xsd:string">
2503     <xsd:enumeration value="pageBottom"/>
2504     <xsd:enumeration value="beneathText"/>
2505     <xsd:enumeration value="sectEnd"/>
2506     <xsd:enumeration value="docEnd"/>
2507   </xsd:restriction>
2508 </xsd:simpleType>
2509 <xsd:complexType name="CT_FtnPos">
2510   <xsd:attribute name="val" type="ST_FtnPos" use="required"/>
2511 </xsd:complexType>
2512 <xsd:simpleType name="ST_EdnPos">
2513   <xsd:restriction base="xsd:string">
2514     <xsd:enumeration value="sectEnd"/>
2515     <xsd:enumeration value="docEnd"/>
2516   </xsd:restriction>
2517 </xsd:simpleType>
2518 <xsd:complexType name="CT_EdnPos">
2519   <xsd:attribute name="val" type="ST_EdnPos" use="required"/>
2520 </xsd:complexType>
2521 <xsd:complexType name="CT_NumFmt">
2522   <xsd:attribute name="val" type="ST_NumberFormat" use="required"/>
2523   <xsd:attribute name="format" type="s:ST_String" use="optional"/>
2524 </xsd:complexType>
2525 <xsd:simpleType name="ST_RestartNumber">
2526   <xsd:restriction base="xsd:string">
2527     <xsd:enumeration value="continuous"/>

```



```

2528         <xsd:enumeration value="eachSect"/>
2529         <xsd:enumeration value="eachPage"/>
2530     </xsd:restriction>
2531 </xsd:simpleType>
2532 <xsd:complexType name="CT_NumRestart">
2533     <xsd:attribute name="val" type="ST_RestartNumber" use="required"/>
2534 </xsd:complexType>
2535 <xsd:complexType name="CT_FtnEdnRef">
2536     <xsd:attribute name="customMarkFollows" type="s:ST_OnOff" use="optional"/>
2537     <xsd:attribute name="id" use="required" type="ST_DecimalNumber"/>
2538 </xsd:complexType>
2539 <xsd:complexType name="CT_FtnEdnSepRef">
2540     <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
2541 </xsd:complexType>
2542 <xsd:complexType name="CT_FtnEdn">
2543     <xsd:sequence>
2544         <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
2545     </xsd:sequence>
2546     <xsd:attribute name="type" type="ST_FtnEdn" use="optional"/>
2547     <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
2548 </xsd:complexType>
2549 <xsd:group name="EG_FtnEdnNumProps">
2550     <xsd:sequence>
2551         <xsd:element name="numStart" type="CT_DecimalNumber" minOccurs="0"/>
2552         <xsd:element name="numRestart" type="CT_NumRestart" minOccurs="0"/>
2553     </xsd:sequence>
2554 </xsd:group>
2555 <xsd:complexType name="CT_FtnProps">
2556     <xsd:sequence>
2557         <xsd:element name="pos" type="CT_FtnPos" minOccurs="0"/>
2558         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0"/>
2559         <xsd:group ref="EG_FtnEdnNumProps" minOccurs="0"/>
2560     </xsd:sequence>
2561 </xsd:complexType>
2562 <xsd:complexType name="CT_EdnProps">
2563     <xsd:sequence>
2564         <xsd:element name="pos" type="CT_EdnPos" minOccurs="0"/>
2565         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0"/>
2566         <xsd:group ref="EG_FtnEdnNumProps" minOccurs="0"/>
2567     </xsd:sequence>
2568 </xsd:complexType>
2569 <xsd:complexType name="CT_FtnDocProps">
2570     <xsd:complexContent>
2571         <xsd:extension base="CT_FtnProps">
2572             <xsd:sequence>
2573                 <xsd:element name="footnote" type="CT_FtnEdnSepRef" minOccurs="0" maxOccurs="3"/>
2574             </xsd:sequence>
2575         </xsd:extension>
2576     </xsd:complexContent>
2577 </xsd:complexType>
2578 <xsd:complexType name="CT_EdnDocProps">
2579     <xsd:complexContent>
2580         <xsd:extension base="CT_EdnProps">

```

```

2581         <xsd:sequence>
2582             <xsd:element name="endnote" type="CT_FtnEdnSepRef" minOccurs="0" maxOccurs="3"/>
2583         </xsd:sequence>
2584     </xsd:extension>
2585 </xsd:complexContent>
2586 </xsd:complexType>
2587 <xsd:complexType name="CT_RecipientData">
2588     <xsd:sequence>
2589         <xsd:element name="active" type="CT_OnOff" minOccurs="0"/>
2590         <xsd:element name="column" type="CT_DecimalNumber" minOccurs="1"/>
2591         <xsd:element name="uniqueTag" type="xsd:base64Binary" minOccurs="1"/>
2592     </xsd:sequence>
2593 </xsd:complexType>
2594 <xsd:complexType name="CT_Recipients">
2595     <xsd:sequence>
2596         <xsd:element name="recipientData" type="CT_RecipientData" minOccurs="1"
2597             maxOccurs="unbounded"/>
2598     </xsd:sequence>
2599 </xsd:complexType>
2600 <xsd:element name="recipients" type="CT_Recipients"/>
2601 <xsd:complexType name="CT_OdsoFieldMapData">
2602     <xsd:sequence>
2603         <xsd:element name="type" type="CT_MailMergeOdsoFMDFieldType" minOccurs="0"/>
2604         <xsd:element name="name" type="CT_String" minOccurs="0"/>
2605         <xsd:element name="mappedName" type="CT_String" minOccurs="0"/>
2606         <xsd:element name="column" type="CT_DecimalNumber" minOccurs="0"/>
2607         <xsd:element name="lid" type="CT_Lang" minOccurs="0"/>
2608         <xsd:element name="dynamicAddress" type="CT_OnOff" minOccurs="0"/>
2609     </xsd:sequence>
2610 </xsd:complexType>
2611 <xsd:simpleType name="ST_MailMergeSourceType">
2612     <xsd:restriction base="xsd:string">
2613         <xsd:enumeration value="database"/>
2614         <xsd:enumeration value="addressBook"/>
2615         <xsd:enumeration value="document1"/>
2616         <xsd:enumeration value="document2"/>
2617         <xsd:enumeration value="text"/>
2618         <xsd:enumeration value="email"/>
2619         <xsd:enumeration value="native"/>
2620         <xsd:enumeration value="legacy"/>
2621         <xsd:enumeration value="master"/>
2622     </xsd:restriction>
2623 </xsd:simpleType>
2624 <xsd:complexType name="CT_MailMergeSourceType">
2625     <xsd:attribute name="val" use="required" type="ST_MailMergeSourceType"/>
2626 </xsd:complexType>
2627 <xsd:complexType name="CT_Odso">
2628     <xsd:sequence>
2629         <xsd:element name="udl" type="CT_String" minOccurs="0"/>
2630         <xsd:element name="table" type="CT_String" minOccurs="0"/>
2631         <xsd:element name="src" type="CT_Rel" minOccurs="0"/>
2632         <xsd:element name="colDelim" type="CT_DecimalNumber" minOccurs="0"/>
2633         <xsd:element name="type" type="CT_MailMergeSourceType" minOccurs="0"/>

```

```

2634     <xsd:element name="fHdr" type="CT_OnOff" minOccurs="0"/>
2635     <xsd:element name="fieldMapData" type="CT_OdsoFieldMapData" minOccurs="0"
2636         maxOccurs="unbounded"/>
2637     <xsd:element name="recipientData" type="CT_Rel" minOccurs="0" maxOccurs="unbounded"/>
2638 </xsd:sequence>
2639 </xsd:complexType>
2640 <xsd:complexType name="CT_MailMerge">
2641     <xsd:sequence>
2642         <xsd:element name="mainDocumentType" type="CT_MailMergeDocType" minOccurs="1"/>
2643         <xsd:element name="linkToQuery" type="CT_OnOff" minOccurs="0"/>
2644         <xsd:element name="dataType" type="CT_MailMergeDataType" minOccurs="1"/>
2645         <xsd:element name="connectString" type="CT_String" minOccurs="0"/>
2646         <xsd:element name="query" type="CT_String" minOccurs="0"/>
2647         <xsd:element name="dataSource" type="CT_Rel" minOccurs="0"/>
2648         <xsd:element name="headerSource" type="CT_Rel" minOccurs="0"/>
2649         <xsd:element name="doNotSuppressBlankLines" type="CT_OnOff" minOccurs="0"/>
2650         <xsd:element name="destination" type="CT_MailMergeDest" minOccurs="0"/>
2651         <xsd:element name="addressFieldName" type="CT_String" minOccurs="0"/>
2652         <xsd:element name="mailSubject" type="CT_String" minOccurs="0"/>
2653         <xsd:element name="mailAsAttachment" type="CT_OnOff" minOccurs="0"/>
2654         <xsd:element name="viewMergedData" type="CT_OnOff" minOccurs="0"/>
2655         <xsd:element name="activeRecord" type="CT_DecimalNumber" minOccurs="0"/>
2656         <xsd:element name="checkErrors" type="CT_DecimalNumber" minOccurs="0"/>
2657         <xsd:element name="odso" type="CT_Odso" minOccurs="0"/>
2658     </xsd:sequence>
2659 </xsd:complexType>
2660 <xsd:simpleType name="ST_TargetScreenSz">
2661     <xsd:restriction base="xsd:string">
2662         <xsd:enumeration value="544x376"/>
2663         <xsd:enumeration value="640x480"/>
2664         <xsd:enumeration value="720x512"/>
2665         <xsd:enumeration value="800x600"/>
2666         <xsd:enumeration value="1024x768"/>
2667         <xsd:enumeration value="1152x882"/>
2668         <xsd:enumeration value="1152x900"/>
2669         <xsd:enumeration value="1280x1024"/>
2670         <xsd:enumeration value="1600x1200"/>
2671         <xsd:enumeration value="1800x1440"/>
2672         <xsd:enumeration value="1920x1200"/>
2673     </xsd:restriction>
2674 </xsd:simpleType>
2675 <xsd:complexType name="CT_TargetScreenSz">
2676     <xsd:attribute name="val" type="ST_TargetScreenSz" use="required"/>
2677 </xsd:complexType>
2678 <xsd:complexType name="CT_Compat">
2679     <xsd:sequence>
2680         <xsd:element name="useSingleBorderforContiguousCells" type="CT_OnOff" minOccurs="0"/>
2681         <xsd:element name="wpJustification" type="CT_OnOff" minOccurs="0"/>
2682         <xsd:element name="noTabHangInd" type="CT_OnOff" minOccurs="0"/>
2683         <xsd:element name="noLeading" type="CT_OnOff" minOccurs="0"/>
2684         <xsd:element name="spaceForUL" type="CT_OnOff" minOccurs="0"/>
2685         <xsd:element name="noColumnBalance" type="CT_OnOff" minOccurs="0"/>
2686         <xsd:element name="balanceSingleByteDoubleByteWidth" type="CT_OnOff" minOccurs="0"/>

```

```

2687 <xsd:element name="noExtraLineSpacing" type="CT OnOff" minOccurs="0"/>
2688 <xsd:element name="doNotLeaveBackslashAlone" type="CT OnOff" minOccurs="0"/>
2689 <xsd:element name="ulTrailSpace" type="CT OnOff" minOccurs="0"/>
2690 <xsd:element name="doNotExpandShiftReturn" type="CT OnOff" minOccurs="0"/>
2691 <xsd:element name="spacingInWholePoints" type="CT OnOff" minOccurs="0"/>
2692 <xsd:element name="lineWrapLikeWord6" type="CT OnOff" minOccurs="0"/>
2693 <xsd:element name="printBodyTextBeforeHeader" type="CT OnOff" minOccurs="0"/>
2694 <xsd:element name="printColBlack" type="CT OnOff" minOccurs="0"/>
2695 <xsd:element name="wpSpaceWidth" type="CT OnOff" minOccurs="0"/>
2696 <xsd:element name="showBreaksInFrames" type="CT OnOff" minOccurs="0"/>
2697 <xsd:element name="subFontBySize" type="CT OnOff" minOccurs="0"/>
2698 <xsd:element name="suppressBottomSpacing" type="CT OnOff" minOccurs="0"/>
2699 <xsd:element name="suppressTopSpacing" type="CT OnOff" minOccurs="0"/>
2700 <xsd:element name="suppressSpacingAtTopOfPage" type="CT OnOff" minOccurs="0"/>
2701 <xsd:element name="suppressTopSpacingWP" type="CT OnOff" minOccurs="0"/>
2702 <xsd:element name="suppressSpBfAfterPgBrk" type="CT OnOff" minOccurs="0"/>
2703 <xsd:element name="swapBordersFacingPages" type="CT OnOff" minOccurs="0"/>
2704 <xsd:element name="convMailMergeEsc" type="CT OnOff" minOccurs="0"/>
2705 <xsd:element name="truncateFontHeightsLikeWP6" type="CT OnOff" minOccurs="0"/>
2706 <xsd:element name="mwSmallCaps" type="CT OnOff" minOccurs="0"/>
2707 <xsd:element name="usePrinterMetrics" type="CT OnOff" minOccurs="0"/>
2708 <xsd:element name="doNotSuppressParagraphBorders" type="CT OnOff" minOccurs="0"/>
2709 <xsd:element name="wrapTrailSpaces" type="CT OnOff" minOccurs="0"/>
2710 <xsd:element name="footnoteLayoutLikeWW8" type="CT OnOff" minOccurs="0"/>
2711 <xsd:element name="shapeLayoutLikeWW8" type="CT OnOff" minOccurs="0"/>
2712 <xsd:element name="alignTablesRowByRow" type="CT OnOff" minOccurs="0"/>
2713 <xsd:element name="forgetLastTabAlignment" type="CT OnOff" minOccurs="0"/>
2714 <xsd:element name="adjustLineHeightInTable" type="CT OnOff" minOccurs="0"/>
2715 <xsd:element name="autoSpaceLikeWord95" type="CT OnOff" minOccurs="0"/>
2716 <xsd:element name="noSpaceRaiseLower" type="CT OnOff" minOccurs="0"/>
2717 <xsd:element name="doNotUseHTMLParagraphAutoSpacing" type="CT OnOff" minOccurs="0"/>
2718 <xsd:element name="layoutRawTableWidth" type="CT OnOff" minOccurs="0"/>
2719 <xsd:element name="layoutTableRowsApart" type="CT OnOff" minOccurs="0"/>
2720 <xsd:element name="useWord97LineBreakRules" type="CT OnOff" minOccurs="0"/>
2721 <xsd:element name="doNotBreakWrappedTables" type="CT OnOff" minOccurs="0"/>
2722 <xsd:element name="doNotSnapToGridInCell" type="CT OnOff" minOccurs="0"/>
2723 <xsd:element name="selectFldWithFirstOrLastChar" type="CT OnOff" minOccurs="0"/>
2724 <xsd:element name="applyBreakingRules" type="CT OnOff" minOccurs="0"/>
2725 <xsd:element name="doNotWrapTextWithPunct" type="CT OnOff" minOccurs="0"/>
2726 <xsd:element name="doNotUseEastAsianBreakRules" type="CT OnOff" minOccurs="0"/>
2727 <xsd:element name="useWord2002TableStyleRules" type="CT OnOff" minOccurs="0"/>
2728 <xsd:element name="growAutofit" type="CT OnOff" minOccurs="0"/>
2729 <xsd:element name="useFELayout" type="CT OnOff" minOccurs="0"/>
2730 <xsd:element name="useNormalStyleForList" type="CT OnOff" minOccurs="0"/>
2731 <xsd:element name="doNotUseIndentAsNumberingTabStop" type="CT OnOff" minOccurs="0"/>
2732 <xsd:element name="useAltKinsokuLineBreakRules" type="CT OnOff" minOccurs="0"/>
2733 <xsd:element name="allowSpaceOfSameStyleInTable" type="CT OnOff" minOccurs="0"/>
2734 <xsd:element name="doNotSuppressIndentation" type="CT OnOff" minOccurs="0"/>
2735 <xsd:element name="doNotAutofitConstrainedTables" type="CT OnOff" minOccurs="0"/>
2736 <xsd:element name="autofitToFirstFixedWidthCell" type="CT OnOff" minOccurs="0"/>
2737 <xsd:element name="underlineTabInNumList" type="CT OnOff" minOccurs="0"/>
2738 <xsd:element name="displayHangulFixedWidth" type="CT OnOff" minOccurs="0"/>
2739 <xsd:element name="splitPgBreakAndParaMark" type="CT OnOff" minOccurs="0"/>

```

```

2740     <xsd:element name="doNotVertAlignCellWithSp" type="CT_OnOff" minOccurs="0"/>
2741     <xsd:element name="doNotBreakConstrainedForcedTable" type="CT_OnOff" minOccurs="0"/>
2742     <xsd:element name="doNotVertAlignInTxbx" type="CT_OnOff" minOccurs="0"/>
2743     <xsd:element name="useAnsiKerningPairs" type="CT_OnOff" minOccurs="0"/>
2744     <xsd:element name="cachedColBalance" type="CT_OnOff" minOccurs="0"/>
2745     <xsd:element name="compatSetting" type="CT_CompatSetting" minOccurs="0"
2746         maxOccurs="unbounded"/>
2747     </xsd:sequence>
2748 </xsd:complexType>
2749 <xsd:complexType name="CT_CompatSetting">
2750     <xsd:attribute name="name" type="s:ST_String"/>
2751     <xsd:attribute name="uri" type="s:ST_String"/>
2752     <xsd:attribute name="val" type="s:ST_String"/>
2753 </xsd:complexType>
2754 <xsd:complexType name="CT_DocVar">
2755     <xsd:attribute name="name" type="s:ST_String" use="required"/>
2756     <xsd:attribute name="val" type="s:ST_String" use="required"/>
2757 </xsd:complexType>
2758 <xsd:complexType name="CT_DocVars">
2759     <xsd:sequence>
2760         <xsd:element name="docVar" type="CT_DocVar" minOccurs="0" maxOccurs="unbounded"/>
2761     </xsd:sequence>
2762 </xsd:complexType>
2763 <xsd:complexType name="CT_DocRsids">
2764     <xsd:sequence>
2765         <xsd:element name="rsidRoot" type="CT_LongHexNumber" minOccurs="0" maxOccurs="1"/>
2766         <xsd:element name="rsid" type="CT_LongHexNumber" minOccurs="0" maxOccurs="unbounded"/>
2767     </xsd:sequence>
2768 </xsd:complexType>
2769 <xsd:simpleType name="ST_CharacterSpacing">
2770     <xsd:restriction base="xsd:string">
2771         <xsd:enumeration value="doNotCompress"/>
2772         <xsd:enumeration value="compressPunctuation"/>
2773         <xsd:enumeration value="compressPunctuationAndJapaneseKana"/>
2774     </xsd:restriction>
2775 </xsd:simpleType>
2776 <xsd:complexType name="CT_CharacterSpacing">
2777     <xsd:attribute name="val" type="ST_CharacterSpacing" use="required"/>
2778 </xsd:complexType>
2779 <xsd:complexType name="CT_SaveThroughXslt">
2780     <xsd:attribute ref="r:id" use="optional"/>
2781     <xsd:attribute name="solutionID" type="s:ST_String" use="optional"/>
2782 </xsd:complexType>
2783 <xsd:complexType name="CT_RPrDefault">
2784     <xsd:sequence>
2785         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
2786     </xsd:sequence>
2787 </xsd:complexType>
2788 <xsd:complexType name="CT_PPrDefault">
2789     <xsd:sequence>
2790         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0"/>
2791     </xsd:sequence>
2792 </xsd:complexType>

```

```

2793 <xsd:complexType name="CT_DocDefaults">
2794   <xsd:sequence>
2795     <xsd:element name="rPrDefault" type="CT_RPrDefault" minOccurs="0"/>
2796     <xsd:element name="pPrDefault" type="CT_PPrDefault" minOccurs="0"/>
2797   </xsd:sequence>
2798 </xsd:complexType>
2799 <xsd:simpleType name="ST_WmlColorSchemeIndex">
2800   <xsd:restriction base="xsd:string">
2801     <xsd:enumeration value="dark1"/>
2802     <xsd:enumeration value="light1"/>
2803     <xsd:enumeration value="dark2"/>
2804     <xsd:enumeration value="light2"/>
2805     <xsd:enumeration value="accent1"/>
2806     <xsd:enumeration value="accent2"/>
2807     <xsd:enumeration value="accent3"/>
2808     <xsd:enumeration value="accent4"/>
2809     <xsd:enumeration value="accent5"/>
2810     <xsd:enumeration value="accent6"/>
2811     <xsd:enumeration value="hyperlink"/>
2812     <xsd:enumeration value="followedHyperlink"/>
2813   </xsd:restriction>
2814 </xsd:simpleType>
2815 <xsd:complexType name="CT_ColorSchemeMapping">
2816   <xsd:attribute name="bg1" type="ST_WmlColorSchemeIndex"/>
2817   <xsd:attribute name="t1" type="ST_WmlColorSchemeIndex"/>
2818   <xsd:attribute name="bg2" type="ST_WmlColorSchemeIndex"/>
2819   <xsd:attribute name="t2" type="ST_WmlColorSchemeIndex"/>
2820   <xsd:attribute name="accent1" type="ST_WmlColorSchemeIndex"/>
2821   <xsd:attribute name="accent2" type="ST_WmlColorSchemeIndex"/>
2822   <xsd:attribute name="accent3" type="ST_WmlColorSchemeIndex"/>
2823   <xsd:attribute name="accent4" type="ST_WmlColorSchemeIndex"/>
2824   <xsd:attribute name="accent5" type="ST_WmlColorSchemeIndex"/>
2825   <xsd:attribute name="accent6" type="ST_WmlColorSchemeIndex"/>
2826   <xsd:attribute name="hyperlink" type="ST_WmlColorSchemeIndex"/>
2827   <xsd:attribute name="followedHyperlink" type="ST_WmlColorSchemeIndex"/>
2828 </xsd:complexType>
2829 <xsd:complexType name="CT_ReadingModeInkLockDown">
2830   <xsd:attribute name="actualPg" type="s:ST_OnOff" use="required"/>
2831   <xsd:attribute name="w" type="ST_PixelsMeasure" use="required"/>
2832   <xsd:attribute name="h" type="ST_PixelsMeasure" use="required"/>
2833   <xsd:attribute name="fontSz" type="ST_DecimalNumberOrPercent" use="required"/>
2834 </xsd:complexType>
2835 <xsd:complexType name="CT_WriteProtection">
2836   <xsd:attribute name="recommended" type="s:ST_OnOff" use="optional"/>
2837   <xsd:attributeGroup ref="AG_Password"/>
2838   <xsd:attributeGroup ref="AG_TransitionalPassword"/>
2839 </xsd:complexType>
2840 <xsd:complexType name="CT_Settings">
2841   <xsd:sequence>
2842     <xsd:element name="writeProtection" type="CT_WriteProtection" minOccurs="0"/>
2843     <xsd:element name="view" type="CT_View" minOccurs="0"/>
2844     <xsd:element name="zoom" type="CT_Zoom" minOccurs="0"/>
2845     <xsd:element name="removePersonalInformation" type="CT_OnOff" minOccurs="0"/>

```

```

2846 <xsd:element name="removeDateAndTime" type="CT OnOff" minOccurs="0"/>
2847 <xsd:element name="doNotDisplayPageBoundaries" type="CT OnOff" minOccurs="0"/>
2848 <xsd:element name="displayBackgroundShape" type="CT OnOff" minOccurs="0"/>
2849 <xsd:element name="printPostScriptOverText" type="CT OnOff" minOccurs="0"/>
2850 <xsd:element name="printFractionalCharacterWidth" type="CT OnOff" minOccurs="0"/>
2851 <xsd:element name="printFormsData" type="CT OnOff" minOccurs="0"/>
2852 <xsd:element name="embedTrueTypeFonts" type="CT OnOff" minOccurs="0"/>
2853 <xsd:element name="embedSystemFonts" type="CT OnOff" minOccurs="0"/>
2854 <xsd:element name="saveSubsetFonts" type="CT OnOff" minOccurs="0"/>
2855 <xsd:element name="saveFormsData" type="CT OnOff" minOccurs="0"/>
2856 <xsd:element name="mirrorMargins" type="CT OnOff" minOccurs="0"/>
2857 <xsd:element name="alignBordersAndEdges" type="CT OnOff" minOccurs="0"/>
2858 <xsd:element name="bordersDoNotSurroundHeader" type="CT OnOff" minOccurs="0"/>
2859 <xsd:element name="bordersDoNotSurroundFooter" type="CT OnOff" minOccurs="0"/>
2860 <xsd:element name="gutterAtTop" type="CT OnOff" minOccurs="0"/>
2861 <xsd:element name="hideSpellingErrors" type="CT OnOff" minOccurs="0"/>
2862 <xsd:element name="hideGrammaticalErrors" type="CT OnOff" minOccurs="0"/>
2863 <xsd:element name="activeWritingStyle" type="CT WritingStyle" minOccurs="0"
2864     maxOccurs="unbounded"/>
2865 <xsd:element name="proofState" type="CT Proof" minOccurs="0"/>
2866 <xsd:element name="formsDesign" type="CT OnOff" minOccurs="0"/>
2867 <xsd:element name="attachedTemplate" type="CT Rel" minOccurs="0"/>
2868 <xsd:element name="linkStyles" type="CT OnOff" minOccurs="0"/>
2869 <xsd:element name="stylePaneFormatFilter" type="CT StylePaneFilter" minOccurs="0"/>
2870 <xsd:element name="stylePaneSortMethod" type="CT StyleSort" minOccurs="0"/>
2871 <xsd:element name="documentType" type="CT DocType" minOccurs="0"/>
2872 <xsd:element name="mailMerge" type="CT MailMerge" minOccurs="0"/>
2873 <xsd:element name="revisionView" type="CT TrackChangesView" minOccurs="0"/>
2874 <xsd:element name="trackRevisions" type="CT OnOff" minOccurs="0"/>
2875 <xsd:element name="doNotTrackMoves" type="CT OnOff" minOccurs="0"/>
2876 <xsd:element name="doNotTrackFormatting" type="CT OnOff" minOccurs="0"/>
2877 <xsd:element name="documentProtection" type="CT DocProtect" minOccurs="0"/>
2878 <xsd:element name="autoFormatOverride" type="CT OnOff" minOccurs="0"/>
2879 <xsd:element name="styleLockTheme" type="CT OnOff" minOccurs="0"/>
2880 <xsd:element name="styleLockQFSet" type="CT OnOff" minOccurs="0"/>
2881 <xsd:element name="defaultTabStop" type="CT TwipsMeasure" minOccurs="0"/>
2882 <xsd:element name="autoHyphenation" type="CT OnOff" minOccurs="0"/>
2883 <xsd:element name="consecutiveHyphenLimit" type="CT DecimalNumber" minOccurs="0"/>
2884 <xsd:element name="hyphenationZone" type="CT TwipsMeasure" minOccurs="0"/>
2885 <xsd:element name="doNotHyphenateCaps" type="CT OnOff" minOccurs="0"/>
2886 <xsd:element name="showEnvelope" type="CT OnOff" minOccurs="0"/>
2887 <xsd:element name="summaryLength" type="CT DecimalNumberOrPrecent" minOccurs="0"/>
2888 <xsd:element name="clickAndTypeStyle" type="CT String" minOccurs="0"/>
2889 <xsd:element name="defaultTableStyle" type="CT String" minOccurs="0"/>
2890 <xsd:element name="evenAndOddHeaders" type="CT OnOff" minOccurs="0"/>
2891 <xsd:element name="bookFoldRevPrinting" type="CT OnOff" minOccurs="0"/>
2892 <xsd:element name="bookFoldPrinting" type="CT OnOff" minOccurs="0"/>
2893 <xsd:element name="bookFoldPrintingSheets" type="CT DecimalNumber" minOccurs="0"/>
2894 <xsd:element name="drawingGridHorizontalSpacing" type="CT TwipsMeasure" minOccurs="0"/>
2895 <xsd:element name="drawingGridVerticalSpacing" type="CT TwipsMeasure" minOccurs="0"/>
2896 <xsd:element name="displayHorizontalDrawingGridEvery" type="CT DecimalNumber"
2897     minOccurs="0"/>

```

```

2898     <xsd:element name="displayVerticalDrawingGridEvery" type="CT_DecimalNumber"
2899         minOccurs="0"/>
2900     <xsd:element name="doNotUseMarginsForDrawingGridOrigin" type="CT_OnOff" minOccurs="0"/>
2901     <xsd:element name="drawingGridHorizontalOrigin" type="CT_TwipsMeasure" minOccurs="0"/>
2902     <xsd:element name="drawingGridVerticalOrigin" type="CT_TwipsMeasure" minOccurs="0"/>
2903     <xsd:element name="doNotShadeFormData" type="CT_OnOff" minOccurs="0"/>
2904     <xsd:element name="noPunctuationKerning" type="CT_OnOff" minOccurs="0"/>
2905     <xsd:element name="characterSpacingControl" type="CT_CharacterSpacing" minOccurs="0"/>
2906     <xsd:element name="printTwoOnOne" type="CT_OnOff" minOccurs="0"/>
2907     <xsd:element name="strictFirstAndLastChars" type="CT_OnOff" minOccurs="0"/>
2908     <xsd:element name="noLineBreaksAfter" type="CT_Kinsoku" minOccurs="0"/>
2909     <xsd:element name="noLineBreaksBefore" type="CT_Kinsoku" minOccurs="0"/>
2910     <xsd:element name="savePreviewPicture" type="CT_OnOff" minOccurs="0"/>
2911     <xsd:element name="doNotValidateAgainstSchema" type="CT_OnOff" minOccurs="0"/>
2912     <xsd:element name="saveInvalidXml" type="CT_OnOff" minOccurs="0"/>
2913     <xsd:element name="ignoreMixedContent" type="CT_OnOff" minOccurs="0"/>
2914     <xsd:element name="alwaysShowPlaceholderText" type="CT_OnOff" minOccurs="0"/>
2915     <xsd:element name="doNotDemarcateInvalidXml" type="CT_OnOff" minOccurs="0"/>
2916     <xsd:element name="saveXmlDataOnly" type="CT_OnOff" minOccurs="0"/>
2917     <xsd:element name="useXSLTWhenSaving" type="CT_OnOff" minOccurs="0"/>
2918     <xsd:element name="saveThroughXslt" type="CT_SaveThroughXslt" minOccurs="0"/>
2919     <xsd:element name="showXMLTags" type="CT_OnOff" minOccurs="0"/>
2920     <xsd:element name="alwaysMergeEmptyNamespace" type="CT_OnOff" minOccurs="0"/>
2921     <xsd:element name="updateFields" type="CT_OnOff" minOccurs="0"/>
2922     <xsd:element name="hdrShapeDefaults" type="CT_ShapeDefaults" minOccurs="0"/>
2923     <xsd:element name="footnotePr" type="CT_FtnDocProps" minOccurs="0"/>
2924     <xsd:element name="endnotePr" type="CT_EdnDocProps" minOccurs="0"/>
2925     <xsd:element name="compat" type="CT_Compat" minOccurs="0"/>
2926     <xsd:element name="docVars" type="CT_DocVars" minOccurs="0"/>
2927     <xsd:element name="rsids" type="CT_DocRsids" minOccurs="0"/>
2928     <xsd:element ref="m:mathPr" minOccurs="0" maxOccurs="1"/>
2929     <xsd:element name="attachedSchema" type="CT_String" minOccurs="0" maxOccurs="unbounded"/>
2930     <xsd:element name="themeFontLang" type="CT_Language" minOccurs="0" maxOccurs="1"/>
2931     <xsd:element name="clrSchemeMapping" type="CT_ColorSchemeMapping" minOccurs="0"/>
2932     <xsd:element name="doNotIncludeSubdocsInStats" type="CT_OnOff" minOccurs="0"/>
2933     <xsd:element name="doNotAutoCompressPictures" type="CT_OnOff" minOccurs="0"/>
2934     <xsd:element name="forceUpgrade" type="CT_Empty" minOccurs="0" maxOccurs="1"/>
2935     <xsd:element name="captions" type="CT_Captions" minOccurs="0" maxOccurs="1"/>
2936     <xsd:element name="readModeInkLockDown" type="CT_ReadingModeInkLockDown" minOccurs="0"/>
2937     <xsd:element name="smartTagType" type="CT_SmartTagType" minOccurs="0"
2938         maxOccurs="unbounded"/>
2939     <xsd:element ref="sl:schemaLibrary" minOccurs="0" maxOccurs="1"/>
2940     <xsd:element name="shapeDefaults" type="CT_ShapeDefaults" minOccurs="0"/>
2941     <xsd:element name="doNotEmbedSmartTags" type="CT_OnOff" minOccurs="0"/>
2942     <xsd:element name="decimalSymbol" type="CT_String" minOccurs="0" maxOccurs="1"/>
2943     <xsd:element name="listSeparator" type="CT_String" minOccurs="0" maxOccurs="1"/>
2944 </xsd:sequence>
2945 </xsd:complexType>
2946 <xsd:complexType name="CT_StyleSort">
2947     <xsd:attribute name="val" type="ST_StyleSort" use="required"/>
2948 </xsd:complexType>
2949 <xsd:complexType name="CT_StylePaneFilter">
2950     <xsd:attribute name="allStyles" type="s:ST_OnOff"/>

```



```

2951 <xsd:attribute name="customStyles" type="s:ST_OnOff"/>
2952 <xsd:attribute name="latentStyles" type="s:ST_OnOff"/>
2953 <xsd:attribute name="stylesInUse" type="s:ST_OnOff"/>
2954 <xsd:attribute name="headingStyles" type="s:ST_OnOff"/>
2955 <xsd:attribute name="numberingStyles" type="s:ST_OnOff"/>
2956 <xsd:attribute name="tableStyles" type="s:ST_OnOff"/>
2957 <xsd:attribute name="directFormattingOnRuns" type="s:ST_OnOff"/>
2958 <xsd:attribute name="directFormattingOnParagraphs" type="s:ST_OnOff"/>
2959 <xsd:attribute name="directFormattingOnNumbering" type="s:ST_OnOff"/>
2960 <xsd:attribute name="directFormattingOnTables" type="s:ST_OnOff"/>
2961 <xsd:attribute name="clearFormatting" type="s:ST_OnOff"/>
2962 <xsd:attribute name="top3HeadingStyles" type="s:ST_OnOff"/>
2963 <xsd:attribute name="visibleStyles" type="s:ST_OnOff"/>
2964 <xsd:attribute name="alternateStyleNames" type="s:ST_OnOff"/>
2965 <xsd:attribute name="val" type="ST_ShortHexNumber"/>
2966 </xsd:complexType>
2967 <xsd:simpleType name="ST_StyleSort">
2968   <xsd:restriction base="xsd:string">
2969     <xsd:enumeration value="name"/>
2970     <xsd:enumeration value="priority"/>
2971     <xsd:enumeration value="default"/>
2972     <xsd:enumeration value="font"/>
2973     <xsd:enumeration value="basedOn"/>
2974     <xsd:enumeration value="type"/>
2975     <xsd:enumeration value="0000"/>
2976     <xsd:enumeration value="0001"/>
2977     <xsd:enumeration value="0002"/>
2978     <xsd:enumeration value="0003"/>
2979     <xsd:enumeration value="0004"/>
2980     <xsd:enumeration value="0005"/>
2981   </xsd:restriction>
2982 </xsd:simpleType>
2983 <xsd:complexType name="CT_WebSettings">
2984   <xsd:sequence>
2985     <xsd:element name="frameset" type="CT_Frameset" minOccurs="0"/>
2986     <xsd:element name="divs" type="CT_Divs" minOccurs="0"/>
2987     <xsd:element name="encoding" type="CT_String" minOccurs="0"/>
2988     <xsd:element name="optimizeForBrowser" type="CT_OptimizeForBrowser" minOccurs="0"/>
2989     <xsd:element name="relyOnVML" type="CT_OnOff" minOccurs="0"/>
2990     <xsd:element name="allowPNG" type="CT_OnOff" minOccurs="0"/>
2991     <xsd:element name="doNotRelyOnCSS" type="CT_OnOff" minOccurs="0"/>
2992     <xsd:element name="doNotSaveAsSingleFile" type="CT_OnOff" minOccurs="0"/>
2993     <xsd:element name="doNotOrganizeInFolder" type="CT_OnOff" minOccurs="0"/>
2994     <xsd:element name="doNotUseLongFileNames" type="CT_OnOff" minOccurs="0"/>
2995     <xsd:element name="pixelsPerInch" type="CT_DecimalNumber" minOccurs="0"/>
2996     <xsd:element name="targetScreenSz" type="CT_TargetScreenSz" minOccurs="0"/>
2997     <xsd:element name="saveSmartTagsAsXml" type="CT_OnOff" minOccurs="0"/>
2998   </xsd:sequence>
2999 </xsd:complexType>
3000 <xsd:simpleType name="ST_FrameScrollbar">
3001   <xsd:restriction base="xsd:string">
3002     <xsd:enumeration value="on"/>
3003     <xsd:enumeration value="off"/>

```

```

3004         <xsd:enumeration value="auto"/>
3005     </xsd:restriction>
3006 </xsd:simpleType>
3007 <xsd:complexType name="CT_FrameScrollbar">
3008     <xsd:attribute name="val" type="ST_FrameScrollbar" use="required"/>
3009 </xsd:complexType>
3010 <xsd:complexType name="CT_OptimizeForBrowser">
3011     <xsd:complexContent>
3012         <xsd:extension base="CT_OnOff">
3013             <xsd:attribute name="target" type="s:ST_String" use="optional"/>
3014         </xsd:extension>
3015     </xsd:complexContent>
3016 </xsd:complexType>
3017 <xsd:complexType name="CT_Frame">
3018     <xsd:sequence>
3019         <xsd:element name="sz" type="CT_String" minOccurs="0"/>
3020         <xsd:element name="name" type="CT_String" minOccurs="0"/>
3021         <xsd:element name="title" type="CT_String" minOccurs="0"/>
3022         <xsd:element name="longDesc" type="CT_Rel" minOccurs="0"/>
3023         <xsd:element name="sourceFileName" type="CT_Rel" minOccurs="0"/>
3024         <xsd:element name="marW" type="CT_PixelsMeasure" minOccurs="0"/>
3025         <xsd:element name="marH" type="CT_PixelsMeasure" minOccurs="0"/>
3026         <xsd:element name="scrollbar" type="CT_FrameScrollbar" minOccurs="0"/>
3027         <xsd:element name="noResizeAllowed" type="CT_OnOff" minOccurs="0"/>
3028         <xsd:element name="linkedToFile" type="CT_OnOff" minOccurs="0"/>
3029     </xsd:sequence>
3030 </xsd:complexType>
3031 <xsd:simpleType name="ST_FrameLayout">
3032     <xsd:restriction base="xsd:string">
3033         <xsd:enumeration value="rows"/>
3034         <xsd:enumeration value="cols"/>
3035         <xsd:enumeration value="none"/>
3036     </xsd:restriction>
3037 </xsd:simpleType>
3038 <xsd:complexType name="CT_FrameLayout">
3039     <xsd:attribute name="val" type="ST_FrameLayout" use="required"/>
3040 </xsd:complexType>
3041 <xsd:complexType name="CT_FramesetSplitbar">
3042     <xsd:sequence>
3043         <xsd:element name="w" type="CT_TwipsMeasure" minOccurs="0"/>
3044         <xsd:element name="color" type="CT_Color" minOccurs="0"/>
3045         <xsd:element name="noBorder" type="CT_OnOff" minOccurs="0"/>
3046         <xsd:element name="flatBorders" type="CT_OnOff" minOccurs="0"/>
3047     </xsd:sequence>
3048 </xsd:complexType>
3049 <xsd:complexType name="CT_Frameset">
3050     <xsd:sequence>
3051         <xsd:element name="sz" type="CT_String" minOccurs="0"/>
3052         <xsd:element name="framesetSplitbar" type="CT_FramesetSplitbar" minOccurs="0"/>
3053         <xsd:element name="frameLayout" type="CT_FrameLayout" minOccurs="0"/>
3054         <xsd:element name="title" type="CT_String" minOccurs="0"/>
3055         <xsd:choice minOccurs="0" maxOccurs="unbounded">
3056             <xsd:element name="frameset" type="CT_Frameset" minOccurs="0" maxOccurs="unbounded"/>

```

```

3057         <xsd:element name="frame" type="CT_Frame" minOccurs="0" maxOccurs="unbounded"/>
3058     </xsd:choice>
3059 </xsd:sequence>
3060 </xsd:complexType>
3061 <xsd:complexType name="CT_NumPicBullet">
3062     <xsd:choice>
3063         <xsd:element name="pict" type="CT_Picture"/>
3064         <xsd:element name="drawing" type="CT_Drawing"/>
3065     </xsd:choice>
3066     <xsd:attribute name="numPicBulletId" type="ST_DecimalNumber" use="required"/>
3067 </xsd:complexType>
3068 <xsd:simpleType name="ST_LevelSuffix">
3069     <xsd:restriction base="xsd:string">
3070         <xsd:enumeration value="tab"/>
3071         <xsd:enumeration value="space"/>
3072         <xsd:enumeration value="nothing"/>
3073     </xsd:restriction>
3074 </xsd:simpleType>
3075 <xsd:complexType name="CT_LevelSuffix">
3076     <xsd:attribute name="val" type="ST_LevelSuffix" use="required"/>
3077 </xsd:complexType>
3078 <xsd:complexType name="CT_LevelText">
3079     <xsd:attribute name="val" type="s:ST_String" use="optional"/>
3080     <xsd:attribute name="null" type="s:ST_OnOff" use="optional"/>
3081 </xsd:complexType>
3082 <xsd:complexType name="CT_LvlLegacy">
3083     <xsd:attribute name="legacy" type="s:ST_OnOff" use="optional"/>
3084     <xsd:attribute name="legacySpace" type="s:ST_TwipsMeasure" use="optional"/>
3085     <xsd:attribute name="legacyIndent" type="ST_SignedTwipsMeasure" use="optional"/>
3086 </xsd:complexType>
3087 <xsd:complexType name="CT_Lvl">
3088     <xsd:sequence>
3089         <xsd:element name="start" type="CT_DecimalNumber" minOccurs="0"/>
3090         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0"/>
3091         <xsd:element name="lvlRestart" type="CT_DecimalNumber" minOccurs="0"/>
3092         <xsd:element name="pStyle" type="CT_String" minOccurs="0"/>
3093         <xsd:element name="isLgl" type="CT_OnOff" minOccurs="0"/>
3094         <xsd:element name="suff" type="CT_LevelSuffix" minOccurs="0"/>
3095         <xsd:element name="lvlText" type="CT_LevelText" minOccurs="0"/>
3096         <xsd:element name="lvlPicBulletId" type="CT_DecimalNumber" minOccurs="0"/>
3097         <xsd:element name="legacy" type="CT_LvlLegacy" minOccurs="0"/>
3098         <xsd:element name="lvlJc" type="CT_Jc" minOccurs="0"/>
3099         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0"/>
3100         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
3101     </xsd:sequence>
3102     <xsd:attribute name="ilvl" type="ST_DecimalNumber" use="required"/>
3103     <xsd:attribute name="tplc" type="ST_LongHexNumber" use="optional"/>
3104     <xsd:attribute name="tentative" type="s:ST_OnOff" use="optional"/>
3105 </xsd:complexType>
3106 <xsd:simpleType name="ST_MultiLevelType">
3107     <xsd:restriction base="xsd:string">
3108         <xsd:enumeration value="singleLevel"/>
3109         <xsd:enumeration value="multilevel"/>

```

```

3110     <xsd:enumeration value="hybridMultilevel"/>
3111   </xsd:restriction>
3112 </xsd:simpleType>
3113 <xsd:complexType name="CT_MultiLevelType">
3114   <xsd:attribute name="val" type="ST_MultiLevelType" use="required"/>
3115 </xsd:complexType>
3116 <xsd:complexType name="CT_AbstractNum">
3117   <xsd:sequence>
3118     <xsd:element name="nsid" type="CT_LongHexNumber" minOccurs="0"/>
3119     <xsd:element name="multiLevelType" type="CT_MultiLevelType" minOccurs="0"/>
3120     <xsd:element name="tmpl" type="CT_LongHexNumber" minOccurs="0"/>
3121     <xsd:element name="name" type="CT_String" minOccurs="0"/>
3122     <xsd:element name="styleLink" type="CT_String" minOccurs="0"/>
3123     <xsd:element name="numStyleLink" type="CT_String" minOccurs="0"/>
3124     <xsd:element name="lvl" type="CT_Lvl" minOccurs="0" maxOccurs="9"/>
3125   </xsd:sequence>
3126   <xsd:attribute name="abstractNumId" type="ST_DecimalNumber" use="required"/>
3127 </xsd:complexType>
3128 <xsd:complexType name="CT_NumLvl">
3129   <xsd:sequence>
3130     <xsd:element name="startOverride" type="CT_DecimalNumber" minOccurs="0"/>
3131     <xsd:element name="lvl" type="CT_Lvl" minOccurs="0" maxOccurs="1"/>
3132   </xsd:sequence>
3133   <xsd:attribute name="ilvl" type="ST_DecimalNumber" use="required"/>
3134 </xsd:complexType>
3135 <xsd:complexType name="CT_Num">
3136   <xsd:sequence>
3137     <xsd:element name="abstractNumId" type="CT_DecimalNumber" minOccurs="1"/>
3138     <xsd:element name="lvlOverride" type="CT_NumLvl" minOccurs="0" maxOccurs="9"/>
3139   </xsd:sequence>
3140   <xsd:attribute name="numId" type="ST_DecimalNumber" use="required"/>
3141 </xsd:complexType>
3142 <xsd:complexType name="CT_Numbering">
3143   <xsd:sequence>
3144     <xsd:element name="numPicBullet" type="CT_NumPicBullet" minOccurs="0"
3145       maxOccurs="unbounded"/>
3146     <xsd:element name="abstractNum" type="CT_AbstractNum" minOccurs="0"
3147       maxOccurs="unbounded"/>
3148     <xsd:element name="num" type="CT_Num" minOccurs="0" maxOccurs="unbounded"/>
3149     <xsd:element name="numIdMacAtCleanup" type="CT_DecimalNumber" minOccurs="0"/>
3150   </xsd:sequence>
3151 </xsd:complexType>
3152 <xsd:simpleType name="ST_TblStyleOverrideType">
3153   <xsd:restriction base="xsd:string">
3154     <xsd:enumeration value="wholeTable"/>
3155     <xsd:enumeration value="firstRow"/>
3156     <xsd:enumeration value="lastRow"/>
3157     <xsd:enumeration value="firstCol"/>
3158     <xsd:enumeration value="lastCol"/>
3159     <xsd:enumeration value="band1Vert"/>
3160     <xsd:enumeration value="band2Vert"/>
3161     <xsd:enumeration value="band1Horz"/>
3162     <xsd:enumeration value="band2Horz"/>

```

```

3163         <xsd:enumeration value="neCell"/>
3164         <xsd:enumeration value="nwCell"/>
3165         <xsd:enumeration value="seCell"/>
3166         <xsd:enumeration value="swCell"/>
3167     </xsd:restriction>
3168 </xsd:simpleType>
3169 <xsd:complexType name="CT_TblStylePr">
3170     <xsd:sequence>
3171         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0"/>
3172         <xsd:element name="rPr" type="CT_RPr" minOccurs="0"/>
3173         <xsd:element name="tblPr" type="CT_TblPrBase" minOccurs="0"/>
3174         <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
3175         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
3176     </xsd:sequence>
3177     <xsd:attribute name="type" type="ST_TblStyleOverrideType" use="required"/>
3178 </xsd:complexType>
3179 <xsd:simpleType name="ST_StyleType">
3180     <xsd:restriction base="xsd:string">
3181         <xsd:enumeration value="paragraph"/>
3182         <xsd:enumeration value="character"/>
3183         <xsd:enumeration value="table"/>
3184         <xsd:enumeration value="numbering"/>
3185     </xsd:restriction>
3186 </xsd:simpleType>
3187 <xsd:complexType name="CT_Style">
3188     <xsd:sequence>
3189         <xsd:element name="name" type="CT_String" minOccurs="0" maxOccurs="1"/>
3190         <xsd:element name="aliases" type="CT_String" minOccurs="0"/>
3191         <xsd:element name="basedOn" type="CT_String" minOccurs="0"/>
3192         <xsd:element name="next" type="CT_String" minOccurs="0"/>
3193         <xsd:element name="link" type="CT_String" minOccurs="0"/>
3194         <xsd:element name="autoRedefine" type="CT_OnOff" minOccurs="0"/>
3195         <xsd:element name="hidden" type="CT_OnOff" minOccurs="0"/>
3196         <xsd:element name="uiPriority" type="CT_DecimalNumber" minOccurs="0"/>
3197         <xsd:element name="semiHidden" type="CT_OnOff" minOccurs="0"/>
3198         <xsd:element name="unhideWhenUsed" type="CT_OnOff" minOccurs="0"/>
3199         <xsd:element name="qFormat" type="CT_OnOff" minOccurs="0"/>
3200         <xsd:element name="locked" type="CT_OnOff" minOccurs="0"/>
3201         <xsd:element name="personal" type="CT_OnOff" minOccurs="0"/>
3202         <xsd:element name="personalCompose" type="CT_OnOff" minOccurs="0"/>
3203         <xsd:element name="personalReply" type="CT_OnOff" minOccurs="0"/>
3204         <xsd:element name="rsid" type="CT_LongHexNumber" minOccurs="0"/>
3205         <xsd:element name="pPr" type="CT_PPrGeneral" minOccurs="0" maxOccurs="1"/>
3206         <xsd:element name="rPr" type="CT_RPr" minOccurs="0" maxOccurs="1"/>
3207         <xsd:element name="tblPr" type="CT_TblPrBase" minOccurs="0" maxOccurs="1"/>
3208         <xsd:element name="trPr" type="CT_TrPr" minOccurs="0" maxOccurs="1"/>
3209         <xsd:element name="tcPr" type="CT_TcPr" minOccurs="0" maxOccurs="1"/>
3210         <xsd:element name="tblStylePr" type="CT_TblStylePr" minOccurs="0" maxOccurs="unbounded"/>
3211     </xsd:sequence>
3212     <xsd:attribute name="type" type="ST_StyleType" use="optional"/>
3213     <xsd:attribute name="styleId" type="s:ST_String" use="optional"/>
3214     <xsd:attribute name="default" type="s:ST_OnOff" use="optional"/>
3215     <xsd:attribute name="customStyle" type="s:ST_OnOff" use="optional"/>

```

```

3216 </xsd:complexType>
3217 <xsd:complexType name="CT_LsdException">
3218   <xsd:attribute name="name" type="s:ST_String" use="required"/>
3219   <xsd:attribute name="locked" type="s:ST_OnOff"/>
3220   <xsd:attribute name="uiPriority" type="ST_DecimalNumber"/>
3221   <xsd:attribute name="semiHidden" type="s:ST_OnOff"/>
3222   <xsd:attribute name="unhideWhenUsed" type="s:ST_OnOff"/>
3223   <xsd:attribute name="qFormat" type="s:ST_OnOff"/>
3224 </xsd:complexType>
3225 <xsd:complexType name="CT_LatentStyles">
3226   <xsd:sequence>
3227     <xsd:element name="lsdException" type="CT_LsdException" minOccurs="0"
3228       maxOccurs="unbounded"/>
3229   </xsd:sequence>
3230   <xsd:attribute name="defLockedState" type="s:ST_OnOff"/>
3231   <xsd:attribute name="defUIPriority" type="ST_DecimalNumber"/>
3232   <xsd:attribute name="defSemiHidden" type="s:ST_OnOff"/>
3233   <xsd:attribute name="defUnhideWhenUsed" type="s:ST_OnOff"/>
3234   <xsd:attribute name="defQFormat" type="s:ST_OnOff"/>
3235   <xsd:attribute name="count" type="ST_DecimalNumber"/>
3236 </xsd:complexType>
3237 <xsd:complexType name="CT_Styles">
3238   <xsd:sequence>
3239     <xsd:element name="docDefaults" type="CT_DocDefaults" minOccurs="0"/>
3240     <xsd:element name="latentStyles" type="CT_LatentStyles" minOccurs="0" maxOccurs="1"/>
3241     <xsd:element name="style" type="CT_Style" minOccurs="0" maxOccurs="unbounded"/>
3242   </xsd:sequence>
3243 </xsd:complexType>
3244 <xsd:complexType name="CT_Panose">
3245   <xsd:attribute name="val" type="s:ST_Panose" use="required"/>
3246 </xsd:complexType>
3247 <xsd:simpleType name="ST_FontFamily">
3248   <xsd:restriction base="xsd:string">
3249     <xsd:enumeration value="decorative"/>
3250     <xsd:enumeration value="modern"/>
3251     <xsd:enumeration value="roman"/>
3252     <xsd:enumeration value="script"/>
3253     <xsd:enumeration value="swiss"/>
3254     <xsd:enumeration value="auto"/>
3255   </xsd:restriction>
3256 </xsd:simpleType>
3257 <xsd:complexType name="CT_FontFamily">
3258   <xsd:attribute name="val" type="ST_FontFamily" use="required"/>
3259 </xsd:complexType>
3260 <xsd:simpleType name="ST_Pitch">
3261   <xsd:restriction base="xsd:string">
3262     <xsd:enumeration value="fixed"/>
3263     <xsd:enumeration value="variable"/>
3264     <xsd:enumeration value="default"/>
3265   </xsd:restriction>
3266 </xsd:simpleType>
3267 <xsd:complexType name="CT_Pitch">
3268   <xsd:attribute name="val" type="ST_Pitch" use="required"/>

```

```

3269 </xsd:complexType>
3270 <xsd:complexType name="CT_FontSig">
3271     <xsd:attribute name="usb0" use="required" type="ST LongHexNumber"/>
3272     <xsd:attribute name="usb1" use="required" type="ST LongHexNumber"/>
3273     <xsd:attribute name="usb2" use="required" type="ST LongHexNumber"/>
3274     <xsd:attribute name="usb3" use="required" type="ST LongHexNumber"/>
3275     <xsd:attribute name="csb0" use="required" type="ST LongHexNumber"/>
3276     <xsd:attribute name="csb1" use="required" type="ST LongHexNumber"/>
3277 </xsd:complexType>
3278 <xsd:complexType name="CT_FontRel">
3279     <xsd:complexContent>
3280         <xsd:extension base="CT_Rel">
3281             <xsd:attribute name="fontKey" type="s:ST Guid"/>
3282             <xsd:attribute name="subsetting" type="s:ST OnOff"/>
3283         </xsd:extension>
3284     </xsd:complexContent>
3285 </xsd:complexType>
3286 <xsd:complexType name="CT_Font">
3287     <xsd:sequence>
3288         <xsd:element name="altName" type="CT String" minOccurs="0" maxOccurs="1"/>
3289         <xsd:element name="panose1" type="CT Panose" minOccurs="0" maxOccurs="1"/>
3290         <xsd:element name="charset" type="CT Charset" minOccurs="0" maxOccurs="1"/>
3291         <xsd:element name="family" type="CT FontFamily" minOccurs="0" maxOccurs="1"/>
3292         <xsd:element name="notTrueType" type="CT OnOff" minOccurs="0" maxOccurs="1"/>
3293         <xsd:element name="pitch" type="CT Pitch" minOccurs="0" maxOccurs="1"/>
3294         <xsd:element name="sig" type="CT FontSig" minOccurs="0" maxOccurs="1"/>
3295         <xsd:element name="embedRegular" type="CT FontRel" minOccurs="0" maxOccurs="1"/>
3296         <xsd:element name="embedBold" type="CT FontRel" minOccurs="0" maxOccurs="1"/>
3297         <xsd:element name="embedItalic" type="CT FontRel" minOccurs="0" maxOccurs="1"/>
3298         <xsd:element name="embedBoldItalic" type="CT FontRel" minOccurs="0" maxOccurs="1"/>
3299     </xsd:sequence>
3300     <xsd:attribute name="name" type="s:ST String" use="required"/>
3301 </xsd:complexType>
3302 <xsd:complexType name="CT_FontsList">
3303     <xsd:sequence>
3304         <xsd:element name="font" type="CT Font" minOccurs="0" maxOccurs="unbounded"/>
3305     </xsd:sequence>
3306 </xsd:complexType>
3307 <xsd:complexType name="CT_DivBdr">
3308     <xsd:sequence>
3309         <xsd:element name="top" type="CT Border" minOccurs="0"/>
3310         <xsd:element name="left" type="CT Border" minOccurs="0"/>
3311         <xsd:element name="bottom" type="CT Border" minOccurs="0"/>
3312         <xsd:element name="right" type="CT Border" minOccurs="0"/>
3313     </xsd:sequence>
3314 </xsd:complexType>
3315 <xsd:complexType name="CT_Div">
3316     <xsd:sequence>
3317         <xsd:element name="blockQuote" type="CT OnOff" minOccurs="0"/>
3318         <xsd:element name="bodyDiv" type="CT OnOff" minOccurs="0"/>
3319         <xsd:element name="marLeft" type="CT SignedTwipsMeasure"/>
3320         <xsd:element name="marRight" type="CT SignedTwipsMeasure"/>
3321         <xsd:element name="marTop" type="CT SignedTwipsMeasure"/>

```

```

3322     <xsd:element name="marBottom" type="CT_SignedTwipsMeasure"/>
3323     <xsd:element name="divBdr" type="CT_DivBdr" minOccurs="0"/>
3324     <xsd:element name="divsChild" type="CT_Divs" minOccurs="0" maxOccurs="unbounded"/>
3325   </xsd:sequence>
3326   <xsd:attribute name="id" type="ST_DecimalNumber" use="required"/>
3327 </xsd:complexType>
3328 <xsd:complexType name="CT_Divs">
3329   <xsd:sequence minOccurs="1" maxOccurs="unbounded">
3330     <xsd:element name="div" type="CT_Div"/>
3331   </xsd:sequence>
3332 </xsd:complexType>
3333 <xsd:complexType name="CT_TxbxContent">
3334   <xsd:group ref="EG_BlockLevelElts" minOccurs="1" maxOccurs="unbounded"/>
3335 </xsd:complexType>
3336 <xsd:element name="txbxContent" type="CT_TxbxContent"/>
3337 <xsd:group name="EG_MathContent">
3338   <xsd:choice>
3339     <xsd:element ref="m:oMathPara"/>
3340     <xsd:element ref="m:oMath"/>
3341   </xsd:choice>
3342 </xsd:group>
3343 <xsd:group name="EG_BlockLevelChunkElts">
3344   <xsd:choice>
3345     <xsd:group ref="EG_ContentBlockContent" minOccurs="0" maxOccurs="unbounded"/>
3346   </xsd:choice>
3347 </xsd:group>
3348 <xsd:group name="EG_BlockLevelElts">
3349   <xsd:choice>
3350     <xsd:group ref="EG_BlockLevelChunkElts" minOccurs="0" maxOccurs="unbounded"/>
3351     <xsd:element name="altChunk" type="CT_AltChunk" minOccurs="0" maxOccurs="unbounded"/>
3352   </xsd:choice>
3353 </xsd:group>
3354 <xsd:group name="EG_RunLevelElts">
3355   <xsd:choice>
3356     <xsd:element name="proofErr" minOccurs="0" type="CT_ProofErr"/>
3357     <xsd:element name="permStart" minOccurs="0" type="CT_PermStart"/>
3358     <xsd:element name="permEnd" minOccurs="0" type="CT_Perm"/>
3359     <xsd:group ref="EG_RangeMarkupElements" minOccurs="0" maxOccurs="unbounded"/>
3360     <xsd:element name="ins" type="CT_RunTrackChange" minOccurs="0"/>
3361     <xsd:element name="del" type="CT_RunTrackChange" minOccurs="0"/>
3362     <xsd:element name="moveFrom" type="CT_RunTrackChange"/>
3363     <xsd:element name="moveTo" type="CT_RunTrackChange"/>
3364     <xsd:group ref="EG_MathContent" minOccurs="0" maxOccurs="unbounded"/>
3365   </xsd:choice>
3366 </xsd:group>
3367 <xsd:complexType name="CT_Body">
3368   <xsd:sequence>
3369     <xsd:group ref="EG_BlockLevelElts" minOccurs="0" maxOccurs="unbounded"/>
3370     <xsd:element name="sectPr" minOccurs="0" maxOccurs="1" type="CT_SectPr"/>
3371   </xsd:sequence>
3372 </xsd:complexType>
3373 <xsd:complexType name="CT_ShapeDefaults">
3374   <xsd:choice maxOccurs="unbounded">

```



```

3375         <xsd:any processContents="lax" namespace="urn:schemas-microsoft-com:office:office"
3376             minOccurs="0" maxOccurs="unbounded"/>
3377     </xsd:choice>
3378 </xsd:complexType>
3379 <xsd:complexType name="CT_Comments">
3380     <xsd:sequence>
3381         <xsd:element name="comment" type="CT_Comment" minOccurs="0" maxOccurs="unbounded"/>
3382     </xsd:sequence>
3383 </xsd:complexType>
3384 <xsd:element name="comments" type="CT_Comments"/>
3385 <xsd:complexType name="CT_Footnotes">
3386     <xsd:sequence maxOccurs="unbounded">
3387         <xsd:element name="footnote" type="CT_FtnEdn" minOccurs="0"/>
3388     </xsd:sequence>
3389 </xsd:complexType>
3390 <xsd:element name="footnotes" type="CT_Footnotes"/>
3391 <xsd:complexType name="CT_Endnotes">
3392     <xsd:sequence maxOccurs="unbounded">
3393         <xsd:element name="endnote" type="CT_FtnEdn" minOccurs="0"/>
3394     </xsd:sequence>
3395 </xsd:complexType>
3396 <xsd:element name="endnotes" type="CT_Endnotes"/>
3397 <xsd:element name="hdr" type="CT_HdrFtr"/>
3398 <xsd:element name="ftr" type="CT_HdrFtr"/>
3399 <xsd:complexType name="CT_SmartTagType">
3400     <xsd:attribute name="namespaceuri" type="s:ST_String"/>
3401     <xsd:attribute name="name" type="s:ST_String"/>
3402     <xsd:attribute name="url" type="s:ST_String"/>
3403 </xsd:complexType>
3404 <xsd:simpleType name="ST_ThemeColor">
3405     <xsd:restriction base="xsd:string">
3406         <xsd:enumeration value="dark1"/>
3407         <xsd:enumeration value="light1"/>
3408         <xsd:enumeration value="dark2"/>
3409         <xsd:enumeration value="light2"/>
3410         <xsd:enumeration value="accent1"/>
3411         <xsd:enumeration value="accent2"/>
3412         <xsd:enumeration value="accent3"/>
3413         <xsd:enumeration value="accent4"/>
3414         <xsd:enumeration value="accent5"/>
3415         <xsd:enumeration value="accent6"/>
3416         <xsd:enumeration value="hyperlink"/>
3417         <xsd:enumeration value="followedHyperlink"/>
3418         <xsd:enumeration value="none"/>
3419         <xsd:enumeration value="background1"/>
3420         <xsd:enumeration value="text1"/>
3421         <xsd:enumeration value="background2"/>
3422         <xsd:enumeration value="text2"/>
3423     </xsd:restriction>
3424 </xsd:simpleType>
3425 <xsd:simpleType name="ST_DocPartBehavior">
3426     <xsd:restriction base="xsd:string">
3427         <xsd:enumeration value="content"/>

```

```

3428         <xsd:enumeration value="p"/>
3429         <xsd:enumeration value="pg"/>
3430     </xsd:restriction>
3431 </xsd:simpleType>
3432 <xsd:complexType name="CT_DocPartBehavior">
3433     <xsd:attribute name="val" use="required" type="ST_DocPartBehavior"/>
3434 </xsd:complexType>
3435 <xsd:complexType name="CT_DocPartBehaviors">
3436     <xsd:choice>
3437         <xsd:element name="behavior" type="CT_DocPartBehavior" maxOccurs="unbounded"/>
3438     </xsd:choice>
3439 </xsd:complexType>
3440 <xsd:simpleType name="ST_DocPartType">
3441     <xsd:restriction base="xsd:string">
3442         <xsd:enumeration value="none"/>
3443         <xsd:enumeration value="normal"/>
3444         <xsd:enumeration value="autoExp"/>
3445         <xsd:enumeration value="toolbar"/>
3446         <xsd:enumeration value="speller"/>
3447         <xsd:enumeration value="formFld"/>
3448         <xsd:enumeration value="bbPlcHdr"/>
3449     </xsd:restriction>
3450 </xsd:simpleType>
3451 <xsd:complexType name="CT_DocPartType">
3452     <xsd:attribute name="val" use="required" type="ST_DocPartType"/>
3453 </xsd:complexType>
3454 <xsd:complexType name="CT_DocPartTypes">
3455     <xsd:choice>
3456         <xsd:element name="type" type="CT_DocPartType" maxOccurs="unbounded"/>
3457     </xsd:choice>
3458     <xsd:attribute name="all" type="s:ST_OnOff" use="optional"/>
3459 </xsd:complexType>
3460 <xsd:simpleType name="ST_DocPartGallery">
3461     <xsd:restriction base="xsd:string">
3462         <xsd:enumeration value="placeholder"/>
3463         <xsd:enumeration value="any"/>
3464         <xsd:enumeration value="default"/>
3465         <xsd:enumeration value="docParts"/>
3466         <xsd:enumeration value="coverPg"/>
3467         <xsd:enumeration value="eq"/>
3468         <xsd:enumeration value="ftrs"/>
3469         <xsd:enumeration value="hdrs"/>
3470         <xsd:enumeration value="pgNum"/>
3471         <xsd:enumeration value="tbls"/>
3472         <xsd:enumeration value="watermarks"/>
3473         <xsd:enumeration value="autoTxt"/>
3474         <xsd:enumeration value="txtBox"/>
3475         <xsd:enumeration value="pgNumT"/>
3476         <xsd:enumeration value="pgNumB"/>
3477         <xsd:enumeration value="pgNumMargins"/>
3478         <xsd:enumeration value="tblOfContents"/>
3479         <xsd:enumeration value="bib"/>
3480         <xsd:enumeration value="custQuickParts"/>

```

```

3481     <xsd:enumeration value="custCoverPg"/>
3482     <xsd:enumeration value="custEq"/>
3483     <xsd:enumeration value="custFtrs"/>
3484     <xsd:enumeration value="custHdrs"/>
3485     <xsd:enumeration value="custPgNum"/>
3486     <xsd:enumeration value="custTbls"/>
3487     <xsd:enumeration value="custWatermarks"/>
3488     <xsd:enumeration value="custAutoTxt"/>
3489     <xsd:enumeration value="custTxtBox"/>
3490     <xsd:enumeration value="custPgNumT"/>
3491     <xsd:enumeration value="custPgNumB"/>
3492     <xsd:enumeration value="custPgNumMargins"/>
3493     <xsd:enumeration value="custTblOfContents"/>
3494     <xsd:enumeration value="custBib"/>
3495     <xsd:enumeration value="custom1"/>
3496     <xsd:enumeration value="custom2"/>
3497     <xsd:enumeration value="custom3"/>
3498     <xsd:enumeration value="custom4"/>
3499     <xsd:enumeration value="custom5"/>
3500 </xsd:restriction>
3501 </xsd:simpleType>
3502 <xsd:complexType name="CT_DocPartGallery">
3503     <xsd:attribute name="val" type="ST_DocPartGallery" use="required"/>
3504 </xsd:complexType>
3505 <xsd:complexType name="CT_DocPartCategory">
3506     <xsd:sequence>
3507         <xsd:element name="name" type="CT_String" minOccurs="1" maxOccurs="1"/>
3508         <xsd:element name="gallery" type="CT_DocPartGallery" minOccurs="1" maxOccurs="1"/>
3509     </xsd:sequence>
3510 </xsd:complexType>
3511 <xsd:complexType name="CT_DocPartName">
3512     <xsd:attribute name="val" type="s:ST_String" use="required"/>
3513     <xsd:attribute name="decorated" type="s:ST_OnOff" use="optional"/>
3514 </xsd:complexType>
3515 <xsd:complexType name="CT_DocPartPr">
3516     <xsd:choice maxOccurs="unbounded">
3517         <xsd:element name="name" type="CT_DocPartName" minOccurs="1"/>
3518         <xsd:element name="style" type="CT_String"/>
3519         <xsd:element name="category" type="CT_DocPartCategory"/>
3520         <xsd:element name="types" type="CT_DocPartTypes"/>
3521         <xsd:element name="behaviors" type="CT_DocPartBehaviors"/>
3522         <xsd:element name="description" type="CT_String"/>
3523         <xsd:element name="guid" type="CT_Guid"/>
3524     </xsd:choice>
3525 </xsd:complexType>
3526 <xsd:complexType name="CT_DocPart">
3527     <xsd:sequence>
3528         <xsd:element name="docPartPr" type="CT_DocPartPr" minOccurs="0"/>
3529         <xsd:element name="docPartBody" type="CT_Body" minOccurs="0"/>
3530     </xsd:sequence>
3531 </xsd:complexType>
3532 <xsd:complexType name="CT_DocParts">
3533     <xsd:choice>

```

```

3534         <xsd:element name="docPart" type="CT_DocPart" minOccurs="1" maxOccurs="unbounded"/>
3535     </xsd:choice>
3536 </xsd:complexType>
3537 <xsd:element name="settings" type="CT_Settings"/>
3538 <xsd:element name="webSettings" type="CT_WebSettings"/>
3539 <xsd:element name="fonts" type="CT_FontsList"/>
3540 <xsd:element name="numbering" type="CT_Numbering"/>
3541 <xsd:element name="styles" type="CT_Styles"/>
3542 <xsd:simpleType name="ST_CaptionPos">
3543     <xsd:restriction base="xsd:string">
3544         <xsd:enumeration value="above"/>
3545         <xsd:enumeration value="below"/>
3546         <xsd:enumeration value="left"/>
3547         <xsd:enumeration value="right"/>
3548     </xsd:restriction>
3549 </xsd:simpleType>
3550 <xsd:complexType name="CT_Caption">
3551     <xsd:attribute name="name" type="s:ST_String" use="required"/>
3552     <xsd:attribute name="pos" type="ST_CaptionPos" use="optional"/>
3553     <xsd:attribute name="chapNum" type="s:ST_OnOff" use="optional"/>
3554     <xsd:attribute name="heading" type="ST_DecimalNumber" use="optional"/>
3555     <xsd:attribute name="noLabel" type="s:ST_OnOff" use="optional"/>
3556     <xsd:attribute name="numFmt" type="ST_NumberFormat" use="optional"/>
3557     <xsd:attribute name="sep" type="ST_ChapterSep" use="optional"/>
3558 </xsd:complexType>
3559 <xsd:complexType name="CT_AutoCaption">
3560     <xsd:attribute name="name" type="s:ST_String" use="required"/>
3561     <xsd:attribute name="caption" type="s:ST_String" use="required"/>
3562 </xsd:complexType>
3563 <xsd:complexType name="CT_AutoCaptions">
3564     <xsd:sequence>
3565         <xsd:element name="autoCaption" type="CT_AutoCaption" minOccurs="1"
3566             maxOccurs="unbounded"/>
3567     </xsd:sequence>
3568 </xsd:complexType>
3569 <xsd:complexType name="CT_Captions">
3570     <xsd:sequence>
3571         <xsd:element name="caption" type="CT_Caption" minOccurs="1" maxOccurs="unbounded"/>
3572         <xsd:element name="autoCaptions" type="CT_AutoCaptions" minOccurs="0" maxOccurs="1"/>
3573     </xsd:sequence>
3574 </xsd:complexType>
3575 <xsd:complexType name="CT_DocumentBase">
3576     <xsd:sequence>
3577         <xsd:element name="background" type="CT_Background" minOccurs="0"/>
3578     </xsd:sequence>
3579 </xsd:complexType>
3580 <xsd:complexType name="CT_Document">
3581     <xsd:complexContent>
3582         <xsd:extension base="CT_DocumentBase">
3583             <xsd:sequence>
3584                 <xsd:element name="body" type="CT_Body" minOccurs="0" maxOccurs="1"/>
3585             </xsd:sequence>
3586             <xsd:attribute name="conformance" type="s:ST_ConformanceClass"/>

```

```

3587         </xsd:extension>
3588     </xsd:complexContent>
3589 </xsd:complexType>
3590 <xsd:complexType name="CT_GlossaryDocument">
3591     <xsd:complexContent>
3592         <xsd:extension base="CT_DocumentBase">
3593             <xsd:sequence>
3594                 <xsd:element name="docParts" type="CT_DocParts" minOccurs="0"/>
3595             </xsd:sequence>
3596         </xsd:extension>
3597     </xsd:complexContent>
3598 </xsd:complexType>
3599 <xsd:element name="document" type="CT_Document"/>
3600 <xsd:element name="glossaryDocument" type="CT_GlossaryDocument"/>
3601 </xsd:schema>

```

A.2 SpreadsheetML

```

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 id="s" namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11      schemaLocation="shared-commonSimpleTypes.xsd"/>
12    <xsd:import id="xdr"
13      namespace="http://schemas.openxmlformats.org/drawingml/2006/spreadsheetDrawing"
14      schemaLocation="dml-spreadsheetDrawing.xsd"/>
15    <xsd:complexType name="CT_AutoFilter">
16      <xsd:sequence>
17        <xsd:element name="filterColumn" minOccurs="0" maxOccurs="unbounded"
18          type="CT_FilterColumn"/>
19        <xsd:element name="sortState" minOccurs="0" maxOccurs="1" type="CT_SortState"/>
20        <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
21      </xsd:sequence>
22      <xsd:attribute name="ref" type="ST_Ref"/>
23    </xsd:complexType>
24    <xsd:complexType name="CT_FilterColumn">
25      <xsd:choice minOccurs="0" maxOccurs="1">
26        <xsd:element name="filters" type="CT_Filters" minOccurs="0" maxOccurs="1"/>
27        <xsd:element name="top10" type="CT_Top10" minOccurs="0" maxOccurs="1"/>
28        <xsd:element name="customFilters" type="CT_CustomFilters" minOccurs="0" maxOccurs="1"/>
29        <xsd:element name="dynamicFilter" type="CT_DynamicFilter" minOccurs="0" maxOccurs="1"/>
30        <xsd:element name="colorFilter" type="CT_ColorFilter" minOccurs="0" maxOccurs="1"/>
31        <xsd:element name="iconFilter" minOccurs="0" maxOccurs="1" type="CT_IconFilter"/>
32        <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
33      </xsd:choice>
34      <xsd:attribute name="colId" type="xsd:unsignedInt" use="required"/>
35      <xsd:attribute name="hiddenButton" type="xsd:boolean" use="optional" default="false"/>

```

```

36     <xsd:attribute name="showButton" type="xsd:boolean" use="optional" default="true"/>
37 </xsd:complexType>
38 <xsd:complexType name="CT_Filters">
39     <xsd:sequence>
40         <xsd:element name="filter" type="CT_Filter" minOccurs="0" maxOccurs="unbounded"/>
41         <xsd:element name="dateGroupItem" type="CT_DateGroupItem" minOccurs="0"
42             maxOccurs="unbounded"/>
43     </xsd:sequence>
44     <xsd:attribute name="blank" type="xsd:boolean" use="optional" default="false"/>
45     <xsd:attribute name="calendarType" type="s:ST_CalendarType" use="optional" default="none"/>
46 </xsd:complexType>
47 <xsd:complexType name="CT_Filter">
48     <xsd:attribute name="val" type="s:ST_Xstring"/>
49 </xsd:complexType>
50 <xsd:complexType name="CT_CustomFilters">
51     <xsd:sequence>
52         <xsd:element name="customFilter" type="CT_CustomFilter" minOccurs="1" maxOccurs="2"/>
53     </xsd:sequence>
54     <xsd:attribute name="and" type="xsd:boolean" use="optional" default="false"/>
55 </xsd:complexType>
56 <xsd:complexType name="CT_CustomFilter">
57     <xsd:attribute name="operator" type="ST_FilterOperator" default="equal" use="optional"/>
58     <xsd:attribute name="val" type="s:ST_Xstring"/>
59 </xsd:complexType>
60 <xsd:complexType name="CT_Top10">
61     <xsd:attribute name="top" type="xsd:boolean" use="optional" default="true"/>
62     <xsd:attribute name="percent" type="xsd:boolean" use="optional" default="false"/>
63     <xsd:attribute name="val" type="xsd:double" use="required"/>
64     <xsd:attribute name="filterVal" type="xsd:double" use="optional"/>
65 </xsd:complexType>
66 <xsd:complexType name="CT_ColorFilter">
67     <xsd:attribute name="dxfId" type="ST_DxfId" use="optional"/>
68     <xsd:attribute name="cellColor" type="xsd:boolean" use="optional" default="true"/>
69 </xsd:complexType>
70 <xsd:complexType name="CT_IconFilter">
71     <xsd:attribute name="iconSet" type="ST_IconSetType" use="required"/>
72     <xsd:attribute name="iconId" type="xsd:unsignedInt" use="optional"/>
73 </xsd:complexType>
74 <xsd:simpleType name="ST_FilterOperator">
75     <xsd:restriction base="xsd:string">
76         <xsd:enumeration value="equal"/>
77         <xsd:enumeration value="lessThan"/>
78         <xsd:enumeration value="lessThanOrEqual"/>
79         <xsd:enumeration value="notEqual"/>
80         <xsd:enumeration value="greaterThanOrEqual"/>
81         <xsd:enumeration value="greaterThan"/>
82     </xsd:restriction>
83 </xsd:simpleType>
84 <xsd:complexType name="CT_DynamicFilter">
85     <xsd:attribute name="type" type="ST_DynamicFilterType" use="required"/>
86     <xsd:attribute name="val" type="xsd:double" use="optional"/>
87     <xsd:attribute name="valIso" type="xsd:dateTime" use="optional"/>
88     <xsd:attribute name="maxVal" type="xsd:double" use="optional"/>

```

```

89     <xsd:attribute name="maxValIso" type="xsd:dateTime" use="optional"/>
90 </xsd:complexType>
91 <xsd:simpleType name="ST_DynamicFilterType">
92     <xsd:restriction base="xsd:string">
93         <xsd:enumeration value="null"/>
94         <xsd:enumeration value="aboveAverage"/>
95         <xsd:enumeration value="belowAverage"/>
96         <xsd:enumeration value="tomorrow"/>
97         <xsd:enumeration value="today"/>
98         <xsd:enumeration value="yesterday"/>
99         <xsd:enumeration value="nextWeek"/>
100        <xsd:enumeration value="thisWeek"/>
101        <xsd:enumeration value="lastWeek"/>
102        <xsd:enumeration value="nextMonth"/>
103        <xsd:enumeration value="thisMonth"/>
104        <xsd:enumeration value="lastMonth"/>
105        <xsd:enumeration value="nextQuarter"/>
106        <xsd:enumeration value="thisQuarter"/>
107        <xsd:enumeration value="lastQuarter"/>
108        <xsd:enumeration value="nextYear"/>
109        <xsd:enumeration value="thisYear"/>
110        <xsd:enumeration value="lastYear"/>
111        <xsd:enumeration value="yearToDate"/>
112        <xsd:enumeration value="Q1"/>
113        <xsd:enumeration value="Q2"/>
114        <xsd:enumeration value="Q3"/>
115        <xsd:enumeration value="Q4"/>
116        <xsd:enumeration value="M1"/>
117        <xsd:enumeration value="M2"/>
118        <xsd:enumeration value="M3"/>
119        <xsd:enumeration value="M4"/>
120        <xsd:enumeration value="M5"/>
121        <xsd:enumeration value="M6"/>
122        <xsd:enumeration value="M7"/>
123        <xsd:enumeration value="M8"/>
124        <xsd:enumeration value="M9"/>
125        <xsd:enumeration value="M10"/>
126        <xsd:enumeration value="M11"/>
127        <xsd:enumeration value="M12"/>
128     </xsd:restriction>
129 </xsd:simpleType>
130 <xsd:simpleType name="ST_IconSetType">
131     <xsd:restriction base="xsd:string">
132         <xsd:enumeration value="3Arrows"/>
133         <xsd:enumeration value="3ArrowsGray"/>
134         <xsd:enumeration value="3Flags"/>
135         <xsd:enumeration value="3TrafficLights1"/>
136         <xsd:enumeration value="3TrafficLights2"/>
137         <xsd:enumeration value="3Signs"/>
138         <xsd:enumeration value="3Symbols"/>
139         <xsd:enumeration value="3Symbols2"/>
140         <xsd:enumeration value="4Arrows"/>
141         <xsd:enumeration value="4ArrowsGray"/>

```

```

142     <xsd:enumeration value="4RedToBlack"/>
143     <xsd:enumeration value="4Rating"/>
144     <xsd:enumeration value="4TrafficLights"/>
145     <xsd:enumeration value="5Arrows"/>
146     <xsd:enumeration value="5ArrowsGray"/>
147     <xsd:enumeration value="5Rating"/>
148     <xsd:enumeration value="5Quarters"/>
149   </xsd:restriction>
150 </xsd:simpleType>
151 <xsd:complexType name="CT_SortState">
152   <xsd:sequence>
153     <xsd:element name="sortCondition" minOccurs="0" maxOccurs="64" type="CT_SortCondition"/>
154     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
155   </xsd:sequence>
156   <xsd:attribute name="columnSort" type="xsd:boolean" use="optional" default="false"/>
157   <xsd:attribute name="caseSensitive" type="xsd:boolean" use="optional" default="false"/>
158   <xsd:attribute name="sortMethod" type="ST_SortMethod" use="optional" default="none"/>
159   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
160 </xsd:complexType>
161 <xsd:complexType name="CT_SortCondition">
162   <xsd:attribute name="descending" type="xsd:boolean" use="optional" default="false"/>
163   <xsd:attribute name="sortBy" type="ST_SortBy" use="optional" default="value"/>
164   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
165   <xsd:attribute name="customList" type="s:ST_Xstring" use="optional"/>
166   <xsd:attribute name="dxfid" type="ST_DxfId" use="optional"/>
167   <xsd:attribute name="iconSet" type="ST_IconSetType" use="optional" default="3Arrows"/>
168   <xsd:attribute name="iconId" type="xsd:unsignedInt" use="optional"/>
169 </xsd:complexType>
170 <xsd:simpleType name="ST_SortBy">
171   <xsd:restriction base="xsd:string">
172     <xsd:enumeration value="value"/>
173     <xsd:enumeration value="cellColor"/>
174     <xsd:enumeration value="fontColor"/>
175     <xsd:enumeration value="icon"/>
176   </xsd:restriction>
177 </xsd:simpleType>
178 <xsd:simpleType name="ST_SortMethod">
179   <xsd:restriction base="xsd:string">
180     <xsd:enumeration value="stroke"/>
181     <xsd:enumeration value="pinYin"/>
182     <xsd:enumeration value="none"/>
183   </xsd:restriction>
184 </xsd:simpleType>
185 <xsd:complexType name="CT_DateGroupItem">
186   <xsd:attribute name="year" type="xsd:unsignedShort" use="required"/>
187   <xsd:attribute name="month" type="xsd:unsignedShort" use="optional"/>
188   <xsd:attribute name="day" type="xsd:unsignedShort" use="optional"/>
189   <xsd:attribute name="hour" type="xsd:unsignedShort" use="optional"/>
190   <xsd:attribute name="minute" type="xsd:unsignedShort" use="optional"/>
191   <xsd:attribute name="second" type="xsd:unsignedShort" use="optional"/>
192   <xsd:attribute name="dateTimeGrouping" type="ST_DateTimeGrouping" use="required"/>
193 </xsd:complexType>
194 <xsd:simpleType name="ST_DateTimeGrouping">

```



```

195     <xsd:restriction base="xsd:string">
196         <xsd:enumeration value="year"/>
197         <xsd:enumeration value="month"/>
198         <xsd:enumeration value="day"/>
199         <xsd:enumeration value="hour"/>
200         <xsd:enumeration value="minute"/>
201         <xsd:enumeration value="second"/>
202     </xsd:restriction>
203 </xsd:simpleType>
204 <xsd:simpleType name="ST_CellRef">
205     <xsd:restriction base="xsd:string"/>
206 </xsd:simpleType>
207 <xsd:simpleType name="ST_Ref">
208     <xsd:restriction base="xsd:string"/>
209 </xsd:simpleType>
210 <xsd:simpleType name="ST_RefA">
211     <xsd:restriction base="xsd:string"/>
212 </xsd:simpleType>
213 <xsd:simpleType name="ST_Sqref">
214     <xsd:list itemType="ST_Ref"/>
215 </xsd:simpleType>
216 <xsd:simpleType name="ST_Formula">
217     <xsd:restriction base="s:ST_Xstring"/>
218 </xsd:simpleType>
219 <xsd:simpleType name="ST_UnsignedIntHex">
220     <xsd:restriction base="xsd:hexBinary">
221         <xsd:length value="4"/>
222     </xsd:restriction>
223 </xsd:simpleType>
224 <xsd:simpleType name="ST_UnsignedShortHex">
225     <xsd:restriction base="xsd:hexBinary">
226         <xsd:length value="2"/>
227     </xsd:restriction>
228 </xsd:simpleType>
229 <xsd:complexType name="CT_XStringElement">
230     <xsd:attribute name="v" type="s:ST_Xstring" use="required"/>
231 </xsd:complexType>
232 <xsd:complexType name="CT_Extension">
233     <xsd:sequence>
234         <xsd:any processContents="lax"/>
235     </xsd:sequence>
236     <xsd:attribute name="uri" type="xsd:token"/>
237 </xsd:complexType>
238 <xsd:complexType name="CT_ObjectAnchor">
239     <xsd:sequence>
240         <xsd:element ref="xdr:from" minOccurs="1" maxOccurs="1"/>
241         <xsd:element ref="xdr:to" minOccurs="1" maxOccurs="1"/>
242     </xsd:sequence>
243     <xsd:attribute name="moveWithCells" type="xsd:boolean" use="optional" default="false"/>
244     <xsd:attribute name="sizeWithCells" type="xsd:boolean" use="optional" default="false"/>
245     <xsd:attribute name="z-order" type="xsd:unsignedInt" use="optional"/>
246 </xsd:complexType>
247 <xsd:group name="EG_ExtensionList">

```

```

248     <xsd:sequence>
249         <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
250     </xsd:sequence>
251 </xsd:group>
252 <xsd:complexType name="CT_ExtensionList">
253     <xsd:sequence>
254         <xsd:group ref="EG_ExtensionList" minOccurs="0"/>
255     </xsd:sequence>
256 </xsd:complexType>
257 <xsd:element name="calcChain" type="CT_CalcChain"/>
258 <xsd:complexType name="CT_CalcChain">
259     <xsd:sequence>
260         <xsd:element name="c" type="CT_CalcCell" minOccurs="1" maxOccurs="unbounded"/>
261         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
262     </xsd:sequence>
263 </xsd:complexType>
264 <xsd:complexType name="CT_CalcCell">
265     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
266     <xsd:attribute name="i" type="xsd:int" use="optional" default="0"/>
267     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
268     <xsd:attribute name="l" type="xsd:boolean" use="optional" default="false"/>
269     <xsd:attribute name="t" type="xsd:boolean" use="optional" default="false"/>
270     <xsd:attribute name="a" type="xsd:boolean" use="optional" default="false"/>
271 </xsd:complexType>
272 <xsd:element name="comments" type="CT_Comments"/>
273 <xsd:complexType name="CT_Comments">
274     <xsd:sequence>
275         <xsd:element name="authors" type="CT_Authors" minOccurs="1" maxOccurs="1"/>
276         <xsd:element name="commentList" type="CT_CommentList" minOccurs="1" maxOccurs="1"/>
277         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
278     </xsd:sequence>
279 </xsd:complexType>
280 <xsd:complexType name="CT_Authors">
281     <xsd:sequence>
282         <xsd:element name="author" type="s:ST_Xstring" minOccurs="0" maxOccurs="unbounded"/>
283     </xsd:sequence>
284 </xsd:complexType>
285 <xsd:complexType name="CT_CommentList">
286     <xsd:sequence>
287         <xsd:element name="comment" type="CT_Comment" minOccurs="0" maxOccurs="unbounded"/>
288     </xsd:sequence>
289 </xsd:complexType>
290 <xsd:complexType name="CT_Comment">
291     <xsd:sequence>
292         <xsd:element name="text" type="CT_Rst" minOccurs="1" maxOccurs="1"/>
293         <xsd:element name="commentPr" type="CT_CommentPr" minOccurs="0" maxOccurs="1"/>
294     </xsd:sequence>
295     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
296     <xsd:attribute name="authorId" type="xsd:unsignedInt" use="required"/>
297     <xsd:attribute name="guid" type="s:ST_Guid" use="optional"/>
298     <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="optional"/>
299 </xsd:complexType>
300 <xsd:complexType name="CT_CommentPr">

```

```

301     <xsd:sequence>
302         <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
303     </xsd:sequence>
304     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
305     <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
306     <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
307     <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
308     <xsd:attribute name="uiObject" type="xsd:boolean" use="optional" default="false"/>
309     <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
310     <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
311     <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
312     <xsd:attribute name="textHAlign" type="ST_TextHAlign" use="optional" default="left"/>
313     <xsd:attribute name="textVAlign" type="ST_TextVAlign" use="optional" default="top"/>
314     <xsd:attribute name="lockText" type="xsd:boolean" use="optional" default="true"/>
315     <xsd:attribute name="justLastX" type="xsd:boolean" use="optional" default="false"/>
316     <xsd:attribute name="autoScale" type="xsd:boolean" use="optional" default="false"/>
317     <xsd:attribute name="rowHidden" type="xsd:boolean" use="optional" default="false"/>
318     <xsd:attribute name="colHidden" type="xsd:boolean" use="optional" default="false"/>
319 </xsd:complexType>
320 <xsd:simpleType name="ST_TextHAlign">
321     <xsd:restriction base="xsd:string">
322         <xsd:enumeration value="left"/>
323         <xsd:enumeration value="center"/>
324         <xsd:enumeration value="right"/>
325         <xsd:enumeration value="justify"/>
326         <xsd:enumeration value="distributed"/>
327     </xsd:restriction>
328 </xsd:simpleType>
329 <xsd:simpleType name="ST_TextVAlign">
330     <xsd:restriction base="xsd:string">
331         <xsd:enumeration value="top"/>
332         <xsd:enumeration value="center"/>
333         <xsd:enumeration value="bottom"/>
334         <xsd:enumeration value="justify"/>
335         <xsd:enumeration value="distributed"/>
336     </xsd:restriction>
337 </xsd:simpleType>
338 <xsd:element name="MapInfo" type="CT_MapInfo"/>
339 <xsd:complexType name="CT_MapInfo">
340     <xsd:sequence>
341         <xsd:element name="Schema" type="CT_Schema" minOccurs="1" maxOccurs="unbounded"/>
342         <xsd:element name="Map" type="CT_Map" minOccurs="1" maxOccurs="unbounded"/>
343     </xsd:sequence>
344     <xsd:attribute name="SelectionNamespaces" type="xsd:string" use="required"/>
345 </xsd:complexType>
346 <xsd:complexType name="CT_Schema" mixed="true">
347     <xsd:sequence>
348         <xsd:any/>
349     </xsd:sequence>
350     <xsd:attribute name="ID" type="xsd:string" use="required"/>
351     <xsd:attribute name="SchemaRef" type="xsd:string" use="optional"/>
352     <xsd:attribute name="Namespace" type="xsd:string" use="optional"/>
353     <xsd:attribute name="SchemaLanguage" type="xsd:token" use="optional"/>

```

```

354 </xsd:complexType>
355 <xsd:complexType name="CT_Map">
356   <xsd:sequence>
357     <xsd:element name="DataBinding" type="CT_DataBinding" minOccurs="0" maxOccurs="1"/>
358   </xsd:sequence>
359   <xsd:attribute name="ID" type="xsd:unsignedInt" use="required"/>
360   <xsd:attribute name="Name" type="xsd:string" use="required"/>
361   <xsd:attribute name="RootElement" type="xsd:string" use="required"/>
362   <xsd:attribute name="SchemaID" type="xsd:string" use="required"/>
363   <xsd:attribute name="ShowImportExportValidationErrors" type="xsd:boolean" use="required"/>
364   <xsd:attribute name="AutoFit" type="xsd:boolean" use="required"/>
365   <xsd:attribute name="Append" type="xsd:boolean" use="required"/>
366   <xsd:attribute name="PreserveSortAFLayout" type="xsd:boolean" use="required"/>
367   <xsd:attribute name="PreserveFormat" type="xsd:boolean" use="required"/>
368 </xsd:complexType>
369 <xsd:complexType name="CT_DataBinding">
370   <xsd:sequence>
371     <xsd:any/>
372   </xsd:sequence>
373   <xsd:attribute name="DataBindingName" type="xsd:string" use="optional"/>
374   <xsd:attribute name="FileBinding" type="xsd:boolean" use="optional"/>
375   <xsd:attribute name="ConnectionID" type="xsd:unsignedInt" use="optional"/>
376   <xsd:attribute name="FileBindingName" type="xsd:string" use="optional"/>
377   <xsd:attribute name="DataBindingLoadMode" type="xsd:unsignedInt" use="required"/>
378 </xsd:complexType>
379 <xsd:element name="connections" type="CT_Connections"/>
380 <xsd:complexType name="CT_Connections">
381   <xsd:sequence>
382     <xsd:element name="connection" minOccurs="1" maxOccurs="unbounded" type="CT_Connection"/>
383   </xsd:sequence>
384 </xsd:complexType>
385 <xsd:complexType name="CT_Connection">
386   <xsd:sequence>
387     <xsd:element name="dbPr" minOccurs="0" maxOccurs="1" type="CT_DbPr"/>
388     <xsd:element name="olapPr" minOccurs="0" maxOccurs="1" type="CT_OlapPr"/>
389     <xsd:element name="webPr" minOccurs="0" maxOccurs="1" type="CT_WebPr"/>
390     <xsd:element name="textPr" minOccurs="0" maxOccurs="1" type="CT_TextPr"/>
391     <xsd:element name="parameters" minOccurs="0" maxOccurs="1" type="CT_Parameters"/>
392     <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
393   </xsd:sequence>
394   <xsd:attribute name="id" use="required" type="xsd:unsignedInt"/>
395   <xsd:attribute name="sourceFile" use="optional" type="s:ST_Xstring"/>
396   <xsd:attribute name="odcFile" use="optional" type="s:ST_Xstring"/>
397   <xsd:attribute name="keepAlive" use="optional" type="xsd:boolean" default="false"/>
398   <xsd:attribute name="interval" use="optional" type="xsd:unsignedInt" default="0"/>
399   <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
400   <xsd:attribute name="description" use="optional" type="s:ST_Xstring"/>
401   <xsd:attribute name="type" use="optional" type="xsd:unsignedInt"/>
402   <xsd:attribute name="reconnectionMethod" use="optional" type="xsd:unsignedInt" default="1"/>
403   <xsd:attribute name="refreshedVersion" use="required" type="xsd:unsignedByte"/>
404   <xsd:attribute name="minRefreshableVersion" use="optional" type="xsd:unsignedByte"
405     default="0"/>
406   <xsd:attribute name="savePassword" use="optional" type="xsd:boolean" default="false"/>

```

```

407 <xsd:attribute name="new" use="optional" type="xsd:boolean" default="false"/>
408 <xsd:attribute name="deleted" use="optional" type="xsd:boolean" default="false"/>
409 <xsd:attribute name="onlyUseConnectionFile" use="optional" type="xsd:boolean"
410     default="false"/>
411 <xsd:attribute name="background" use="optional" type="xsd:boolean" default="false"/>
412 <xsd:attribute name="refreshOnLoad" use="optional" type="xsd:boolean" default="false"/>
413 <xsd:attribute name="saveData" use="optional" type="xsd:boolean" default="false"/>
414 <xsd:attribute name="credentials" use="optional" type="ST_CredMethod" default="integrated"/>
415 <xsd:attribute name="singleSignOnId" use="optional" type="s:ST_Xstring"/>
416 </xsd:complexType>
417 <xsd:simpleType name="ST_CredMethod">
418     <xsd:restriction base="xsd:string">
419         <xsd:enumeration value="integrated"/>
420         <xsd:enumeration value="none"/>
421         <xsd:enumeration value="stored"/>
422         <xsd:enumeration value="prompt"/>
423     </xsd:restriction>
424 </xsd:simpleType>
425 <xsd:complexType name="CT_DbPr">
426     <xsd:attribute name="connection" use="required" type="s:ST_Xstring"/>
427     <xsd:attribute name="command" use="optional" type="s:ST_Xstring"/>
428     <xsd:attribute name="serverCommand" use="optional" type="s:ST_Xstring"/>
429     <xsd:attribute name="commandType" use="optional" type="xsd:unsignedInt" default="2"/>
430 </xsd:complexType>
431 <xsd:complexType name="CT_OlapPr">
432     <xsd:attribute name="local" use="optional" type="xsd:boolean" default="false"/>
433     <xsd:attribute name="localConnection" use="optional" type="s:ST_Xstring"/>
434     <xsd:attribute name="localRefresh" use="optional" type="xsd:boolean" default="true"/>
435     <xsd:attribute name="sendLocale" use="optional" type="xsd:boolean" default="false"/>
436     <xsd:attribute name="rowDrillCount" use="optional" type="xsd:unsignedInt"/>
437     <xsd:attribute name="serverFill" use="optional" type="xsd:boolean" default="true"/>
438     <xsd:attribute name="serverNumberFormat" use="optional" type="xsd:boolean" default="true"/>
439     <xsd:attribute name="serverFont" use="optional" type="xsd:boolean" default="true"/>
440     <xsd:attribute name="serverFontColor" use="optional" type="xsd:boolean" default="true"/>
441 </xsd:complexType>
442 <xsd:complexType name="CT_WebPr">
443     <xsd:sequence>
444         <xsd:element name="tables" minOccurs="0" maxOccurs="1" type="CT_Tables"/>
445     </xsd:sequence>
446     <xsd:attribute name="xml" use="optional" type="xsd:boolean" default="false"/>
447     <xsd:attribute name="sourceData" use="optional" type="xsd:boolean" default="false"/>
448     <xsd:attribute name="parsePre" use="optional" type="xsd:boolean" default="false"/>
449     <xsd:attribute name="consecutive" use="optional" type="xsd:boolean" default="false"/>
450     <xsd:attribute name="firstRow" use="optional" type="xsd:boolean" default="false"/>
451     <xsd:attribute name="xl97" use="optional" type="xsd:boolean" default="false"/>
452     <xsd:attribute name="textDates" use="optional" type="xsd:boolean" default="false"/>
453     <xsd:attribute name="xl2000" use="optional" type="xsd:boolean" default="false"/>
454     <xsd:attribute name="url" use="optional" type="s:ST_Xstring"/>
455     <xsd:attribute name="post" use="optional" type="s:ST_Xstring"/>
456     <xsd:attribute name="htmlTables" use="optional" type="xsd:boolean" default="false"/>
457     <xsd:attribute name="htmlFormat" use="optional" type="ST_HtmlFmt" default="none"/>
458     <xsd:attribute name="editPage" use="optional" type="s:ST_Xstring"/>
459 </xsd:complexType>

```

```

460 <xsd:simpleType name="ST_HtmlFmt">
461   <xsd:restriction base="xsd:string">
462     <xsd:enumeration value="none"/>
463     <xsd:enumeration value="rtf"/>
464     <xsd:enumeration value="all"/>
465   </xsd:restriction>
466 </xsd:simpleType>
467 <xsd:complexType name="CT_Parameters">
468   <xsd:sequence>
469     <xsd:element name="parameter" minOccurs="1" maxOccurs="unbounded" type="CT_Parameter"/>
470   </xsd:sequence>
471   <xsd:attribute name="count" use="optional" type="xsd:unsignedInt"/>
472 </xsd:complexType>
473 <xsd:complexType name="CT_Parameter">
474   <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
475   <xsd:attribute name="sqlType" use="optional" type="xsd:int" default="0"/>
476   <xsd:attribute name="parameterType" use="optional" type="ST_ParameterType" default="prompt"/>
477   <xsd:attribute name="refreshOnChange" use="optional" type="xsd:boolean" default="false"/>
478   <xsd:attribute name="prompt" use="optional" type="s:ST_Xstring"/>
479   <xsd:attribute name="boolean" use="optional" type="xsd:boolean"/>
480   <xsd:attribute name="double" use="optional" type="xsd:double"/>
481   <xsd:attribute name="integer" use="optional" type="xsd:int"/>
482   <xsd:attribute name="string" use="optional" type="s:ST_Xstring"/>
483   <xsd:attribute name="cell" use="optional" type="s:ST_Xstring"/>
484 </xsd:complexType>
485 <xsd:simpleType name="ST_ParameterType">
486   <xsd:restriction base="xsd:string">
487     <xsd:enumeration value="prompt"/>
488     <xsd:enumeration value="value"/>
489     <xsd:enumeration value="cell"/>
490   </xsd:restriction>
491 </xsd:simpleType>
492 <xsd:complexType name="CT_Tables">
493   <xsd:choice minOccurs="1" maxOccurs="unbounded">
494     <xsd:element name="m" type="CT_TableMissing"/>
495     <xsd:element name="s" type="CT_XStringElement"/>
496     <xsd:element name="x" type="CT_Index"/>
497   </xsd:choice>
498   <xsd:attribute name="count" use="optional" type="xsd:unsignedInt"/>
499 </xsd:complexType>
500 <xsd:complexType name="CT_TableMissing"/>
501 <xsd:complexType name="CT_TextPr">
502   <xsd:sequence>
503     <xsd:element name="textFields" minOccurs="0" maxOccurs="1" type="CT_TextFields"/>
504   </xsd:sequence>
505   <xsd:attribute name="prompt" use="optional" type="xsd:boolean" default="true"/>
506   <xsd:attribute name="fileType" use="optional" type="ST_FileType" default="win"/>
507   <xsd:attribute name="codePage" use="optional" type="xsd:unsignedInt" default="1252"/>
508   <xsd:attribute name="characterSet" use="optional" type="xsd:string"/>
509   <xsd:attribute name="firstRow" use="optional" type="xsd:unsignedInt" default="1"/>
510   <xsd:attribute name="sourceFile" use="optional" type="s:ST_Xstring" default=""/>
511   <xsd:attribute name="delimited" use="optional" type="xsd:boolean" default="true"/>
512   <xsd:attribute name="decimal" use="optional" type="s:ST_Xstring" default="."/>

```

```

513     <xsd:attribute name="thousands" use="optional" type="s:ST_Xstring" default=","/>
514     <xsd:attribute name="tab" use="optional" type="xsd:boolean" default="true"/>
515     <xsd:attribute name="space" use="optional" type="xsd:boolean" default="false"/>
516     <xsd:attribute name="comma" use="optional" type="xsd:boolean" default="false"/>
517     <xsd:attribute name="semicolon" use="optional" type="xsd:boolean" default="false"/>
518     <xsd:attribute name="consecutive" use="optional" type="xsd:boolean" default="false"/>
519     <xsd:attribute name="qualifier" use="optional" type="ST_Qualifier" default="doubleQuote"/>
520     <xsd:attribute name="delimiter" use="optional" type="s:ST_Xstring"/>
521 </xsd:complexType>
522 <xsd:simpleType name="ST_FileType">
523     <xsd:restriction base="xsd:string">
524         <xsd:enumeration value="mac"/>
525         <xsd:enumeration value="win"/>
526         <xsd:enumeration value="dos"/>
527         <xsd:enumeration value="lin"/>
528         <xsd:enumeration value="other"/>
529     </xsd:restriction>
530 </xsd:simpleType>
531 <xsd:simpleType name="ST_Qualifier">
532     <xsd:restriction base="xsd:string">
533         <xsd:enumeration value="doubleQuote"/>
534         <xsd:enumeration value="singleQuote"/>
535         <xsd:enumeration value="none"/>
536     </xsd:restriction>
537 </xsd:simpleType>
538 <xsd:complexType name="CT_TextFields">
539     <xsd:sequence>
540         <xsd:element name="textField" minOccurs="1" maxOccurs="unbounded" type="CT_TextField"/>
541     </xsd:sequence>
542     <xsd:attribute name="count" use="optional" type="xsd:unsignedInt" default="1"/>
543 </xsd:complexType>
544 <xsd:complexType name="CT_TextField">
545     <xsd:attribute name="type" use="optional" type="ST_ExternalConnectionType" default="general"/>
546     <xsd:attribute name="position" use="optional" type="xsd:unsignedInt" default="0"/>
547 </xsd:complexType>
548 <xsd:simpleType name="ST_ExternalConnectionType">
549     <xsd:restriction base="xsd:string">
550         <xsd:enumeration value="general"/>
551         <xsd:enumeration value="text"/>
552         <xsd:enumeration value="MDY"/>
553         <xsd:enumeration value="DMY"/>
554         <xsd:enumeration value="YMD"/>
555         <xsd:enumeration value="MYD"/>
556         <xsd:enumeration value="DYM"/>
557         <xsd:enumeration value="YDM"/>
558         <xsd:enumeration value="skip"/>
559         <xsd:enumeration value="EMD"/>
560     </xsd:restriction>
561 </xsd:simpleType>
562 <xsd:element name="pivotCacheDefinition" type="CT_PivotCacheDefinition"/>
563 <xsd:element name="pivotCacheRecords" type="CT_PivotCacheRecords"/>
564 <xsd:element name="pivotTableDefinition" type="CT_pivotTableDefinition"/>
565 <xsd:complexType name="CT_PivotCacheDefinition">

```

```

566 <xsd:sequence>
567   <xsd:element name="cacheSource" type="CT_CacheSource" minOccurs="1" maxOccurs="1"/>
568   <xsd:element name="cacheFields" type="CT_CacheFields" minOccurs="1" maxOccurs="1"/>
569   <xsd:element name="cacheHierarchies" minOccurs="0" type="CT_CacheHierarchies"/>
570   <xsd:element name="kpis" minOccurs="0" type="CT_PCDKPIs"/>
571   <xsd:element name="tupleCache" minOccurs="0" type="CT_TupleCache"/>
572   <xsd:element name="calculatedItems" minOccurs="0" type="CT_CalculatedItems"/>
573   <xsd:element name="calculatedMembers" type="CT_CalculatedMembers" minOccurs="0"/>
574   <xsd:element name="dimensions" type="CT_Dimensions" minOccurs="0"/>
575   <xsd:element name="measureGroups" type="CT_MeasureGroups" minOccurs="0"/>
576   <xsd:element name="maps" type="CT_MeasureDimensionMaps" minOccurs="0"/>
577   <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
578 </xsd:sequence>
579 <xsd:attribute ref="r:id" use="optional"/>
580 <xsd:attribute name="invalid" type="xsd:boolean" use="optional" default="false"/>
581 <xsd:attribute name="saveData" type="xsd:boolean" use="optional" default="true"/>
582 <xsd:attribute name="refreshOnLoad" type="xsd:boolean" use="optional" default="false"/>
583 <xsd:attribute name="optimizeMemory" type="xsd:boolean" use="optional" default="false"/>
584 <xsd:attribute name="enableRefresh" type="xsd:boolean" use="optional" default="true"/>
585 <xsd:attribute name="refreshedBy" type="s:ST_Xstring" use="optional"/>
586 <xsd:attribute name="refreshedDate" type="xsd:double" use="optional"/>
587 <xsd:attribute name="refreshedDateIso" type="xsd:dateTime" use="optional"/>
588 <xsd:attribute name="backgroundQuery" type="xsd:boolean" default="false"/>
589 <xsd:attribute name="missingItemsLimit" type="xsd:unsignedInt" use="optional"/>
590 <xsd:attribute name="createdVersion" type="xsd:unsignedByte" use="optional" default="0"/>
591 <xsd:attribute name="refreshedVersion" type="xsd:unsignedByte" use="optional" default="0"/>
592 <xsd:attribute name="minRefreshableVersion" type="xsd:unsignedByte" use="optional"
593   default="0"/>
594 <xsd:attribute name="recordCount" type="xsd:unsignedInt" use="optional"/>
595 <xsd:attribute name="upgradeOnRefresh" type="xsd:boolean" use="optional" default="false"/>
596 <xsd:attribute name="tupleCache" type="xsd:boolean" use="optional" default="false"/>
597 <xsd:attribute name="supportSubquery" type="xsd:boolean" use="optional" default="false"/>
598 <xsd:attribute name="supportAdvancedDrill" type="xsd:boolean" use="optional" default="false"/>
599 </xsd:complexType>
600 <xsd:complexType name="CT_CacheFields">
601   <xsd:sequence>
602     <xsd:element name="cacheField" type="CT_CacheField" minOccurs="0" maxOccurs="unbounded"/>
603   </xsd:sequence>
604   <xsd:attribute name="count" type="xsd:unsignedInt"/>
605 </xsd:complexType>
606 <xsd:complexType name="CT_CacheField">
607   <xsd:sequence>
608     <xsd:element name="sharedItems" type="CT_SharedItems" minOccurs="0" maxOccurs="1"/>
609     <xsd:element name="fieldGroup" minOccurs="0" type="CT_FieldGroup"/>
610     <xsd:element name="mpMap" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
611     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
612   </xsd:sequence>
613   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
614   <xsd:attribute name="caption" type="s:ST_Xstring" use="optional"/>
615   <xsd:attribute name="propertyName" type="s:ST_Xstring" use="optional"/>
616   <xsd:attribute name="serverField" type="xsd:boolean" use="optional" default="false"/>
617   <xsd:attribute name="uniqueList" type="xsd:boolean" use="optional" default="true"/>
618   <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>

```



```

619     <xsd:attribute name="formula" type="s:ST Xstring" use="optional"/>
620     <xsd:attribute name="sqlType" type="xsd:int" use="optional" default="0"/>
621     <xsd:attribute name="hierarchy" type="xsd:int" use="optional" default="0"/>
622     <xsd:attribute name="level" type="xsd:unsignedInt" use="optional" default="0"/>
623     <xsd:attribute name="databaseField" type="xsd:boolean" default="true"/>
624     <xsd:attribute name="mappingCount" type="xsd:unsignedInt" use="optional"/>
625     <xsd:attribute name="memberPropertyField" type="xsd:boolean" use="optional" default="false"/>
626 </xsd:complexType>
627 <xsd:complexType name="CT_CacheSource">
628     <xsd:choice minOccurs="0" maxOccurs="1">
629         <xsd:element name="worksheetSource" type="CT WorksheetSource" minOccurs="1"
630             maxOccurs="1"/>
631         <xsd:element name="consolidation" type="CT Consolidation" minOccurs="1" maxOccurs="1"/>
632         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0"/>
633     </xsd:choice>
634     <xsd:attribute name="type" type="ST SourceType" use="required"/>
635     <xsd:attribute name="connectionId" type="xsd:unsignedInt" default="0" use="optional"/>
636 </xsd:complexType>
637 <xsd:simpleType name="ST_SourceType">
638     <xsd:restriction base="xsd:string">
639         <xsd:enumeration value="worksheet"/>
640         <xsd:enumeration value="external"/>
641         <xsd:enumeration value="consolidation"/>
642         <xsd:enumeration value="scenario"/>
643     </xsd:restriction>
644 </xsd:simpleType>
645 <xsd:complexType name="CT_WorksheetSource">
646     <xsd:attribute name="ref" type="ST Ref" use="optional"/>
647     <xsd:attribute name="name" type="s:ST Xstring" use="optional"/>
648     <xsd:attribute name="sheet" type="s:ST Xstring" use="optional"/>
649     <xsd:attribute ref="r:id" use="optional"/>
650 </xsd:complexType>
651 <xsd:complexType name="CT_Consolidation">
652     <xsd:sequence>
653         <xsd:element name="pages" type="CT Pages" minOccurs="0" maxOccurs="1"/>
654         <xsd:element name="rangeSets" type="CT RangeSets" minOccurs="1" maxOccurs="1"/>
655     </xsd:sequence>
656     <xsd:attribute name="autoPage" type="xsd:boolean" default="true" use="optional"/>
657 </xsd:complexType>
658 <xsd:complexType name="CT_Pages">
659     <xsd:sequence>
660         <xsd:element name="page" type="CT PCDSPage" minOccurs="1" maxOccurs="4"/>
661     </xsd:sequence>
662     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
663 </xsd:complexType>
664 <xsd:complexType name="CT_PCDSPage">
665     <xsd:sequence>
666         <xsd:element name="pageItem" type="CT PageItem" minOccurs="0" maxOccurs="unbounded"/>
667     </xsd:sequence>
668     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
669 </xsd:complexType>
670 <xsd:complexType name="CT_PageItem">
671     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>

```

```

672 </xsd:complexType>
673 <xsd:complexType name="CT_RangeSets">
674   <xsd:sequence>
675     <xsd:element name="rangeSet" type="CT_RangeSet" minOccurs="1" maxOccurs="unbounded"/>
676   </xsd:sequence>
677   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
678 </xsd:complexType>
679 <xsd:complexType name="CT_RangeSet">
680   <xsd:attribute name="i1" type="xsd:unsignedInt" use="optional"/>
681   <xsd:attribute name="i2" type="xsd:unsignedInt" use="optional"/>
682   <xsd:attribute name="i3" type="xsd:unsignedInt" use="optional"/>
683   <xsd:attribute name="i4" type="xsd:unsignedInt" use="optional"/>
684   <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
685   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
686   <xsd:attribute name="sheet" type="s:ST_Xstring" use="optional"/>
687   <xsd:attribute ref="r:id" use="optional"/>
688 </xsd:complexType>
689 <xsd:complexType name="CT_SharedItems">
690   <xsd:choice minOccurs="0" maxOccurs="unbounded">
691     <xsd:element name="m" type="CT_Missing" minOccurs="1" maxOccurs="1"/>
692     <xsd:element name="n" type="CT_Number" minOccurs="1" maxOccurs="1"/>
693     <xsd:element name="b" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
694     <xsd:element name="e" type="CT_Error" minOccurs="1" maxOccurs="1"/>
695     <xsd:element name="s" type="CT_String" minOccurs="1" maxOccurs="1"/>
696     <xsd:element name="d" type="CT_DateTime" minOccurs="1" maxOccurs="1"/>
697   </xsd:choice>
698   <xsd:attribute name="containsSemiMixedTypes" type="xsd:boolean" use="optional"
699     default="true"/>
700   <xsd:attribute name="containsNonDate" type="xsd:boolean" use="optional" default="true"/>
701   <xsd:attribute name="containsDate" type="xsd:boolean" use="optional" default="false"/>
702   <xsd:attribute name="containsString" type="xsd:boolean" use="optional" default="true"/>
703   <xsd:attribute name="containsBlank" type="xsd:boolean" use="optional" default="false"/>
704   <xsd:attribute name="containsMixedTypes" type="xsd:boolean" use="optional" default="false"/>
705   <xsd:attribute name="containsNumber" type="xsd:boolean" use="optional" default="false"/>
706   <xsd:attribute name="containsInteger" type="xsd:boolean" use="optional" default="false"/>
707   <xsd:attribute name="minValue" type="xsd:double" use="optional"/>
708   <xsd:attribute name="maxValue" type="xsd:double" use="optional"/>
709   <xsd:attribute name="minDate" type="xsd:dateTime" use="optional"/>
710   <xsd:attribute name="maxDate" type="xsd:dateTime" use="optional"/>
711   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
712   <xsd:attribute name="longText" type="xsd:boolean" use="optional" default="false"/>
713 </xsd:complexType>
714 <xsd:complexType name="CT_Missing">
715   <xsd:sequence>
716     <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
717     <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
718   </xsd:sequence>
719   <xsd:attribute name="u" type="xsd:boolean"/>
720   <xsd:attribute name="f" type="xsd:boolean"/>
721   <xsd:attribute name="c" type="s:ST_Xstring"/>
722   <xsd:attribute name="cp" type="xsd:unsignedInt"/>
723   <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
724   <xsd:attribute name="bc" type="ST_UnsignedIntHex" use="optional"/>

```

```

725     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
726     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
727     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
728     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
729     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
730 </xsd:complexType>
731 <xsd:complexType name="CT_Number">
732     <xsd:sequence>
733         <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
734         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
735     </xsd:sequence>
736     <xsd:attribute name="v" use="required" type="xsd:double"/>
737     <xsd:attribute name="u" type="xsd:boolean"/>
738     <xsd:attribute name="f" type="xsd:boolean"/>
739     <xsd:attribute name="c" type="s:ST_Xstring"/>
740     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
741     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
742     <xsd:attribute name="bc" type="ST UnsignedIntHex" use="optional"/>
743     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
744     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
745     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
746     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
747     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
748 </xsd:complexType>
749 <xsd:complexType name="CT_Boolean">
750     <xsd:sequence>
751         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
752     </xsd:sequence>
753     <xsd:attribute name="v" use="required" type="xsd:boolean"/>
754     <xsd:attribute name="u" type="xsd:boolean"/>
755     <xsd:attribute name="f" type="xsd:boolean"/>
756     <xsd:attribute name="c" type="s:ST_Xstring"/>
757     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
758 </xsd:complexType>
759 <xsd:complexType name="CT_Error">
760     <xsd:sequence>
761         <xsd:element name="tpls" minOccurs="0" type="CT_Tuples"/>
762         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
763     </xsd:sequence>
764     <xsd:attribute name="v" use="required" type="s:ST_Xstring"/>
765     <xsd:attribute name="u" type="xsd:boolean"/>
766     <xsd:attribute name="f" type="xsd:boolean"/>
767     <xsd:attribute name="c" type="s:ST_Xstring"/>
768     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
769     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
770     <xsd:attribute name="bc" type="ST UnsignedIntHex" use="optional"/>
771     <xsd:attribute name="fc" type="ST UnsignedIntHex" use="optional"/>
772     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
773     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
774     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
775     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
776 </xsd:complexType>
777 <xsd:complexType name="CT_String">

```

```

778     <xsd:sequence>
779         <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
780         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
781     </xsd:sequence>
782     <xsd:attribute name="v" use="required" type="s:ST_Xstring"/>
783     <xsd:attribute name="u" type="xsd:boolean"/>
784     <xsd:attribute name="f" type="xsd:boolean"/>
785     <xsd:attribute name="c" type="s:ST_Xstring"/>
786     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
787     <xsd:attribute name="in" type="xsd:unsignedInt" use="optional"/>
788     <xsd:attribute name="bc" type="ST_UnsignedIntHex" use="optional"/>
789     <xsd:attribute name="fc" type="ST_UnsignedIntHex" use="optional"/>
790     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
791     <xsd:attribute name="un" type="xsd:boolean" use="optional" default="false"/>
792     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
793     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
794 </xsd:complexType>
795 <xsd:complexType name="CT_DateTime">
796     <xsd:sequence>
797         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
798     </xsd:sequence>
799     <xsd:attribute name="v" use="required" type="xsd:dateTime"/>
800     <xsd:attribute name="u" type="xsd:boolean"/>
801     <xsd:attribute name="f" type="xsd:boolean"/>
802     <xsd:attribute name="c" type="s:ST_Xstring"/>
803     <xsd:attribute name="cp" type="xsd:unsignedInt"/>
804 </xsd:complexType>
805 <xsd:complexType name="CT_FieldGroup">
806     <xsd:sequence>
807         <xsd:element name="rangePr" minOccurs="0" type="CT_RangePr"/>
808         <xsd:element name="discretePr" minOccurs="0" type="CT_DiscretePr"/>
809         <xsd:element name="groupItems" minOccurs="0" type="CT_GroupItems"/>
810     </xsd:sequence>
811     <xsd:attribute name="par" type="xsd:unsignedInt" use="optional"/>
812     <xsd:attribute name="base" type="xsd:unsignedInt" use="optional"/>
813 </xsd:complexType>
814 <xsd:complexType name="CT_RangePr">
815     <xsd:attribute name="autoStart" type="xsd:boolean" default="true"/>
816     <xsd:attribute name="autoEnd" type="xsd:boolean" default="true"/>
817     <xsd:attribute name="groupBy" type="ST_GroupBy" default="range"/>
818     <xsd:attribute name="startNum" type="xsd:double"/>
819     <xsd:attribute name="endNum" type="xsd:double"/>
820     <xsd:attribute name="startDate" type="xsd:dateTime"/>
821     <xsd:attribute name="endDate" type="xsd:dateTime"/>
822     <xsd:attribute name="groupInterval" type="xsd:double" default="1"/>
823 </xsd:complexType>
824 <xsd:simpleType name="ST_GroupBy">
825     <xsd:restriction base="xsd:string">
826         <xsd:enumeration value="range"/>
827         <xsd:enumeration value="seconds"/>
828         <xsd:enumeration value="minutes"/>
829         <xsd:enumeration value="hours"/>
830         <xsd:enumeration value="days"/>

```

```

831         <xsd:enumeration value="months"/>
832         <xsd:enumeration value="quarters"/>
833         <xsd:enumeration value="years"/>
834     </xsd:restriction>
835 </xsd:simpleType>
836 <xsd:complexType name="CT_DiscretePr">
837     <xsd:sequence>
838         <xsd:element name="x" maxOccurs="unbounded" type="CT_Index"/>
839     </xsd:sequence>
840     <xsd:attribute name="count" type="xsd:unsignedInt"/>
841 </xsd:complexType>
842 <xsd:complexType name="CT_GroupItems">
843     <xsd:choice maxOccurs="unbounded">
844         <xsd:element name="m" type="CT_Missing"/>
845         <xsd:element name="n" type="CT_Number"/>
846         <xsd:element name="b" type="CT_Boolean"/>
847         <xsd:element name="e" type="CT_Error"/>
848         <xsd:element name="s" type="CT_String"/>
849         <xsd:element name="d" type="CT_DateTime"/>
850     </xsd:choice>
851     <xsd:attribute name="count" type="xsd:unsignedInt"/>
852 </xsd:complexType>
853 <xsd:complexType name="CT_PivotCacheRecords">
854     <xsd:sequence>
855         <xsd:element name="r" minOccurs="0" maxOccurs="unbounded" type="CT_Record"/>
856         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
857     </xsd:sequence>
858     <xsd:attribute name="count" type="xsd:unsignedInt"/>
859 </xsd:complexType>
860 <xsd:complexType name="CT_Record">
861     <xsd:choice maxOccurs="unbounded">
862         <xsd:element name="m" type="CT_Missing"/>
863         <xsd:element name="n" type="CT_Number"/>
864         <xsd:element name="b" type="CT_Boolean"/>
865         <xsd:element name="e" type="CT_Error"/>
866         <xsd:element name="s" type="CT_String"/>
867         <xsd:element name="d" type="CT_DateTime"/>
868         <xsd:element name="x" type="CT_Index"/>
869     </xsd:choice>
870 </xsd:complexType>
871 <xsd:complexType name="CT_PCDKPIs">
872     <xsd:sequence>
873         <xsd:element name="kpi" minOccurs="0" maxOccurs="unbounded" type="CT_PCDKPI"/>
874     </xsd:sequence>
875     <xsd:attribute name="count" type="xsd:unsignedInt"/>
876 </xsd:complexType>
877 <xsd:complexType name="CT_PCDKPI">
878     <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
879     <xsd:attribute name="caption" use="optional" type="s:ST_Xstring"/>
880     <xsd:attribute name="displayFolder" type="s:ST_Xstring"/>
881     <xsd:attribute name="measureGroup" type="s:ST_Xstring"/>
882     <xsd:attribute name="parent" type="s:ST_Xstring"/>
883     <xsd:attribute name="value" use="required" type="s:ST_Xstring"/>

```

```

884     <xsd:attribute name="goal" type="s:ST Xstring"/>
885     <xsd:attribute name="status" type="s:ST Xstring"/>
886     <xsd:attribute name="trend" type="s:ST Xstring"/>
887     <xsd:attribute name="weight" type="s:ST Xstring"/>
888     <xsd:attribute name="time" type="s:ST Xstring"/>
889 </xsd:complexType>
890 <xsd:complexType name="CT_CacheHierarchies">
891     <xsd:sequence>
892         <xsd:element name="cacheHierarchy" minOccurs="0" maxOccurs="unbounded"
893             type="CT_CacheHierarchy"/>
894     </xsd:sequence>
895     <xsd:attribute name="count" type="xsd:unsignedInt"/>
896 </xsd:complexType>
897 <xsd:complexType name="CT_CacheHierarchy">
898     <xsd:sequence>
899         <xsd:element name="fieldsUsage" minOccurs="0" type="CT_FieldsUsage"/>
900         <xsd:element name="groupLevels" minOccurs="0" type="CT_GroupLevels"/>
901         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
902     </xsd:sequence>
903     <xsd:attribute name="uniqueName" use="required" type="s:ST Xstring"/>
904     <xsd:attribute name="caption" use="optional" type="s:ST Xstring"/>
905     <xsd:attribute name="measure" type="xsd:boolean" default="false"/>
906     <xsd:attribute name="set" type="xsd:boolean" default="false"/>
907     <xsd:attribute name="parentSet" type="xsd:unsignedInt" use="optional"/>
908     <xsd:attribute name="iconSet" type="xsd:int" default="0"/>
909     <xsd:attribute name="attribute" type="xsd:boolean" default="false"/>
910     <xsd:attribute name="time" type="xsd:boolean" default="false"/>
911     <xsd:attribute name="keyAttribute" type="xsd:boolean" default="false"/>
912     <xsd:attribute name="defaultMemberUniqueName" type="s:ST Xstring"/>
913     <xsd:attribute name="allUniqueName" type="s:ST Xstring"/>
914     <xsd:attribute name="allCaption" type="s:ST Xstring"/>
915     <xsd:attribute name="dimensionUniqueName" type="s:ST Xstring"/>
916     <xsd:attribute name="displayFolder" type="s:ST Xstring"/>
917     <xsd:attribute name="measureGroup" type="s:ST Xstring"/>
918     <xsd:attribute name="measures" type="xsd:boolean" default="false"/>
919     <xsd:attribute name="count" use="required" type="xsd:unsignedInt"/>
920     <xsd:attribute name="oneField" type="xsd:boolean" default="false"/>
921     <xsd:attribute name="memberValueDatatype" use="optional" type="xsd:unsignedShort"/>
922     <xsd:attribute name="unbalanced" use="optional" type="xsd:boolean"/>
923     <xsd:attribute name="unbalancedGroup" use="optional" type="xsd:boolean"/>
924     <xsd:attribute name="hidden" type="xsd:boolean" default="false"/>
925 </xsd:complexType>
926 <xsd:complexType name="CT_FieldsUsage">
927     <xsd:sequence>
928         <xsd:element name="fieldUsage" minOccurs="0" maxOccurs="unbounded" type="CT_FieldUsage"/>
929     </xsd:sequence>
930     <xsd:attribute name="count" type="xsd:unsignedInt"/>
931 </xsd:complexType>
932 <xsd:complexType name="CT_FieldUsage">
933     <xsd:attribute name="x" use="required" type="xsd:int"/>
934 </xsd:complexType>
935 <xsd:complexType name="CT_GroupLevels">
936     <xsd:sequence>

```

```

937     <xsd:element name="groupLevel" maxOccurs="unbounded" type="CT_GroupLevel1"/>
938   </xsd:sequence>
939   <xsd:attribute name="count" type="xsd:unsignedInt"/>
940 </xsd:complexType>
941 <xsd:complexType name="CT_GroupLevel">
942   <xsd:sequence>
943     <xsd:element name="groups" minOccurs="0" type="CT_Groups"/>
944     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
945   </xsd:sequence>
946   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
947   <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
948   <xsd:attribute name="user" type="xsd:boolean" default="false"/>
949   <xsd:attribute name="customRollUp" type="xsd:boolean" default="false"/>
950 </xsd:complexType>
951 <xsd:complexType name="CT_Groups">
952   <xsd:sequence>
953     <xsd:element name="group" maxOccurs="unbounded" type="CT_LevelGroup"/>
954   </xsd:sequence>
955   <xsd:attribute name="count" type="xsd:unsignedInt"/>
956 </xsd:complexType>
957 <xsd:complexType name="CT_LevelGroup">
958   <xsd:sequence>
959     <xsd:element name="groupMembers" type="CT_GroupMembers"/>
960   </xsd:sequence>
961   <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
962   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
963   <xsd:attribute name="caption" use="required" type="s:ST_Xstring"/>
964   <xsd:attribute name="uniqueParent" type="s:ST_Xstring"/>
965   <xsd:attribute name="id" type="xsd:int"/>
966 </xsd:complexType>
967 <xsd:complexType name="CT_GroupMembers">
968   <xsd:sequence>
969     <xsd:element name="groupMember" maxOccurs="unbounded" type="CT_GroupMember"/>
970   </xsd:sequence>
971   <xsd:attribute name="count" type="xsd:unsignedInt"/>
972 </xsd:complexType>
973 <xsd:complexType name="CT_GroupMember">
974   <xsd:attribute name="uniqueName" use="required" type="s:ST_Xstring"/>
975   <xsd:attribute name="group" type="xsd:boolean" default="false"/>
976 </xsd:complexType>
977 <xsd:complexType name="CT_TupleCache">
978   <xsd:sequence>
979     <xsd:element name="entries" minOccurs="0" type="CT_PCSDTCEntries"/>
980     <xsd:element name="sets" minOccurs="0" type="CT_Sets"/>
981     <xsd:element name="queryCache" minOccurs="0" type="CT_QueryCache"/>
982     <xsd:element name="serverFormats" minOccurs="0" maxOccurs="1" type="CT_ServerFormats"/>
983     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
984   </xsd:sequence>
985 </xsd:complexType>
986 <xsd:complexType name="CT_ServerFormat">
987   <xsd:attribute name="culture" use="optional" type="s:ST_Xstring"/>
988   <xsd:attribute name="format" use="optional" type="s:ST_Xstring"/>
989 </xsd:complexType>

```

```

990 <xsd:complexType name="CT_ServerFormats">
991   <xsd:sequence>
992     <xsd:element name="serverFormat" type="CT_ServerFormat" minOccurs="0"
993       maxOccurs="unbounded"/>
994   </xsd:sequence>
995   <xsd:attribute name="count" type="xsd:unsignedInt"/>
996 </xsd:complexType>
997 <xsd:complexType name="CT_PCSDTCEntries">
998   <xsd:choice maxOccurs="unbounded">
999     <xsd:element name="m" type="CT_Missing"/>
1000    <xsd:element name="n" type="CT_Number"/>
1001    <xsd:element name="e" type="CT_Error"/>
1002    <xsd:element name="s" type="CT_String"/>
1003  </xsd:choice>
1004  <xsd:attribute name="count" type="xsd:unsignedInt"/>
1005 </xsd:complexType>
1006 <xsd:complexType name="CT_Tuples">
1007   <xsd:sequence>
1008     <xsd:element name="tpl" type="CT_Tuple" minOccurs="1" maxOccurs="unbounded"/>
1009   </xsd:sequence>
1010   <xsd:attribute name="c" type="xsd:unsignedInt" use="optional"/>
1011 </xsd:complexType>
1012 <xsd:complexType name="CT_Tuple">
1013   <xsd:attribute name="fld" type="xsd:unsignedInt"/>
1014   <xsd:attribute name="hier" type="xsd:unsignedInt"/>
1015   <xsd:attribute name="item" type="xsd:unsignedInt" use="required"/>
1016 </xsd:complexType>
1017 <xsd:complexType name="CT_Sets">
1018   <xsd:sequence>
1019     <xsd:element name="set" maxOccurs="unbounded" type="CT_Set"/>
1020   </xsd:sequence>
1021   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1022 </xsd:complexType>
1023 <xsd:complexType name="CT_Set">
1024   <xsd:sequence>
1025     <xsd:element name="tpls" minOccurs="0" maxOccurs="unbounded" type="CT_Tuples"/>
1026     <xsd:element name="sortByTuple" minOccurs="0" type="CT_Tuples"/>
1027   </xsd:sequence>
1028   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1029   <xsd:attribute name="maxRank" use="required" type="xsd:int"/>
1030   <xsd:attribute name="setDefinition" use="required" type="s:ST_Xstring"/>
1031   <xsd:attribute name="sortType" type="ST_SortType" default="none"/>
1032   <xsd:attribute name="queryFailed" type="xsd:boolean" default="false"/>
1033 </xsd:complexType>
1034 <xsd:simpleType name="ST_SortType">
1035   <xsd:restriction base="xsd:string">
1036     <xsd:enumeration value="none"/>
1037     <xsd:enumeration value="ascending"/>
1038     <xsd:enumeration value="descending"/>
1039     <xsd:enumeration value="ascendingAlpha"/>
1040     <xsd:enumeration value="descendingAlpha"/>
1041     <xsd:enumeration value="ascendingNatural"/>
1042     <xsd:enumeration value="descendingNatural"/>

```



```

1043     </xsd:restriction>
1044 </xsd:simpleType>
1045 <xsd:complexType name="CT_QueryCache">
1046     <xsd:sequence>
1047         <xsd:element name="query" maxOccurs="unbounded" type="CT_Query"/>
1048     </xsd:sequence>
1049     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1050 </xsd:complexType>
1051 <xsd:complexType name="CT_Query">
1052     <xsd:sequence>
1053         <xsd:element name="tpls" minOccurs="0" type="CT_Tuples"/>
1054     </xsd:sequence>
1055     <xsd:attribute name="mdx" use="required" type="s:ST_Xstring"/>
1056 </xsd:complexType>
1057 <xsd:complexType name="CT_CalculatedItems">
1058     <xsd:sequence>
1059         <xsd:element name="calculatedItem" maxOccurs="unbounded" type="CT_CalculatedItem"/>
1060     </xsd:sequence>
1061     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1062 </xsd:complexType>
1063 <xsd:complexType name="CT_CalculatedItem">
1064     <xsd:sequence>
1065         <xsd:element name="pivotArea" type="CT_PivotArea"/>
1066         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1067     </xsd:sequence>
1068     <xsd:attribute name="field" type="xsd:unsignedInt" use="optional"/>
1069     <xsd:attribute name="formula" type="s:ST_Xstring"/>
1070 </xsd:complexType>
1071 <xsd:complexType name="CT_CalculatedMembers">
1072     <xsd:sequence>
1073         <xsd:element name="calculatedMember" maxOccurs="unbounded" type="CT_CalculatedMember"/>
1074     </xsd:sequence>
1075     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1076 </xsd:complexType>
1077 <xsd:complexType name="CT_CalculatedMember">
1078     <xsd:sequence minOccurs="0">
1079         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1080     </xsd:sequence>
1081     <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1082     <xsd:attribute name="mdx" use="required" type="s:ST_Xstring"/>
1083     <xsd:attribute name="memberName" type="s:ST_Xstring"/>
1084     <xsd:attribute name="hierarchy" type="s:ST_Xstring"/>
1085     <xsd:attribute name="parent" type="s:ST_Xstring"/>
1086     <xsd:attribute name="solveOrder" type="xsd:int" default="0"/>
1087     <xsd:attribute name="set" type="xsd:boolean" default="false"/>
1088 </xsd:complexType>
1089 <xsd:complexType name="CT_pivotTableDefinition">
1090     <xsd:sequence>
1091         <xsd:element name="location" type="CT_Location"/>
1092         <xsd:element name="pivotFields" type="CT_PivotFields" minOccurs="0"/>
1093         <xsd:element name="rowFields" type="CT_RowFields" minOccurs="0"/>
1094         <xsd:element name="rowItems" type="CT_rowItems" minOccurs="0"/>
1095         <xsd:element name="colFields" type="CT_ColFields" minOccurs="0"/>

```

```

1096     <xsd:element name="colItems" type="CT_colItems" minOccurs="0"/>
1097     <xsd:element name="pageFields" type="CT_PageFields" minOccurs="0"/>
1098     <xsd:element name="dataFields" type="CT_DataFields" minOccurs="0"/>
1099     <xsd:element name="formats" type="CT_Formats" minOccurs="0"/>
1100     <xsd:element name="conditionalFormats" type="CT_ConditionalFormats" minOccurs="0"/>
1101     <xsd:element name="chartFormats" type="CT_ChartFormats" minOccurs="0"/>
1102     <xsd:element name="pivotHierarchies" type="CT_PivotHierarchies" minOccurs="0"/>
1103     <xsd:element name="pivotTableStyleInfo" minOccurs="0" maxOccurs="1"
1104         type="CT_PivotTableStyle"/>
1105     <xsd:element name="filters" minOccurs="0" maxOccurs="1" type="CT_PivotFilters"/>
1106     <xsd:element name="rowHierarchiesUsage" type="CT_RowHierarchiesUsage" minOccurs="0"
1107         maxOccurs="1"/>
1108     <xsd:element name="colHierarchiesUsage" type="CT_ColHierarchiesUsage" minOccurs="0"
1109         maxOccurs="1"/>
1110     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1111 </xsd:sequence>
1112 <xsd:attribute name="name" use="required" type="s:ST_Xstring"/>
1113 <xsd:attribute name="cacheId" use="required" type="xsd:unsignedInt"/>
1114 <xsd:attribute name="dataOnRows" type="xsd:boolean" default="false"/>
1115 <xsd:attribute name="dataPosition" type="xsd:unsignedInt" use="optional"/>
1116 <xsd:attributeGroup ref="AG_AutoFormat"/>
1117 <xsd:attribute name="dataCaption" use="required" type="s:ST_Xstring"/>
1118 <xsd:attribute name="grandTotalCaption" type="s:ST_Xstring"/>
1119 <xsd:attribute name="errorCaption" type="s:ST_Xstring"/>
1120 <xsd:attribute name="showError" type="xsd:boolean" default="false"/>
1121 <xsd:attribute name="missingCaption" type="s:ST_Xstring"/>
1122 <xsd:attribute name="showMissing" type="xsd:boolean" default="true"/>
1123 <xsd:attribute name="pageStyle" type="s:ST_Xstring"/>
1124 <xsd:attribute name="pivotTableStyle" type="s:ST_Xstring"/>
1125 <xsd:attribute name="vacatedStyle" type="s:ST_Xstring"/>
1126 <xsd:attribute name="tag" type="s:ST_Xstring"/>
1127 <xsd:attribute name="updatedVersion" type="xsd:unsignedByte" default="0"/>
1128 <xsd:attribute name="minRefreshableVersion" type="xsd:unsignedByte" default="0"/>
1129 <xsd:attribute name="asteriskTotals" type="xsd:boolean" default="false"/>
1130 <xsd:attribute name="showItems" type="xsd:boolean" default="true"/>
1131 <xsd:attribute name="editData" type="xsd:boolean" default="false"/>
1132 <xsd:attribute name="disableFieldList" type="xsd:boolean" default="false"/>
1133 <xsd:attribute name="showCalcMbrs" type="xsd:boolean" default="true"/>
1134 <xsd:attribute name="visualTotals" type="xsd:boolean" default="true"/>
1135 <xsd:attribute name="showMultipleLabel" type="xsd:boolean" default="true"/>
1136 <xsd:attribute name="showDataDropDown" type="xsd:boolean" default="true"/>
1137 <xsd:attribute name="showDrill" type="xsd:boolean" default="true"/>
1138 <xsd:attribute name="printDrill" type="xsd:boolean" default="false"/>
1139 <xsd:attribute name="showMemberPropertyTips" type="xsd:boolean" default="true"/>
1140 <xsd:attribute name="showDataTips" type="xsd:boolean" default="true"/>
1141 <xsd:attribute name="enableWizard" type="xsd:boolean" default="true"/>
1142 <xsd:attribute name="enableDrill" type="xsd:boolean" default="true"/>
1143 <xsd:attribute name="enableFieldProperties" type="xsd:boolean" default="true"/>
1144 <xsd:attribute name="preserveFormatting" type="xsd:boolean" default="true"/>
1145 <xsd:attribute name="useAutoFormatting" type="xsd:boolean" default="false"/>
1146 <xsd:attribute name="pageWrap" type="xsd:unsignedInt" default="0"/>
1147 <xsd:attribute name="pageOverThenDown" type="xsd:boolean" default="false"/>
1148 <xsd:attribute name="subtotalHiddenItems" type="xsd:boolean" default="false"/>

```

```

1149 <xsd:attribute name="rowGrandTotals" type="xsd:boolean" default="true"/>
1150 <xsd:attribute name="colGrandTotals" type="xsd:boolean" default="true"/>
1151 <xsd:attribute name="fieldPrintTitles" type="xsd:boolean" default="false"/>
1152 <xsd:attribute name="itemPrintTitles" type="xsd:boolean" default="false"/>
1153 <xsd:attribute name="mergeItem" type="xsd:boolean" default="false"/>
1154 <xsd:attribute name="showDropZones" type="xsd:boolean" default="true"/>
1155 <xsd:attribute name="createdVersion" type="xsd:unsignedByte" default="0"/>
1156 <xsd:attribute name="indent" type="xsd:unsignedInt" default="1"/>
1157 <xsd:attribute name="showEmptyRow" type="xsd:boolean" default="false"/>
1158 <xsd:attribute name="showEmptyCol" type="xsd:boolean" default="false"/>
1159 <xsd:attribute name="showHeaders" type="xsd:boolean" default="true"/>
1160 <xsd:attribute name="compact" type="xsd:boolean" default="true"/>
1161 <xsd:attribute name="outline" type="xsd:boolean" default="false"/>
1162 <xsd:attribute name="outlineData" type="xsd:boolean" default="false"/>
1163 <xsd:attribute name="compactData" type="xsd:boolean" default="true"/>
1164 <xsd:attribute name="published" type="xsd:boolean" default="false"/>
1165 <xsd:attribute name="gridDropZones" type="xsd:boolean" default="false"/>
1166 <xsd:attribute name="immersive" type="xsd:boolean" default="true"/>
1167 <xsd:attribute name="multipleFieldFilters" type="xsd:boolean" default="true"/>
1168 <xsd:attribute name="chartFormat" type="xsd:unsignedInt" default="0"/>
1169 <xsd:attribute name="rowHeaderCaption" type="s:ST Xstring"/>
1170 <xsd:attribute name="colHeaderCaption" type="s:ST Xstring"/>
1171 <xsd:attribute name="fieldListSortAscending" type="xsd:boolean" default="false"/>
1172 <xsd:attribute name="mdxSubqueries" type="xsd:boolean" default="false"/>
1173 <xsd:attribute name="customListSort" type="xsd:boolean" use="optional" default="true"/>
1174 </xsd:complexType>
1175 <xsd:complexType name="CT_Location">
1176 <xsd:attribute name="ref" use="required" type="ST Ref"/>
1177 <xsd:attribute name="firstHeaderRow" use="required" type="xsd:unsignedInt"/>
1178 <xsd:attribute name="firstDataRow" use="required" type="xsd:unsignedInt"/>
1179 <xsd:attribute name="firstDataCol" use="required" type="xsd:unsignedInt"/>
1180 <xsd:attribute name="rowPageCount" type="xsd:unsignedInt" default="0"/>
1181 <xsd:attribute name="colPageCount" type="xsd:unsignedInt" default="0"/>
1182 </xsd:complexType>
1183 <xsd:complexType name="CT_PivotFields">
1184 <xsd:sequence>
1185 <xsd:element name="pivotField" maxOccurs="unbounded" type="CT_PivotField"/>
1186 </xsd:sequence>
1187 <xsd:attribute name="count" type="xsd:unsignedInt"/>
1188 </xsd:complexType>
1189 <xsd:complexType name="CT_PivotField">
1190 <xsd:sequence>
1191 <xsd:element name="items" minOccurs="0" type="CT_Items"/>
1192 <xsd:element name="autoSortScope" minOccurs="0" type="CT_AutoSortScope"/>
1193 <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1194 </xsd:sequence>
1195 <xsd:attribute name="name" type="s:ST Xstring"/>
1196 <xsd:attribute name="axis" use="optional" type="ST Axis"/>
1197 <xsd:attribute name="dataField" type="xsd:boolean" default="false"/>
1198 <xsd:attribute name="subtotalCaption" type="s:ST Xstring"/>
1199 <xsd:attribute name="showDropDowns" type="xsd:boolean" default="true"/>
1200 <xsd:attribute name="hiddenLevel" type="xsd:boolean" default="false"/>
1201 <xsd:attribute name="uniqueMemberProperty" type="s:ST Xstring"/>

```

```

1202 <xsd:attribute name="compact" type="xsd:boolean" default="true"/>
1203 <xsd:attribute name="allDrilled" type="xsd:boolean" default="false"/>
1204 <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
1205 <xsd:attribute name="outline" type="xsd:boolean" default="true"/>
1206 <xsd:attribute name="subtotalTop" type="xsd:boolean" default="true"/>
1207 <xsd:attribute name="dragToRow" type="xsd:boolean" default="true"/>
1208 <xsd:attribute name="dragToCol" type="xsd:boolean" default="true"/>
1209 <xsd:attribute name="multipleItemSelectionAllowed" type="xsd:boolean" default="false"/>
1210 <xsd:attribute name="dragToPage" type="xsd:boolean" default="true"/>
1211 <xsd:attribute name="dragToData" type="xsd:boolean" default="true"/>
1212 <xsd:attribute name="dragOff" type="xsd:boolean" default="true"/>
1213 <xsd:attribute name="showAll" type="xsd:boolean" default="true"/>
1214 <xsd:attribute name="insertBlankRow" type="xsd:boolean" default="false"/>
1215 <xsd:attribute name="serverField" type="xsd:boolean" default="false"/>
1216 <xsd:attribute name="insertPageBreak" type="xsd:boolean" default="false"/>
1217 <xsd:attribute name="autoShow" type="xsd:boolean" default="false"/>
1218 <xsd:attribute name="topAutoShow" type="xsd:boolean" default="true"/>
1219 <xsd:attribute name="hideNewItems" type="xsd:boolean" default="false"/>
1220 <xsd:attribute name="measureFilter" type="xsd:boolean" default="false"/>
1221 <xsd:attribute name="includeNewItemsInFilter" type="xsd:boolean" default="false"/>
1222 <xsd:attribute name="itemPageCount" type="xsd:unsignedInt" default="10"/>
1223 <xsd:attribute name="sortType" type="ST_FieldSortType" default="manual"/>
1224 <xsd:attribute name="dataSourceSort" type="xsd:boolean" use="optional"/>
1225 <xsd:attribute name="nonAutoSortDefault" type="xsd:boolean" default="false"/>
1226 <xsd:attribute name="rankBy" type="xsd:unsignedInt" use="optional"/>
1227 <xsd:attribute name="defaultSubtotal" type="xsd:boolean" default="true"/>
1228 <xsd:attribute name="sumSubtotal" type="xsd:boolean" default="false"/>
1229 <xsd:attribute name="countASubtotal" type="xsd:boolean" default="false"/>
1230 <xsd:attribute name="avgSubtotal" type="xsd:boolean" default="false"/>
1231 <xsd:attribute name="maxSubtotal" type="xsd:boolean" default="false"/>
1232 <xsd:attribute name="minSubtotal" type="xsd:boolean" default="false"/>
1233 <xsd:attribute name="productSubtotal" type="xsd:boolean" default="false"/>
1234 <xsd:attribute name="countSubtotal" type="xsd:boolean" default="false"/>
1235 <xsd:attribute name="stdDevSubtotal" type="xsd:boolean" default="false"/>
1236 <xsd:attribute name="stdDevPSubtotal" type="xsd:boolean" default="false"/>
1237 <xsd:attribute name="varSubtotal" type="xsd:boolean" default="false"/>
1238 <xsd:attribute name="varPSubtotal" type="xsd:boolean" default="false"/>
1239 <xsd:attribute name="showPropCell" type="xsd:boolean" use="optional" default="false"/>
1240 <xsd:attribute name="showPropTip" type="xsd:boolean" use="optional" default="false"/>
1241 <xsd:attribute name="showPropAsCaption" type="xsd:boolean" use="optional" default="false"/>
1242 <xsd:attribute name="defaultAttributeDrillState" type="xsd:boolean" use="optional"
1243     default="false"/>
1244 </xsd:complexType>
1245 <xsd:complexType name="CT_AutoSortScope">
1246     <xsd:sequence>
1247         <xsd:element name="pivotArea" type="CT_PivotArea"/>
1248     </xsd:sequence>
1249 </xsd:complexType>
1250 <xsd:complexType name="CT_Items">
1251     <xsd:sequence>
1252         <xsd:element name="item" maxOccurs="unbounded" type="CT_Item"/>
1253     </xsd:sequence>
1254     <xsd:attribute name="count" type="xsd:unsignedInt"/>

```

```

1255 </xsd:complexType>
1256 <xsd:complexType name="CT_Item">
1257     <xsd:attribute name="n" type="s:ST_Xstring"/>
1258     <xsd:attribute name="t" type="ST_ItemType" default="data"/>
1259     <xsd:attribute name="h" type="xsd:boolean" default="false"/>
1260     <xsd:attribute name="s" type="xsd:boolean" default="false"/>
1261     <xsd:attribute name="sd" type="xsd:boolean" default="true"/>
1262     <xsd:attribute name="f" type="xsd:boolean" default="false"/>
1263     <xsd:attribute name="m" type="xsd:boolean" default="false"/>
1264     <xsd:attribute name="c" type="xsd:boolean" default="false"/>
1265     <xsd:attribute name="x" type="xsd:unsignedInt" use="optional"/>
1266     <xsd:attribute name="d" type="xsd:boolean" default="false"/>
1267     <xsd:attribute name="e" type="xsd:boolean" default="true"/>
1268 </xsd:complexType>
1269 <xsd:complexType name="CT_PageFields">
1270     <xsd:sequence>
1271         <xsd:element name="pageField" maxOccurs="unbounded" type="CT_PageField"/>
1272     </xsd:sequence>
1273     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1274 </xsd:complexType>
1275 <xsd:complexType name="CT_PageField">
1276     <xsd:sequence minOccurs="0">
1277         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1278     </xsd:sequence>
1279     <xsd:attribute name="fld" use="required" type="xsd:int"/>
1280     <xsd:attribute name="item" use="optional" type="xsd:unsignedInt"/>
1281     <xsd:attribute name="hier" type="xsd:int"/>
1282     <xsd:attribute name="name" type="s:ST_Xstring"/>
1283     <xsd:attribute name="cap" type="s:ST_Xstring"/>
1284 </xsd:complexType>
1285 <xsd:complexType name="CT_DataFields">
1286     <xsd:sequence>
1287         <xsd:element name="dataField" maxOccurs="unbounded" type="CT_DataField"/>
1288     </xsd:sequence>
1289     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1290 </xsd:complexType>
1291 <xsd:complexType name="CT_DataField">
1292     <xsd:sequence>
1293         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1294     </xsd:sequence>
1295     <xsd:attribute name="name" use="optional" type="s:ST_Xstring"/>
1296     <xsd:attribute name="fld" type="xsd:unsignedInt" use="required"/>
1297     <xsd:attribute name="subtotal" type="ST_DataConsolidateFunction" default="sum"/>
1298     <xsd:attribute name="showDataAs" type="ST_ShowDataAs" default="normal"/>
1299     <xsd:attribute name="baseField" type="xsd:int" default="-1"/>
1300     <xsd:attribute name="baseItem" type="xsd:unsignedInt" default="1048832"/>
1301     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
1302 </xsd:complexType>
1303 <xsd:complexType name="CT_rowItems">
1304     <xsd:sequence>
1305         <xsd:element name="i" maxOccurs="unbounded" type="CT_I"/>
1306     </xsd:sequence>
1307     <xsd:attribute name="count" type="xsd:unsignedInt"/>

```

```

1308 </xsd:complexType>
1309 <xsd:complexType name="CT_colItems">
1310   <xsd:sequence>
1311     <xsd:element name="i" maxOccurs="unbounded" type="CT_I"/>
1312   </xsd:sequence>
1313   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1314 </xsd:complexType>
1315 <xsd:complexType name="CT_I">
1316   <xsd:sequence>
1317     <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_X"/>
1318   </xsd:sequence>
1319   <xsd:attribute name="t" type="ST_ItemType" default="data"/>
1320   <xsd:attribute name="r" type="xsd:unsignedInt" default="0"/>
1321   <xsd:attribute name="i" type="xsd:unsignedInt" default="0"/>
1322 </xsd:complexType>
1323 <xsd:complexType name="CT_X">
1324   <xsd:attribute name="v" type="xsd:int" default="0"/>
1325 </xsd:complexType>
1326 <xsd:complexType name="CT_RowFields">
1327   <xsd:sequence>
1328     <xsd:element name="field" maxOccurs="unbounded" type="CT_Field"/>
1329   </xsd:sequence>
1330   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1331 </xsd:complexType>
1332 <xsd:complexType name="CT_ColFields">
1333   <xsd:sequence>
1334     <xsd:element name="field" maxOccurs="unbounded" type="CT_Field"/>
1335   </xsd:sequence>
1336   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1337 </xsd:complexType>
1338 <xsd:complexType name="CT_Field">
1339   <xsd:attribute name="x" type="xsd:int" use="required"/>
1340 </xsd:complexType>
1341 <xsd:complexType name="CT_Formats">
1342   <xsd:sequence>
1343     <xsd:element name="format" maxOccurs="unbounded" type="CT_Format"/>
1344   </xsd:sequence>
1345   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1346 </xsd:complexType>
1347 <xsd:complexType name="CT_Format">
1348   <xsd:sequence>
1349     <xsd:element name="pivotArea" type="CT_PivotArea"/>
1350     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1351   </xsd:sequence>
1352   <xsd:attribute name="action" type="ST_FormatAction" default="formatting"/>
1353   <xsd:attribute name="dxId" type="ST_DxfId" use="optional"/>
1354 </xsd:complexType>
1355 <xsd:complexType name="CT_ConditionalFormats">
1356   <xsd:sequence>
1357     <xsd:element name="conditionalFormat" maxOccurs="unbounded" type="CT_ConditionalFormat"/>
1358   </xsd:sequence>
1359   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1360 </xsd:complexType>

```

```

1361 <xsd:complexType name="CT_ConditionalFormat">
1362   <xsd:sequence>
1363     <xsd:element name="pivotAreas" type="CT_PivotAreas"/>
1364     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1365   </xsd:sequence>
1366   <xsd:attribute name="scope" type="ST_Scope" default="selection"/>
1367   <xsd:attribute name="type" type="ST_Type" default="none"/>
1368   <xsd:attribute name="priority" use="required" type="xsd:unsignedInt"/>
1369 </xsd:complexType>
1370 <xsd:complexType name="CT_PivotAreas">
1371   <xsd:sequence>
1372     <xsd:element name="pivotArea" minOccurs="0" maxOccurs="unbounded" type="CT_PivotArea"/>
1373   </xsd:sequence>
1374   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1375 </xsd:complexType>
1376 <xsd:simpleType name="ST_Scope">
1377   <xsd:restriction base="xsd:string">
1378     <xsd:enumeration value="selection"/>
1379     <xsd:enumeration value="data"/>
1380     <xsd:enumeration value="field"/>
1381   </xsd:restriction>
1382 </xsd:simpleType>
1383 <xsd:simpleType name="ST_Type">
1384   <xsd:restriction base="xsd:string">
1385     <xsd:enumeration value="none"/>
1386     <xsd:enumeration value="all"/>
1387     <xsd:enumeration value="row"/>
1388     <xsd:enumeration value="column"/>
1389   </xsd:restriction>
1390 </xsd:simpleType>
1391 <xsd:complexType name="CT_ChartFormats">
1392   <xsd:sequence>
1393     <xsd:element name="chartFormat" maxOccurs="unbounded" type="CT_ChartFormat"/>
1394   </xsd:sequence>
1395   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1396 </xsd:complexType>
1397 <xsd:complexType name="CT_ChartFormat">
1398   <xsd:sequence>
1399     <xsd:element name="pivotArea" type="CT_PivotArea"/>
1400   </xsd:sequence>
1401   <xsd:attribute name="chart" use="required" type="xsd:unsignedInt"/>
1402   <xsd:attribute name="format" use="required" type="xsd:unsignedInt"/>
1403   <xsd:attribute name="series" type="xsd:boolean" default="false"/>
1404 </xsd:complexType>
1405 <xsd:complexType name="CT_PivotHierarchies">
1406   <xsd:sequence>
1407     <xsd:element name="pivotHierarchy" maxOccurs="unbounded" type="CT_PivotHierarchy"/>
1408   </xsd:sequence>
1409   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1410 </xsd:complexType>
1411 <xsd:complexType name="CT_PivotHierarchy">
1412   <xsd:sequence>
1413     <xsd:element name="mps" minOccurs="0" type="CT_MemberProperties"/>

```

```

1414     <xsd:element name="members" minOccurs="0" maxOccurs="unbounded" type="CT_Members"/>
1415     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1416 </xsd:sequence>
1417 <xsd:attribute name="outline" type="xsd:boolean" default="false"/>
1418 <xsd:attribute name="multipleItemSelectionAllowed" type="xsd:boolean" default="false"/>
1419 <xsd:attribute name="subtotalTop" type="xsd:boolean" default="false"/>
1420 <xsd:attribute name="showInFieldList" type="xsd:boolean" default="true"/>
1421 <xsd:attribute name="dragToRow" type="xsd:boolean" default="true"/>
1422 <xsd:attribute name="dragToCol" type="xsd:boolean" default="true"/>
1423 <xsd:attribute name="dragToPage" type="xsd:boolean" default="true"/>
1424 <xsd:attribute name="dragToData" type="xsd:boolean" default="false"/>
1425 <xsd:attribute name="dragOff" type="xsd:boolean" default="true"/>
1426 <xsd:attribute name="includeNewItemInFilter" type="xsd:boolean" default="false"/>
1427 <xsd:attribute name="caption" type="s:ST_Xstring" use="optional"/>
1428 </xsd:complexType>
1429 <xsd:complexType name="CT_RowHierarchiesUsage">
1430   <xsd:sequence>
1431     <xsd:element name="rowHierarchyUsage" minOccurs="1" maxOccurs="unbounded"
1432       type="CT_HierarchyUsage"/>
1433   </xsd:sequence>
1434   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1435 </xsd:complexType>
1436 <xsd:complexType name="CT_ColHierarchiesUsage">
1437   <xsd:sequence>
1438     <xsd:element name="colHierarchyUsage" minOccurs="1" maxOccurs="unbounded"
1439       type="CT_HierarchyUsage"/>
1440   </xsd:sequence>
1441   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1442 </xsd:complexType>
1443 <xsd:complexType name="CT_HierarchyUsage">
1444   <xsd:attribute name="hierarchyUsage" type="xsd:int" use="required"/>
1445 </xsd:complexType>
1446 <xsd:complexType name="CT_MemberProperties">
1447   <xsd:sequence>
1448     <xsd:element name="mp" maxOccurs="unbounded" type="CT_MemberProperty"/>
1449   </xsd:sequence>
1450   <xsd:attribute name="count" type="xsd:unsignedInt"/>
1451 </xsd:complexType>
1452 <xsd:complexType name="CT_MemberProperty">
1453   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
1454   <xsd:attribute name="showCell" type="xsd:boolean" use="optional" default="false"/>
1455   <xsd:attribute name="showTip" type="xsd:boolean" use="optional" default="false"/>
1456   <xsd:attribute name="showAsCaption" type="xsd:boolean" use="optional" default="false"/>
1457   <xsd:attribute name="nameLen" type="xsd:unsignedInt" use="optional"/>
1458   <xsd:attribute name="pPos" type="xsd:unsignedInt" use="optional"/>
1459   <xsd:attribute name="pLen" type="xsd:unsignedInt" use="optional"/>
1460   <xsd:attribute name="level" type="xsd:unsignedInt" use="optional"/>
1461   <xsd:attribute name="field" use="required" type="xsd:unsignedInt"/>
1462 </xsd:complexType>
1463 <xsd:complexType name="CT_Members">
1464   <xsd:sequence>
1465     <xsd:element name="member" maxOccurs="unbounded" type="CT_Member"/>
1466   </xsd:sequence>

```



```

1467     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1468     <xsd:attribute name="level" use="optional" type="xsd:unsignedInt"/>
1469 </xsd:complexType>
1470 <xsd:complexType name="CT_Member">
1471     <xsd:attribute name="name" use="required" type="s:ST Xstring"/>
1472 </xsd:complexType>
1473 <xsd:complexType name="CT_Dimensions">
1474     <xsd:sequence>
1475         <xsd:element name="dimension" minOccurs="0" maxOccurs="unbounded"
1476             type="CT_PivotDimension"/>
1477     </xsd:sequence>
1478     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1479 </xsd:complexType>
1480 <xsd:complexType name="CT_PivotDimension">
1481     <xsd:attribute name="measure" type="xsd:boolean" default="false"/>
1482     <xsd:attribute name="name" use="required" type="s:ST Xstring"/>
1483     <xsd:attribute name="uniqueName" use="required" type="s:ST Xstring"/>
1484     <xsd:attribute name="caption" use="required" type="s:ST Xstring"/>
1485 </xsd:complexType>
1486 <xsd:complexType name="CT_MeasureGroups">
1487     <xsd:sequence>
1488         <xsd:element name="measureGroup" minOccurs="0" maxOccurs="unbounded"
1489             type="CT_MeasureGroup"/>
1490     </xsd:sequence>
1491     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1492 </xsd:complexType>
1493 <xsd:complexType name="CT_MeasureDimensionMaps">
1494     <xsd:sequence>
1495         <xsd:element name="map" minOccurs="0" maxOccurs="unbounded"
1496             type="CT_MeasureDimensionMap"/>
1497     </xsd:sequence>
1498     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1499 </xsd:complexType>
1500 <xsd:complexType name="CT_MeasureGroup">
1501     <xsd:attribute name="name" use="required" type="s:ST Xstring"/>
1502     <xsd:attribute name="caption" use="required" type="s:ST Xstring"/>
1503 </xsd:complexType>
1504 <xsd:complexType name="CT_MeasureDimensionMap">
1505     <xsd:attribute name="measureGroup" use="optional" type="xsd:unsignedInt"/>
1506     <xsd:attribute name="dimension" use="optional" type="xsd:unsignedInt"/>
1507 </xsd:complexType>
1508 <xsd:complexType name="CT_PivotTableStyle">
1509     <xsd:attribute name="name" type="xsd:string"/>
1510     <xsd:attribute name="showRowHeaders" type="xsd:boolean"/>
1511     <xsd:attribute name="showColHeaders" type="xsd:boolean"/>
1512     <xsd:attribute name="showRowStripes" type="xsd:boolean"/>
1513     <xsd:attribute name="showColStripes" type="xsd:boolean"/>
1514     <xsd:attribute name="showLastColumn" type="xsd:boolean" use="optional"/>
1515 </xsd:complexType>
1516 <xsd:complexType name="CT_PivotFilters">
1517     <xsd:sequence>
1518         <xsd:element name="filter" minOccurs="0" maxOccurs="unbounded" type="CT_PivotFilter"/>
1519     </xsd:sequence>

```

```

1520     <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
1521 </xsd:complexType>
1522 <xsd:complexType name="CT_PivotFilter">
1523     <xsd:sequence>
1524         <xsd:element name="autoFilter" minOccurs="1" maxOccurs="1" type="CT_AutoFilter"/>
1525         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1526     </xsd:sequence>
1527     <xsd:attribute name="fld" use="required" type="xsd:unsignedInt"/>
1528     <xsd:attribute name="mpFld" type="xsd:unsignedInt" use="optional"/>
1529     <xsd:attribute name="type" use="required" type="ST_PivotFilterType"/>
1530     <xsd:attribute name="evalOrder" use="optional" type="xsd:int" default="0"/>
1531     <xsd:attribute name="id" use="required" type="xsd:unsignedInt"/>
1532     <xsd:attribute name="iMeasureHier" use="optional" type="xsd:unsignedInt"/>
1533     <xsd:attribute name="iMeasureFld" use="optional" type="xsd:unsignedInt"/>
1534     <xsd:attribute name="name" type="s:ST_Xstring"/>
1535     <xsd:attribute name="description" type="s:ST_Xstring"/>
1536     <xsd:attribute name="stringValue1" type="s:ST_Xstring"/>
1537     <xsd:attribute name="stringValue2" type="s:ST_Xstring"/>
1538 </xsd:complexType>
1539 <xsd:simpleType name="ST_ShowDataAs">
1540     <xsd:restriction base="xsd:string">
1541         <xsd:enumeration value="normal"/>
1542         <xsd:enumeration value="difference"/>
1543         <xsd:enumeration value="percent"/>
1544         <xsd:enumeration value="percentDiff"/>
1545         <xsd:enumeration value="runTotal"/>
1546         <xsd:enumeration value="percentOfRow"/>
1547         <xsd:enumeration value="percentOfCol"/>
1548         <xsd:enumeration value="percentOfTotal"/>
1549         <xsd:enumeration value="index"/>
1550     </xsd:restriction>
1551 </xsd:simpleType>
1552 <xsd:simpleType name="ST_ItemType">
1553     <xsd:restriction base="xsd:string">
1554         <xsd:enumeration value="data"/>
1555         <xsd:enumeration value="default"/>
1556         <xsd:enumeration value="sum"/>
1557         <xsd:enumeration value="countA"/>
1558         <xsd:enumeration value="avg"/>
1559         <xsd:enumeration value="max"/>
1560         <xsd:enumeration value="min"/>
1561         <xsd:enumeration value="product"/>
1562         <xsd:enumeration value="count"/>
1563         <xsd:enumeration value="stdDev"/>
1564         <xsd:enumeration value="stdDevP"/>
1565         <xsd:enumeration value="var"/>
1566         <xsd:enumeration value="varP"/>
1567         <xsd:enumeration value="grand"/>
1568         <xsd:enumeration value="blank"/>
1569     </xsd:restriction>
1570 </xsd:simpleType>
1571 <xsd:simpleType name="ST_FormatAction">
1572     <xsd:restriction base="xsd:string">

```

```

1573         <xsd:enumeration value="blank"/>
1574         <xsd:enumeration value="formatting"/>
1575         <xsd:enumeration value="drill"/>
1576         <xsd:enumeration value="formula"/>
1577     </xsd:restriction>
1578 </xsd:simpleType>
1579 <xsd:simpleType name="ST_FieldSortType">
1580     <xsd:restriction base="xsd:string">
1581         <xsd:enumeration value="manual"/>
1582         <xsd:enumeration value="ascending"/>
1583         <xsd:enumeration value="descending"/>
1584     </xsd:restriction>
1585 </xsd:simpleType>
1586 <xsd:simpleType name="ST_PivotFilterType">
1587     <xsd:restriction base="xsd:string">
1588         <xsd:enumeration value="unknown"/>
1589         <xsd:enumeration value="count"/>
1590         <xsd:enumeration value="percent"/>
1591         <xsd:enumeration value="sum"/>
1592         <xsd:enumeration value="captionEqual"/>
1593         <xsd:enumeration value="captionNotEqual"/>
1594         <xsd:enumeration value="captionBeginsWith"/>
1595         <xsd:enumeration value="captionNotBeginsWith"/>
1596         <xsd:enumeration value="captionEndsWith"/>
1597         <xsd:enumeration value="captionNotEndsWith"/>
1598         <xsd:enumeration value="captionContains"/>
1599         <xsd:enumeration value="captionNotContains"/>
1600         <xsd:enumeration value="captionGreaterThan"/>
1601         <xsd:enumeration value="captionGreaterThanOrEqual"/>
1602         <xsd:enumeration value="captionLessThan"/>
1603         <xsd:enumeration value="captionLessThanOrEqual"/>
1604         <xsd:enumeration value="captionBetween"/>
1605         <xsd:enumeration value="captionNotBetween"/>
1606         <xsd:enumeration value="valueEqual"/>
1607         <xsd:enumeration value="valueNotEqual"/>
1608         <xsd:enumeration value="valueGreaterThan"/>
1609         <xsd:enumeration value="valueGreaterThanOrEqual"/>
1610         <xsd:enumeration value="valueLessThan"/>
1611         <xsd:enumeration value="valueLessThanOrEqual"/>
1612         <xsd:enumeration value="valueBetween"/>
1613         <xsd:enumeration value="valueNotBetween"/>
1614         <xsd:enumeration value="dateEqual"/>
1615         <xsd:enumeration value="dateNotEqual"/>
1616         <xsd:enumeration value="dateOlderThan"/>
1617         <xsd:enumeration value="dateOlderThanOrEqual"/>
1618         <xsd:enumeration value="dateNewerThan"/>
1619         <xsd:enumeration value="dateNewerThanOrEqual"/>
1620         <xsd:enumeration value="dateBetween"/>
1621         <xsd:enumeration value="dateNotBetween"/>
1622         <xsd:enumeration value="tomorrow"/>
1623         <xsd:enumeration value="today"/>
1624         <xsd:enumeration value="yesterday"/>
1625         <xsd:enumeration value="nextWeek"/>

```

```

1626     <xsd:enumeration value="thisWeek"/>
1627     <xsd:enumeration value="lastWeek"/>
1628     <xsd:enumeration value="nextMonth"/>
1629     <xsd:enumeration value="thisMonth"/>
1630     <xsd:enumeration value="lastMonth"/>
1631     <xsd:enumeration value="nextQuarter"/>
1632     <xsd:enumeration value="thisQuarter"/>
1633     <xsd:enumeration value="lastQuarter"/>
1634     <xsd:enumeration value="nextYear"/>
1635     <xsd:enumeration value="thisYear"/>
1636     <xsd:enumeration value="lastYear"/>
1637     <xsd:enumeration value="yearToDate"/>
1638     <xsd:enumeration value="Q1"/>
1639     <xsd:enumeration value="Q2"/>
1640     <xsd:enumeration value="Q3"/>
1641     <xsd:enumeration value="Q4"/>
1642     <xsd:enumeration value="M1"/>
1643     <xsd:enumeration value="M2"/>
1644     <xsd:enumeration value="M3"/>
1645     <xsd:enumeration value="M4"/>
1646     <xsd:enumeration value="M5"/>
1647     <xsd:enumeration value="M6"/>
1648     <xsd:enumeration value="M7"/>
1649     <xsd:enumeration value="M8"/>
1650     <xsd:enumeration value="M9"/>
1651     <xsd:enumeration value="M10"/>
1652     <xsd:enumeration value="M11"/>
1653     <xsd:enumeration value="M12"/>
1654 </xsd:restriction>
1655 </xsd:simpleType>
1656 <xsd:complexType name="CT_PivotArea">
1657     <xsd:sequence>
1658         <xsd:element name="references" minOccurs="0" type="CT_PivotAreaReferences"/>
1659         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1660     </xsd:sequence>
1661     <xsd:attribute name="field" use="optional" type="xsd:int"/>
1662     <xsd:attribute name="type" type="ST_PivotAreaType" default="normal"/>
1663     <xsd:attribute name="dataOnly" type="xsd:boolean" default="true"/>
1664     <xsd:attribute name="labelOnly" type="xsd:boolean" default="false"/>
1665     <xsd:attribute name="grandRow" type="xsd:boolean" default="false"/>
1666     <xsd:attribute name="grandCol" type="xsd:boolean" default="false"/>
1667     <xsd:attribute name="cacheIndex" type="xsd:boolean" default="false"/>
1668     <xsd:attribute name="outline" type="xsd:boolean" default="true"/>
1669     <xsd:attribute name="offset" type="ST_Ref"/>
1670     <xsd:attribute name="collapsedLevelsAreSubtotals" type="xsd:boolean" default="false"/>
1671     <xsd:attribute name="axis" type="ST_Axis" use="optional"/>
1672     <xsd:attribute name="fieldPosition" type="xsd:unsignedInt" use="optional"/>
1673 </xsd:complexType>
1674 <xsd:simpleType name="ST_PivotAreaType">
1675     <xsd:restriction base="xsd:string">
1676         <xsd:enumeration value="none"/>
1677         <xsd:enumeration value="normal"/>
1678         <xsd:enumeration value="data"/>

```

```

1679         <xsd:enumeration value="all"/>
1680         <xsd:enumeration value="origin"/>
1681         <xsd:enumeration value="button"/>
1682         <xsd:enumeration value="topEnd"/>
1683         <xsd:enumeration value="topRight"/>
1684     </xsd:restriction>
1685 </xsd:simpleType>
1686 <xsd:complexType name="CT_PivotAreaReferences">
1687     <xsd:sequence>
1688         <xsd:element name="reference" maxOccurs="unbounded" type="CT_PivotAreaReference"/>
1689     </xsd:sequence>
1690     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1691 </xsd:complexType>
1692 <xsd:complexType name="CT_PivotAreaReference">
1693     <xsd:sequence>
1694         <xsd:element name="x" minOccurs="0" maxOccurs="unbounded" type="CT_Index"/>
1695         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1696     </xsd:sequence>
1697     <xsd:attribute name="field" use="optional" type="xsd:unsignedInt"/>
1698     <xsd:attribute name="count" type="xsd:unsignedInt"/>
1699     <xsd:attribute name="selected" type="xsd:boolean" default="true"/>
1700     <xsd:attribute name="byPosition" type="xsd:boolean" default="false"/>
1701     <xsd:attribute name="relative" type="xsd:boolean" default="false"/>
1702     <xsd:attribute name="defaultSubtotal" type="xsd:boolean" default="false"/>
1703     <xsd:attribute name="sumSubtotal" type="xsd:boolean" default="false"/>
1704     <xsd:attribute name="countASubtotal" type="xsd:boolean" default="false"/>
1705     <xsd:attribute name="avgSubtotal" type="xsd:boolean" default="false"/>
1706     <xsd:attribute name="maxSubtotal" type="xsd:boolean" default="false"/>
1707     <xsd:attribute name="minSubtotal" type="xsd:boolean" default="false"/>
1708     <xsd:attribute name="productSubtotal" type="xsd:boolean" default="false"/>
1709     <xsd:attribute name="countSubtotal" type="xsd:boolean" default="false"/>
1710     <xsd:attribute name="stdDevSubtotal" type="xsd:boolean" default="false"/>
1711     <xsd:attribute name="stdDevPSubtotal" type="xsd:boolean" default="false"/>
1712     <xsd:attribute name="varSubtotal" type="xsd:boolean" default="false"/>
1713     <xsd:attribute name="varPSubtotal" type="xsd:boolean" default="false"/>
1714 </xsd:complexType>
1715 <xsd:complexType name="CT_Index">
1716     <xsd:attribute name="v" use="required" type="xsd:unsignedInt"/>
1717 </xsd:complexType>
1718 <xsd:simpleType name="ST_Axis">
1719     <xsd:restriction base="xsd:string">
1720         <xsd:enumeration value="axisRow"/>
1721         <xsd:enumeration value="axisCol"/>
1722         <xsd:enumeration value="axisPage"/>
1723         <xsd:enumeration value="axisValues"/>
1724     </xsd:restriction>
1725 </xsd:simpleType>
1726 <xsd:element name="queryTable" type="CT_QueryTable"/>
1727 <xsd:complexType name="CT_QueryTable">
1728     <xsd:sequence>
1729         <xsd:element name="queryTableRefresh" type="CT_QueryTableRefresh" minOccurs="0"
1730             maxOccurs="1"/>
1731         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

1732     </xsd:sequence>
1733     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
1734     <xsd:attribute name="headers" type="xsd:boolean" use="optional" default="true"/>
1735     <xsd:attribute name="rowNumbers" type="xsd:boolean" use="optional" default="false"/>
1736     <xsd:attribute name="disableRefresh" type="xsd:boolean" use="optional" default="false"/>
1737     <xsd:attribute name="backgroundRefresh" type="xsd:boolean" use="optional" default="true"/>
1738     <xsd:attribute name="firstBackgroundRefresh" type="xsd:boolean" use="optional"
1739         default="false"/>
1740     <xsd:attribute name="refreshOnLoad" type="xsd:boolean" use="optional" default="false"/>
1741     <xsd:attribute name="growShrinkType" type="ST GrowShrinkType" use="optional"
1742         default="insertDelete"/>
1743     <xsd:attribute name="fillFormulas" type="xsd:boolean" use="optional" default="false"/>
1744     <xsd:attribute name="removeDataOnSave" type="xsd:boolean" use="optional" default="false"/>
1745     <xsd:attribute name="disableEdit" type="xsd:boolean" use="optional" default="false"/>
1746     <xsd:attribute name="preserveFormatting" type="xsd:boolean" use="optional" default="true"/>
1747     <xsd:attribute name="adjustColumnWidth" type="xsd:boolean" use="optional" default="true"/>
1748     <xsd:attribute name="intermediate" type="xsd:boolean" use="optional" default="false"/>
1749     <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="required"/>
1750     <xsd:attributeGroup ref="AG AutoFormat"/>
1751 </xsd:complexType>
1752 <xsd:complexType name="CT_QueryTableRefresh">
1753     <xsd:sequence>
1754         <xsd:element name="queryTableFields" type="CT_QueryTableFields" minOccurs="1"
1755             maxOccurs="1"/>
1756         <xsd:element name="queryTableDeletedFields" type="CT_QueryTableDeletedFields"
1757             minOccurs="0" maxOccurs="1"/>
1758         <xsd:element name="sortState" minOccurs="0" maxOccurs="1" type="CT_SortState"/>
1759         <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
1760     </xsd:sequence>
1761     <xsd:attribute name="preserveSortFilterLayout" type="xsd:boolean" use="optional"
1762         default="true"/>
1763     <xsd:attribute name="fieldIdWrapped" type="xsd:boolean" use="optional" default="false"/>
1764     <xsd:attribute name="headersInLastRefresh" type="xsd:boolean" use="optional" default="true"/>
1765     <xsd:attribute name="minimumVersion" type="xsd:unsignedByte" use="optional" default="0"/>
1766     <xsd:attribute name="nextId" type="xsd:unsignedInt" use="optional" default="1"/>
1767     <xsd:attribute name="unboundColumnsLeft" type="xsd:unsignedInt" use="optional" default="0"/>
1768     <xsd:attribute name="unboundColumnsRight" type="xsd:unsignedInt" use="optional" default="0"/>
1769 </xsd:complexType>
1770 <xsd:complexType name="CT_QueryTableDeletedFields">
1771     <xsd:sequence>
1772         <xsd:element name="deletedField" type="CT_DeletedField" minOccurs="1"
1773             maxOccurs="unbounded"/>
1774     </xsd:sequence>
1775     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1776 </xsd:complexType>
1777 <xsd:complexType name="CT_DeletedField">
1778     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
1779 </xsd:complexType>
1780 <xsd:complexType name="CT_QueryTableFields">
1781     <xsd:sequence>
1782         <xsd:element name="queryTableField" type="CT_QueryTableField" minOccurs="0"
1783             maxOccurs="unbounded"/>
1784     </xsd:sequence>

```

```

1785     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
1786 </xsd:complexType>
1787 <xsd:complexType name="CT_QueryTableField">
1788     <xsd:sequence minOccurs="0">
1789         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1790     </xsd:sequence>
1791     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
1792     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
1793     <xsd:attribute name="dataBound" type="xsd:boolean" use="optional" default="true"/>
1794     <xsd:attribute name="rowNumbers" type="xsd:boolean" use="optional" default="false"/>
1795     <xsd:attribute name="fillFormulas" type="xsd:boolean" use="optional" default="false"/>
1796     <xsd:attribute name="clipped" type="xsd:boolean" use="optional" default="false"/>
1797     <xsd:attribute name="tableColumnId" type="xsd:unsignedInt" default="0"/>
1798 </xsd:complexType>
1799 <xsd:simpleType name="ST_GrowShrinkType">
1800     <xsd:restriction base="xsd:string">
1801         <xsd:enumeration value="insertDelete"/>
1802         <xsd:enumeration value="insertClear"/>
1803         <xsd:enumeration value="overwriteClear"/>
1804     </xsd:restriction>
1805 </xsd:simpleType>
1806 <xsd:element name="sst" type="CT_Sst"/>
1807 <xsd:complexType name="CT_Sst">
1808     <xsd:sequence>
1809         <xsd:element name="si" type="CT_Rst" minOccurs="0" maxOccurs="unbounded"/>
1810         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1811     </xsd:sequence>
1812     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1813     <xsd:attribute name="uniqueCount" type="xsd:unsignedInt" use="optional"/>
1814 </xsd:complexType>
1815 <xsd:simpleType name="ST_PhoneticType">
1816     <xsd:restriction base="xsd:string">
1817         <xsd:enumeration value="halfwidthKatakana"/>
1818         <xsd:enumeration value="fullwidthKatakana"/>
1819         <xsd:enumeration value="Hiragana"/>
1820         <xsd:enumeration value="noConversion"/>
1821     </xsd:restriction>
1822 </xsd:simpleType>
1823 <xsd:simpleType name="ST_PhoneticAlignment">
1824     <xsd:restriction base="xsd:string">
1825         <xsd:enumeration value="noControl"/>
1826         <xsd:enumeration value="left"/>
1827         <xsd:enumeration value="center"/>
1828         <xsd:enumeration value="distributed"/>
1829     </xsd:restriction>
1830 </xsd:simpleType>
1831 <xsd:complexType name="CT_PhoneticRun">
1832     <xsd:sequence>
1833         <xsd:element name="t" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
1834     </xsd:sequence>
1835     <xsd:attribute name="sb" type="xsd:unsignedInt" use="required"/>
1836     <xsd:attribute name="eb" type="xsd:unsignedInt" use="required"/>
1837 </xsd:complexType>

```

```

1838 <xsd:complexType name="CT_RElt">
1839   <xsd:sequence>
1840     <xsd:element name="rPr" type="CT_RPrElt" minOccurs="0" maxOccurs="1"/>
1841     <xsd:element name="t" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
1842   </xsd:sequence>
1843 </xsd:complexType>
1844 <xsd:complexType name="CT_RPrElt">
1845   <xsd:choice maxOccurs="unbounded">
1846     <xsd:element name="rFont" type="CT_FontName" minOccurs="0" maxOccurs="1"/>
1847     <xsd:element name="charset" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
1848     <xsd:element name="family" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
1849     <xsd:element name="b" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1850     <xsd:element name="i" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1851     <xsd:element name="strike" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1852     <xsd:element name="outline" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1853     <xsd:element name="shadow" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1854     <xsd:element name="condense" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1855     <xsd:element name="extend" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
1856     <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1857     <xsd:element name="sz" type="CT_FontSize" minOccurs="0" maxOccurs="1"/>
1858     <xsd:element name="u" type="CT_UnderlineProperty" minOccurs="0" maxOccurs="1"/>
1859     <xsd:element name="vertAlign" type="CT_VerticalAlignFontProperty" minOccurs="0"
1860       maxOccurs="1"/>
1861     <xsd:element name="scheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
1862   </xsd:choice>
1863 </xsd:complexType>
1864 <xsd:complexType name="CT_Rst">
1865   <xsd:sequence>
1866     <xsd:element name="t" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
1867     <xsd:element name="r" type="CT_RElt" minOccurs="0" maxOccurs="unbounded"/>
1868     <xsd:element name="rPh" type="CT_PhoneticRun" minOccurs="0" maxOccurs="unbounded"/>
1869     <xsd:element name="phoneticPr" minOccurs="0" maxOccurs="1" type="CT_PhoneticPr"/>
1870   </xsd:sequence>
1871 </xsd:complexType>
1872 <xsd:complexType name="CT_PhoneticPr">
1873   <xsd:attribute name="fontId" type="ST_FontId" use="required"/>
1874   <xsd:attribute name="type" type="ST_PhoneticType" use="optional" default="fullwidthKatakana"/>
1875   <xsd:attribute name="alignment" type="ST_PhoneticAlignment" use="optional" default="left"/>
1876 </xsd:complexType>
1877 <xsd:element name="headers" type="CT_RevisionHeaders"/>
1878 <xsd:element name="revisions" type="CT_Revisions"/>
1879 <xsd:complexType name="CT_RevisionHeaders">
1880   <xsd:sequence>
1881     <xsd:element name="header" type="CT_RevisionHeader" minOccurs="1" maxOccurs="unbounded"/>
1882   </xsd:sequence>
1883   <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
1884   <xsd:attribute name="lastGuid" type="s:ST_Guid" use="optional"/>
1885   <xsd:attribute name="shared" type="xsd:boolean" default="true"/>
1886   <xsd:attribute name="diskRevisions" type="xsd:boolean" default="false"/>
1887   <xsd:attribute name="history" type="xsd:boolean" default="true"/>
1888   <xsd:attribute name="trackRevisions" type="xsd:boolean" default="true"/>
1889   <xsd:attribute name="exclusive" type="xsd:boolean" default="false"/>
1890   <xsd:attribute name="revisionId" type="xsd:unsignedInt" default="0"/>

```



```

1891 <xsd:attribute name="version" type="xsd:int" default="1"/>
1892 <xsd:attribute name="keepChangeHistory" type="xsd:boolean" use="optional" default="true"/>
1893 <xsd:attribute name="protected" type="xsd:boolean" use="optional" default="false"/>
1894 <xsd:attribute name="preserveHistory" type="xsd:unsignedInt" default="30"/>
1895 </xsd:complexType>
1896 <xsd:complexType name="CT_Revisions">
1897   <xsd:choice maxOccurs="unbounded">
1898     <xsd:element name="rrc" type="CT_RevisionRowColumn" minOccurs="0" maxOccurs="unbounded"/>
1899     <xsd:element name="rm" type="CT_RevisionMove" minOccurs="0" maxOccurs="unbounded"/>
1900     <xsd:element name="rcv" type="CT_RevisionCustomView" minOccurs="0" maxOccurs="unbounded"/>
1901     <xsd:element name="rsnm" type="CT_RevisionSheetRename" minOccurs="0"
1902       maxOccurs="unbounded"/>
1903     <xsd:element name="ris" type="CT_RevisionInsertSheet" minOccurs="0"
1904       maxOccurs="unbounded"/>
1905     <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1906     <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1907       maxOccurs="unbounded"/>
1908     <xsd:element name="raf" type="CT_RevisionAutoFormatting" minOccurs="0"
1909       maxOccurs="unbounded"/>
1910     <xsd:element name="rdn" type="CT_RevisionDefinedName" minOccurs="0"
1911       maxOccurs="unbounded"/>
1912     <xsd:element name="rcmt" type="CT_RevisionComment" minOccurs="0" maxOccurs="unbounded"/>
1913     <xsd:element name="rqt" type="CT_RevisionQueryTableField" minOccurs="0"
1914       maxOccurs="unbounded"/>
1915     <xsd:element name="rcft" type="CT_RevisionConflict" minOccurs="0" maxOccurs="unbounded"/>
1916   </xsd:choice>
1917 </xsd:complexType>
1918 <xsd:attributeGroup name="AG_RevData">
1919   <xsd:attribute name="rId" type="xsd:unsignedInt" use="required"/>
1920   <xsd:attribute name="ua" type="xsd:boolean" use="optional" default="false"/>
1921   <xsd:attribute name="ra" type="xsd:boolean" use="optional" default="false"/>
1922 </xsd:attributeGroup>
1923 <xsd:complexType name="CT_RevisionHeader">
1924   <xsd:sequence>
1925     <xsd:element name="sheetIdMap" minOccurs="1" maxOccurs="1" type="CT_SheetIdMap"/>
1926     <xsd:element name="reviewedList" minOccurs="0" maxOccurs="1" type="CT_ReviewedRevisions"/>
1927     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
1928   </xsd:sequence>
1929   <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
1930   <xsd:attribute name="dateTime" type="xsd:dateTime" use="required"/>
1931   <xsd:attribute name="maxSheetId" type="xsd:unsignedInt" use="required"/>
1932   <xsd:attribute name="userName" type="s:ST_Xstring" use="required"/>
1933   <xsd:attribute ref="r:id" use="required"/>
1934   <xsd:attribute name="minRId" type="xsd:unsignedInt" use="optional"/>
1935   <xsd:attribute name="maxRId" type="xsd:unsignedInt" use="optional"/>
1936 </xsd:complexType>
1937 <xsd:complexType name="CT_SheetIdMap">
1938   <xsd:sequence>
1939     <xsd:element name="sheetId" type="CT_SheetId" minOccurs="1" maxOccurs="unbounded"/>
1940   </xsd:sequence>
1941   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1942 </xsd:complexType>
1943 <xsd:complexType name="CT_SheetId">

```

```

1944     <xsd:attribute name="val" type="xsd:unsignedInt" use="required"/>
1945 </xsd:complexType>
1946 <xsd:complexType name="CT_ReviewedRevisions">
1947     <xsd:sequence>
1948         <xsd:element name="reviewed" type="CT_Reviewed" minOccurs="1" maxOccurs="unbounded"/>
1949     </xsd:sequence>
1950     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
1951 </xsd:complexType>
1952 <xsd:complexType name="CT_Reviewed">
1953     <xsd:attribute name="rId" type="xsd:unsignedInt" use="required"/>
1954 </xsd:complexType>
1955 <xsd:complexType name="CT_UndoInfo">
1956     <xsd:attribute name="index" type="xsd:unsignedInt" use="required"/>
1957     <xsd:attribute name="exp" type="ST_FormulaExpression" use="required"/>
1958     <xsd:attribute name="ref3D" type="xsd:boolean" use="optional" default="false"/>
1959     <xsd:attribute name="array" type="xsd:boolean" use="optional" default="false"/>
1960     <xsd:attribute name="v" type="xsd:boolean" use="optional" default="false"/>
1961     <xsd:attribute name="nf" type="xsd:boolean" use="optional" default="false"/>
1962     <xsd:attribute name="cs" type="xsd:boolean" use="optional" default="false"/>
1963     <xsd:attribute name="dr" type="ST_RefA" use="required"/>
1964     <xsd:attribute name="dn" type="s:ST_Xstring" use="optional"/>
1965     <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
1966     <xsd:attribute name="sId" type="xsd:unsignedInt" use="optional"/>
1967 </xsd:complexType>
1968 <xsd:complexType name="CT_RevisionRowColumn">
1969     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1970         <xsd:element name="undo" type="CT_UndoInfo" minOccurs="0" maxOccurs="unbounded"/>
1971         <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1972         <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1973             maxOccurs="unbounded"/>
1974     </xsd:choice>
1975     <xsd:attributeGroup ref="AG_RevData"/>
1976     <xsd:attribute name="sId" type="xsd:unsignedInt" use="required"/>
1977     <xsd:attribute name="eol" type="xsd:boolean" use="optional" default="false"/>
1978     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
1979     <xsd:attribute name="action" type="ST_rwColActionType" use="required"/>
1980     <xsd:attribute name="edge" type="xsd:boolean" use="optional" default="false"/>
1981 </xsd:complexType>
1982 <xsd:complexType name="CT_RevisionMove">
1983     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1984         <xsd:element name="undo" type="CT_UndoInfo" minOccurs="0" maxOccurs="unbounded"/>
1985         <xsd:element name="rcc" type="CT_RevisionCellChange" minOccurs="0" maxOccurs="unbounded"/>
1986         <xsd:element name="rfmt" type="CT_RevisionFormatting" minOccurs="0"
1987             maxOccurs="unbounded"/>
1988     </xsd:choice>
1989     <xsd:attributeGroup ref="AG_RevData"/>
1990     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
1991     <xsd:attribute name="source" type="ST_Ref" use="required"/>
1992     <xsd:attribute name="destination" type="ST_Ref" use="required"/>
1993     <xsd:attribute name="sourceSheetId" type="xsd:unsignedInt" use="optional" default="0"/>
1994 </xsd:complexType>
1995 <xsd:complexType name="CT_RevisionCustomView">
1996     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>

```

```

1997     <xsd:attribute name="action" type="ST_RevisionAction" use="required"/>
1998 </xsd:complexType>
1999 <xsd:complexType name="CT_RevisionSheetRename">
2000     <xsd:sequence>
2001         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2002     </xsd:sequence>
2003     <xsd:attributeGroup ref="AG_RevData"/>
2004     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2005     <xsd:attribute name="oldName" type="s:ST_Xstring" use="required"/>
2006     <xsd:attribute name="newName" type="s:ST_Xstring" use="required"/>
2007 </xsd:complexType>
2008 <xsd:complexType name="CT_RevisionInsertSheet">
2009     <xsd:attributeGroup ref="AG_RevData"/>
2010     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2011     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2012     <xsd:attribute name="sheetPosition" type="xsd:unsignedInt" use="required"/>
2013 </xsd:complexType>
2014 <xsd:complexType name="CT_RevisionCellChange">
2015     <xsd:sequence>
2016         <xsd:element name="oc" type="CT_Cell" minOccurs="0" maxOccurs="1"/>
2017         <xsd:element name="nc" type="CT_Cell" minOccurs="1" maxOccurs="1"/>
2018         <xsd:element name="odxf" type="CT_Dxf" minOccurs="0" maxOccurs="1"/>
2019         <xsd:element name="ndxf" type="CT_Dxf" minOccurs="0" maxOccurs="1"/>
2020         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2021     </xsd:sequence>
2022     <xsd:attributeGroup ref="AG_RevData"/>
2023     <xsd:attribute name="sId" type="xsd:unsignedInt" use="required"/>
2024     <xsd:attribute name="odxf" type="xsd:boolean" default="false"/>
2025     <xsd:attribute name="xfDxf" type="xsd:boolean" use="optional" default="false"/>
2026     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
2027     <xsd:attribute name="dxf" type="xsd:boolean" default="false"/>
2028     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
2029     <xsd:attribute name="quotePrefix" type="xsd:boolean" use="optional" default="false"/>
2030     <xsd:attribute name="oldQuotePrefix" type="xsd:boolean" use="optional" default="false"/>
2031     <xsd:attribute name="ph" type="xsd:boolean" default="false"/>
2032     <xsd:attribute name="oldPh" type="xsd:boolean" default="false"/>
2033     <xsd:attribute name="endOfListFormulaUpdate" type="xsd:boolean" default="false"/>
2034 </xsd:complexType>
2035 <xsd:complexType name="CT_RevisionFormatting">
2036     <xsd:sequence>
2037         <xsd:element name="dxf" type="CT_Dxf" minOccurs="0" maxOccurs="1"/>
2038         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2039     </xsd:sequence>
2040     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2041     <xsd:attribute name="xfDxf" type="xsd:boolean" use="optional" default="false"/>
2042     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
2043     <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
2044     <xsd:attribute name="start" type="xsd:unsignedInt" use="optional"/>
2045     <xsd:attribute name="length" type="xsd:unsignedInt" use="optional"/>
2046 </xsd:complexType>
2047 <xsd:complexType name="CT_RevisionAutoFormatting">
2048     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2049     <xsd:attributeGroup ref="AG_AutoFormat"/>

```

```

2050     <xsd:attribute name="ref" type="ST Ref" use="required"/>
2051 </xsd:complexType>
2052 <xsd:complexType name="CT_RevisionComment">
2053     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2054     <xsd:attribute name="cell" type="ST CellRef" use="required"/>
2055     <xsd:attribute name="guid" type="s:ST Guid" use="required"/>
2056     <xsd:attribute name="action" type="ST RevisionAction" default="add"/>
2057     <xsd:attribute name="alwaysShow" type="xsd:boolean" use="optional" default="false"/>
2058     <xsd:attribute name="old" type="xsd:boolean" use="optional" default="false"/>
2059     <xsd:attribute name="hiddenRow" type="xsd:boolean" use="optional" default="false"/>
2060     <xsd:attribute name="hiddenColumn" type="xsd:boolean" use="optional" default="false"/>
2061     <xsd:attribute name="author" type="s:ST Xstring" use="required"/>
2062     <xsd:attribute name="oldLength" type="xsd:unsignedInt" default="0"/>
2063     <xsd:attribute name="newLength" type="xsd:unsignedInt" default="0"/>
2064 </xsd:complexType>
2065 <xsd:complexType name="CT_RevisionDefinedName">
2066     <xsd:sequence>
2067         <xsd:element name="formula" type="ST Formula" minOccurs="0" maxOccurs="1"/>
2068         <xsd:element name="oldFormula" type="ST Formula" minOccurs="0" maxOccurs="1"/>
2069         <xsd:element name="extLst" minOccurs="0" type="CT ExtensionList"/>
2070     </xsd:sequence>
2071     <xsd:attributeGroup ref="AG RevData"/>
2072     <xsd:attribute name="localSheetId" type="xsd:unsignedInt" use="optional"/>
2073     <xsd:attribute name="customView" type="xsd:boolean" use="optional" default="false"/>
2074     <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
2075     <xsd:attribute name="function" type="xsd:boolean" use="optional" default="false"/>
2076     <xsd:attribute name="oldFunction" type="xsd:boolean" default="false"/>
2077     <xsd:attribute name="functionGroupId" type="xsd:unsignedByte" use="optional"/>
2078     <xsd:attribute name="oldFunctionGroupId" type="xsd:unsignedByte" use="optional"/>
2079     <xsd:attribute name="shortcutKey" type="xsd:unsignedByte" use="optional"/>
2080     <xsd:attribute name="oldShortcutKey" type="xsd:unsignedByte" use="optional"/>
2081     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2082     <xsd:attribute name="oldHidden" type="xsd:boolean" use="optional" default="false"/>
2083     <xsd:attribute name="customMenu" type="s:ST Xstring" use="optional"/>
2084     <xsd:attribute name="oldCustomMenu" type="s:ST Xstring" use="optional"/>
2085     <xsd:attribute name="description" type="s:ST Xstring" use="optional"/>
2086     <xsd:attribute name="oldDescription" type="s:ST Xstring" use="optional"/>
2087     <xsd:attribute name="help" type="s:ST Xstring" use="optional"/>
2088     <xsd:attribute name="oldHelp" type="s:ST Xstring" use="optional"/>
2089     <xsd:attribute name="statusBar" type="s:ST Xstring" use="optional"/>
2090     <xsd:attribute name="oldStatusBar" type="s:ST Xstring" use="optional"/>
2091     <xsd:attribute name="comment" type="s:ST Xstring" use="optional"/>
2092     <xsd:attribute name="oldComment" type="s:ST Xstring" use="optional"/>
2093 </xsd:complexType>
2094 <xsd:complexType name="CT_RevisionConflict">
2095     <xsd:attributeGroup ref="AG RevData"/>
2096     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="optional"/>
2097 </xsd:complexType>
2098 <xsd:complexType name="CT_RevisionQueryTableField">
2099     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
2100     <xsd:attribute name="ref" type="ST Ref" use="required"/>
2101     <xsd:attribute name="fieldId" type="xsd:unsignedInt" use="required"/>
2102 </xsd:complexType>

```

```

2103 <xsd:simpleType name="ST_rwColActionType">
2104   <xsd:restriction base="xsd:string">
2105     <xsd:enumeration value="insertRow"/>
2106     <xsd:enumeration value="deleteRow"/>
2107     <xsd:enumeration value="insertCol"/>
2108     <xsd:enumeration value="deleteCol"/>
2109   </xsd:restriction>
2110 </xsd:simpleType>
2111 <xsd:simpleType name="ST_RevisionAction">
2112   <xsd:restriction base="xsd:string">
2113     <xsd:enumeration value="add"/>
2114     <xsd:enumeration value="delete"/>
2115   </xsd:restriction>
2116 </xsd:simpleType>
2117 <xsd:simpleType name="ST_FormulaExpression">
2118   <xsd:restriction base="xsd:string">
2119     <xsd:enumeration value="ref"/>
2120     <xsd:enumeration value="refError"/>
2121     <xsd:enumeration value="area"/>
2122     <xsd:enumeration value="areaError"/>
2123     <xsd:enumeration value="computedArea"/>
2124   </xsd:restriction>
2125 </xsd:simpleType>
2126 <xsd:element name="users" type="CT_Users"/>
2127 <xsd:complexType name="CT_Users">
2128   <xsd:sequence>
2129     <xsd:element name="userInfo" minOccurs="0" maxOccurs="256" type="CT_SharedUser"/>
2130   </xsd:sequence>
2131   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2132 </xsd:complexType>
2133 <xsd:complexType name="CT_SharedUser">
2134   <xsd:sequence>
2135     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2136   </xsd:sequence>
2137   <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
2138   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2139   <xsd:attribute name="id" type="xsd:int" use="required"/>
2140   <xsd:attribute name="dateTime" type="xsd:dateTime" use="required"/>
2141 </xsd:complexType>
2142 <xsd:element name="worksheet" type="CT_Worksheet"/>
2143 <xsd:element name="chartsheet" type="CT_Chartsheet"/>
2144 <xsd:element name="dialogsheet" type="CT_Dialogsheet"/>
2145 <xsd:complexType name="CT_Macrosheet">
2146   <xsd:sequence>
2147     <xsd:element name="sheetPr" type="CT_SheetPr" minOccurs="0" maxOccurs="1"/>
2148     <xsd:element name="dimension" type="CT_SheetDimension" minOccurs="0" maxOccurs="1"/>
2149     <xsd:element name="sheetViews" type="CT_SheetViews" minOccurs="0" maxOccurs="1"/>
2150     <xsd:element name="sheetFormatPr" type="CT_SheetFormatPr" minOccurs="0" maxOccurs="1"/>
2151     <xsd:element name="cols" type="CT_Cols" minOccurs="0" maxOccurs="unbounded"/>
2152     <xsd:element name="sheetData" type="CT_SheetData" minOccurs="1" maxOccurs="1"/>
2153     <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2154       maxOccurs="1"/>
2155     <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>

```

```

2156     <xsd:element name="sortState" type="CT_SortState" minOccurs="0" maxOccurs="1"/>
2157     <xsd:element name="dataConsolidate" type="CT_DataConsolidate" minOccurs="0"
2158       maxOccurs="1"/>
2159     <xsd:element name="customSheetViews" type="CT_CustomSheetViews" minOccurs="0"
2160       maxOccurs="1"/>
2161     <xsd:element name="phoneticPr" type="CT_PhoneticPr" minOccurs="0" maxOccurs="1"/>
2162     <xsd:element name="conditionalFormatting" type="CT_ConditionalFormatting" minOccurs="0"
2163       maxOccurs="unbounded"/>
2164     <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2165     <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2166     <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2167     <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2168     <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2169     <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2170     <xsd:element name="customProperties" type="CT_CustomProperties" minOccurs="0"
2171       maxOccurs="1"/>
2172     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0" maxOccurs="1"/>
2173     <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2174     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2175     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2176     <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
2177     <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2178     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2179   </xsd:sequence>
2180 </xsd:complexType>
2181 <xsd:complexType name="CT_Dialogsheet">
2182   <xsd:sequence>
2183     <xsd:element name="sheetPr" minOccurs="0" type="CT_SheetPr"/>
2184     <xsd:element name="sheetViews" minOccurs="0" type="CT_SheetViews"/>
2185     <xsd:element name="sheetFormatPr" minOccurs="0" type="CT_SheetFormatPr"/>
2186     <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2187       maxOccurs="1"/>
2188     <xsd:element name="customSheetViews" minOccurs="0" type="CT_CustomSheetViews"/>
2189     <xsd:element name="printOptions" minOccurs="0" type="CT_PrintOptions"/>
2190     <xsd:element name="pageMargins" minOccurs="0" type="CT_PageMargins"/>
2191     <xsd:element name="pageSetup" minOccurs="0" type="CT_PageSetup"/>
2192     <xsd:element name="headerFooter" minOccurs="0" type="CT_HeaderFooter"/>
2193     <xsd:element name="drawing" minOccurs="0" type="CT_Drawing"/>
2194     <xsd:element name="legacyDrawing" minOccurs="0" type="CT_LegacyDrawing"/>
2195     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2196     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2197     <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2198     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2199   </xsd:sequence>
2200 </xsd:complexType>
2201 <xsd:complexType name="CT_Worksheet">
2202   <xsd:sequence>
2203     <xsd:element name="sheetPr" type="CT_SheetPr" minOccurs="0" maxOccurs="1"/>
2204     <xsd:element name="dimension" type="CT_SheetDimension" minOccurs="0" maxOccurs="1"/>
2205     <xsd:element name="sheetViews" type="CT_SheetViews" minOccurs="0" maxOccurs="1"/>
2206     <xsd:element name="sheetFormatPr" type="CT_SheetFormatPr" minOccurs="0" maxOccurs="1"/>
2207     <xsd:element name="cols" type="CT_Cols" minOccurs="0" maxOccurs="unbounded"/>
2208     <xsd:element name="sheetData" type="CT_SheetData" minOccurs="1" maxOccurs="1"/>

```

```

2209     <xsd:element name="sheetCalcPr" type="CT_SheetCalcPr" minOccurs="0" maxOccurs="1"/>
2210     <xsd:element name="sheetProtection" type="CT_SheetProtection" minOccurs="0"
2211         maxOccurs="1"/>
2212     <xsd:element name="protectedRanges" type="CT_ProtectedRanges" minOccurs="0"
2213         maxOccurs="1"/>
2214     <xsd:element name="scenarios" type="CT_Scenarios" minOccurs="0" maxOccurs="1"/>
2215     <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
2216     <xsd:element name="sortState" type="CT_SortState" minOccurs="0" maxOccurs="1"/>
2217     <xsd:element name="dataConsolidate" type="CT_DataConsolidate" minOccurs="0"
2218         maxOccurs="1"/>
2219     <xsd:element name="customSheetViews" type="CT_CustomSheetViews" minOccurs="0"
2220         maxOccurs="1"/>
2221     <xsd:element name="mergeCells" type="CT_MergeCells" minOccurs="0" maxOccurs="1"/>
2222     <xsd:element name="phoneticPr" type="CT_PhoneticPr" minOccurs="0" maxOccurs="1"/>
2223     <xsd:element name="conditionalFormatting" type="CT_ConditionalFormatting" minOccurs="0"
2224         maxOccurs="unbounded"/>
2225     <xsd:element name="dataValidations" type="CT_DataValidations" minOccurs="0"
2226         maxOccurs="1"/>
2227     <xsd:element name="hyperlinks" type="CT_Hyperlinks" minOccurs="0" maxOccurs="1"/>
2228     <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2229     <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2230     <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2231     <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2232     <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2233     <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2234     <xsd:element name="customProperties" type="CT_CustomProperties" minOccurs="0"
2235         maxOccurs="1"/>
2236     <xsd:element name="cellWatches" type="CT_CellWatches" minOccurs="0" maxOccurs="1"/>
2237     <xsd:element name="ignoredErrors" type="CT_IgnoredErrors" minOccurs="0" maxOccurs="1"/>
2238     <xsd:element name="smartTags" type="CT_SmartTags" minOccurs="0" maxOccurs="1"/>
2239     <xsd:element name="drawing" type="CT_Drawing" minOccurs="0" maxOccurs="1"/>
2240     <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2241     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
2242     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
2243     <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
2244     <xsd:element name="oleObjects" type="CT_OleObjects" minOccurs="0" maxOccurs="1"/>
2245     <xsd:element name="controls" type="CT_Controls" minOccurs="0" maxOccurs="1"/>
2246     <xsd:element name="webPublishItems" type="CT_WebPublishItems" minOccurs="0"
2247         maxOccurs="1"/>
2248     <xsd:element name="tableParts" type="CT_TableParts" minOccurs="0" maxOccurs="1"/>
2249     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
2250 </xsd:sequence>
2251 </xsd:complexType>
2252 <xsd:complexType name="CT_SheetData">
2253     <xsd:sequence>
2254         <xsd:element name="row" type="CT_Row" minOccurs="0" maxOccurs="unbounded"/>
2255     </xsd:sequence>
2256 </xsd:complexType>
2257 <xsd:complexType name="CT_SheetCalcPr">
2258     <xsd:attribute name="fullCalcOnLoad" type="xsd:boolean" use="optional" default="false"/>
2259 </xsd:complexType>
2260 <xsd:complexType name="CT_SheetFormatPr">
2261     <xsd:attribute name="baseColWidth" type="xsd:unsignedInt" use="optional" default="8"/>

```

```

2262     <xsd:attribute name="defaultColWidth" type="xsd:double" use="optional"/>
2263     <xsd:attribute name="defaultRowHeight" type="xsd:double" use="required"/>
2264     <xsd:attribute name="customHeight" type="xsd:boolean" use="optional" default="false"/>
2265     <xsd:attribute name="zeroHeight" type="xsd:boolean" use="optional" default="false"/>
2266     <xsd:attribute name="thickTop" type="xsd:boolean" use="optional" default="false"/>
2267     <xsd:attribute name="thickBottom" type="xsd:boolean" use="optional" default="false"/>
2268     <xsd:attribute name="outlineLevelRow" type="xsd:unsignedByte" use="optional" default="0"/>
2269     <xsd:attribute name="outlineLevelCol" type="xsd:unsignedByte" use="optional" default="0"/>
2270 </xsd:complexType>
2271 <xsd:complexType name="CT_Cols">
2272     <xsd:sequence>
2273         <xsd:element name="col" type="CT_Col" minOccurs="1" maxOccurs="unbounded"/>
2274     </xsd:sequence>
2275 </xsd:complexType>
2276 <xsd:complexType name="CT_Col">
2277     <xsd:attribute name="min" type="xsd:unsignedInt" use="required"/>
2278     <xsd:attribute name="max" type="xsd:unsignedInt" use="required"/>
2279     <xsd:attribute name="width" type="xsd:double" use="optional"/>
2280     <xsd:attribute name="style" type="xsd:unsignedInt" use="optional" default="0"/>
2281     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2282     <xsd:attribute name="bestFit" type="xsd:boolean" use="optional" default="false"/>
2283     <xsd:attribute name="customWidth" type="xsd:boolean" use="optional" default="false"/>
2284     <xsd:attribute name="phonetic" type="xsd:boolean" use="optional" default="false"/>
2285     <xsd:attribute name="outlineLevel" type="xsd:unsignedByte" use="optional" default="0"/>
2286     <xsd:attribute name="collapsed" type="xsd:boolean" use="optional" default="false"/>
2287 </xsd:complexType>
2288 <xsd:simpleType name="ST_CellSpan">
2289     <xsd:restriction base="xsd:string"/>
2290 </xsd:simpleType>
2291 <xsd:simpleType name="ST_CellSpans">
2292     <xsd:list itemType="ST_CellSpan"/>
2293 </xsd:simpleType>
2294 <xsd:complexType name="CT_Row">
2295     <xsd:sequence>
2296         <xsd:element name="c" type="CT_Cell" minOccurs="0" maxOccurs="unbounded"/>
2297         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2298     </xsd:sequence>
2299     <xsd:attribute name="r" type="xsd:unsignedInt" use="optional"/>
2300     <xsd:attribute name="spans" type="ST_CellSpans" use="optional"/>
2301     <xsd:attribute name="s" type="xsd:unsignedInt" use="optional" default="0"/>
2302     <xsd:attribute name="customFormat" type="xsd:boolean" use="optional" default="false"/>
2303     <xsd:attribute name="ht" type="xsd:double" use="optional"/>
2304     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2305     <xsd:attribute name="customHeight" type="xsd:boolean" use="optional" default="false"/>
2306     <xsd:attribute name="outlineLevel" type="xsd:unsignedByte" use="optional" default="0"/>
2307     <xsd:attribute name="collapsed" type="xsd:boolean" use="optional" default="false"/>
2308     <xsd:attribute name="thickTop" type="xsd:boolean" use="optional" default="false"/>
2309     <xsd:attribute name="thickBot" type="xsd:boolean" use="optional" default="false"/>
2310     <xsd:attribute name="ph" type="xsd:boolean" use="optional" default="false"/>
2311 </xsd:complexType>
2312 <xsd:complexType name="CT_Cell">
2313     <xsd:sequence>
2314         <xsd:element name="f" type="CT_CellFormula" minOccurs="0" maxOccurs="1"/>

```



```

2315     <xsd:element name="v" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2316     <xsd:element name="is" type="CT_Rst" minOccurs="0" maxOccurs="1"/>
2317     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2318 </xsd:sequence>
2319 <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
2320 <xsd:attribute name="s" type="xsd:unsignedInt" use="optional" default="0"/>
2321 <xsd:attribute name="t" type="ST_CellType" use="optional" default="n"/>
2322 <xsd:attribute name="cm" type="xsd:unsignedInt" use="optional" default="0"/>
2323 <xsd:attribute name="vm" type="xsd:unsignedInt" use="optional" default="0"/>
2324 <xsd:attribute name="ph" type="xsd:boolean" use="optional" default="false"/>
2325 </xsd:complexType>
2326 <xsd:simpleType name="ST_CellType">
2327     <xsd:restriction base="xsd:string">
2328         <xsd:enumeration value="b"/>
2329         <xsd:enumeration value="d"/>
2330         <xsd:enumeration value="n"/>
2331         <xsd:enumeration value="e"/>
2332         <xsd:enumeration value="s"/>
2333         <xsd:enumeration value="str"/>
2334         <xsd:enumeration value="inlineStr"/>
2335     </xsd:restriction>
2336 </xsd:simpleType>
2337 <xsd:simpleType name="ST_CellFormulaType">
2338     <xsd:restriction base="xsd:string">
2339         <xsd:enumeration value="normal"/>
2340         <xsd:enumeration value="array"/>
2341         <xsd:enumeration value="dataTable"/>
2342         <xsd:enumeration value="shared"/>
2343     </xsd:restriction>
2344 </xsd:simpleType>
2345 <xsd:complexType name="CT_SheetPr">
2346     <xsd:sequence>
2347         <xsd:element name="tabColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
2348         <xsd:element name="outlinePr" type="CT_OutlinePr" minOccurs="0" maxOccurs="1"/>
2349         <xsd:element name="pageSetUpPr" type="CT_PageSetUpPr" minOccurs="0" maxOccurs="1"/>
2350     </xsd:sequence>
2351     <xsd:attribute name="syncHorizontal" type="xsd:boolean" use="optional" default="false"/>
2352     <xsd:attribute name="syncVertical" type="xsd:boolean" use="optional" default="false"/>
2353     <xsd:attribute name="syncRef" type="ST_Ref" use="optional"/>
2354     <xsd:attribute name="transitionEvaluation" type="xsd:boolean" use="optional" default="false"/>
2355     <xsd:attribute name="transitionEntry" type="xsd:boolean" use="optional" default="false"/>
2356     <xsd:attribute name="published" type="xsd:boolean" use="optional" default="true"/>
2357     <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
2358     <xsd:attribute name="filterMode" type="xsd:boolean" use="optional" default="false"/>
2359     <xsd:attribute name="enableFormatConditionsCalculation" type="xsd:boolean" use="optional"
2360         default="true"/>
2361 </xsd:complexType>
2362 <xsd:complexType name="CT_SheetDimension">
2363     <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2364 </xsd:complexType>
2365 <xsd:complexType name="CT_SheetViews">
2366     <xsd:sequence>
2367         <xsd:element name="sheetView" type="CT_SheetView" minOccurs="1" maxOccurs="unbounded"/>

```

```

2368     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
2369   </xsd:sequence>
2370 </xsd:complexType>
2371 <xsd:complexType name="CT_SheetView">
2372   <xsd:sequence>
2373     <xsd:element name="pane" type="CT_Pane" minOccurs="0" maxOccurs="1"/>
2374     <xsd:element name="selection" type="CT_Selection" minOccurs="0" maxOccurs="4"/>
2375     <xsd:element name="pivotSelection" type="CT_PivotSelection" minOccurs="0" maxOccurs="4"/>
2376     <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
2377   </xsd:sequence>
2378   <xsd:attribute name="windowProtection" type="xsd:boolean" use="optional" default="false"/>
2379   <xsd:attribute name="showFormulas" type="xsd:boolean" use="optional" default="false"/>
2380   <xsd:attribute name="showGridLines" type="xsd:boolean" use="optional" default="true"/>
2381   <xsd:attribute name="showRowColHeaders" type="xsd:boolean" use="optional" default="true"/>
2382   <xsd:attribute name="showZeros" type="xsd:boolean" use="optional" default="true"/>
2383   <xsd:attribute name="rightToLeft" type="xsd:boolean" use="optional" default="false"/>
2384   <xsd:attribute name="tabSelected" type="xsd:boolean" use="optional" default="false"/>
2385   <xsd:attribute name="showRuler" type="xsd:boolean" use="optional" default="true"/>
2386   <xsd:attribute name="showOutlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2387   <xsd:attribute name="defaultGridColor" type="xsd:boolean" use="optional" default="true"/>
2388   <xsd:attribute name="showWhiteSpace" type="xsd:boolean" use="optional" default="true"/>
2389   <xsd:attribute name="view" type="ST_SheetViewType" use="optional" default="normal"/>
2390   <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2391   <xsd:attribute name="colorId" type="xsd:unsignedInt" use="optional" default="64"/>
2392   <xsd:attribute name="zoomScale" type="xsd:unsignedInt" use="optional" default="100"/>
2393   <xsd:attribute name="zoomScaleNormal" type="xsd:unsignedInt" use="optional" default="0"/>
2394   <xsd:attribute name="zoomScaleSheetLayoutView" type="xsd:unsignedInt" use="optional"
2395     default="0"/>
2396   <xsd:attribute name="zoomScalePageLayoutView" type="xsd:unsignedInt" use="optional"
2397     default="0"/>
2398   <xsd:attribute name="workbookViewId" type="xsd:unsignedInt" use="required"/>
2399 </xsd:complexType>
2400 <xsd:complexType name="CT_Pane">
2401   <xsd:attribute name="xSplit" type="xsd:double" use="optional" default="0"/>
2402   <xsd:attribute name="ySplit" type="xsd:double" use="optional" default="0"/>
2403   <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2404   <xsd:attribute name="activePane" type="ST_Pane" use="optional" default="topLeft"/>
2405   <xsd:attribute name="state" type="ST_PaneState" use="optional" default="split"/>
2406 </xsd:complexType>
2407 <xsd:complexType name="CT_PivotSelection">
2408   <xsd:sequence>
2409     <xsd:element name="pivotArea" type="CT_PivotArea"/>
2410   </xsd:sequence>
2411   <xsd:attribute name="pane" type="ST_Pane" use="optional" default="topLeft"/>
2412   <xsd:attribute name="showHeader" type="xsd:boolean" default="false"/>
2413   <xsd:attribute name="label" type="xsd:boolean" default="false"/>
2414   <xsd:attribute name="data" type="xsd:boolean" default="false"/>
2415   <xsd:attribute name="extendable" type="xsd:boolean" default="false"/>
2416   <xsd:attribute name="count" type="xsd:unsignedInt" default="0"/>
2417   <xsd:attribute name="axis" type="ST_Axis" use="optional"/>
2418   <xsd:attribute name="dimension" type="xsd:unsignedInt" default="0"/>
2419   <xsd:attribute name="start" type="xsd:unsignedInt" default="0"/>
2420   <xsd:attribute name="min" type="xsd:unsignedInt" default="0"/>

```

```

2421     <xsd:attribute name="max" type="xsd:unsignedInt" default="0"/>
2422     <xsd:attribute name="activeRow" type="xsd:unsignedInt" default="0"/>
2423     <xsd:attribute name="activeCol" type="xsd:unsignedInt" default="0"/>
2424     <xsd:attribute name="previousRow" type="xsd:unsignedInt" default="0"/>
2425     <xsd:attribute name="previousCol" type="xsd:unsignedInt" default="0"/>
2426     <xsd:attribute name="click" type="xsd:unsignedInt" default="0"/>
2427     <xsd:attribute ref="r:id" use="optional"/>
2428 </xsd:complexType>
2429 <xsd:complexType name="CT_Selection">
2430     <xsd:attribute name="pane" type="ST_Pane" use="optional" default="topLeft"/>
2431     <xsd:attribute name="activeCell" type="ST_CellRef" use="optional"/>
2432     <xsd:attribute name="activeCellId" type="xsd:unsignedInt" use="optional" default="0"/>
2433     <xsd:attribute name="sqref" type="ST_Sqref" use="optional" default="A1"/>
2434 </xsd:complexType>
2435 <xsd:simpleType name="ST_Pane">
2436     <xsd:restriction base="xsd:string">
2437         <xsd:enumeration value="bottomRight"/>
2438         <xsd:enumeration value="topRight"/>
2439         <xsd:enumeration value="bottomLeft"/>
2440         <xsd:enumeration value="topLeft"/>
2441     </xsd:restriction>
2442 </xsd:simpleType>
2443 <xsd:complexType name="CT_PageBreak">
2444     <xsd:sequence>
2445         <xsd:element name="brk" type="CT_Break" minOccurs="0" maxOccurs="unbounded"/>
2446     </xsd:sequence>
2447     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
2448     <xsd:attribute name="manualBreakCount" type="xsd:unsignedInt" use="optional" default="0"/>
2449 </xsd:complexType>
2450 <xsd:complexType name="CT_Break">
2451     <xsd:attribute name="id" type="xsd:unsignedInt" use="optional" default="0"/>
2452     <xsd:attribute name="min" type="xsd:unsignedInt" use="optional" default="0"/>
2453     <xsd:attribute name="max" type="xsd:unsignedInt" use="optional" default="0"/>
2454     <xsd:attribute name="man" type="xsd:boolean" use="optional" default="false"/>
2455     <xsd:attribute name="pt" type="xsd:boolean" use="optional" default="false"/>
2456 </xsd:complexType>
2457 <xsd:simpleType name="ST_SheetViewType">
2458     <xsd:restriction base="xsd:string">
2459         <xsd:enumeration value="normal"/>
2460         <xsd:enumeration value="pageBreakPreview"/>
2461         <xsd:enumeration value="pageLayout"/>
2462     </xsd:restriction>
2463 </xsd:simpleType>
2464 <xsd:complexType name="CT_OutlinePr">
2465     <xsd:attribute name="applyStyles" type="xsd:boolean" use="optional" default="false"/>
2466     <xsd:attribute name="summaryBelow" type="xsd:boolean" use="optional" default="true"/>
2467     <xsd:attribute name="summaryRight" type="xsd:boolean" use="optional" default="true"/>
2468     <xsd:attribute name="showOutlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2469 </xsd:complexType>
2470 <xsd:complexType name="CT_PageSetUpPr">
2471     <xsd:attribute name="autoPageBreaks" type="xsd:boolean" use="optional" default="true"/>
2472     <xsd:attribute name="fitToPage" type="xsd:boolean" use="optional" default="false"/>
2473 </xsd:complexType>

```

```

2474 <xsd:complexType name="CT_DataConsolidate">
2475   <xsd:sequence>
2476     <xsd:element name="dataRefs" type="CT_DataRefs" minOccurs="0" maxOccurs="1"/>
2477   </xsd:sequence>
2478   <xsd:attribute name="function" type="ST_DataConsolidateFunction" use="optional"
2479     default="sum"/>
2480   <xsd:attribute name="startLabels" type="xsd:boolean" use="optional" default="false"/>
2481   <xsd:attribute name="leftLabels" type="xsd:boolean" use="optional" default="false"/>
2482   <xsd:attribute name="topLabels" type="xsd:boolean" use="optional" default="false"/>
2483   <xsd:attribute name="link" type="xsd:boolean" use="optional" default="false"/>
2484 </xsd:complexType>
2485 <xsd:simpleType name="ST_DataConsolidateFunction">
2486   <xsd:restriction base="xsd:string">
2487     <xsd:enumeration value="average"/>
2488     <xsd:enumeration value="count"/>
2489     <xsd:enumeration value="countNums"/>
2490     <xsd:enumeration value="max"/>
2491     <xsd:enumeration value="min"/>
2492     <xsd:enumeration value="product"/>
2493     <xsd:enumeration value="stdDev"/>
2494     <xsd:enumeration value="stdDevp"/>
2495     <xsd:enumeration value="sum"/>
2496     <xsd:enumeration value="var"/>
2497     <xsd:enumeration value="varp"/>
2498   </xsd:restriction>
2499 </xsd:simpleType>
2500 <xsd:complexType name="CT_DataRefs">
2501   <xsd:sequence>
2502     <xsd:element name="dataRef" type="CT_DataRef" minOccurs="0" maxOccurs="unbounded"/>
2503   </xsd:sequence>
2504   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2505 </xsd:complexType>
2506 <xsd:complexType name="CT_DataRef">
2507   <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
2508   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
2509   <xsd:attribute name="sheet" type="s:ST_Xstring" use="optional"/>
2510   <xsd:attribute ref="r:id" use="optional"/>
2511 </xsd:complexType>
2512 <xsd:complexType name="CT_MergeCells">
2513   <xsd:sequence>
2514     <xsd:element name="mergeCell" type="CT_MergeCell" minOccurs="1" maxOccurs="unbounded"/>
2515   </xsd:sequence>
2516   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2517 </xsd:complexType>
2518 <xsd:complexType name="CT_MergeCell">
2519   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
2520 </xsd:complexType>
2521 <xsd:complexType name="CT_SmartTags">
2522   <xsd:sequence>
2523     <xsd:element name="cellSmartTags" type="CT_CellSmartTags" minOccurs="1"
2524       maxOccurs="unbounded"/>
2525   </xsd:sequence>
2526 </xsd:complexType>

```

```

2527 <xsd:complexType name="CT_CellSmartTags">
2528     <xsd:sequence>
2529         <xsd:element name="cellSmartTag" type="CT_CellSmartTag" minOccurs="1"
2530             maxOccurs="unbounded"/>
2531     </xsd:sequence>
2532     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
2533 </xsd:complexType>
2534 <xsd:complexType name="CT_CellSmartTag">
2535     <xsd:sequence>
2536         <xsd:element name="cellSmartTagPr" minOccurs="0" maxOccurs="unbounded"
2537             type="CT_CellSmartTagPr"/>
2538     </xsd:sequence>
2539     <xsd:attribute name="type" type="xsd:unsignedInt" use="required"/>
2540     <xsd:attribute name="deleted" type="xsd:boolean" use="optional" default="false"/>
2541     <xsd:attribute name="xmlBased" type="xsd:boolean" use="optional" default="false"/>
2542 </xsd:complexType>
2543 <xsd:complexType name="CT_CellSmartTagPr">
2544     <xsd:attribute name="key" type="s:ST_Xstring" use="required"/>
2545     <xsd:attribute name="val" type="s:ST_Xstring" use="required"/>
2546 </xsd:complexType>
2547 <xsd:complexType name="CT_Drawing">
2548     <xsd:attribute ref="r:id" use="required"/>
2549 </xsd:complexType>
2550 <xsd:complexType name="CT_LegacyDrawing">
2551     <xsd:attribute ref="r:id" use="required"/>
2552 </xsd:complexType>
2553 <xsd:complexType name="CT_DrawingHF">
2554     <xsd:attribute ref="r:id" use="required"/>
2555     <xsd:attribute name="lho" type="xsd:unsignedInt" use="optional"/>
2556     <xsd:attribute name="lhe" type="xsd:unsignedInt" use="optional"/>
2557     <xsd:attribute name="lhf" type="xsd:unsignedInt" use="optional"/>
2558     <xsd:attribute name="cho" type="xsd:unsignedInt" use="optional"/>
2559     <xsd:attribute name="che" type="xsd:unsignedInt" use="optional"/>
2560     <xsd:attribute name="chf" type="xsd:unsignedInt" use="optional"/>
2561     <xsd:attribute name="rho" type="xsd:unsignedInt" use="optional"/>
2562     <xsd:attribute name="rhe" type="xsd:unsignedInt" use="optional"/>
2563     <xsd:attribute name="rhf" type="xsd:unsignedInt" use="optional"/>
2564     <xsd:attribute name="lfo" type="xsd:unsignedInt" use="optional"/>
2565     <xsd:attribute name="lfe" type="xsd:unsignedInt" use="optional"/>
2566     <xsd:attribute name="lff" type="xsd:unsignedInt" use="optional"/>
2567     <xsd:attribute name="cfo" type="xsd:unsignedInt" use="optional"/>
2568     <xsd:attribute name="cfe" type="xsd:unsignedInt" use="optional"/>
2569     <xsd:attribute name="cff" type="xsd:unsignedInt" use="optional"/>
2570     <xsd:attribute name="rfo" type="xsd:unsignedInt" use="optional"/>
2571     <xsd:attribute name="rfe" type="xsd:unsignedInt" use="optional"/>
2572     <xsd:attribute name="rff" type="xsd:unsignedInt" use="optional"/>
2573 </xsd:complexType>
2574 <xsd:complexType name="CT_CustomSheetViews">
2575     <xsd:sequence>
2576         <xsd:element name="customSheetView" minOccurs="1" maxOccurs="unbounded"
2577             type="CT_CustomSheetView"/>
2578     </xsd:sequence>
2579 </xsd:complexType>

```

```

2580 <xsd:complexType name="CT_CustomSheetView">
2581   <xsd:sequence>
2582     <xsd:element name="pane" type="CT_Pane" minOccurs="0" maxOccurs="1"/>
2583     <xsd:element name="selection" type="CT_Selection" minOccurs="0" maxOccurs="1"/>
2584     <xsd:element name="rowBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2585     <xsd:element name="colBreaks" type="CT_PageBreak" minOccurs="0" maxOccurs="1"/>
2586     <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
2587     <xsd:element name="printOptions" type="CT_PrintOptions" minOccurs="0" maxOccurs="1"/>
2588     <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
2589     <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
2590     <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
2591     <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2592   </xsd:sequence>
2593   <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
2594   <xsd:attribute name="scale" type="xsd:unsignedInt" default="100"/>
2595   <xsd:attribute name="colorId" type="xsd:unsignedInt" default="64"/>
2596   <xsd:attribute name="showPageBreaks" type="xsd:boolean" use="optional" default="false"/>
2597   <xsd:attribute name="showFormulas" type="xsd:boolean" use="optional" default="false"/>
2598   <xsd:attribute name="showGridLines" type="xsd:boolean" use="optional" default="true"/>
2599   <xsd:attribute name="showRowCol" type="xsd:boolean" use="optional" default="true"/>
2600   <xsd:attribute name="outlineSymbols" type="xsd:boolean" use="optional" default="true"/>
2601   <xsd:attribute name="zeroValues" type="xsd:boolean" use="optional" default="true"/>
2602   <xsd:attribute name="fitToPage" type="xsd:boolean" use="optional" default="false"/>
2603   <xsd:attribute name="printArea" type="xsd:boolean" use="optional" default="false"/>
2604   <xsd:attribute name="filter" type="xsd:boolean" use="optional" default="false"/>
2605   <xsd:attribute name="showAutoFilter" type="xsd:boolean" use="optional" default="false"/>
2606   <xsd:attribute name="hiddenRows" type="xsd:boolean" use="optional" default="false"/>
2607   <xsd:attribute name="hiddenColumns" type="xsd:boolean" use="optional" default="false"/>
2608   <xsd:attribute name="state" type="ST_SheetState" default="visible"/>
2609   <xsd:attribute name="filterUnique" type="xsd:boolean" use="optional" default="false"/>
2610   <xsd:attribute name="view" type="ST_SheetViewType" default="normal"/>
2611   <xsd:attribute name="showRuler" type="xsd:boolean" use="optional" default="true"/>
2612   <xsd:attribute name="topLeftCell" type="ST_CellRef" use="optional"/>
2613 </xsd:complexType>
2614 <xsd:complexType name="CT_DataValidations">
2615   <xsd:sequence>
2616     <xsd:element name="dataValidation" type="CT_DataValidation" minOccurs="1"
2617       maxOccurs="unbounded"/>
2618   </xsd:sequence>
2619   <xsd:attribute name="disablePrompts" type="xsd:boolean" use="optional" default="false"/>
2620   <xsd:attribute name="xWindow" type="xsd:unsignedInt" use="optional"/>
2621   <xsd:attribute name="yWindow" type="xsd:unsignedInt" use="optional"/>
2622   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2623 </xsd:complexType>
2624 <xsd:complexType name="CT_DataValidation">
2625   <xsd:sequence>
2626     <xsd:element name="formula1" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2627     <xsd:element name="formula2" type="ST_Formula" minOccurs="0" maxOccurs="1"/>
2628   </xsd:sequence>
2629   <xsd:attribute name="type" type="ST_DataValidationType" use="optional" default="none"/>
2630   <xsd:attribute name="errorStyle" type="ST_DataValidationErrorStyle" use="optional"
2631     default="stop"/>

```

```

2632 <xsd:attribute name="imeMode" type="ST_DataValidationImeMode" use="optional"
2633     default="noControl"/>
2634 <xsd:attribute name="operator" type="ST_DataValidationOperator" use="optional"
2635     default="between"/>
2636 <xsd:attribute name="allowBlank" type="xsd:boolean" use="optional" default="false"/>
2637 <xsd:attribute name="showDropDown" type="xsd:boolean" use="optional" default="false"/>
2638 <xsd:attribute name="showInputMessage" type="xsd:boolean" use="optional" default="false"/>
2639 <xsd:attribute name="showErrorMessage" type="xsd:boolean" use="optional" default="false"/>
2640 <xsd:attribute name="errorTitle" type="s:ST_Xstring" use="optional"/>
2641 <xsd:attribute name="error" type="s:ST_Xstring" use="optional"/>
2642 <xsd:attribute name="promptTitle" type="s:ST_Xstring" use="optional"/>
2643 <xsd:attribute name="prompt" type="s:ST_Xstring" use="optional"/>
2644 <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
2645 </xsd:complexType>
2646 <xsd:simpleType name="ST_DataValidationType">
2647     <xsd:restriction base="xsd:string">
2648         <xsd:enumeration value="none"/>
2649         <xsd:enumeration value="whole"/>
2650         <xsd:enumeration value="decimal"/>
2651         <xsd:enumeration value="list"/>
2652         <xsd:enumeration value="date"/>
2653         <xsd:enumeration value="time"/>
2654         <xsd:enumeration value="textLength"/>
2655         <xsd:enumeration value="custom"/>
2656     </xsd:restriction>
2657 </xsd:simpleType>
2658 <xsd:simpleType name="ST_DataValidationOperator">
2659     <xsd:restriction base="xsd:string">
2660         <xsd:enumeration value="between"/>
2661         <xsd:enumeration value="notBetween"/>
2662         <xsd:enumeration value="equal"/>
2663         <xsd:enumeration value="notEqual"/>
2664         <xsd:enumeration value="lessThan"/>
2665         <xsd:enumeration value="lessThanOrEqual"/>
2666         <xsd:enumeration value="greaterThan"/>
2667         <xsd:enumeration value="greaterThanOrEqual"/>
2668     </xsd:restriction>
2669 </xsd:simpleType>
2670 <xsd:simpleType name="ST_DataValidationErrorStyle">
2671     <xsd:restriction base="xsd:string">
2672         <xsd:enumeration value="stop"/>
2673         <xsd:enumeration value="warning"/>
2674         <xsd:enumeration value="information"/>
2675     </xsd:restriction>
2676 </xsd:simpleType>
2677 <xsd:simpleType name="ST_DataValidationImeMode">
2678     <xsd:restriction base="xsd:string">
2679         <xsd:enumeration value="noControl"/>
2680         <xsd:enumeration value="off"/>
2681         <xsd:enumeration value="on"/>
2682         <xsd:enumeration value="disabled"/>
2683         <xsd:enumeration value="hiragana"/>
2684         <xsd:enumeration value="fullKatakana"/>

```

```

2685         <xsd:enumeration value="halfKatakana"/>
2686         <xsd:enumeration value="fullAlpha"/>
2687         <xsd:enumeration value="halfAlpha"/>
2688         <xsd:enumeration value="fullHangul"/>
2689         <xsd:enumeration value="halfHangul"/>
2690     </xsd:restriction>
2691 </xsd:simpleType>
2692 <xsd:simpleType name="ST_CfType">
2693     <xsd:restriction base="xsd:string">
2694         <xsd:enumeration value="expression"/>
2695         <xsd:enumeration value="cellIs"/>
2696         <xsd:enumeration value="colorScale"/>
2697         <xsd:enumeration value="dataBar"/>
2698         <xsd:enumeration value="iconSet"/>
2699         <xsd:enumeration value="top10"/>
2700         <xsd:enumeration value="uniqueValues"/>
2701         <xsd:enumeration value="duplicateValues"/>
2702         <xsd:enumeration value="containsText"/>
2703         <xsd:enumeration value="notContainsText"/>
2704         <xsd:enumeration value="beginsWith"/>
2705         <xsd:enumeration value="endsWith"/>
2706         <xsd:enumeration value="containsBlanks"/>
2707         <xsd:enumeration value="notContainsBlanks"/>
2708         <xsd:enumeration value="containsErrors"/>
2709         <xsd:enumeration value="notContainsErrors"/>
2710         <xsd:enumeration value="timePeriod"/>
2711         <xsd:enumeration value="aboveAverage"/>
2712     </xsd:restriction>
2713 </xsd:simpleType>
2714 <xsd:simpleType name="ST_TimePeriod">
2715     <xsd:restriction base="xsd:string">
2716         <xsd:enumeration value="today"/>
2717         <xsd:enumeration value="yesterday"/>
2718         <xsd:enumeration value="tomorrow"/>
2719         <xsd:enumeration value="last7Days"/>
2720         <xsd:enumeration value="thisMonth"/>
2721         <xsd:enumeration value="lastMonth"/>
2722         <xsd:enumeration value="nextMonth"/>
2723         <xsd:enumeration value="thisWeek"/>
2724         <xsd:enumeration value="lastWeek"/>
2725         <xsd:enumeration value="nextWeek"/>
2726     </xsd:restriction>
2727 </xsd:simpleType>
2728 <xsd:simpleType name="ST_ConditionalFormattingOperator">
2729     <xsd:restriction base="xsd:string">
2730         <xsd:enumeration value="lessThan"/>
2731         <xsd:enumeration value="lessThanOrEqual"/>
2732         <xsd:enumeration value="equal"/>
2733         <xsd:enumeration value="notEqual"/>
2734         <xsd:enumeration value="greaterThanOrEqual"/>
2735         <xsd:enumeration value="greaterThan"/>
2736         <xsd:enumeration value="between"/>
2737         <xsd:enumeration value="notBetween"/>

```



```

2738         <xsd:enumeration value="containsText"/>
2739         <xsd:enumeration value="notContains"/>
2740         <xsd:enumeration value="beginsWith"/>
2741         <xsd:enumeration value="endsWith"/>
2742     </xsd:restriction>
2743 </xsd:simpleType>
2744 <xsd:simpleType name="ST_CfvoType">
2745     <xsd:restriction base="xsd:string">
2746         <xsd:enumeration value="num"/>
2747         <xsd:enumeration value="percent"/>
2748         <xsd:enumeration value="max"/>
2749         <xsd:enumeration value="min"/>
2750         <xsd:enumeration value="formula"/>
2751         <xsd:enumeration value="percentile"/>
2752     </xsd:restriction>
2753 </xsd:simpleType>
2754 <xsd:complexType name="CT_ConditionalFormatting">
2755     <xsd:sequence>
2756         <xsd:element name="cfRule" type="CT_CfRule" minOccurs="1" maxOccurs="unbounded"/>
2757         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2758     </xsd:sequence>
2759     <xsd:attribute name="pivot" type="xsd:boolean" default="false"/>
2760     <xsd:attribute name="sqref" type="ST_Sqref"/>
2761 </xsd:complexType>
2762 <xsd:complexType name="CT_CfRule">
2763     <xsd:sequence>
2764         <xsd:element name="formula" type="ST_Formula" minOccurs="0" maxOccurs="3"/>
2765         <xsd:element name="colorScale" type="CT_ColorScale" minOccurs="0" maxOccurs="1"/>
2766         <xsd:element name="dataBar" type="CT_DataBar" minOccurs="0" maxOccurs="1"/>
2767         <xsd:element name="iconSet" type="CT_IconSet" minOccurs="0" maxOccurs="1"/>
2768         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
2769     </xsd:sequence>
2770     <xsd:attribute name="type" type="ST_CfType"/>
2771     <xsd:attribute name="dxfId" type="ST_DxfId" use="optional"/>
2772     <xsd:attribute name="priority" type="xsd:int" use="required"/>
2773     <xsd:attribute name="stopIfTrue" type="xsd:boolean" use="optional" default="false"/>
2774     <xsd:attribute name="aboveAverage" type="xsd:boolean" use="optional" default="true"/>
2775     <xsd:attribute name="percent" type="xsd:boolean" use="optional" default="false"/>
2776     <xsd:attribute name="bottom" type="xsd:boolean" use="optional" default="false"/>
2777     <xsd:attribute name="operator" type="ST_ConditionalFormattingOperator" use="optional"/>
2778     <xsd:attribute name="text" type="xsd:string" use="optional"/>
2779     <xsd:attribute name="timePeriod" type="ST_TimePeriod" use="optional"/>
2780     <xsd:attribute name="rank" type="xsd:unsignedInt" use="optional"/>
2781     <xsd:attribute name="stdDev" type="xsd:int" use="optional"/>
2782     <xsd:attribute name="equalAverage" type="xsd:boolean" use="optional" default="false"/>
2783 </xsd:complexType>
2784 <xsd:complexType name="CT_Hyperlinks">
2785     <xsd:sequence>
2786         <xsd:element name="hyperlink" type="CT_Hyperlink" minOccurs="1" maxOccurs="unbounded"/>
2787     </xsd:sequence>
2788 </xsd:complexType>
2789 <xsd:complexType name="CT_Hyperlink">
2790     <xsd:attribute name="ref" type="ST_Ref" use="required"/>

```

```

2791     <xsd:attribute ref="r:id" use="optional"/>
2792     <xsd:attribute name="location" type="s:ST_Xstring" use="optional"/>
2793     <xsd:attribute name="tooltip" type="s:ST_Xstring" use="optional"/>
2794     <xsd:attribute name="display" type="s:ST_Xstring" use="optional"/>
2795 </xsd:complexType>
2796 <xsd:complexType name="CT_CellFormula">
2797     <xsd:simpleContent>
2798         <xsd:extension base="ST_Formula">
2799             <xsd:attribute name="t" type="ST_CellFormulaType" use="optional" default="normal"/>
2800             <xsd:attribute name="aca" type="xsd:boolean" use="optional" default="false"/>
2801             <xsd:attribute name="ref" type="ST_Ref" use="optional"/>
2802             <xsd:attribute name="dt2D" type="xsd:boolean" use="optional" default="false"/>
2803             <xsd:attribute name="dtr" type="xsd:boolean" use="optional" default="false"/>
2804             <xsd:attribute name="del1" type="xsd:boolean" use="optional" default="false"/>
2805             <xsd:attribute name="del2" type="xsd:boolean" use="optional" default="false"/>
2806             <xsd:attribute name="r1" type="ST_CellRef" use="optional"/>
2807             <xsd:attribute name="r2" type="ST_CellRef" use="optional"/>
2808             <xsd:attribute name="ca" type="xsd:boolean" use="optional" default="false"/>
2809             <xsd:attribute name="si" type="xsd:unsignedInt" use="optional"/>
2810             <xsd:attribute name="bx" type="xsd:boolean" use="optional" default="false"/>
2811         </xsd:extension>
2812     </xsd:simpleContent>
2813 </xsd:complexType>
2814 <xsd:complexType name="CT_ColorScale">
2815     <xsd:sequence>
2816         <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="unbounded"/>
2817         <xsd:element name="color" type="CT_Color" minOccurs="2" maxOccurs="unbounded"/>
2818     </xsd:sequence>
2819 </xsd:complexType>
2820 <xsd:complexType name="CT_DataBar">
2821     <xsd:sequence>
2822         <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="2"/>
2823         <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="1"/>
2824     </xsd:sequence>
2825     <xsd:attribute name="minLength" type="xsd:unsignedInt" use="optional" default="10"/>
2826     <xsd:attribute name="maxLength" type="xsd:unsignedInt" use="optional" default="90"/>
2827     <xsd:attribute name="showValue" type="xsd:boolean" use="optional" default="true"/>
2828 </xsd:complexType>
2829 <xsd:complexType name="CT_IconSet">
2830     <xsd:sequence>
2831         <xsd:element name="cfvo" type="CT_Cfvo" minOccurs="2" maxOccurs="unbounded"/>
2832     </xsd:sequence>
2833     <xsd:attribute name="iconSet" type="ST_IconSetType" use="optional" default="3TrafficLights1"/>
2834     <xsd:attribute name="showValue" type="xsd:boolean" use="optional" default="true"/>
2835     <xsd:attribute name="percent" type="xsd:boolean" default="true"/>
2836     <xsd:attribute name="reverse" type="xsd:boolean" use="optional" default="false"/>
2837 </xsd:complexType>
2838 <xsd:complexType name="CT_Cfvo">
2839     <xsd:sequence>
2840         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
2841     </xsd:sequence>
2842     <xsd:attribute name="type" type="ST_CfvoType" use="required"/>
2843     <xsd:attribute name="val" type="s:ST_Xstring" use="optional"/>

```

```

2844     <xsd:attribute name="gte" type="xsd:boolean" use="optional" default="true"/>
2845 </xsd:complexType>
2846 <xsd:complexType name="CT_PageMargins">
2847     <xsd:attribute name="left" type="xsd:double" use="required"/>
2848     <xsd:attribute name="right" type="xsd:double" use="required"/>
2849     <xsd:attribute name="top" type="xsd:double" use="required"/>
2850     <xsd:attribute name="bottom" type="xsd:double" use="required"/>
2851     <xsd:attribute name="header" type="xsd:double" use="required"/>
2852     <xsd:attribute name="footer" type="xsd:double" use="required"/>
2853 </xsd:complexType>
2854 <xsd:complexType name="CT_PrintOptions">
2855     <xsd:attribute name="horizontalCentered" type="xsd:boolean" use="optional" default="false"/>
2856     <xsd:attribute name="verticalCentered" type="xsd:boolean" use="optional" default="false"/>
2857     <xsd:attribute name="headings" type="xsd:boolean" use="optional" default="false"/>
2858     <xsd:attribute name="gridLines" type="xsd:boolean" use="optional" default="false"/>
2859     <xsd:attribute name="gridLinesSet" type="xsd:boolean" use="optional" default="true"/>
2860 </xsd:complexType>
2861 <xsd:complexType name="CT_PageSetup">
2862     <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>
2863     <xsd:attribute name="paperHeight" type="s:ST_PositiveUniversalMeasure" use="optional"/>
2864     <xsd:attribute name="paperWidth" type="s:ST_PositiveUniversalMeasure" use="optional"/>
2865     <xsd:attribute name="scale" type="xsd:unsignedInt" use="optional" default="100"/>
2866     <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
2867     <xsd:attribute name="fitToWidth" type="xsd:unsignedInt" use="optional" default="1"/>
2868     <xsd:attribute name="fitToHeight" type="xsd:unsignedInt" use="optional" default="1"/>
2869     <xsd:attribute name="pageOrder" type="ST_PageOrder" use="optional" default="downThenOver"/>
2870     <xsd:attribute name="orientation" type="ST_Orientation" use="optional" default="default"/>
2871     <xsd:attribute name="usePrinterDefaults" type="xsd:boolean" use="optional" default="true"/>
2872     <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
2873     <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
2874     <xsd:attribute name="cellComments" type="ST_CellComments" use="optional" default="none"/>
2875     <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
2876     <xsd:attribute name="errors" type="ST_PrintError" use="optional" default="displayed"/>
2877     <xsd:attribute name="horizontalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
2878     <xsd:attribute name="verticalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
2879     <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
2880     <xsd:attribute ref="r:id" use="optional"/>
2881 </xsd:complexType>
2882 <xsd:simpleType name="ST_PageOrder">
2883     <xsd:restriction base="xsd:string">
2884         <xsd:enumeration value="downThenOver"/>
2885         <xsd:enumeration value="overThenDown"/>
2886     </xsd:restriction>
2887 </xsd:simpleType>
2888 <xsd:simpleType name="ST_Orientation">
2889     <xsd:restriction base="xsd:string">
2890         <xsd:enumeration value="default"/>
2891         <xsd:enumeration value="portrait"/>
2892         <xsd:enumeration value="landscape"/>
2893     </xsd:restriction>
2894 </xsd:simpleType>
2895 <xsd:simpleType name="ST_CellComments">
2896     <xsd:restriction base="xsd:string">

```

```

2897     <xsd:enumeration value="none"/>
2898     <xsd:enumeration value="asDisplayed"/>
2899     <xsd:enumeration value="atEnd"/>
2900   </xsd:restriction>
2901 </xsd:simpleType>
2902 <xsd:complexType name="CT_HeaderFooter">
2903   <xsd:sequence>
2904     <xsd:element name="oddHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2905     <xsd:element name="oddFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2906     <xsd:element name="evenHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2907     <xsd:element name="evenFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2908     <xsd:element name="firstHeader" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2909     <xsd:element name="firstFooter" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
2910   </xsd:sequence>
2911   <xsd:attribute name="differentOddEven" type="xsd:boolean" default="false"/>
2912   <xsd:attribute name="differentFirst" type="xsd:boolean" default="false"/>
2913   <xsd:attribute name="scaleWithDoc" type="xsd:boolean" default="true"/>
2914   <xsd:attribute name="alignWithMargins" type="xsd:boolean" default="true"/>
2915 </xsd:complexType>
2916 <xsd:simpleType name="ST_PrintError">
2917   <xsd:restriction base="xsd:string">
2918     <xsd:enumeration value="displayed"/>
2919     <xsd:enumeration value="blank"/>
2920     <xsd:enumeration value="dash"/>
2921     <xsd:enumeration value="NA"/>
2922   </xsd:restriction>
2923 </xsd:simpleType>
2924 <xsd:complexType name="CT_Scenarios">
2925   <xsd:sequence>
2926     <xsd:element name="scenario" type="CT_Scenario" minOccurs="1" maxOccurs="unbounded"/>
2927   </xsd:sequence>
2928   <xsd:attribute name="current" type="xsd:unsignedInt" use="optional"/>
2929   <xsd:attribute name="show" type="xsd:unsignedInt" use="optional"/>
2930   <xsd:attribute name="sqref" type="ST_Sqref" use="optional"/>
2931 </xsd:complexType>
2932 <xsd:complexType name="CT_SheetProtection">
2933   <xsd:attribute name="password" type="ST_UnsignedShortHex" use="optional"/>
2934   <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
2935   <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
2936   <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
2937   <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
2938   <xsd:attribute name="sheet" type="xsd:boolean" use="optional" default="false"/>
2939   <xsd:attribute name="objects" type="xsd:boolean" use="optional" default="false"/>
2940   <xsd:attribute name="scenarios" type="xsd:boolean" use="optional" default="false"/>
2941   <xsd:attribute name="formatCells" type="xsd:boolean" use="optional" default="true"/>
2942   <xsd:attribute name="formatColumns" type="xsd:boolean" use="optional" default="true"/>
2943   <xsd:attribute name="formatRows" type="xsd:boolean" use="optional" default="true"/>
2944   <xsd:attribute name="insertColumns" type="xsd:boolean" use="optional" default="true"/>
2945   <xsd:attribute name="insertRows" type="xsd:boolean" use="optional" default="true"/>
2946   <xsd:attribute name="insertHyperlinks" type="xsd:boolean" use="optional" default="true"/>
2947   <xsd:attribute name="deleteColumns" type="xsd:boolean" use="optional" default="true"/>
2948   <xsd:attribute name="deleteRows" type="xsd:boolean" use="optional" default="true"/>
2949   <xsd:attribute name="selectLockedCells" type="xsd:boolean" use="optional" default="false"/>

```

```

2950     <xsd:attribute name="sort" type="xsd:boolean" use="optional" default="true"/>
2951     <xsd:attribute name="autoFilter" type="xsd:boolean" use="optional" default="true"/>
2952     <xsd:attribute name="pivotTables" type="xsd:boolean" use="optional" default="true"/>
2953     <xsd:attribute name="selectUnlockedCells" type="xsd:boolean" use="optional" default="false"/>
2954 </xsd:complexType>
2955 <xsd:complexType name="CT_ProtectedRanges">
2956     <xsd:sequence>
2957         <xsd:element name="protectedRange" type="CT_ProtectedRange" minOccurs="1"
2958             maxOccurs="unbounded"/>
2959     </xsd:sequence>
2960 </xsd:complexType>
2961 <xsd:complexType name="CT_ProtectedRange">
2962     <xsd:sequence>
2963         <xsd:element name="securityDescriptor" type="xsd:string" minOccurs="0"
2964             maxOccurs="unbounded"/>
2965     </xsd:sequence>
2966     <xsd:attribute name="password" type="ST_UnsignedShortHex" use="optional"/>
2967     <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
2968     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2969     <xsd:attribute name="securityDescriptor" type="xsd:string" use="optional"/>
2970     <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
2971     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
2972     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
2973     <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
2974 </xsd:complexType>
2975 <xsd:complexType name="CT_Scenario">
2976     <xsd:sequence>
2977         <xsd:element name="inputCells" type="CT_InputCells" minOccurs="1" maxOccurs="unbounded"/>
2978     </xsd:sequence>
2979     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
2980     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="false"/>
2981     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
2982     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
2983     <xsd:attribute name="user" type="s:ST_Xstring" use="optional"/>
2984     <xsd:attribute name="comment" type="s:ST_Xstring" use="optional"/>
2985 </xsd:complexType>
2986 <xsd:complexType name="CT_InputCells">
2987     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
2988     <xsd:attribute name="deleted" type="xsd:boolean" use="optional" default="false"/>
2989     <xsd:attribute name="undone" type="xsd:boolean" use="optional" default="false"/>
2990     <xsd:attribute name="val" type="s:ST_Xstring" use="required"/>
2991     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
2992 </xsd:complexType>
2993 <xsd:complexType name="CT_CellWatches">
2994     <xsd:sequence>
2995         <xsd:element name="cellWatch" type="CT_CellWatch" minOccurs="1" maxOccurs="unbounded"/>
2996     </xsd:sequence>
2997 </xsd:complexType>
2998 <xsd:complexType name="CT_CellWatch">
2999     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
3000 </xsd:complexType>
3001 <xsd:complexType name="CT_Chartsheet">
3002     <xsd:sequence>

```

```

3003     <xsd:element name="sheetPr" type="CT_ChartsheetPr" minOccurs="0" maxOccurs="1"/>
3004     <xsd:element name="sheetViews" type="CT_ChartsheetViews" minOccurs="1" maxOccurs="1"/>
3005     <xsd:element name="sheetProtection" type="CT_ChartsheetProtection" minOccurs="0"
3006         maxOccurs="1"/>
3007     <xsd:element name="customSheetViews" type="CT_CustomChartsheetViews" minOccurs="0"
3008         maxOccurs="1"/>
3009     <xsd:element name="pageMargins" minOccurs="0" type="CT_PageMargins"/>
3010     <xsd:element name="pageSetup" type="CT_CsPageSetup" minOccurs="0" maxOccurs="1"/>
3011     <xsd:element name="headerFooter" minOccurs="0" type="CT_HeaderFooter"/>
3012     <xsd:element name="drawing" type="CT_Drawing" minOccurs="1" maxOccurs="1"/>
3013     <xsd:element name="legacyDrawing" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
3014     <xsd:element name="legacyDrawingHF" type="CT_LegacyDrawing" minOccurs="0" maxOccurs="1"/>
3015     <xsd:element name="drawingHF" type="CT_DrawingHF" minOccurs="0" maxOccurs="1"/>
3016     <xsd:element name="picture" type="CT_SheetBackgroundPicture" minOccurs="0" maxOccurs="1"/>
3017     <xsd:element name="webPublishItems" type="CT_WebPublishItems" minOccurs="0"
3018         maxOccurs="1"/>
3019     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3020 </xsd:sequence>
3021 </xsd:complexType>
3022 <xsd:complexType name="CT_ChartsheetPr">
3023     <xsd:sequence>
3024         <xsd:element name="tabColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3025     </xsd:sequence>
3026     <xsd:attribute name="published" type="xsd:boolean" use="optional" default="true"/>
3027     <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
3028 </xsd:complexType>
3029 <xsd:complexType name="CT_ChartsheetViews">
3030     <xsd:sequence>
3031         <xsd:element name="sheetView" type="CT_ChartsheetView" minOccurs="1"
3032             maxOccurs="unbounded"/>
3033         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3034     </xsd:sequence>
3035 </xsd:complexType>
3036 <xsd:complexType name="CT_ChartsheetView">
3037     <xsd:sequence>
3038         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3039     </xsd:sequence>
3040     <xsd:attribute name="tabSelected" type="xsd:boolean" use="optional" default="false"/>
3041     <xsd:attribute name="zoomScale" type="xsd:unsignedInt" default="100" use="optional"/>
3042     <xsd:attribute name="workbookViewId" type="xsd:unsignedInt" use="required"/>
3043     <xsd:attribute name="zoomToFit" type="xsd:boolean" use="optional" default="false"/>
3044 </xsd:complexType>
3045 <xsd:complexType name="CT_ChartsheetProtection">
3046     <xsd:attribute name="password" type="ST_UnsignedShortHex" use="optional"/>
3047     <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
3048     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
3049     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
3050     <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
3051     <xsd:attribute name="content" type="xsd:boolean" use="optional" default="false"/>
3052     <xsd:attribute name="objects" type="xsd:boolean" use="optional" default="false"/>
3053 </xsd:complexType>
3054 <xsd:complexType name="CT_CsPageSetup">
3055     <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>

```

```

3056 <xsd:attribute name="paperHeight" type="s:ST PositiveUniversalMeasure" use="optional"/>
3057 <xsd:attribute name="paperWidth" type="s:ST PositiveUniversalMeasure" use="optional"/>
3058 <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
3059 <xsd:attribute name="orientation" type="ST Orientation" use="optional" default="default"/>
3060 <xsd:attribute name="usePrinterDefaults" type="xsd:boolean" use="optional" default="true"/>
3061 <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
3062 <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
3063 <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
3064 <xsd:attribute name="horizontalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
3065 <xsd:attribute name="verticalDpi" type="xsd:unsignedInt" use="optional" default="600"/>
3066 <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
3067 <xsd:attribute ref="r:id" use="optional"/>
3068 </xsd:complexType>
3069 <xsd:complexType name="CT_CustomChartsheetViews">
3070 <xsd:sequence>
3071 <xsd:element name="customSheetView" minOccurs="0" maxOccurs="unbounded"
3072 type="CT CustomChartsheetView"/>
3073 </xsd:sequence>
3074 </xsd:complexType>
3075 <xsd:complexType name="CT_CustomChartsheetView">
3076 <xsd:sequence>
3077 <xsd:element name="pageMargins" type="CT PageMargins" minOccurs="0" maxOccurs="1"/>
3078 <xsd:element name="pageSetup" type="CT CsPageSetup" minOccurs="0" maxOccurs="1"/>
3079 <xsd:element name="headerFooter" type="CT HeaderFooter" minOccurs="0" maxOccurs="1"/>
3080 </xsd:sequence>
3081 <xsd:attribute name="guid" type="s:ST Guid" use="required"/>
3082 <xsd:attribute name="scale" type="xsd:unsignedInt" default="100"/>
3083 <xsd:attribute name="state" type="ST SheetState" default="visible"/>
3084 <xsd:attribute name="zoomToFit" type="xsd:boolean" use="optional" default="false"/>
3085 </xsd:complexType>
3086 <xsd:complexType name="CT_CustomProperties">
3087 <xsd:sequence>
3088 <xsd:element name="customPr" type="CT CustomProperty" minOccurs="1"
3089 maxOccurs="unbounded"/>
3090 </xsd:sequence>
3091 </xsd:complexType>
3092 <xsd:complexType name="CT_CustomProperty">
3093 <xsd:attribute name="name" type="s:ST Xstring" use="required"/>
3094 <xsd:attribute ref="r:id" use="required"/>
3095 </xsd:complexType>
3096 <xsd:complexType name="CT_OleObjects">
3097 <xsd:sequence>
3098 <xsd:element name="oleObject" type="CT OleObject" minOccurs="1" maxOccurs="unbounded"/>
3099 </xsd:sequence>
3100 </xsd:complexType>
3101 <xsd:complexType name="CT_OleObject">
3102 <xsd:sequence>
3103 <xsd:element name="objectPr" type="CT ObjectPr" minOccurs="0" maxOccurs="1"/>
3104 </xsd:sequence>
3105 <xsd:attribute name="progId" type="xsd:string" use="optional"/>
3106 <xsd:attribute name="dvAspect" type="ST DvAspect" use="optional" default="DVASPECT_CONTENT"/>
3107 <xsd:attribute name="link" type="s:ST Xstring" use="optional"/>
3108 <xsd:attribute name="oleUpdate" type="ST OleUpdate" use="optional"/>

```

```

3109     <xsd:attribute name="autoLoad" type="xsd:boolean" use="optional" default="false"/>
3110     <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="required"/>
3111     <xsd:attribute ref="r:id" use="optional"/>
3112 </xsd:complexType>
3113 <xsd:complexType name="CT_ObjectPr">
3114     <xsd:sequence>
3115         <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
3116     </xsd:sequence>
3117     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
3118     <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
3119     <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
3120     <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
3121     <xsd:attribute name="uiObject" type="xsd:boolean" use="optional" default="false"/>
3122     <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
3123     <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
3124     <xsd:attribute name="autoPict" type="xsd:boolean" use="optional" default="true"/>
3125     <xsd:attribute name="macro" type="ST_Formula" use="optional"/>
3126     <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
3127     <xsd:attribute name="dde" type="xsd:boolean" use="optional" default="false"/>
3128     <xsd:attribute ref="r:id" use="optional"/>
3129 </xsd:complexType>
3130 <xsd:simpleType name="ST_DvAspect">
3131     <xsd:restriction base="xsd:string">
3132         <xsd:enumeration value="DVASPECT_CONTENT"/>
3133         <xsd:enumeration value="DVASPECT_ICON"/>
3134     </xsd:restriction>
3135 </xsd:simpleType>
3136 <xsd:simpleType name="ST_OleUpdate">
3137     <xsd:restriction base="xsd:string">
3138         <xsd:enumeration value="OLEUPDATE_ALWAYS"/>
3139         <xsd:enumeration value="OLEUPDATE_ONCALL"/>
3140     </xsd:restriction>
3141 </xsd:simpleType>
3142 <xsd:complexType name="CT_WebPublishItems">
3143     <xsd:sequence>
3144         <xsd:element name="webPublishItem" type="CT_WebPublishItem" minOccurs="1"
3145             maxOccurs="unbounded"/>
3146     </xsd:sequence>
3147     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3148 </xsd:complexType>
3149 <xsd:complexType name="CT_WebPublishItem">
3150     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3151     <xsd:attribute name="divId" type="s:ST_Xstring" use="required"/>
3152     <xsd:attribute name="sourceType" type="ST_WebSourceType" use="required"/>
3153     <xsd:attribute name="sourceRef" type="ST_Ref" use="optional"/>
3154     <xsd:attribute name="sourceObject" type="s:ST_Xstring" use="optional"/>
3155     <xsd:attribute name="destinationFile" type="s:ST_Xstring" use="required"/>
3156     <xsd:attribute name="title" type="s:ST_Xstring" use="optional"/>
3157     <xsd:attribute name="autoRepublish" type="xsd:boolean" use="optional" default="false"/>
3158 </xsd:complexType>
3159 <xsd:complexType name="CT_Controls">
3160     <xsd:sequence>
3161         <xsd:element name="control" type="CT_Control" minOccurs="1" maxOccurs="unbounded"/>

```



```

3162     </xsd:sequence>
3163 </xsd:complexType>
3164 <xsd:complexType name="CT_Control">
3165     <xsd:sequence>
3166         <xsd:element name="controlPr" type="CT_ControlPr" minOccurs="0" maxOccurs="1"/>
3167     </xsd:sequence>
3168     <xsd:attribute name="shapeId" type="xsd:unsignedInt" use="required"/>
3169     <xsd:attribute ref="r:id" use="required"/>
3170     <xsd:attribute name="name" type="xsd:string" use="optional"/>
3171 </xsd:complexType>
3172 <xsd:complexType name="CT_ControlPr">
3173     <xsd:sequence>
3174         <xsd:element name="anchor" type="CT_ObjectAnchor" minOccurs="1" maxOccurs="1"/>
3175     </xsd:sequence>
3176     <xsd:attribute name="locked" type="xsd:boolean" use="optional" default="true"/>
3177     <xsd:attribute name="defaultSize" type="xsd:boolean" use="optional" default="true"/>
3178     <xsd:attribute name="print" type="xsd:boolean" use="optional" default="true"/>
3179     <xsd:attribute name="disabled" type="xsd:boolean" use="optional" default="false"/>
3180     <xsd:attribute name="recalcAlways" type="xsd:boolean" use="optional" default="false"/>
3181     <xsd:attribute name="uiObject" type="xsd:boolean" use="optional" default="false"/>
3182     <xsd:attribute name="autoFill" type="xsd:boolean" use="optional" default="true"/>
3183     <xsd:attribute name="autoLine" type="xsd:boolean" use="optional" default="true"/>
3184     <xsd:attribute name="autoPict" type="xsd:boolean" use="optional" default="true"/>
3185     <xsd:attribute name="macro" type="ST_Formula" use="optional"/>
3186     <xsd:attribute name="altText" type="s:ST_Xstring" use="optional"/>
3187     <xsd:attribute name="linkedCell" type="ST_Formula" use="optional"/>
3188     <xsd:attribute name="listFillRange" type="ST_Formula" use="optional"/>
3189     <xsd:attribute name="cf" type="s:ST_Xstring" use="optional" default="pict"/>
3190     <xsd:attribute ref="r:id" use="optional"/>
3191 </xsd:complexType>
3192 <xsd:simpleType name="ST_WebSourceType">
3193     <xsd:restriction base="xsd:string">
3194         <xsd:enumeration value="sheet"/>
3195         <xsd:enumeration value="printArea"/>
3196         <xsd:enumeration value="autoFilter"/>
3197         <xsd:enumeration value="range"/>
3198         <xsd:enumeration value="chart"/>
3199         <xsd:enumeration value="pivotTable"/>
3200         <xsd:enumeration value="query"/>
3201         <xsd:enumeration value="label"/>
3202     </xsd:restriction>
3203 </xsd:simpleType>
3204 <xsd:complexType name="CT_IgnoredErrors">
3205     <xsd:sequence>
3206         <xsd:element name="ignoredError" type="CT_IgnoredError" minOccurs="1"
3207             maxOccurs="unbounded"/>
3208         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3209     </xsd:sequence>
3210 </xsd:complexType>
3211 <xsd:complexType name="CT_IgnoredError">
3212     <xsd:attribute name="sqref" type="ST_Sqref" use="required"/>
3213     <xsd:attribute name="evalError" type="xsd:boolean" use="optional" default="false"/>
3214     <xsd:attribute name="twoDigitTextYear" type="xsd:boolean" use="optional" default="false"/>

```

```

3215     <xsd:attribute name="numberStoredAsText" type="xsd:boolean" use="optional" default="false"/>
3216     <xsd:attribute name="formula" type="xsd:boolean" use="optional" default="false"/>
3217     <xsd:attribute name="formulaRange" type="xsd:boolean" use="optional" default="false"/>
3218     <xsd:attribute name="unlockedFormula" type="xsd:boolean" use="optional" default="false"/>
3219     <xsd:attribute name="emptyCellReference" type="xsd:boolean" use="optional" default="false"/>
3220     <xsd:attribute name="listDataValidation" type="xsd:boolean" use="optional" default="false"/>
3221     <xsd:attribute name="calculatedColumn" type="xsd:boolean" use="optional" default="false"/>
3222 </xsd:complexType>
3223 <xsd:simpleType name="ST_PaneState">
3224     <xsd:restriction base="xsd:string">
3225         <xsd:enumeration value="split"/>
3226         <xsd:enumeration value="frozen"/>
3227         <xsd:enumeration value="frozenSplit"/>
3228     </xsd:restriction>
3229 </xsd:simpleType>
3230 <xsd:complexType name="CT_TableParts">
3231     <xsd:sequence>
3232         <xsd:element name="tablePart" type="CT_TablePart" minOccurs="0" maxOccurs="unbounded"/>
3233     </xsd:sequence>
3234     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3235 </xsd:complexType>
3236 <xsd:complexType name="CT_TablePart">
3237     <xsd:attribute ref="r:id" use="required"/>
3238 </xsd:complexType>
3239 <xsd:element name="metadata" type="CT_Metadata"/>
3240 <xsd:complexType name="CT_Metadata">
3241     <xsd:sequence>
3242         <xsd:element name="metadataTypes" type="CT_MetadataTypes" minOccurs="0" maxOccurs="1"/>
3243         <xsd:element name="metadataStrings" type="CT_MetadataStrings" minOccurs="0"
3244             maxOccurs="1"/>
3245         <xsd:element name="mdxMetadata" type="CT_MdxMetadata" minOccurs="0" maxOccurs="1"/>
3246         <xsd:element name="futureMetadata" type="CT_FutureMetadata" minOccurs="0"
3247             maxOccurs="unbounded"/>
3248         <xsd:element name="cellMetadata" type="CT_MetadataBlocks" minOccurs="0" maxOccurs="1"/>
3249         <xsd:element name="valueMetadata" type="CT_MetadataBlocks" minOccurs="0" maxOccurs="1"/>
3250         <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3251     </xsd:sequence>
3252 </xsd:complexType>
3253 <xsd:complexType name="CT_MetadataTypes">
3254     <xsd:sequence>
3255         <xsd:element name="metadataType" type="CT_MetadataType" minOccurs="1"
3256             maxOccurs="unbounded"/>
3257     </xsd:sequence>
3258     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3259 </xsd:complexType>
3260 <xsd:complexType name="CT_MetadataType">
3261     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3262     <xsd:attribute name="minSupportedVersion" type="xsd:unsignedInt" use="required"/>
3263     <xsd:attribute name="ghostRow" type="xsd:boolean" use="optional" default="false"/>
3264     <xsd:attribute name="ghostCol" type="xsd:boolean" use="optional" default="false"/>
3265     <xsd:attribute name="edit" type="xsd:boolean" use="optional" default="false"/>
3266     <xsd:attribute name="delete" type="xsd:boolean" use="optional" default="false"/>
3267     <xsd:attribute name="copy" type="xsd:boolean" use="optional" default="false"/>

```

```

3268 <xsd:attribute name="pasteAll" type="xsd:boolean" use="optional" default="false"/>
3269 <xsd:attribute name="pasteFormulas" type="xsd:boolean" use="optional" default="false"/>
3270 <xsd:attribute name="pasteValues" type="xsd:boolean" use="optional" default="false"/>
3271 <xsd:attribute name="pasteFormats" type="xsd:boolean" use="optional" default="false"/>
3272 <xsd:attribute name="pasteComments" type="xsd:boolean" use="optional" default="false"/>
3273 <xsd:attribute name="pasteDataValidation" type="xsd:boolean" use="optional" default="false"/>
3274 <xsd:attribute name="pasteBorders" type="xsd:boolean" use="optional" default="false"/>
3275 <xsd:attribute name="pasteColWidths" type="xsd:boolean" use="optional" default="false"/>
3276 <xsd:attribute name="pasteNumberFormats" type="xsd:boolean" use="optional" default="false"/>
3277 <xsd:attribute name="merge" type="xsd:boolean" use="optional" default="false"/>
3278 <xsd:attribute name="splitFirst" type="xsd:boolean" use="optional" default="false"/>
3279 <xsd:attribute name="splitAll" type="xsd:boolean" use="optional" default="false"/>
3280 <xsd:attribute name="rowColShift" type="xsd:boolean" use="optional" default="false"/>
3281 <xsd:attribute name="clearAll" type="xsd:boolean" default="false"/>
3282 <xsd:attribute name="clearFormats" type="xsd:boolean" use="optional" default="false"/>
3283 <xsd:attribute name="clearContents" type="xsd:boolean" use="optional" default="false"/>
3284 <xsd:attribute name="clearComments" type="xsd:boolean" use="optional" default="false"/>
3285 <xsd:attribute name="assign" type="xsd:boolean" use="optional" default="false"/>
3286 <xsd:attribute name="coerce" type="xsd:boolean" use="optional" default="false"/>
3287 <xsd:attribute name="adjust" type="xsd:boolean" use="optional" default="false"/>
3288 <xsd:attribute name="cellMeta" type="xsd:boolean" use="optional" default="false"/>
3289 </xsd:complexType>
3290 <xsd:complexType name="CT_MetadataBlocks">
3291 <xsd:sequence>
3292 <xsd:element name="bk" type="CT_MetadataBlock" minOccurs="1" maxOccurs="unbounded"/>
3293 </xsd:sequence>
3294 <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3295 </xsd:complexType>
3296 <xsd:complexType name="CT_MetadataBlock">
3297 <xsd:sequence>
3298 <xsd:element name="rc" type="CT_MetadataRecord" minOccurs="1" maxOccurs="unbounded"/>
3299 </xsd:sequence>
3300 </xsd:complexType>
3301 <xsd:complexType name="CT_MetadataRecord">
3302 <xsd:attribute name="t" type="xsd:unsignedInt" use="required"/>
3303 <xsd:attribute name="v" type="xsd:unsignedInt" use="required"/>
3304 </xsd:complexType>
3305 <xsd:complexType name="CT_FutureMetadata">
3306 <xsd:sequence>
3307 <xsd:element name="bk" type="CT_FutureMetadataBlock" minOccurs="0" maxOccurs="unbounded"/>
3308 <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3309 </xsd:sequence>
3310 <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3311 <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3312 </xsd:complexType>
3313 <xsd:complexType name="CT_FutureMetadataBlock">
3314 <xsd:sequence>
3315 <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="CT_ExtensionList"/>
3316 </xsd:sequence>
3317 </xsd:complexType>
3318 <xsd:complexType name="CT_MdxMetadata">
3319 <xsd:sequence>
3320 <xsd:element name="mdx" type="CT_Mdx" minOccurs="1" maxOccurs="unbounded"/>

```

```

3321     </xsd:sequence>
3322     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3323 </xsd:complexType>
3324 <xsd:complexType name="CT_Mdx">
3325     <xsd:choice minOccurs="1" maxOccurs="1">
3326         <xsd:element name="t" type="CT_MdxTuple"/>
3327         <xsd:element name="ms" type="CT_MdxSet"/>
3328         <xsd:element name="p" type="CT_MdxMemberProp"/>
3329         <xsd:element name="k" type="CT_MdxKPI"/>
3330     </xsd:choice>
3331     <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3332     <xsd:attribute name="f" type="ST_MdxFunctionType" use="required"/>
3333 </xsd:complexType>
3334 <xsd:simpleType name="ST_MdxFunctionType">
3335     <xsd:restriction base="xsd:string">
3336         <xsd:enumeration value="m"/>
3337         <xsd:enumeration value="v"/>
3338         <xsd:enumeration value="s"/>
3339         <xsd:enumeration value="c"/>
3340         <xsd:enumeration value="r"/>
3341         <xsd:enumeration value="p"/>
3342         <xsd:enumeration value="k"/>
3343     </xsd:restriction>
3344 </xsd:simpleType>
3345 <xsd:complexType name="CT_MdxTuple">
3346     <xsd:sequence>
3347         <xsd:element name="n" type="CT_MetadataStringIndex" minOccurs="0" maxOccurs="unbounded"/>
3348     </xsd:sequence>
3349     <xsd:attribute name="c" type="xsd:unsignedInt" use="optional" default="0"/>
3350     <xsd:attribute name="ct" type="s:ST_Xstring" use="optional"/>
3351     <xsd:attribute name="si" type="xsd:unsignedInt" use="optional"/>
3352     <xsd:attribute name="fi" type="xsd:unsignedInt" use="optional"/>
3353     <xsd:attribute name="bc" type="ST_UnsignedIntHex" use="optional"/>
3354     <xsd:attribute name="fc" type="ST_UnsignedIntHex" use="optional"/>
3355     <xsd:attribute name="i" type="xsd:boolean" use="optional" default="false"/>
3356     <xsd:attribute name="u" type="xsd:boolean" use="optional" default="false"/>
3357     <xsd:attribute name="st" type="xsd:boolean" use="optional" default="false"/>
3358     <xsd:attribute name="b" type="xsd:boolean" use="optional" default="false"/>
3359 </xsd:complexType>
3360 <xsd:complexType name="CT_MdxSet">
3361     <xsd:sequence>
3362         <xsd:element name="n" type="CT_MetadataStringIndex" minOccurs="0" maxOccurs="unbounded"/>
3363     </xsd:sequence>
3364     <xsd:attribute name="ns" type="xsd:unsignedInt" use="required"/>
3365     <xsd:attribute name="c" type="xsd:unsignedInt" use="optional" default="0"/>
3366     <xsd:attribute name="o" type="ST_MdxSetOrder" use="optional" default="u"/>
3367 </xsd:complexType>
3368 <xsd:simpleType name="ST_MdxSetOrder">
3369     <xsd:restriction base="xsd:string">
3370         <xsd:enumeration value="u"/>
3371         <xsd:enumeration value="a"/>
3372         <xsd:enumeration value="d"/>
3373         <xsd:enumeration value="aa"/>

```

```

3374         <xsd:enumeration value="ad"/>
3375         <xsd:enumeration value="na"/>
3376         <xsd:enumeration value="nd"/>
3377     </xsd:restriction>
3378 </xsd:simpleType>
3379 <xsd:complexType name="CT_MdxMemeberProp">
3380     <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3381     <xsd:attribute name="np" type="xsd:unsignedInt" use="required"/>
3382 </xsd:complexType>
3383 <xsd:complexType name="CT_MdxKPI">
3384     <xsd:attribute name="n" type="xsd:unsignedInt" use="required"/>
3385     <xsd:attribute name="np" type="xsd:unsignedInt" use="required"/>
3386     <xsd:attribute name="p" type="ST_MdxKPIProperty" use="required"/>
3387 </xsd:complexType>
3388 <xsd:simpleType name="ST_MdxKPIProperty">
3389     <xsd:restriction base="xsd:string">
3390         <xsd:enumeration value="v"/>
3391         <xsd:enumeration value="g"/>
3392         <xsd:enumeration value="s"/>
3393         <xsd:enumeration value="t"/>
3394         <xsd:enumeration value="w"/>
3395         <xsd:enumeration value="m"/>
3396     </xsd:restriction>
3397 </xsd:simpleType>
3398 <xsd:complexType name="CT_MetadataStringIndex">
3399     <xsd:attribute name="x" type="xsd:unsignedInt" use="required"/>
3400     <xsd:attribute name="s" type="xsd:boolean" use="optional" default="false"/>
3401 </xsd:complexType>
3402 <xsd:complexType name="CT_MetadataStrings">
3403     <xsd:sequence>
3404         <xsd:element name="s" type="CT_XStringElement" minOccurs="1" maxOccurs="unbounded"/>
3405     </xsd:sequence>
3406     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional" default="0"/>
3407 </xsd:complexType>
3408 <xsd:element name="singleXmlCells" type="CT_SingleXmlCells"/>
3409 <xsd:complexType name="CT_SingleXmlCells">
3410     <xsd:sequence>
3411         <xsd:element name="singleXmlCell" type="CT_SingleXmlCell" maxOccurs="unbounded"/>
3412     </xsd:sequence>
3413 </xsd:complexType>
3414 <xsd:complexType name="CT_SingleXmlCell">
3415     <xsd:sequence>
3416         <xsd:element name="xmlCellPr" type="CT_XmlCellPr" minOccurs="1" maxOccurs="1"/>
3417         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3418     </xsd:sequence>
3419     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3420     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
3421     <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="required"/>
3422 </xsd:complexType>
3423 <xsd:complexType name="CT_XmlCellPr">
3424     <xsd:sequence>
3425         <xsd:element name="xmlPr" type="CT_XmlPr" minOccurs="1" maxOccurs="1"/>
3426         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>

```

```

3427     </xsd:sequence>
3428     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3429     <xsd:attribute name="uniqueName" type="s:ST_Xstring" use="optional"/>
3430 </xsd:complexType>
3431 <xsd:complexType name="CT_XmlPr">
3432     <xsd:sequence>
3433         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3434     </xsd:sequence>
3435     <xsd:attribute name="mapId" type="xsd:unsignedInt" use="required"/>
3436     <xsd:attribute name="xpath" type="s:ST_Xstring" use="required"/>
3437     <xsd:attribute name="xmlDataType" type="ST_XmlDataType" use="required"/>
3438 </xsd:complexType>
3439 <xsd:element name="styleSheet" type="CT_Stylesheet"/>
3440 <xsd:complexType name="CT_Stylesheet">
3441     <xsd:sequence>
3442         <xsd:element name="numFmts" type="CT_NumFmts" minOccurs="0" maxOccurs="1"/>
3443         <xsd:element name="fonts" type="CT_Fonts" minOccurs="0" maxOccurs="1"/>
3444         <xsd:element name="fills" type="CT_Fills" minOccurs="0" maxOccurs="1"/>
3445         <xsd:element name="borders" type="CT_Borders" minOccurs="0" maxOccurs="1"/>
3446         <xsd:element name="cellStyleXfs" type="CT_CellStyleXfs" minOccurs="0" maxOccurs="1"/>
3447         <xsd:element name="cellXfs" type="CT_CellXfs" minOccurs="0" maxOccurs="1"/>
3448         <xsd:element name="cellStyles" type="CT_CellStyles" minOccurs="0" maxOccurs="1"/>
3449         <xsd:element name="dxfs" type="CT_Dxfs" minOccurs="0" maxOccurs="1"/>
3450         <xsd:element name="tableStyles" type="CT_TableStyles" minOccurs="0" maxOccurs="1"/>
3451         <xsd:element name="colors" type="CT_Colors" minOccurs="0" maxOccurs="1"/>
3452         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3453     </xsd:sequence>
3454 </xsd:complexType>
3455 <xsd:complexType name="CT_CellAlignment">
3456     <xsd:attribute name="horizontal" type="ST_HorizontalAlignment" use="optional"/>
3457     <xsd:attribute name="vertical" type="ST_VerticalAlignment" use="optional"/>
3458     <xsd:attribute name="textRotation" type="xsd:unsignedInt" use="optional"/>
3459     <xsd:attribute name="wrapText" type="xsd:boolean" use="optional"/>
3460     <xsd:attribute name="indent" type="xsd:unsignedInt" use="optional"/>
3461     <xsd:attribute name="relativeIndent" type="xsd:int" use="optional"/>
3462     <xsd:attribute name="justifyLastLine" type="xsd:boolean" use="optional"/>
3463     <xsd:attribute name="shrinkToFit" type="xsd:boolean" use="optional"/>
3464     <xsd:attribute name="readingOrder" type="xsd:unsignedInt" use="optional"/>
3465 </xsd:complexType>
3466 <xsd:simpleType name="ST_BorderStyle">
3467     <xsd:restriction base="xsd:string">
3468         <xsd:enumeration value="none"/>
3469         <xsd:enumeration value="thin"/>
3470         <xsd:enumeration value="medium"/>
3471         <xsd:enumeration value="dashed"/>
3472         <xsd:enumeration value="dotted"/>
3473         <xsd:enumeration value="thick"/>
3474         <xsd:enumeration value="double"/>
3475         <xsd:enumeration value="hair"/>
3476         <xsd:enumeration value="mediumDashed"/>
3477         <xsd:enumeration value="dashDot"/>
3478         <xsd:enumeration value="mediumDashDot"/>
3479         <xsd:enumeration value="dashDotDot"/>

```

```

3480         <xsd:enumeration value="mediumDashDotDot"/>
3481         <xsd:enumeration value="slantDashDot"/>
3482     </xsd:restriction>
3483 </xsd:simpleType>
3484 <xsd:complexType name="CT_Borders">
3485     <xsd:sequence>
3486         <xsd:element name="border" type="CT_Border" minOccurs="0" maxOccurs="unbounded"/>
3487     </xsd:sequence>
3488     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3489 </xsd:complexType>
3490 <xsd:complexType name="CT_Border">
3491     <xsd:sequence>
3492         <xsd:element name="start" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3493         <xsd:element name="end" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3494         <xsd:element name="left" type="CT_BorderPr" minOccurs="0"/>
3495         <xsd:element name="right" type="CT_BorderPr" minOccurs="0"/>
3496         <xsd:element name="top" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3497         <xsd:element name="bottom" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3498         <xsd:element name="diagonal" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3499         <xsd:element name="vertical" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3500         <xsd:element name="horizontal" type="CT_BorderPr" minOccurs="0" maxOccurs="1"/>
3501     </xsd:sequence>
3502     <xsd:attribute name="diagonalUp" type="xsd:boolean" use="optional"/>
3503     <xsd:attribute name="diagonalDown" type="xsd:boolean" use="optional"/>
3504     <xsd:attribute name="outline" type="xsd:boolean" use="optional" default="true"/>
3505 </xsd:complexType>
3506 <xsd:complexType name="CT_BorderPr">
3507     <xsd:sequence>
3508         <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3509     </xsd:sequence>
3510     <xsd:attribute name="style" type="ST_BorderStyle" use="optional" default="none"/>
3511 </xsd:complexType>
3512 <xsd:complexType name="CT_CellProtection">
3513     <xsd:attribute name="locked" type="xsd:boolean" use="optional"/>
3514     <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
3515 </xsd:complexType>
3516 <xsd:complexType name="CT_Fonts">
3517     <xsd:sequence>
3518         <xsd:element name="font" type="CT_Font" minOccurs="0" maxOccurs="unbounded"/>
3519     </xsd:sequence>
3520     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3521 </xsd:complexType>
3522 <xsd:complexType name="CT_Fills">
3523     <xsd:sequence>
3524         <xsd:element name="fill" type="CT_Fill" minOccurs="0" maxOccurs="unbounded"/>
3525     </xsd:sequence>
3526     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3527 </xsd:complexType>
3528 <xsd:complexType name="CT_Fill">
3529     <xsd:choice minOccurs="1" maxOccurs="1">
3530         <xsd:element name="patternFill" type="CT_PatternFill" minOccurs="0" maxOccurs="1"/>
3531         <xsd:element name="gradientFill" type="CT_GradientFill" minOccurs="0" maxOccurs="1"/>
3532     </xsd:choice>

```

```

3533 </xsd:complexType>
3534 <xsd:complexType name="CT_PatternFill">
3535   <xsd:sequence>
3536     <xsd:element name="fgColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3537     <xsd:element name="bgColor" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3538   </xsd:sequence>
3539   <xsd:attribute name="patternType" type="ST_PatternType" use="optional"/>
3540 </xsd:complexType>
3541 <xsd:complexType name="CT_Color">
3542   <xsd:attribute name="auto" type="xsd:boolean" use="optional"/>
3543   <xsd:attribute name="indexed" type="xsd:unsignedInt" use="optional"/>
3544   <xsd:attribute name="rgb" type="ST_UnsignedIntHex" use="optional"/>
3545   <xsd:attribute name="theme" type="xsd:unsignedInt" use="optional"/>
3546   <xsd:attribute name="tint" type="xsd:double" use="optional" default="0.0"/>
3547 </xsd:complexType>
3548 <xsd:simpleType name="ST_PatternType">
3549   <xsd:restriction base="xsd:string">
3550     <xsd:enumeration value="none"/>
3551     <xsd:enumeration value="solid"/>
3552     <xsd:enumeration value="mediumGray"/>
3553     <xsd:enumeration value="darkGray"/>
3554     <xsd:enumeration value="lightGray"/>
3555     <xsd:enumeration value="darkHorizontal"/>
3556     <xsd:enumeration value="darkVertical"/>
3557     <xsd:enumeration value="darkDown"/>
3558     <xsd:enumeration value="darkUp"/>
3559     <xsd:enumeration value="darkGrid"/>
3560     <xsd:enumeration value="darkTrellis"/>
3561     <xsd:enumeration value="lightHorizontal"/>
3562     <xsd:enumeration value="lightVertical"/>
3563     <xsd:enumeration value="lightDown"/>
3564     <xsd:enumeration value="lightUp"/>
3565     <xsd:enumeration value="lightGrid"/>
3566     <xsd:enumeration value="lightTrellis"/>
3567     <xsd:enumeration value="gray125"/>
3568     <xsd:enumeration value="gray0625"/>
3569   </xsd:restriction>
3570 </xsd:simpleType>
3571 <xsd:complexType name="CT_GradientFill">
3572   <xsd:sequence>
3573     <xsd:element name="stop" type="CT_GradientStop" minOccurs="0" maxOccurs="unbounded"/>
3574   </xsd:sequence>
3575   <xsd:attribute name="type" type="ST_GradientType" use="optional" default="linear"/>
3576   <xsd:attribute name="degree" type="xsd:double" use="optional" default="0"/>
3577   <xsd:attribute name="left" type="xsd:double" use="optional" default="0"/>
3578   <xsd:attribute name="right" type="xsd:double" use="optional" default="0"/>
3579   <xsd:attribute name="top" type="xsd:double" use="optional" default="0"/>
3580   <xsd:attribute name="bottom" type="xsd:double" use="optional" default="0"/>
3581 </xsd:complexType>
3582 <xsd:complexType name="CT_GradientStop">
3583   <xsd:sequence>
3584     <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="1"/>
3585   </xsd:sequence>

```



```

3586     <xsd:attribute name="position" type="xsd:double" use="required"/>
3587 </xsd:complexType>
3588 <xsd:simpleType name="ST_GradientType">
3589     <xsd:restriction base="xsd:string">
3590         <xsd:enumeration value="linear"/>
3591         <xsd:enumeration value="path"/>
3592     </xsd:restriction>
3593 </xsd:simpleType>
3594 <xsd:simpleType name="ST_HorizontalAlignment">
3595     <xsd:restriction base="xsd:string">
3596         <xsd:enumeration value="general"/>
3597         <xsd:enumeration value="left"/>
3598         <xsd:enumeration value="center"/>
3599         <xsd:enumeration value="right"/>
3600         <xsd:enumeration value="fill"/>
3601         <xsd:enumeration value="justify"/>
3602         <xsd:enumeration value="centerContinuous"/>
3603         <xsd:enumeration value="distributed"/>
3604     </xsd:restriction>
3605 </xsd:simpleType>
3606 <xsd:simpleType name="ST_VerticalAlignment">
3607     <xsd:restriction base="xsd:string">
3608         <xsd:enumeration value="top"/>
3609         <xsd:enumeration value="center"/>
3610         <xsd:enumeration value="bottom"/>
3611         <xsd:enumeration value="justify"/>
3612         <xsd:enumeration value="distributed"/>
3613     </xsd:restriction>
3614 </xsd:simpleType>
3615 <xsd:complexType name="CT_NumFmts">
3616     <xsd:sequence>
3617         <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="unbounded"/>
3618     </xsd:sequence>
3619     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3620 </xsd:complexType>
3621 <xsd:complexType name="CT_NumFmt">
3622     <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="required"/>
3623     <xsd:attribute name="formatCode" type="s:ST_Xstring" use="required"/>
3624 </xsd:complexType>
3625 <xsd:complexType name="CT_CellStyleXfs">
3626     <xsd:sequence>
3627         <xsd:element name="xf" type="CT_Xf" minOccurs="1" maxOccurs="unbounded"/>
3628     </xsd:sequence>
3629     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3630 </xsd:complexType>
3631 <xsd:complexType name="CT_CellXfs">
3632     <xsd:sequence>
3633         <xsd:element name="xf" type="CT_Xf" minOccurs="1" maxOccurs="unbounded"/>
3634     </xsd:sequence>
3635     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3636 </xsd:complexType>
3637 <xsd:complexType name="CT_Xf">
3638     <xsd:sequence>

```

```

3639     <xsd:element name="alignment" type="CT_CellAlignment" minOccurs="0" maxOccurs="1"/>
3640     <xsd:element name="protection" type="CT_CellProtection" minOccurs="0" maxOccurs="1"/>
3641     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3642   </xsd:sequence>
3643   <xsd:attribute name="numFmtId" type="ST_NumFmtId" use="optional"/>
3644   <xsd:attribute name="fontId" type="ST_FontId" use="optional"/>
3645   <xsd:attribute name="fillId" type="ST_FillId" use="optional"/>
3646   <xsd:attribute name="borderId" type="ST_BorderId" use="optional"/>
3647   <xsd:attribute name="xfId" type="ST_CellStyleXfId" use="optional"/>
3648   <xsd:attribute name="quotePrefix" type="xsd:boolean" use="optional" default="false"/>
3649   <xsd:attribute name="pivotButton" type="xsd:boolean" use="optional" default="false"/>
3650   <xsd:attribute name="applyNumberFormat" type="xsd:boolean" use="optional"/>
3651   <xsd:attribute name="applyFont" type="xsd:boolean" use="optional"/>
3652   <xsd:attribute name="applyFill" type="xsd:boolean" use="optional"/>
3653   <xsd:attribute name="applyBorder" type="xsd:boolean" use="optional"/>
3654   <xsd:attribute name="applyAlignment" type="xsd:boolean" use="optional"/>
3655   <xsd:attribute name="applyProtection" type="xsd:boolean" use="optional"/>
3656 </xsd:complexType>
3657 <xsd:complexType name="CT_CellStyles">
3658   <xsd:sequence>
3659     <xsd:element name="cellStyle" type="CT_CellStyle" minOccurs="1" maxOccurs="unbounded"/>
3660   </xsd:sequence>
3661   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3662 </xsd:complexType>
3663 <xsd:complexType name="CT_CellStyle">
3664   <xsd:sequence>
3665     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3666   </xsd:sequence>
3667   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
3668   <xsd:attribute name="xfId" type="ST_CellStyleXfId" use="required"/>
3669   <xsd:attribute name="builtinId" type="xsd:unsignedInt" use="optional"/>
3670   <xsd:attribute name="iLevel" type="xsd:unsignedInt" use="optional"/>
3671   <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
3672   <xsd:attribute name="customBuiltin" type="xsd:boolean" use="optional"/>
3673 </xsd:complexType>
3674 <xsd:complexType name="CT_Dxf">
3675   <xsd:sequence>
3676     <xsd:element name="dxf" type="CT_Dxf" minOccurs="0" maxOccurs="unbounded"/>
3677   </xsd:sequence>
3678   <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3679 </xsd:complexType>
3680 <xsd:complexType name="CT_Dxf">
3681   <xsd:sequence>
3682     <xsd:element name="font" type="CT_Font" minOccurs="0" maxOccurs="1"/>
3683     <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
3684     <xsd:element name="fill" type="CT_Fill" minOccurs="0" maxOccurs="1"/>
3685     <xsd:element name="alignment" type="CT_CellAlignment" minOccurs="0" maxOccurs="1"/>
3686     <xsd:element name="border" type="CT_Border" minOccurs="0" maxOccurs="1"/>
3687     <xsd:element name="protection" type="CT_CellProtection" minOccurs="0" maxOccurs="1"/>
3688     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3689   </xsd:sequence>
3690 </xsd:complexType>
3691 <xsd:simpleType name="ST_NumFmtId">

```

```

3692     <xsd:restriction base="xsd:unsignedInt"/>
3693 </xsd:simpleType>
3694 <xsd:simpleType name="ST_FontId">
3695     <xsd:restriction base="xsd:unsignedInt"/>
3696 </xsd:simpleType>
3697 <xsd:simpleType name="ST_FillId">
3698     <xsd:restriction base="xsd:unsignedInt"/>
3699 </xsd:simpleType>
3700 <xsd:simpleType name="ST_BorderId">
3701     <xsd:restriction base="xsd:unsignedInt"/>
3702 </xsd:simpleType>
3703 <xsd:simpleType name="ST_CellStyleXfId">
3704     <xsd:restriction base="xsd:unsignedInt"/>
3705 </xsd:simpleType>
3706 <xsd:simpleType name="ST_DxfId">
3707     <xsd:restriction base="xsd:unsignedInt"/>
3708 </xsd:simpleType>
3709 <xsd:complexType name="CT_Colors">
3710     <xsd:sequence>
3711         <xsd:element name="indexedColors" type="CT_IndexedColors" minOccurs="0" maxOccurs="1"/>
3712         <xsd:element name="mruColors" type="CT_MRUColors" minOccurs="0" maxOccurs="1"/>
3713     </xsd:sequence>
3714 </xsd:complexType>
3715 <xsd:complexType name="CT_IndexedColors">
3716     <xsd:sequence>
3717         <xsd:element name="rgbColor" type="CT_RgbColor" minOccurs="1" maxOccurs="unbounded"/>
3718     </xsd:sequence>
3719 </xsd:complexType>
3720 <xsd:complexType name="CT_MRUColors">
3721     <xsd:sequence>
3722         <xsd:element name="color" type="CT_Color" minOccurs="1" maxOccurs="unbounded"/>
3723     </xsd:sequence>
3724 </xsd:complexType>
3725 <xsd:complexType name="CT_RgbColor">
3726     <xsd:attribute name="rgb" type="ST_UnsignedIntHex" use="optional"/>
3727 </xsd:complexType>
3728 <xsd:complexType name="CT_TableStyles">
3729     <xsd:sequence>
3730         <xsd:element name="tableStyle" type="CT_TableStyle" minOccurs="0" maxOccurs="unbounded"/>
3731     </xsd:sequence>
3732     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
3733     <xsd:attribute name="defaultTableStyle" type="xsd:string" use="optional"/>
3734     <xsd:attribute name="defaultPivotStyle" type="xsd:string" use="optional"/>
3735 </xsd:complexType>
3736 <xsd:complexType name="CT_TableStyle">
3737     <xsd:sequence>
3738         <xsd:element name="tableStyleElement" type="CT_TableStyleElement" minOccurs="0"
3739             maxOccurs="unbounded"/>
3740     </xsd:sequence>
3741     <xsd:attribute name="name" type="xsd:string" use="required"/>
3742     <xsd:attribute name="pivot" type="xsd:boolean" use="optional" default="true"/>
3743     <xsd:attribute name="table" type="xsd:boolean" use="optional" default="true"/>
3744     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>

```

```

3745 </xsd:complexType>
3746 <xsd:complexType name="CT_TableStyleElement">
3747   <xsd:attribute name="type" type="ST_TableStyleType" use="required"/>
3748   <xsd:attribute name="size" type="xsd:unsignedInt" use="optional" default="1"/>
3749   <xsd:attribute name="dxId" type="ST_DxId" use="optional"/>
3750 </xsd:complexType>
3751 <xsd:simpleType name="ST_TableStyleType">
3752   <xsd:restriction base="xsd:string">
3753     <xsd:enumeration value="wholeTable"/>
3754     <xsd:enumeration value="headerRow"/>
3755     <xsd:enumeration value="totalRow"/>
3756     <xsd:enumeration value="firstColumn"/>
3757     <xsd:enumeration value="lastColumn"/>
3758     <xsd:enumeration value="firstRowStripe"/>
3759     <xsd:enumeration value="secondRowStripe"/>
3760     <xsd:enumeration value="firstColumnStripe"/>
3761     <xsd:enumeration value="secondColumnStripe"/>
3762     <xsd:enumeration value="firstHeaderCell"/>
3763     <xsd:enumeration value="lastHeaderCell"/>
3764     <xsd:enumeration value="firstTotalCell"/>
3765     <xsd:enumeration value="lastTotalCell"/>
3766     <xsd:enumeration value="firstSubtotalColumn"/>
3767     <xsd:enumeration value="secondSubtotalColumn"/>
3768     <xsd:enumeration value="thirdSubtotalColumn"/>
3769     <xsd:enumeration value="firstSubtotalRow"/>
3770     <xsd:enumeration value="secondSubtotalRow"/>
3771     <xsd:enumeration value="thirdSubtotalRow"/>
3772     <xsd:enumeration value="blankRow"/>
3773     <xsd:enumeration value="firstColumnSubheading"/>
3774     <xsd:enumeration value="secondColumnSubheading"/>
3775     <xsd:enumeration value="thirdColumnSubheading"/>
3776     <xsd:enumeration value="firstRowSubheading"/>
3777     <xsd:enumeration value="secondRowSubheading"/>
3778     <xsd:enumeration value="thirdRowSubheading"/>
3779     <xsd:enumeration value="pageFieldLabels"/>
3780     <xsd:enumeration value="pageFieldValues"/>
3781   </xsd:restriction>
3782 </xsd:simpleType>
3783 <xsd:complexType name="CT_BooleanProperty">
3784   <xsd:attribute name="val" type="xsd:boolean" use="optional" default="true"/>
3785 </xsd:complexType>
3786 <xsd:complexType name="CT_FontSize">
3787   <xsd:attribute name="val" type="xsd:double" use="required"/>
3788 </xsd:complexType>
3789 <xsd:complexType name="CT_IntProperty">
3790   <xsd:attribute name="val" type="xsd:int" use="required"/>
3791 </xsd:complexType>
3792 <xsd:complexType name="CT_FontName">
3793   <xsd:attribute name="val" type="s:ST_Xstring" use="required"/>
3794 </xsd:complexType>
3795 <xsd:complexType name="CT_VerticalAlignFontProperty">
3796   <xsd:attribute name="val" type="s:ST_VerticalAlignRun" use="required"/>
3797 </xsd:complexType>

```

```

3798 <xsd:complexType name="CT_FontScheme">
3799   <xsd:attribute name="val" type="ST_FontScheme" use="required"/>
3800 </xsd:complexType>
3801 <xsd:simpleType name="ST_FontScheme">
3802   <xsd:restriction base="xsd:string">
3803     <xsd:enumeration value="none"/>
3804     <xsd:enumeration value="major"/>
3805     <xsd:enumeration value="minor"/>
3806   </xsd:restriction>
3807 </xsd:simpleType>
3808 <xsd:complexType name="CT_UnderlineProperty">
3809   <xsd:attribute name="val" type="ST_UnderlineValues" use="optional" default="single"/>
3810 </xsd:complexType>
3811 <xsd:simpleType name="ST_UnderlineValues">
3812   <xsd:restriction base="xsd:string">
3813     <xsd:enumeration value="single"/>
3814     <xsd:enumeration value="double"/>
3815     <xsd:enumeration value="singleAccounting"/>
3816     <xsd:enumeration value="doubleAccounting"/>
3817     <xsd:enumeration value="none"/>
3818   </xsd:restriction>
3819 </xsd:simpleType>
3820 <xsd:complexType name="CT_Font">
3821   <xsd:choice maxOccurs="unbounded">
3822     <xsd:element name="name" type="CT_FontName" minOccurs="0" maxOccurs="1"/>
3823     <xsd:element name="charset" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
3824     <xsd:element name="family" type="CT_IntProperty" minOccurs="0" maxOccurs="1"/>
3825     <xsd:element name="b" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3826     <xsd:element name="i" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3827     <xsd:element name="strike" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3828     <xsd:element name="outline" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3829     <xsd:element name="shadow" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3830     <xsd:element name="condense" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3831     <xsd:element name="extend" type="CT_BooleanProperty" minOccurs="0" maxOccurs="1"/>
3832     <xsd:element name="color" type="CT_Color" minOccurs="0" maxOccurs="1"/>
3833     <xsd:element name="sz" type="CT_FontSize" minOccurs="0" maxOccurs="1"/>
3834     <xsd:element name="u" type="CT_UnderlineProperty" minOccurs="0" maxOccurs="1"/>
3835     <xsd:element name="vertAlign" type="CT_VerticalAlignFontProperty" minOccurs="0"
3836       maxOccurs="1"/>
3837     <xsd:element name="scheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
3838   </xsd:choice>
3839 </xsd:complexType>
3840 <xsd:attributeGroup name="AG_AutoFormat">
3841   <xsd:attribute name="autoFormatId" type="xsd:unsignedInt"/>
3842   <xsd:attribute name="applyNumberFormats" type="xsd:boolean"/>
3843   <xsd:attribute name="applyBorderFormats" type="xsd:boolean"/>
3844   <xsd:attribute name="applyFontFormats" type="xsd:boolean"/>
3845   <xsd:attribute name="applyPatternFormats" type="xsd:boolean"/>
3846   <xsd:attribute name="applyAlignmentFormats" type="xsd:boolean"/>
3847   <xsd:attribute name="applyWidthHeightFormats" type="xsd:boolean"/>
3848 </xsd:attributeGroup>
3849 <xsd:element name="externalLink" type="CT_ExternalLink"/>
3850 <xsd:complexType name="CT_ExternalLink">

```

```

3851     <xsd:choice>
3852         <xsd:element name="externalBook" type="CT_ExternalBook" minOccurs="0" maxOccurs="1"/>
3853         <xsd:element name="ddeLink" type="CT_DdeLink" minOccurs="0" maxOccurs="1"/>
3854         <xsd:element name="oleLink" type="CT_OleLink" minOccurs="0" maxOccurs="1"/>
3855         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
3856     </xsd:choice>
3857 </xsd:complexType>
3858 <xsd:complexType name="CT_ExternalBook">
3859     <xsd:sequence>
3860         <xsd:element name="sheetNames" type="CT_ExternalSheetNames" minOccurs="0" maxOccurs="1"/>
3861         <xsd:element name="definedNames" type="CT_ExternalDefinedNames" minOccurs="0"
3862             maxOccurs="1"/>
3863         <xsd:element name="sheetDataSet" type="CT_ExternalSheetDataSet" minOccurs="0"
3864             maxOccurs="1"/>
3865     </xsd:sequence>
3866     <xsd:attribute ref="r:id" use="required"/>
3867 </xsd:complexType>
3868 <xsd:complexType name="CT_ExternalSheetNames">
3869     <xsd:sequence>
3870         <xsd:element name="sheetName" minOccurs="1" maxOccurs="unbounded"
3871             type="CT_ExternalSheetName"/>
3872     </xsd:sequence>
3873 </xsd:complexType>
3874 <xsd:complexType name="CT_ExternalSheetName">
3875     <xsd:attribute name="val" type="s:ST_Xstring"/>
3876 </xsd:complexType>
3877 <xsd:complexType name="CT_ExternalDefinedNames">
3878     <xsd:sequence>
3879         <xsd:element name="definedName" type="CT_ExternalDefinedName" minOccurs="0"
3880             maxOccurs="unbounded"/>
3881     </xsd:sequence>
3882 </xsd:complexType>
3883 <xsd:complexType name="CT_ExternalDefinedName">
3884     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3885     <xsd:attribute name="refersTo" type="s:ST_Xstring" use="optional"/>
3886     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="optional"/>
3887 </xsd:complexType>
3888 <xsd:complexType name="CT_ExternalSheetDataSet">
3889     <xsd:sequence>
3890         <xsd:element name="sheetData" type="CT_ExternalSheetData" minOccurs="1"
3891             maxOccurs="unbounded"/>
3892     </xsd:sequence>
3893 </xsd:complexType>
3894 <xsd:complexType name="CT_ExternalSheetData">
3895     <xsd:sequence>
3896         <xsd:element name="row" type="CT_ExternalRow" minOccurs="0" maxOccurs="unbounded"/>
3897     </xsd:sequence>
3898     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
3899     <xsd:attribute name="refreshError" type="xsd:boolean" use="optional" default="false"/>
3900 </xsd:complexType>
3901 <xsd:complexType name="CT_ExternalRow">
3902     <xsd:sequence>
3903         <xsd:element name="cell" type="CT_ExternalCell" minOccurs="0" maxOccurs="unbounded"/>

```

```

3904     </xsd:sequence>
3905     <xsd:attribute name="r" type="xsd:unsignedInt" use="required"/>
3906 </xsd:complexType>
3907 <xsd:complexType name="CT_ExternalCell">
3908     <xsd:sequence>
3909         <xsd:element name="v" type="s:ST_Xstring" minOccurs="0" maxOccurs="1"/>
3910     </xsd:sequence>
3911     <xsd:attribute name="r" type="ST_CellRef" use="optional"/>
3912     <xsd:attribute name="t" type="ST_CellType" use="optional" default="n"/>
3913     <xsd:attribute name="vm" type="xsd:unsignedInt" use="optional" default="0"/>
3914 </xsd:complexType>
3915 <xsd:complexType name="CT_DdeLink">
3916     <xsd:sequence>
3917         <xsd:element name="ddeItems" type="CT_DdeItems" minOccurs="0" maxOccurs="1"/>
3918     </xsd:sequence>
3919     <xsd:attribute name="ddeService" type="s:ST_Xstring" use="required"/>
3920     <xsd:attribute name="ddeTopic" type="s:ST_Xstring" use="required"/>
3921 </xsd:complexType>
3922 <xsd:complexType name="CT_DdeItems">
3923     <xsd:sequence>
3924         <xsd:element name="ddeItem" type="CT_DdeItem" minOccurs="0" maxOccurs="unbounded"/>
3925     </xsd:sequence>
3926 </xsd:complexType>
3927 <xsd:complexType name="CT_DdeItem">
3928     <xsd:sequence>
3929         <xsd:element name="values" type="CT_DdeValues" minOccurs="0" maxOccurs="1"/>
3930     </xsd:sequence>
3931     <xsd:attribute name="name" type="s:ST_Xstring" default="0"/>
3932     <xsd:attribute name="ole" type="xsd:boolean" use="optional" default="false"/>
3933     <xsd:attribute name="advise" type="xsd:boolean" use="optional" default="false"/>
3934     <xsd:attribute name="preferPic" type="xsd:boolean" use="optional" default="false"/>
3935 </xsd:complexType>
3936 <xsd:complexType name="CT_DdeValues">
3937     <xsd:sequence>
3938         <xsd:element name="value" minOccurs="1" maxOccurs="unbounded" type="CT_DdeValue"/>
3939     </xsd:sequence>
3940     <xsd:attribute name="rows" type="xsd:unsignedInt" use="optional" default="1"/>
3941     <xsd:attribute name="cols" type="xsd:unsignedInt" use="optional" default="1"/>
3942 </xsd:complexType>
3943 <xsd:complexType name="CT_DdeValue">
3944     <xsd:sequence>
3945         <xsd:element name="val" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
3946     </xsd:sequence>
3947     <xsd:attribute name="t" type="ST_DdeValueType" use="optional" default="n"/>
3948 </xsd:complexType>
3949 <xsd:simpleType name="ST_DdeValueType">
3950     <xsd:restriction base="xsd:string">
3951         <xsd:enumeration value="nil"/>
3952         <xsd:enumeration value="b"/>
3953         <xsd:enumeration value="n"/>
3954         <xsd:enumeration value="e"/>
3955         <xsd:enumeration value="str"/>
3956     </xsd:restriction>

```

```

3957 </xsd:simpleType>
3958 <xsd:complexType name="CT_OleLink">
3959   <xsd:sequence>
3960     <xsd:element name="oleItems" type="CT_OleItems" minOccurs="0" maxOccurs="1"/>
3961   </xsd:sequence>
3962   <xsd:attribute ref="r:id" use="required"/>
3963   <xsd:attribute name="progId" type="s:ST_Xstring" use="required"/>
3964 </xsd:complexType>
3965 <xsd:complexType name="CT_OleItems">
3966   <xsd:sequence>
3967     <xsd:element name="oleItem" type="CT_OleItem" minOccurs="0" maxOccurs="unbounded"/>
3968   </xsd:sequence>
3969 </xsd:complexType>
3970 <xsd:complexType name="CT_OleItem">
3971   <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
3972   <xsd:attribute name="icon" type="xsd:boolean" use="optional" default="false"/>
3973   <xsd:attribute name="advise" type="xsd:boolean" use="optional" default="false"/>
3974   <xsd:attribute name="preferPic" type="xsd:boolean" use="optional" default="false"/>
3975 </xsd:complexType>
3976 <xsd:element name="table" type="CT_Table"/>
3977 <xsd:complexType name="CT_Table">
3978   <xsd:sequence>
3979     <xsd:element name="autoFilter" type="CT_AutoFilter" minOccurs="0" maxOccurs="1"/>
3980     <xsd:element name="sortState" type="CT_SortState" minOccurs="0" maxOccurs="1"/>
3981     <xsd:element name="tableColumns" type="CT_TableColumns" minOccurs="1" maxOccurs="1"/>
3982     <xsd:element name="tableStyleInfo" type="CT_TableStyleInfo" minOccurs="0" maxOccurs="1"/>
3983     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
3984   </xsd:sequence>
3985   <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
3986   <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
3987   <xsd:attribute name="displayName" type="s:ST_Xstring" use="required"/>
3988   <xsd:attribute name="comment" type="s:ST_Xstring" use="optional"/>
3989   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
3990   <xsd:attribute name="tableType" type="ST_TableType" use="optional" default="worksheet"/>
3991   <xsd:attribute name="headerRowCount" type="xsd:unsignedInt" use="optional" default="1"/>
3992   <xsd:attribute name="insertRow" type="xsd:boolean" use="optional" default="false"/>
3993   <xsd:attribute name="insertRowShift" type="xsd:boolean" use="optional" default="false"/>
3994   <xsd:attribute name="totalsRowCount" type="xsd:unsignedInt" use="optional" default="0"/>
3995   <xsd:attribute name="totalsRowShown" type="xsd:boolean" use="optional" default="true"/>
3996   <xsd:attribute name="published" type="xsd:boolean" use="optional" default="false"/>
3997   <xsd:attribute name="headerRowDxfId" type="ST_DxfId" use="optional"/>
3998   <xsd:attribute name="dataDxfId" type="ST_DxfId" use="optional"/>
3999   <xsd:attribute name="totalsRowDxfId" type="ST_DxfId" use="optional"/>
4000   <xsd:attribute name="headerRowBorderDxfId" type="ST_DxfId" use="optional"/>
4001   <xsd:attribute name="tableBorderDxfId" type="ST_DxfId" use="optional"/>
4002   <xsd:attribute name="totalsRowBorderDxfId" type="ST_DxfId" use="optional"/>
4003   <xsd:attribute name="headerRowCellStyle" type="s:ST_Xstring" use="optional"/>
4004   <xsd:attribute name="dataCellStyle" type="s:ST_Xstring" use="optional"/>
4005   <xsd:attribute name="totalsRowCellStyle" type="s:ST_Xstring" use="optional"/>
4006   <xsd:attribute name="connectionId" type="xsd:unsignedInt" use="optional"/>
4007 </xsd:complexType>
4008 <xsd:simpleType name="ST_TableType">
4009   <xsd:restriction base="xsd:string">

```



```

4010     <xsd:enumeration value="worksheet"/>
4011     <xsd:enumeration value="xml"/>
4012     <xsd:enumeration value="queryTable"/>
4013 </xsd:restriction>
4014 </xsd:simpleType>
4015 <xsd:complexType name="CT_TableStyleInfo">
4016     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
4017     <xsd:attribute name="showFirstColumn" type="xsd:boolean" use="optional"/>
4018     <xsd:attribute name="showLastColumn" type="xsd:boolean" use="optional"/>
4019     <xsd:attribute name="showRowStripes" type="xsd:boolean" use="optional"/>
4020     <xsd:attribute name="showColumnStripes" type="xsd:boolean" use="optional"/>
4021 </xsd:complexType>
4022 <xsd:complexType name="CT_TableColumns">
4023     <xsd:sequence>
4024         <xsd:element name="tableColumn" type="CT_TableColumn" minOccurs="1"
4025             maxOccurs="unbounded"/>
4026     </xsd:sequence>
4027     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
4028 </xsd:complexType>
4029 <xsd:complexType name="CT_TableColumn">
4030     <xsd:sequence>
4031         <xsd:element name="calculatedColumnFormula" type="CT_TableFormula" minOccurs="0"
4032             maxOccurs="1"/>
4033         <xsd:element name="totalsRowFormula" type="CT_TableFormula" minOccurs="0" maxOccurs="1"/>
4034         <xsd:element name="xmlColumnPr" type="CT_XmlColumnPr" minOccurs="0" maxOccurs="1"/>
4035         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4036     </xsd:sequence>
4037     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
4038     <xsd:attribute name="uniqueName" type="s:ST_Xstring" use="optional"/>
4039     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
4040     <xsd:attribute name="totalsRowFunction" type="ST_TotalsRowFunction" use="optional"
4041         default="none"/>
4042     <xsd:attribute name="totalsRowLabel" type="s:ST_Xstring" use="optional"/>
4043     <xsd:attribute name="queryTableFieldId" type="xsd:unsignedInt" use="optional"/>
4044     <xsd:attribute name="headerRowDxfId" type="ST_DxfId" use="optional"/>
4045     <xsd:attribute name="dataDxfId" type="ST_DxfId" use="optional"/>
4046     <xsd:attribute name="totalsRowDxfId" type="ST_DxfId" use="optional"/>
4047     <xsd:attribute name="headerRowCellStyle" type="s:ST_Xstring" use="optional"/>
4048     <xsd:attribute name="dataCellStyle" type="s:ST_Xstring" use="optional"/>
4049     <xsd:attribute name="totalsRowCellStyle" type="s:ST_Xstring" use="optional"/>
4050 </xsd:complexType>
4051 <xsd:complexType name="CT_TableFormula">
4052     <xsd:simpleContent>
4053         <xsd:extension base="ST_Formula">
4054             <xsd:attribute name="array" type="xsd:boolean" default="false"/>
4055         </xsd:extension>
4056     </xsd:simpleContent>
4057 </xsd:complexType>
4058 <xsd:simpleType name="ST_TotalsRowFunction">
4059     <xsd:restriction base="xsd:string">
4060         <xsd:enumeration value="none"/>
4061         <xsd:enumeration value="sum"/>
4062         <xsd:enumeration value="min"/>

```

```

4063         <xsd:enumeration value="max"/>
4064         <xsd:enumeration value="average"/>
4065         <xsd:enumeration value="count"/>
4066         <xsd:enumeration value="countNums"/>
4067         <xsd:enumeration value="stdDev"/>
4068         <xsd:enumeration value="var"/>
4069         <xsd:enumeration value="custom"/>
4070     </xsd:restriction>
4071 </xsd:simpleType>
4072 <xsd:complexType name="CT_XmlColumnPr">
4073     <xsd:sequence>
4074         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4075     </xsd:sequence>
4076     <xsd:attribute name="mapId" type="xsd:unsignedInt" use="required"/>
4077     <xsd:attribute name="xpath" type="s:ST_Xstring" use="required"/>
4078     <xsd:attribute name="denormalized" type="xsd:boolean" use="optional" default="false"/>
4079     <xsd:attribute name="xmlDataType" type="ST_XmlDataType" use="required"/>
4080 </xsd:complexType>
4081 <xsd:simpleType name="ST_XmlDataType">
4082     <xsd:restriction base="xsd:string"/>
4083 </xsd:simpleType>
4084 <xsd:element name="volTypes" type="CT_VolTypes"/>
4085 <xsd:complexType name="CT_VolTypes">
4086     <xsd:sequence>
4087         <xsd:element name="volType" type="CT_VolType" minOccurs="1" maxOccurs="unbounded"/>
4088         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
4089     </xsd:sequence>
4090 </xsd:complexType>
4091 <xsd:complexType name="CT_VolType">
4092     <xsd:sequence>
4093         <xsd:element name="main" type="CT_VolMain" minOccurs="1" maxOccurs="unbounded"/>
4094     </xsd:sequence>
4095     <xsd:attribute name="type" type="ST_VolDepType" use="required"/>
4096 </xsd:complexType>
4097 <xsd:complexType name="CT_VolMain">
4098     <xsd:sequence>
4099         <xsd:element name="tp" type="CT_VolTopic" minOccurs="1" maxOccurs="unbounded"/>
4100     </xsd:sequence>
4101     <xsd:attribute name="first" type="s:ST_Xstring" use="required"/>
4102 </xsd:complexType>
4103 <xsd:complexType name="CT_VolTopic">
4104     <xsd:sequence>
4105         <xsd:element name="v" type="s:ST_Xstring" minOccurs="1" maxOccurs="1"/>
4106         <xsd:element name="stp" type="s:ST_Xstring" minOccurs="0" maxOccurs="unbounded"/>
4107         <xsd:element name="tr" type="CT_VolTopicRef" minOccurs="1" maxOccurs="unbounded"/>
4108     </xsd:sequence>
4109     <xsd:attribute name="t" type="ST_VolValueType" use="optional" default="n"/>
4110 </xsd:complexType>
4111 <xsd:complexType name="CT_VolTopicRef">
4112     <xsd:attribute name="r" type="ST_CellRef" use="required"/>
4113     <xsd:attribute name="s" type="xsd:unsignedInt" use="required"/>
4114 </xsd:complexType>
4115 <xsd:simpleType name="ST_VolDepType">

```

```

4116     <xsd:restriction base="xsd:string">
4117         <xsd:enumeration value="realTimeData"/>
4118         <xsd:enumeration value="olapFunctions"/>
4119     </xsd:restriction>
4120 </xsd:simpleType>
4121 <xsd:simpleType name="ST_VolValueType">
4122     <xsd:restriction base="xsd:string">
4123         <xsd:enumeration value="b"/>
4124         <xsd:enumeration value="n"/>
4125         <xsd:enumeration value="e"/>
4126         <xsd:enumeration value="s"/>
4127     </xsd:restriction>
4128 </xsd:simpleType>
4129 <xsd:element name="workbook" type="CT_Workbook"/>
4130 <xsd:complexType name="CT_Workbook">
4131     <xsd:sequence>
4132         <xsd:element name="fileVersion" type="CT_FileVersion" minOccurs="0" maxOccurs="1"/>
4133         <xsd:element name="fileSharing" type="CT_FileSharing" minOccurs="0" maxOccurs="1"/>
4134         <xsd:element name="workbookPr" type="CT_WorkbookPr" minOccurs="0" maxOccurs="1"/>
4135         <xsd:element name="workbookProtection" type="CT_WorkbookProtection" minOccurs="0"
4136             maxOccurs="1"/>
4137         <xsd:element name="bookViews" type="CT_BookViews" minOccurs="0" maxOccurs="1"/>
4138         <xsd:element name="sheets" type="CT_Sheets" minOccurs="1" maxOccurs="1"/>
4139         <xsd:element name="functionGroups" type="CT_FunctionGroups" minOccurs="0" maxOccurs="1"/>
4140         <xsd:element name="externalReferences" type="CT_ExternalReferences" minOccurs="0"
4141             maxOccurs="1"/>
4142         <xsd:element name="definedNames" type="CT_DefinedNames" minOccurs="0" maxOccurs="1"/>
4143         <xsd:element name="calcPr" type="CT_CalcPr" minOccurs="0" maxOccurs="1"/>
4144         <xsd:element name="oleSize" type="CT_OleSize" minOccurs="0" maxOccurs="1"/>
4145         <xsd:element name="customWorkbookViews" type="CT_CustomWorkbookViews" minOccurs="0"
4146             maxOccurs="1"/>
4147         <xsd:element name="pivotCaches" type="CT_PivotCaches" minOccurs="0" maxOccurs="1"/>
4148         <xsd:element name="smartTagPr" type="CT_SmartTagPr" minOccurs="0" maxOccurs="1"/>
4149         <xsd:element name="smartTagTypes" type="CT_SmartTagTypes" minOccurs="0" maxOccurs="1"/>
4150         <xsd:element name="webPublishing" type="CT_WebPublishing" minOccurs="0" maxOccurs="1"/>
4151         <xsd:element name="fileRecoveryPr" type="CT_FileRecoveryPr" minOccurs="0"
4152             maxOccurs="unbounded"/>
4153         <xsd:element name="webPublishObjects" type="CT_WebPublishObjects" minOccurs="0"
4154             maxOccurs="1"/>
4155         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4156     </xsd:sequence>
4157     <xsd:attribute name="conformance" type="s:ST_ConformanceClass"/>
4158 </xsd:complexType>
4159 <xsd:complexType name="CT_FileVersion">
4160     <xsd:attribute name="appName" type="xsd:string" use="optional"/>
4161     <xsd:attribute name="lastEdited" type="xsd:string" use="optional"/>
4162     <xsd:attribute name="lowestEdited" type="xsd:string" use="optional"/>
4163     <xsd:attribute name="rupBuild" type="xsd:string" use="optional"/>
4164     <xsd:attribute name="codeName" type="s:ST_Guid" use="optional"/>
4165 </xsd:complexType>
4166 <xsd:complexType name="CT_BookViews">
4167     <xsd:sequence>
4168         <xsd:element name="workbookView" type="CT_BookView" minOccurs="1" maxOccurs="unbounded"/>

```

```

4169     </xsd:sequence>
4170 </xsd:complexType>
4171 <xsd:complexType name="CT_BookView">
4172     <xsd:sequence>
4173         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
4174     </xsd:sequence>
4175     <xsd:attribute name="visibility" type="ST_Visibility" use="optional" default="visible"/>
4176     <xsd:attribute name="minimized" type="xsd:boolean" use="optional" default="false"/>
4177     <xsd:attribute name="showHorizontalScroll" type="xsd:boolean" use="optional" default="true"/>
4178     <xsd:attribute name="showVerticalScroll" type="xsd:boolean" use="optional" default="true"/>
4179     <xsd:attribute name="showSheetTabs" type="xsd:boolean" use="optional" default="true"/>
4180     <xsd:attribute name="xWindow" type="xsd:int" use="optional"/>
4181     <xsd:attribute name="yWindow" type="xsd:int" use="optional"/>
4182     <xsd:attribute name="windowWidth" type="xsd:unsignedInt" use="optional"/>
4183     <xsd:attribute name="windowHeight" type="xsd:unsignedInt" use="optional"/>
4184     <xsd:attribute name="tabRatio" type="xsd:unsignedInt" use="optional" default="600"/>
4185     <xsd:attribute name="firstSheet" type="xsd:unsignedInt" use="optional" default="0"/>
4186     <xsd:attribute name="activeTab" type="xsd:unsignedInt" use="optional" default="0"/>
4187     <xsd:attribute name="autoFilterDateGrouping" type="xsd:boolean" use="optional"
4188         default="true"/>
4189 </xsd:complexType>
4190 <xsd:simpleType name="ST_Visibility">
4191     <xsd:restriction base="xsd:string">
4192         <xsd:enumeration value="visible"/>
4193         <xsd:enumeration value="hidden"/>
4194         <xsd:enumeration value="veryHidden"/>
4195     </xsd:restriction>
4196 </xsd:simpleType>
4197 <xsd:complexType name="CT_CustomWorkbookViews">
4198     <xsd:sequence>
4199         <xsd:element name="customWorkbookView" minOccurs="1" maxOccurs="unbounded"
4200             type="CT_CustomWorkbookView"/>
4201     </xsd:sequence>
4202 </xsd:complexType>
4203 <xsd:complexType name="CT_CustomWorkbookView">
4204     <xsd:sequence>
4205         <xsd:element name="extLst" minOccurs="0" type="CT_ExtensionList"/>
4206     </xsd:sequence>
4207     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
4208     <xsd:attribute name="guid" type="s:ST_Guid" use="required"/>
4209     <xsd:attribute name="autoUpdate" type="xsd:boolean" use="optional" default="false"/>
4210     <xsd:attribute name="mergeInterval" type="xsd:unsignedInt" use="optional"/>
4211     <xsd:attribute name="changesSavedWin" type="xsd:boolean" use="optional" default="false"/>
4212     <xsd:attribute name="onlySync" type="xsd:boolean" use="optional" default="false"/>
4213     <xsd:attribute name="personalView" type="xsd:boolean" use="optional" default="false"/>
4214     <xsd:attribute name="includePrintSettings" type="xsd:boolean" use="optional" default="true"/>
4215     <xsd:attribute name="includeHiddenRowCol" type="xsd:boolean" use="optional" default="true"/>
4216     <xsd:attribute name="maximized" type="xsd:boolean" use="optional" default="false"/>
4217     <xsd:attribute name="minimized" type="xsd:boolean" use="optional" default="false"/>
4218     <xsd:attribute name="showHorizontalScroll" type="xsd:boolean" use="optional" default="true"/>
4219     <xsd:attribute name="showVerticalScroll" type="xsd:boolean" use="optional" default="true"/>
4220     <xsd:attribute name="showSheetTabs" type="xsd:boolean" use="optional" default="true"/>
4221     <xsd:attribute name="xWindow" type="xsd:int" use="optional" default="0"/>

```

```

4222     <xsd:attribute name="yWindow" type="xsd:int" use="optional" default="0"/>
4223     <xsd:attribute name="windowWidth" type="xsd:unsignedInt" use="required"/>
4224     <xsd:attribute name="windowHeight" type="xsd:unsignedInt" use="required"/>
4225     <xsd:attribute name="tabRatio" type="xsd:unsignedInt" use="optional" default="600"/>
4226     <xsd:attribute name="activeSheetId" type="xsd:unsignedInt" use="required"/>
4227     <xsd:attribute name="showFormulaBar" type="xsd:boolean" use="optional" default="true"/>
4228     <xsd:attribute name="showStatusbar" type="xsd:boolean" use="optional" default="true"/>
4229     <xsd:attribute name="showComments" type="ST_Comments" use="optional" default="commIndicator"/>
4230     <xsd:attribute name="showObjects" type="ST_Objects" use="optional" default="all"/>
4231 </xsd:complexType>
4232 <xsd:simpleType name="ST_Comments">
4233     <xsd:restriction base="xsd:string">
4234         <xsd:enumeration value="commNone"/>
4235         <xsd:enumeration value="commIndicator"/>
4236         <xsd:enumeration value="commIndAndComment"/>
4237     </xsd:restriction>
4238 </xsd:simpleType>
4239 <xsd:simpleType name="ST_Objects">
4240     <xsd:restriction base="xsd:string">
4241         <xsd:enumeration value="all"/>
4242         <xsd:enumeration value="placeholders"/>
4243         <xsd:enumeration value="none"/>
4244     </xsd:restriction>
4245 </xsd:simpleType>
4246 <xsd:complexType name="CT_Sheets">
4247     <xsd:sequence>
4248         <xsd:element name="sheet" type="CT_Sheet" minOccurs="1" maxOccurs="unbounded"/>
4249     </xsd:sequence>
4250 </xsd:complexType>
4251 <xsd:complexType name="CT_Sheet">
4252     <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
4253     <xsd:attribute name="sheetId" type="xsd:unsignedInt" use="required"/>
4254     <xsd:attribute name="state" type="ST_SheetState" use="optional" default="visible"/>
4255     <xsd:attribute ref="r:id" use="required"/>
4256 </xsd:complexType>
4257 <xsd:simpleType name="ST_SheetState">
4258     <xsd:restriction base="xsd:string">
4259         <xsd:enumeration value="visible"/>
4260         <xsd:enumeration value="hidden"/>
4261         <xsd:enumeration value="veryHidden"/>
4262     </xsd:restriction>
4263 </xsd:simpleType>
4264 <xsd:complexType name="CT_WorkbookPr">
4265     <xsd:attribute name="date1904" type="xsd:boolean" use="optional" default="false"/>
4266     <xsd:attribute name="dateCompatibility" type="xsd:boolean" use="optional" default="true"/>
4267     <xsd:attribute name="showObjects" type="ST_Objects" use="optional" default="all"/>
4268     <xsd:attribute name="showBorderUnselectedTables" type="xsd:boolean" use="optional"
4269         default="true"/>
4270     <xsd:attribute name="filterPrivacy" type="xsd:boolean" use="optional" default="false"/>
4271     <xsd:attribute name="promptedSolutions" type="xsd:boolean" use="optional" default="false"/>
4272     <xsd:attribute name="showInkAnnotation" type="xsd:boolean" use="optional" default="true"/>
4273     <xsd:attribute name="backupFile" type="xsd:boolean" use="optional" default="false"/>

```

```

4274     <xsd:attribute name="saveExternalLinkValues" type="xsd:boolean" use="optional"
4275         default="true"/>
4276     <xsd:attribute name="updateLinks" type="ST_UpdateLinks" use="optional" default="userSet"/>
4277     <xsd:attribute name="codeName" type="xsd:string" use="optional"/>
4278     <xsd:attribute name="hidePivotFieldList" type="xsd:boolean" use="optional" default="false"/>
4279     <xsd:attribute name="showPivotChartFilter" type="xsd:boolean" default="false"/>
4280     <xsd:attribute name="allowRefreshQuery" type="xsd:boolean" use="optional" default="false"/>
4281     <xsd:attribute name="publishItems" type="xsd:boolean" use="optional" default="false"/>
4282     <xsd:attribute name="checkCompatibility" type="xsd:boolean" use="optional" default="false"/>
4283     <xsd:attribute name="autoCompressPictures" type="xsd:boolean" use="optional" default="true"/>
4284     <xsd:attribute name="refreshAllConnections" type="xsd:boolean" use="optional"
4285         default="false"/>
4286     <xsd:attribute name="defaultThemeVersion" type="xsd:unsignedInt" use="optional"/>
4287 </xsd:complexType>
4288 <xsd:simpleType name="ST_UpdateLinks">
4289     <xsd:restriction base="xsd:string">
4290         <xsd:enumeration value="userSet"/>
4291         <xsd:enumeration value="never"/>
4292         <xsd:enumeration value="always"/>
4293     </xsd:restriction>
4294 </xsd:simpleType>
4295 <xsd:complexType name="CT_SmartTagPr">
4296     <xsd:attribute name="embed" type="xsd:boolean" use="optional" default="false"/>
4297     <xsd:attribute name="show" type="ST_SmartTagShow" use="optional" default="all"/>
4298 </xsd:complexType>
4299 <xsd:simpleType name="ST_SmartTagShow">
4300     <xsd:restriction base="xsd:string">
4301         <xsd:enumeration value="all"/>
4302         <xsd:enumeration value="none"/>
4303         <xsd:enumeration value="noIndicator"/>
4304     </xsd:restriction>
4305 </xsd:simpleType>
4306 <xsd:complexType name="CT_SmartTagTypes">
4307     <xsd:sequence>
4308         <xsd:element name="smartTagType" type="CT_SmartTagType" minOccurs="0"
4309             maxOccurs="unbounded"/>
4310     </xsd:sequence>
4311 </xsd:complexType>
4312 <xsd:complexType name="CT_SmartTagType">
4313     <xsd:attribute name="namespaceUri" type="s:ST_Xstring" use="optional"/>
4314     <xsd:attribute name="name" type="s:ST_Xstring" use="optional"/>
4315     <xsd:attribute name="url" type="s:ST_Xstring" use="optional"/>
4316 </xsd:complexType>
4317 <xsd:complexType name="CT_FileRecoveryPr">
4318     <xsd:attribute name="autoRecover" type="xsd:boolean" use="optional" default="true"/>
4319     <xsd:attribute name="crashSave" type="xsd:boolean" use="optional" default="false"/>
4320     <xsd:attribute name="dataExtractLoad" type="xsd:boolean" use="optional" default="false"/>
4321     <xsd:attribute name="repairLoad" type="xsd:boolean" use="optional" default="false"/>
4322 </xsd:complexType>
4323 <xsd:complexType name="CT_CalcPr">
4324     <xsd:attribute name="calcId" type="xsd:unsignedInt"/>
4325     <xsd:attribute name="calcMode" type="ST_CalcMode" use="optional" default="auto"/>
4326     <xsd:attribute name="fullCalcOnLoad" type="xsd:boolean" use="optional" default="false"/>

```

```

4327     <xsd:attribute name="refMode" type="ST_RefMode" use="optional" default="A1"/>
4328     <xsd:attribute name="iterate" type="xsd:boolean" use="optional" default="false"/>
4329     <xsd:attribute name="iterateCount" type="xsd:unsignedInt" use="optional" default="100"/>
4330     <xsd:attribute name="iterateDelta" type="xsd:double" use="optional" default="0.001"/>
4331     <xsd:attribute name="fullPrecision" type="xsd:boolean" use="optional" default="true"/>
4332     <xsd:attribute name="calcCompleted" type="xsd:boolean" use="optional" default="true"/>
4333     <xsd:attribute name="calcOnSave" type="xsd:boolean" use="optional" default="true"/>
4334     <xsd:attribute name="concurrentCalc" type="xsd:boolean" use="optional" default="true"/>
4335     <xsd:attribute name="concurrentManualCount" type="xsd:unsignedInt" use="optional"/>
4336     <xsd:attribute name="forceFullCalc" type="xsd:boolean" use="optional"/>
4337 </xsd:complexType>
4338 <xsd:simpleType name="ST_CalcMode">
4339     <xsd:restriction base="xsd:string">
4340         <xsd:enumeration value="manual"/>
4341         <xsd:enumeration value="auto"/>
4342         <xsd:enumeration value="autoNoTable"/>
4343     </xsd:restriction>
4344 </xsd:simpleType>
4345 <xsd:simpleType name="ST_RefMode">
4346     <xsd:restriction base="xsd:string">
4347         <xsd:enumeration value="A1"/>
4348         <xsd:enumeration value="R1C1"/>
4349     </xsd:restriction>
4350 </xsd:simpleType>
4351 <xsd:complexType name="CT_DefinedNames">
4352     <xsd:sequence>
4353         <xsd:element name="definedName" type="CT_DefinedName" minOccurs="0"
4354             maxOccurs="unbounded"/>
4355     </xsd:sequence>
4356 </xsd:complexType>
4357 <xsd:complexType name="CT_DefinedName">
4358     <xsd:simpleContent>
4359         <xsd:extension base="ST_Formula">
4360             <xsd:attribute name="name" type="s:ST_Xstring" use="required"/>
4361             <xsd:attribute name="comment" type="s:ST_Xstring" use="optional"/>
4362             <xsd:attribute name="customMenu" type="s:ST_Xstring" use="optional"/>
4363             <xsd:attribute name="description" type="s:ST_Xstring" use="optional"/>
4364             <xsd:attribute name="help" type="s:ST_Xstring" use="optional"/>
4365             <xsd:attribute name="statusBar" type="s:ST_Xstring" use="optional"/>
4366             <xsd:attribute name="localSheetId" type="xsd:unsignedInt" use="optional"/>
4367             <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
4368             <xsd:attribute name="function" type="xsd:boolean" use="optional" default="false"/>
4369             <xsd:attribute name="vbProcedure" type="xsd:boolean" use="optional" default="false"/>
4370             <xsd:attribute name="xlm" type="xsd:boolean" use="optional" default="false"/>
4371             <xsd:attribute name="functionGroupId" type="xsd:unsignedInt" use="optional"/>
4372             <xsd:attribute name="shortcutKey" type="s:ST_Xstring" use="optional"/>
4373             <xsd:attribute name="publishToServer" type="xsd:boolean" use="optional"
4374                 default="false"/>
4375             <xsd:attribute name="workbookParameter" type="xsd:boolean" use="optional"
4376                 default="false"/>
4377         </xsd:extension>
4378     </xsd:simpleContent>
4379 </xsd:complexType>

```

```

4380 <xsd:complexType name="CT_ExternalReferences">
4381   <xsd:sequence>
4382     <xsd:element name="externalReference" type="CT_ExternalReference" minOccurs="1"
4383       maxOccurs="unbounded"/>
4384   </xsd:sequence>
4385 </xsd:complexType>
4386 <xsd:complexType name="CT_ExternalReference">
4387   <xsd:attribute ref="r:id" use="required"/>
4388 </xsd:complexType>
4389 <xsd:complexType name="CT_SheetBackgroundPicture">
4390   <xsd:attribute ref="r:id" use="required"/>
4391 </xsd:complexType>
4392 <xsd:complexType name="CT_PivotCaches">
4393   <xsd:sequence>
4394     <xsd:element name="pivotCache" type="CT_PivotCache" minOccurs="1" maxOccurs="unbounded"/>
4395   </xsd:sequence>
4396 </xsd:complexType>
4397 <xsd:complexType name="CT_PivotCache">
4398   <xsd:attribute name="cacheId" type="xsd:unsignedInt" use="required"/>
4399   <xsd:attribute ref="r:id" use="required"/>
4400 </xsd:complexType>
4401 <xsd:complexType name="CT_FileSharing">
4402   <xsd:attribute name="readOnlyRecommended" type="xsd:boolean" use="optional" default="false"/>
4403   <xsd:attribute name="userName" type="s:ST_Xstring"/>
4404   <xsd:attribute name="reservationPassword" type="ST_UnsignedShortHex"/>
4405   <xsd:attribute name="algorithmName" type="s:ST_Xstring" use="optional"/>
4406   <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
4407   <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
4408   <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
4409 </xsd:complexType>
4410 <xsd:complexType name="CT_OleSize">
4411   <xsd:attribute name="ref" type="ST_Ref" use="required"/>
4412 </xsd:complexType>
4413 <xsd:complexType name="CT_WorkbookProtection">
4414   <xsd:attribute name="workbookPassword" type="ST_UnsignedShortHex" use="optional"/>
4415   <xsd:attribute name="workbookPasswordCharacterSet" type="xsd:string" use="optional"/>
4416   <xsd:attribute name="revisionsPassword" type="ST_UnsignedShortHex" use="optional"/>
4417   <xsd:attribute name="revisionsPasswordCharacterSet" type="xsd:string" use="optional"/>
4418   <xsd:attribute name="lockStructure" type="xsd:boolean" use="optional" default="false"/>
4419   <xsd:attribute name="lockWindows" type="xsd:boolean" use="optional" default="false"/>
4420   <xsd:attribute name="lockRevision" type="xsd:boolean" use="optional" default="false"/>
4421   <xsd:attribute name="revisionsAlgorithmName" type="s:ST_Xstring" use="optional"/>
4422   <xsd:attribute name="revisionsHashValue" type="xsd:base64Binary" use="optional"/>
4423   <xsd:attribute name="revisionsSaltValue" type="xsd:base64Binary" use="optional"/>
4424   <xsd:attribute name="revisionsSpinCount" type="xsd:unsignedInt" use="optional"/>
4425   <xsd:attribute name="workbookAlgorithmName" type="s:ST_Xstring" use="optional"/>
4426   <xsd:attribute name="workbookHashValue" type="xsd:base64Binary" use="optional"/>
4427   <xsd:attribute name="workbookSaltValue" type="xsd:base64Binary" use="optional"/>
4428   <xsd:attribute name="workbookSpinCount" type="xsd:unsignedInt" use="optional"/>
4429 </xsd:complexType>
4430 <xsd:complexType name="CT_WebPublishing">
4431   <xsd:attribute name="css" type="xsd:boolean" use="optional" default="true"/>
4432   <xsd:attribute name="thicket" type="xsd:boolean" use="optional" default="true"/>

```



```

4433     <xsd:attribute name="longFileNames" type="xsd:boolean" use="optional" default="true"/>
4434     <xsd:attribute name="vml" type="xsd:boolean" use="optional" default="false"/>
4435     <xsd:attribute name="allowPng" type="xsd:boolean" use="optional" default="false"/>
4436     <xsd:attribute name="targetScreenSize" type="ST_TargetScreenSize" use="optional"
4437         default="800x600"/>
4438     <xsd:attribute name="dpi" type="xsd:unsignedInt" use="optional" default="96"/>
4439     <xsd:attribute name="codePage" type="xsd:unsignedInt" use="optional"/>
4440     <xsd:attribute name="characterSet" type="xsd:string" use="optional"/>
4441 </xsd:complexType>
4442 <xsd:simpleType name="ST_TargetScreenSize">
4443     <xsd:restriction base="xsd:string">
4444         <xsd:enumeration value="544x376"/>
4445         <xsd:enumeration value="640x480"/>
4446         <xsd:enumeration value="720x512"/>
4447         <xsd:enumeration value="800x600"/>
4448         <xsd:enumeration value="1024x768"/>
4449         <xsd:enumeration value="1152x882"/>
4450         <xsd:enumeration value="1152x900"/>
4451         <xsd:enumeration value="1280x1024"/>
4452         <xsd:enumeration value="1600x1200"/>
4453         <xsd:enumeration value="1800x1440"/>
4454         <xsd:enumeration value="1920x1200"/>
4455     </xsd:restriction>
4456 </xsd:simpleType>
4457 <xsd:complexType name="CT_FunctionGroups">
4458     <xsd:sequence maxOccurs="unbounded">
4459         <xsd:element name="functionGroup" type="CT_FunctionGroup" minOccurs="0"/>
4460     </xsd:sequence>
4461     <xsd:attribute name="builtInGroupCount" type="xsd:unsignedInt" default="16" use="optional"/>
4462 </xsd:complexType>
4463 <xsd:complexType name="CT_FunctionGroup">
4464     <xsd:attribute name="name" type="s:ST_Xstring"/>
4465 </xsd:complexType>
4466 <xsd:complexType name="CT_WebPublishObjects">
4467     <xsd:sequence>
4468         <xsd:element name="webPublishObject" type="CT_WebPublishObject" minOccurs="1"
4469             maxOccurs="unbounded"/>
4470     </xsd:sequence>
4471     <xsd:attribute name="count" type="xsd:unsignedInt" use="optional"/>
4472 </xsd:complexType>
4473 <xsd:complexType name="CT_WebPublishObject">
4474     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
4475     <xsd:attribute name="divId" type="s:ST_Xstring" use="required"/>
4476     <xsd:attribute name="sourceObject" type="s:ST_Xstring" use="optional"/>
4477     <xsd:attribute name="destinationFile" type="s:ST_Xstring" use="required"/>
4478     <xsd:attribute name="title" type="s:ST_Xstring" use="optional"/>
4479     <xsd:attribute name="autoRepublish" type="xsd:boolean" use="optional" default="false"/>
4480 </xsd:complexType>
4481 </xsd:schema>

```

A.3 PresentationML

```

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 id="s" 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_ShapeID" 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="attrNameLst" 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="fm1a" 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="fm1a"/>
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="fullScrn" type="xsd:boolean" use="optional" default="false"/>
622 </xsd:complexType>
623 <xsd:attributeGroup name="AG_TLBuild">
624     <xsd:attribute name="spid" type="a:ST_ShapeID" 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"/>
771   </xsd:sequence>
772   <xsd:attribute name="uri" type="xsd:token"/>
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="required"/>
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:choice minOccurs="1" maxOccurs="1">
852         <xsd:element name="embed" type="CT_OleObjectEmbed"/>
853         <xsd:element name="link" type="CT_OleObjectLink"/>
854         <xsd:element name="pic" type="CT_Picture"/>
855     </xsd:choice>
856     <xsd:attributeGroup ref="AG_Ole"/>
857     <xsd:attribute name="progId" type="xsd:string" use="optional"/>
858 </xsd:complexType>
859 <xsd:element name="oleObj" type="CT_OleObject"/>
860 <xsd:complexType name="CT_Control">
861     <xsd:sequence>
862         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
863         <xsd:element name="pic" type="CT_Picture" minOccurs="0" maxOccurs="1"/>
864     </xsd:sequence>
865     <xsd:attributeGroup ref="AG_Ole"/>
866 </xsd:complexType>
867 <xsd:complexType name="CT_ControlList">
868     <xsd:sequence>
869         <xsd:element name="control" type="CT_Control" minOccurs="0" maxOccurs="unbounded"/>
870     </xsd:sequence>
871 </xsd:complexType>
872 <xsd:simpleType name="ST_SlideId">
873     <xsd:restriction base="xsd:unsignedInt">
874         <xsd:minInclusive value="256"/>
875         <xsd:maxExclusive value="2147483648"/>
876     </xsd:restriction>
877 </xsd:simpleType>
878 <xsd:complexType name="CT_SlideIdListEntry">
879     <xsd:sequence>
880         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
881     </xsd:sequence>
882     <xsd:attribute name="id" type="ST_SlideId" use="required"/>
883     <xsd:attribute ref="r:id" use="required"/>
884 </xsd:complexType>
885 <xsd:complexType name="CT_SlideIdList">
886     <xsd:sequence>
887         <xsd:element name="sldId" type="CT_SlideIdListEntry" minOccurs="0" maxOccurs="unbounded"/>
888     </xsd:sequence>
889 </xsd:complexType>
890 <xsd:simpleType name="ST_SlideMasterId">
891     <xsd:restriction base="xsd:unsignedInt">
892         <xsd:minInclusive value="2147483648"/>
893     </xsd:restriction>
894 </xsd:simpleType>
895 <xsd:complexType name="CT_SlideMasterIdListEntry">
896     <xsd:sequence>
897         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
898     </xsd:sequence>
899     <xsd:attribute name="id" type="ST_SlideMasterId" use="optional"/>
900     <xsd:attribute ref="r:id" use="required"/>

```

```

901 </xsd:complexType>
902 <xsd:complexType name="CT_SlideMasterIdList">
903   <xsd:sequence>
904     <xsd:element name="sldMasterId" type="CT_SlideMasterIdListEntry" minOccurs="0"
905       maxOccurs="unbounded"/>
906   </xsd:sequence>
907 </xsd:complexType>
908 <xsd:complexType name="CT_NotesMasterIdListEntry">
909   <xsd:sequence>
910     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
911   </xsd:sequence>
912   <xsd:attribute ref="r:id" use="required"/>
913 </xsd:complexType>
914 <xsd:complexType name="CT_NotesMasterIdList">
915   <xsd:sequence>
916     <xsd:element name="notesMasterId" type="CT_NotesMasterIdListEntry" minOccurs="0"
917       maxOccurs="1"/>
918   </xsd:sequence>
919 </xsd:complexType>
920 <xsd:complexType name="CT_HandoutMasterIdListEntry">
921   <xsd:sequence>
922     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
923   </xsd:sequence>
924   <xsd:attribute ref="r:id" use="required"/>
925 </xsd:complexType>
926 <xsd:complexType name="CT_HandoutMasterIdList">
927   <xsd:sequence>
928     <xsd:element name="handoutMasterId" type="CT_HandoutMasterIdListEntry" minOccurs="0"
929       maxOccurs="1"/>
930   </xsd:sequence>
931 </xsd:complexType>
932 <xsd:complexType name="CT_EmbeddedFontDataId">
933   <xsd:attribute ref="r:id" use="required"/>
934 </xsd:complexType>
935 <xsd:complexType name="CT_EmbeddedFontListEntry">
936   <xsd:sequence>
937     <xsd:element name="font" type="a:CT_TextFont" minOccurs="1" maxOccurs="1"/>
938     <xsd:element name="regular" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
939     <xsd:element name="bold" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
940     <xsd:element name="italic" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
941     <xsd:element name="boldItalic" type="CT_EmbeddedFontDataId" minOccurs="0" maxOccurs="1"/>
942   </xsd:sequence>
943 </xsd:complexType>
944 <xsd:complexType name="CT_EmbeddedFontList">
945   <xsd:sequence>
946     <xsd:element name="embeddedFont" type="CT_EmbeddedFontListEntry" minOccurs="0"
947       maxOccurs="unbounded"/>
948   </xsd:sequence>
949 </xsd:complexType>
950 <xsd:complexType name="CT_SmartTags">
951   <xsd:attribute ref="r:id" use="required"/>
952 </xsd:complexType>
953 <xsd:complexType name="CT_CustomShow">

```

```

954     <xsd:sequence>
955         <xsd:element name="sldLst" type="CT_SlideRelationshipList" minOccurs="1" maxOccurs="1"/>
956         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
957     </xsd:sequence>
958     <xsd:attribute name="name" type="ST_Name" use="required"/>
959     <xsd:attribute name="id" type="xsd:unsignedInt" use="required"/>
960 </xsd:complexType>
961 <xsd:complexType name="CT_CustomShowList">
962     <xsd:sequence>
963         <xsd:element name="custShow" type="CT_CustomShow" minOccurs="0" maxOccurs="unbounded"/>
964     </xsd:sequence>
965 </xsd:complexType>
966 <xsd:simpleType name="ST_PhotoAlbumLayout">
967     <xsd:restriction base="xsd:token">
968         <xsd:enumeration value="fitToSlide"/>
969         <xsd:enumeration value="1pic"/>
970         <xsd:enumeration value="2pic"/>
971         <xsd:enumeration value="4pic"/>
972         <xsd:enumeration value="1picTitle"/>
973         <xsd:enumeration value="2picTitle"/>
974         <xsd:enumeration value="4picTitle"/>
975     </xsd:restriction>
976 </xsd:simpleType>
977 <xsd:simpleType name="ST_PhotoAlbumFrameShape">
978     <xsd:restriction base="xsd:token">
979         <xsd:enumeration value="frameStyle1"/>
980         <xsd:enumeration value="frameStyle2"/>
981         <xsd:enumeration value="frameStyle3"/>
982         <xsd:enumeration value="frameStyle4"/>
983         <xsd:enumeration value="frameStyle5"/>
984         <xsd:enumeration value="frameStyle6"/>
985         <xsd:enumeration value="frameStyle7"/>
986     </xsd:restriction>
987 </xsd:simpleType>
988 <xsd:complexType name="CT_PhotoAlbum">
989     <xsd:sequence>
990         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
991     </xsd:sequence>
992     <xsd:attribute name="bw" type="xsd:boolean" use="optional" default="false"/>
993     <xsd:attribute name="showCaptions" type="xsd:boolean" use="optional" default="false"/>
994     <xsd:attribute name="layout" type="ST_PhotoAlbumLayout" use="optional" default="fitToSlide"/>
995     <xsd:attribute name="frame" type="ST_PhotoAlbumFrameShape" use="optional"
996         default="frameStyle1"/>
997 </xsd:complexType>
998 <xsd:simpleType name="ST_SlideSizeCoordinate">
999     <xsd:restriction base="a:ST_PositiveCoordinate32">
1000         <xsd:minInclusive value="914400"/>
1001         <xsd:maxInclusive value="51206400"/>
1002     </xsd:restriction>
1003 </xsd:simpleType>
1004 <xsd:simpleType name="ST_SlideSizeType">
1005     <xsd:restriction base="xsd:token">
1006         <xsd:enumeration value="screen4x3"/>

```

```

1007     <xsd:enumeration value="letter"/>
1008     <xsd:enumeration value="A4"/>
1009     <xsd:enumeration value="35mm"/>
1010     <xsd:enumeration value="overhead"/>
1011     <xsd:enumeration value="banner"/>
1012     <xsd:enumeration value="custom"/>
1013     <xsd:enumeration value="ledger"/>
1014     <xsd:enumeration value="A3"/>
1015     <xsd:enumeration value="B4ISO"/>
1016     <xsd:enumeration value="B5ISO"/>
1017     <xsd:enumeration value="B4JIS"/>
1018     <xsd:enumeration value="B5JIS"/>
1019     <xsd:enumeration value="hagakiCard"/>
1020     <xsd:enumeration value="screen16x9"/>
1021     <xsd:enumeration value="screen16x10"/>
1022 </xsd:restriction>
1023 </xsd:simpleType>
1024 <xsd:complexType name="CT_SlideSize">
1025     <xsd:attribute name="cx" type="ST_SlideSizeCoordinate" use="required"/>
1026     <xsd:attribute name="cy" type="ST_SlideSizeCoordinate" use="required"/>
1027     <xsd:attribute name="type" type="ST_SlideSizeType" use="optional" default="custom"/>
1028 </xsd:complexType>
1029 <xsd:complexType name="CT_Kinsoku">
1030     <xsd:attribute name="lang" type="xsd:string" use="optional"/>
1031     <xsd:attribute name="invalStChars" type="xsd:string" use="required"/>
1032     <xsd:attribute name="invalEndChars" type="xsd:string" use="required"/>
1033 </xsd:complexType>
1034 <xsd:simpleType name="ST_BookmarkIdSeed">
1035     <xsd:restriction base="xsd:unsignedInt">
1036         <xsd:minInclusive value="1"/>
1037         <xsd:maxExclusive value="2147483648"/>
1038     </xsd:restriction>
1039 </xsd:simpleType>
1040 <xsd:complexType name="CT_ModifyVerifier">
1041     <xsd:attribute name="algorithmName" type="xsd:string" use="optional"/>
1042     <xsd:attribute name="hashValue" type="xsd:base64Binary" use="optional"/>
1043     <xsd:attribute name="saltValue" type="xsd:base64Binary" use="optional"/>
1044     <xsd:attribute name="spinValue" type="xsd:unsignedInt" use="optional"/>
1045     <xsd:attribute name="cryptProviderType" type="s:ST_CryptProv" use="optional"/>
1046     <xsd:attribute name="cryptAlgorithmClass" type="s:ST_AlgorithmClass" use="optional"/>
1047     <xsd:attribute name="cryptAlgorithmType" type="s:ST_AlgorithmType" use="optional"/>
1048     <xsd:attribute name="cryptAlgorithmSid" type="xsd:unsignedInt" use="optional"/>
1049     <xsd:attribute name="spinCount" type="xsd:unsignedInt" use="optional"/>
1050     <xsd:attribute name="saltData" type="xsd:base64Binary" use="optional"/>
1051     <xsd:attribute name="hashData" type="xsd:base64Binary" use="optional"/>
1052     <xsd:attribute name="cryptProvider" type="xsd:string" use="optional"/>
1053     <xsd:attribute name="algIdExt" type="xsd:unsignedInt" use="optional"/>
1054     <xsd:attribute name="algIdExtSource" type="xsd:string" use="optional"/>
1055     <xsd:attribute name="cryptProviderTypeExt" type="xsd:unsignedInt" use="optional"/>
1056     <xsd:attribute name="cryptProviderTypeExtSource" type="xsd:string" use="optional"/>
1057 </xsd:complexType>
1058 <xsd:complexType name="CT_Presentation">
1059     <xsd:sequence>

```

```

1060     <xsd:element name="sldMasterIdLst" type="CT_SlideMasterIdList" minOccurs="0"
1061       maxOccurs="1"/>
1062     <xsd:element name="notesMasterIdLst" type="CT_NotesMasterIdList" minOccurs="0"
1063       maxOccurs="1"/>
1064     <xsd:element name="handoutMasterIdLst" type="CT_HandoutMasterIdList" minOccurs="0"
1065       maxOccurs="1"/>
1066     <xsd:element name="sldIdLst" type="CT_SlideIdList" minOccurs="0" maxOccurs="1"/>
1067     <xsd:element name="sldSz" type="CT_SlideSize" minOccurs="0" maxOccurs="1"/>
1068     <xsd:element name="notesSz" type="a:CT_PositiveSize2D" minOccurs="1" maxOccurs="1"/>
1069     <xsd:element name="smartTags" type="CT_SmartTags" minOccurs="0" maxOccurs="1"/>
1070     <xsd:element name="embeddedFontLst" type="CT_EmbeddedFontList" minOccurs="0"
1071       maxOccurs="1"/>
1072     <xsd:element name="custShowLst" type="CT_CustomShowList" minOccurs="0" maxOccurs="1"/>
1073     <xsd:element name="photoAlbum" type="CT_PhotoAlbum" minOccurs="0" maxOccurs="1"/>
1074     <xsd:element name="custDataLst" type="CT_CustomerDataList" minOccurs="0" maxOccurs="1"/>
1075     <xsd:element name="kinsoku" type="CT_Kinsoku" minOccurs="0"/>
1076     <xsd:element name="defaultTextStyle" type="a:CT_TextListStyle" minOccurs="0"
1077       maxOccurs="1"/>
1078     <xsd:element name="modifyVerifier" type="CT_ModifyVerifier" minOccurs="0" maxOccurs="1"/>
1079     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1080   </xsd:sequence>
1081   <xsd:attribute name="serverZoom" type="a:ST_Percentage" use="optional" default="50%"/>
1082   <xsd:attribute name="firstSlideNum" type="xsd:int" use="optional" default="1"/>
1083   <xsd:attribute name="showSpecialPlsOnTitleSld" type="xsd:boolean" use="optional"
1084     default="true"/>
1085   <xsd:attribute name="rtl" type="xsd:boolean" use="optional" default="false"/>
1086   <xsd:attribute name="removePersonalInfoOnSave" type="xsd:boolean" use="optional"
1087     default="false"/>
1088   <xsd:attribute name="compatMode" type="xsd:boolean" use="optional" default="false"/>
1089   <xsd:attribute name="strictFirstAndLastChars" type="xsd:boolean" use="optional"
1090     default="true"/>
1091   <xsd:attribute name="embedTrueTypeFonts" type="xsd:boolean" use="optional" default="false"/>
1092   <xsd:attribute name="saveSubsetFonts" type="xsd:boolean" use="optional" default="false"/>
1093   <xsd:attribute name="autoCompressPictures" type="xsd:boolean" use="optional" default="true"/>
1094   <xsd:attribute name="bookmarkIdSeed" type="ST_BookmarkIdSeed" use="optional" default="1"/>
1095   <xsd:attribute name="conformance" type="s:ST_ConformanceClass"/>
1096 </xsd:complexType>
1097 <xsd:element name="presentation" type="CT_Presentation"/>
1098 <xsd:complexType name="CT_HtmlPublishProperties">
1099   <xsd:sequence>
1100     <xsd:group ref="EG_SlideListChoice" minOccurs="1" maxOccurs="1"/>
1101     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1102   </xsd:sequence>
1103   <xsd:attribute name="showSpeakerNotes" type="xsd:boolean" use="optional" default="true"/>
1104   <xsd:attribute name="target" type="xsd:string" use="optional"/>
1105   <xsd:attribute name="title" type="xsd:string" use="optional" default=""/>
1106   <xsd:attribute ref="r:id" use="required"/>
1107 </xsd:complexType>
1108 <xsd:simpleType name="ST_WebColorType">
1109   <xsd:restriction base="xsd:token">
1110     <xsd:enumeration value="none"/>
1111     <xsd:enumeration value="browser"/>
1112     <xsd:enumeration value="presentationText"/>

```

```

1113         <xsd:enumeration value="presentationAccent"/>
1114         <xsd:enumeration value="whiteTextOnBlack"/>
1115         <xsd:enumeration value="blackTextOnWhite"/>
1116     </xsd:restriction>
1117 </xsd:simpleType>
1118 <xsd:simpleType name="ST_WebScreenSize">
1119     <xsd:restriction base="xsd:token">
1120         <xsd:enumeration value="544x376"/>
1121         <xsd:enumeration value="640x480"/>
1122         <xsd:enumeration value="720x512"/>
1123         <xsd:enumeration value="800x600"/>
1124         <xsd:enumeration value="1024x768"/>
1125         <xsd:enumeration value="1152x882"/>
1126         <xsd:enumeration value="1152x900"/>
1127         <xsd:enumeration value="1280x1024"/>
1128         <xsd:enumeration value="1600x1200"/>
1129         <xsd:enumeration value="1800x1400"/>
1130         <xsd:enumeration value="1920x1200"/>
1131     </xsd:restriction>
1132 </xsd:simpleType>
1133 <xsd:simpleType name="ST_WebEncoding">
1134     <xsd:restriction base="xsd:string"/>
1135 </xsd:simpleType>
1136 <xsd:complexType name="CT_WebProperties">
1137     <xsd:sequence>
1138         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1139     </xsd:sequence>
1140     <xsd:attribute name="showAnimation" type="xsd:boolean" use="optional" default="false"/>
1141     <xsd:attribute name="resizeGraphics" type="xsd:boolean" use="optional" default="true"/>
1142     <xsd:attribute name="allowPng" type="xsd:boolean" use="optional" default="false"/>
1143     <xsd:attribute name="relyOnVml" type="xsd:boolean" use="optional" default="false"/>
1144     <xsd:attribute name="organizeInFolders" type="xsd:boolean" use="optional" default="true"/>
1145     <xsd:attribute name="useLongFileNames" type="xsd:boolean" use="optional" default="true"/>
1146     <xsd:attribute name="imgSz" type="ST_WebScreenSize" use="optional" default="800x600"/>
1147     <xsd:attribute name="encoding" type="ST_WebEncoding" use="optional" default=""/>
1148     <xsd:attribute name="clr" type="ST_WebColorType" use="optional" default="whiteTextOnBlack"/>
1149 </xsd:complexType>
1150 <xsd:simpleType name="ST_PrintWhat">
1151     <xsd:restriction base="xsd:token">
1152         <xsd:enumeration value="slides"/>
1153         <xsd:enumeration value="handouts1"/>
1154         <xsd:enumeration value="handouts2"/>
1155         <xsd:enumeration value="handouts3"/>
1156         <xsd:enumeration value="handouts4"/>
1157         <xsd:enumeration value="handouts6"/>
1158         <xsd:enumeration value="handouts9"/>
1159         <xsd:enumeration value="notes"/>
1160         <xsd:enumeration value="outline"/>
1161     </xsd:restriction>
1162 </xsd:simpleType>
1163 <xsd:simpleType name="ST_PrintColorMode">
1164     <xsd:restriction base="xsd:token">
1165         <xsd:enumeration value="bw"/>

```

```

1166         <xsd:enumeration value="gray"/>
1167         <xsd:enumeration value="clr"/>
1168     </xsd:restriction>
1169 </xsd:simpleType>
1170 <xsd:complexType name="CT_PrintProperties">
1171     <xsd:sequence>
1172         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1173     </xsd:sequence>
1174     <xsd:attribute name="prnWhat" type="ST_PrintWhat" use="optional" default="slides"/>
1175     <xsd:attribute name="clrMode" type="ST_PrintColorMode" use="optional" default="clr"/>
1176     <xsd:attribute name="hiddenSlides" type="xsd:boolean" use="optional" default="false"/>
1177     <xsd:attribute name="scaleToFitPaper" type="xsd:boolean" use="optional" default="false"/>
1178     <xsd:attribute name="frameSlides" type="xsd:boolean" use="optional" default="false"/>
1179 </xsd:complexType>
1180 <xsd:complexType name="CT_ShowInfoBrowse">
1181     <xsd:attribute name="showScrollbar" type="xsd:boolean" use="optional" default="true"/>
1182 </xsd:complexType>
1183 <xsd:complexType name="CT_ShowInfoKiosk">
1184     <xsd:attribute name="restart" type="xsd:unsignedInt" use="optional" default="300000"/>
1185 </xsd:complexType>
1186 <xsd:group name="EG_ShowType">
1187     <xsd:choice>
1188         <xsd:element name="present" type="CT_Empty"/>
1189         <xsd:element name="browse" type="CT_ShowInfoBrowse"/>
1190         <xsd:element name="kiosk" type="CT_ShowInfoKiosk"/>
1191     </xsd:choice>
1192 </xsd:group>
1193 <xsd:complexType name="CT_ShowProperties">
1194     <xsd:sequence minOccurs="0" maxOccurs="1">
1195         <xsd:group ref="EG_ShowType" minOccurs="0" maxOccurs="1"/>
1196         <xsd:group ref="EG_SlideListChoice" minOccurs="0" maxOccurs="1"/>
1197         <xsd:element name="penClr" type="a:CT_Color" minOccurs="0" maxOccurs="1"/>
1198         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1199     </xsd:sequence>
1200     <xsd:attribute name="loop" type="xsd:boolean" use="optional" default="false"/>
1201     <xsd:attribute name="showNarration" type="xsd:boolean" use="optional" default="false"/>
1202     <xsd:attribute name="showAnimation" type="xsd:boolean" use="optional" default="true"/>
1203     <xsd:attribute name="useTimings" type="xsd:boolean" use="optional" default="true"/>
1204 </xsd:complexType>
1205 <xsd:complexType name="CT_PresentationProperties">
1206     <xsd:sequence>
1207         <xsd:element name="htmlPubPr" type="CT_HtmlPublishProperties" minOccurs="0"
1208             maxOccurs="1"/>
1209         <xsd:element name="webPr" type="CT_WebProperties" minOccurs="0" maxOccurs="1"/>
1210         <xsd:element name="prnPr" type="CT_PrintProperties" minOccurs="0" maxOccurs="1"/>
1211         <xsd:element name="showPr" type="CT_ShowProperties" minOccurs="0" maxOccurs="1"/>
1212         <xsd:element name="clrMru" type="a:CT_ColorMRU" minOccurs="0" maxOccurs="1"/>
1213         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1214     </xsd:sequence>
1215 </xsd:complexType>
1216 <xsd:element name="presentationPr" type="CT_PresentationProperties"/>
1217 <xsd:complexType name="CT_HeaderFooter">
1218     <xsd:sequence>

```



```

1219     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1220 </xsd:sequence>
1221 <xsd:attribute name="sldNum" type="xsd:boolean" use="optional" default="true"/>
1222 <xsd:attribute name="hdr" type="xsd:boolean" use="optional" default="true"/>
1223 <xsd:attribute name="ftr" type="xsd:boolean" use="optional" default="true"/>
1224 <xsd:attribute name="dt" type="xsd:boolean" use="optional" default="true"/>
1225 </xsd:complexType>
1226 <xsd:simpleType name="ST_PlaceholderType">
1227     <xsd:restriction base="xsd:token">
1228         <xsd:enumeration value="title"/>
1229         <xsd:enumeration value="body"/>
1230         <xsd:enumeration value="ctrTitle"/>
1231         <xsd:enumeration value="subTitle"/>
1232         <xsd:enumeration value="dt"/>
1233         <xsd:enumeration value="sldNum"/>
1234         <xsd:enumeration value="ftr"/>
1235         <xsd:enumeration value="hdr"/>
1236         <xsd:enumeration value="obj"/>
1237         <xsd:enumeration value="chart"/>
1238         <xsd:enumeration value="tbl"/>
1239         <xsd:enumeration value="clipArt"/>
1240         <xsd:enumeration value="dgm"/>
1241         <xsd:enumeration value="media"/>
1242         <xsd:enumeration value="sldImg"/>
1243         <xsd:enumeration value="pic"/>
1244     </xsd:restriction>
1245 </xsd:simpleType>
1246 <xsd:simpleType name="ST_PlaceholderSize">
1247     <xsd:restriction base="xsd:token">
1248         <xsd:enumeration value="full"/>
1249         <xsd:enumeration value="half"/>
1250         <xsd:enumeration value="quarter"/>
1251     </xsd:restriction>
1252 </xsd:simpleType>
1253 <xsd:complexType name="CT_Placeholder">
1254     <xsd:sequence>
1255         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1256     </xsd:sequence>
1257     <xsd:attribute name="type" type="ST_PlaceholderType" use="optional" default="obj"/>
1258     <xsd:attribute name="orient" type="ST_Direction" use="optional" default="horz"/>
1259     <xsd:attribute name="sz" type="ST_PlaceholderSize" use="optional" default="full"/>
1260     <xsd:attribute name="idx" type="xsd:unsignedInt" use="optional" default="0"/>
1261     <xsd:attribute name="hasCustomPrompt" type="xsd:boolean" use="optional" default="false"/>
1262 </xsd:complexType>
1263 <xsd:complexType name="CT_ApplicationNonVisualDrawingProps">
1264     <xsd:sequence>
1265         <xsd:element name="ph" type="CT_Placeholder" minOccurs="0" maxOccurs="1"/>
1266         <xsd:group ref="a:EG_Media" minOccurs="0" maxOccurs="1"/>
1267         <xsd:element name="custDataLst" type="CT_CustomerDataList" minOccurs="0" maxOccurs="1"/>
1268         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1269     </xsd:sequence>
1270     <xsd:attribute name="isPhoto" type="xsd:boolean" use="optional" default="false"/>
1271     <xsd:attribute name="userDrawn" type="xsd:boolean" use="optional" default="false"/>

```

```

1272 </xsd:complexType>
1273 <xsd:complexType name="CT_ShapeNonVisual">
1274   <xsd:sequence>
1275     <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1276     <xsd:element name="cNvSpPr" type="a:CT_NonVisualDrawingShapeProps" minOccurs="1"
1277       maxOccurs="1"/>
1278     <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1279       maxOccurs="1"/>
1280   </xsd:sequence>
1281 </xsd:complexType>
1282 <xsd:complexType name="CT_Shape">
1283   <xsd:sequence>
1284     <xsd:element name="nvSpPr" type="CT_ShapeNonVisual" minOccurs="1" maxOccurs="1"/>
1285     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1286     <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1287     <xsd:element name="txBody" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1288     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1289   </xsd:sequence>
1290   <xsd:attribute name="useBgFill" type="xsd:boolean" use="optional" default="false"/>
1291 </xsd:complexType>
1292 <xsd:complexType name="CT_ConnectorNonVisual">
1293   <xsd:sequence>
1294     <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1295     <xsd:element name="cNvCxnSpPr" type="a:CT_NonVisualConnectorProperties" minOccurs="1"
1296       maxOccurs="1"/>
1297     <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1298       maxOccurs="1"/>
1299   </xsd:sequence>
1300 </xsd:complexType>
1301 <xsd:complexType name="CT_Connector">
1302   <xsd:sequence>
1303     <xsd:element name="nvCxnSpPr" type="CT_ConnectorNonVisual" minOccurs="1" maxOccurs="1"/>
1304     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1305     <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1306     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1307   </xsd:sequence>
1308 </xsd:complexType>
1309 <xsd:complexType name="CT_PictureNonVisual">
1310   <xsd:sequence>
1311     <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1312     <xsd:element name="cNvPicPr" type="a:CT_NonVisualPictureProperties" minOccurs="1"
1313       maxOccurs="1"/>
1314     <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1315       maxOccurs="1"/>
1316   </xsd:sequence>
1317 </xsd:complexType>
1318 <xsd:complexType name="CT_Picture">
1319   <xsd:sequence>
1320     <xsd:element name="nvPicPr" type="CT_PictureNonVisual" minOccurs="1" maxOccurs="1"/>
1321     <xsd:element name="blipFill" type="a:CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
1322     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1323     <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1324     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>

```

```

1325     </xsd:sequence>
1326 </xsd:complexType>
1327 <xsd:complexType name="CT_GraphicalObjectFrameNonVisual">
1328     <xsd:sequence>
1329         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1330         <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
1331             minOccurs="1" maxOccurs="1"/>
1332         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1333             maxOccurs="1"/>
1334     </xsd:sequence>
1335 </xsd:complexType>
1336 <xsd:complexType name="CT_GraphicalObjectFrame">
1337     <xsd:sequence>
1338         <xsd:element name="nvGraphicFramePr" type="CT_GraphicalObjectFrameNonVisual" minOccurs="1"
1339             maxOccurs="1"/>
1340         <xsd:element name="xfrm" type="a:CT_Transform2D" minOccurs="1" maxOccurs="1"/>
1341         <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
1342         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1343     </xsd:sequence>
1344 </xsd:complexType>
1345 <xsd:complexType name="CT_GroupShapeNonVisual">
1346     <xsd:sequence>
1347         <xsd:element name="cNvPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1348         <xsd:element name="cNvGrpSpPr" type="a:CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
1349             maxOccurs="1"/>
1350         <xsd:element name="nvPr" type="CT_ApplicationNonVisualDrawingProps" minOccurs="1"
1351             maxOccurs="1"/>
1352     </xsd:sequence>
1353 </xsd:complexType>
1354 <xsd:complexType name="CT_GroupShape">
1355     <xsd:sequence>
1356         <xsd:element name="nvGrpSpPr" type="CT_GroupShapeNonVisual" minOccurs="1" maxOccurs="1"/>
1357         <xsd:element name="grpSpPr" type="a:CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
1358         <xsd:choice minOccurs="0" maxOccurs="unbounded">
1359             <xsd:element name="sp" type="CT_Shape"/>
1360             <xsd:element name="grpSp" type="CT_GroupShape"/>
1361             <xsd:element name="graphicFrame" type="CT_GraphicalObjectFrame"/>
1362             <xsd:element name="cxnSp" type="CT_Connector"/>
1363             <xsd:element name="pic" type="CT_Picture"/>
1364             <xsd:element name="contentPart" type="CT_Rel"/>
1365         </xsd:choice>
1366         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1367     </xsd:sequence>
1368 </xsd:complexType>
1369 <xsd:complexType name="CT_Rel">
1370     <xsd:attribute ref="r:id" use="required"/>
1371 </xsd:complexType>
1372 <xsd:group name="EG_TopLevelSlide">
1373     <xsd:sequence>
1374         <xsd:element name="clrMap" type="a:CT_ColorMapping" minOccurs="1" maxOccurs="1"/>
1375     </xsd:sequence>
1376 </xsd:group>
1377 <xsd:group name="EG_ChildSlide">

```

```

1378     <xsd:sequence>
1379         <xsd:element name="clrMapOvr" type="a:CT_ColorMappingOverride" minOccurs="0"
1380             maxOccurs="1"/>
1381     </xsd:sequence>
1382 </xsd:group>
1383 <xsd:attributeGroup name="AG_ChildSlide">
1384     <xsd:attribute name="showMasterSp" type="xsd:boolean" use="optional" default="true"/>
1385     <xsd:attribute name="showMasterPhAnim" type="xsd:boolean" use="optional" default="true"/>
1386 </xsd:attributeGroup>
1387 <xsd:complexType name="CT_BackgroundProperties">
1388     <xsd:sequence>
1389         <xsd:group ref="a:EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1390         <xsd:group ref="a:EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
1391         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1392     </xsd:sequence>
1393     <xsd:attribute name="shadeToTitle" type="xsd:boolean" use="optional" default="false"/>
1394 </xsd:complexType>
1395 <xsd:group name="EG_Background">
1396     <xsd:choice>
1397         <xsd:element name="bgPr" type="CT_BackgroundProperties"/>
1398         <xsd:element name="bgRef" type="a:CT_StyleMatrixReference"/>
1399     </xsd:choice>
1400 </xsd:group>
1401 <xsd:complexType name="CT_Background">
1402     <xsd:sequence>
1403         <xsd:group ref="EG_Background"/>
1404     </xsd:sequence>
1405     <xsd:attribute name="bwMode" type="a:ST_BlackWhiteMode" use="optional" default="white"/>
1406 </xsd:complexType>
1407 <xsd:complexType name="CT_CommonSlideData">
1408     <xsd:sequence>
1409         <xsd:element name="bg" type="CT_Background" minOccurs="0" maxOccurs="1"/>
1410         <xsd:element name="spTree" type="CT_GroupShape" minOccurs="1" maxOccurs="1"/>
1411         <xsd:element name="custDataLst" type="CT_CustomerDataList" minOccurs="0" maxOccurs="1"/>
1412         <xsd:element name="controls" type="CT_ControlList" minOccurs="0" maxOccurs="1"/>
1413         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1414     </xsd:sequence>
1415     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
1416 </xsd:complexType>
1417 <xsd:complexType name="CT_Slide">
1418     <xsd:sequence minOccurs="1" maxOccurs="1">
1419         <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1420         <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1421         <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1422         <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1423         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1424     </xsd:sequence>
1425     <xsd:attributeGroup ref="AG_ChildSlide"/>
1426     <xsd:attribute name="show" type="xsd:boolean" use="optional" default="true"/>
1427 </xsd:complexType>
1428 <xsd:element name="sld" type="CT_Slide"/>
1429 <xsd:simpleType name="ST_SlideLayoutType">
1430     <xsd:restriction base="xsd:token">

```

```

1431     <xsd:enumeration value="title"/>
1432     <xsd:enumeration value="tx"/>
1433     <xsd:enumeration value="twoColTx"/>
1434     <xsd:enumeration value="tbl"/>
1435     <xsd:enumeration value="txAndChart"/>
1436     <xsd:enumeration value="chartAndTx"/>
1437     <xsd:enumeration value="dgm"/>
1438     <xsd:enumeration value="chart"/>
1439     <xsd:enumeration value="txAndClipArt"/>
1440     <xsd:enumeration value="clipArtAndTx"/>
1441     <xsd:enumeration value="titleOnly"/>
1442     <xsd:enumeration value="blank"/>
1443     <xsd:enumeration value="txAndObj"/>
1444     <xsd:enumeration value="objAndTx"/>
1445     <xsd:enumeration value="objOnly"/>
1446     <xsd:enumeration value="obj"/>
1447     <xsd:enumeration value="txAndMedia"/>
1448     <xsd:enumeration value="mediaAndTx"/>
1449     <xsd:enumeration value="objOverTx"/>
1450     <xsd:enumeration value="txOverObj"/>
1451     <xsd:enumeration value="txAndTwoObj"/>
1452     <xsd:enumeration value="twoObjAndTx"/>
1453     <xsd:enumeration value="twoObjOverTx"/>
1454     <xsd:enumeration value="fourObj"/>
1455     <xsd:enumeration value="vertTx"/>
1456     <xsd:enumeration value="clipArtAndVertTx"/>
1457     <xsd:enumeration value="vertTitleAndTx"/>
1458     <xsd:enumeration value="vertTitleAndTxOverChart"/>
1459     <xsd:enumeration value="twoObj"/>
1460     <xsd:enumeration value="objAndTwoObj"/>
1461     <xsd:enumeration value="twoObjAndObj"/>
1462     <xsd:enumeration value="cust"/>
1463     <xsd:enumeration value="secHead"/>
1464     <xsd:enumeration value="twoTxTwoObj"/>
1465     <xsd:enumeration value="objTx"/>
1466     <xsd:enumeration value="picTx"/>
1467 </xsd:restriction>
1468 </xsd:simpleType>
1469 <xsd:complexType name="CT_SlideLayout">
1470     <xsd:sequence minOccurs="1" maxOccurs="1">
1471         <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1472         <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1473         <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1474         <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1475         <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1476         <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1477     </xsd:sequence>
1478     <xsd:attributeGroup ref="AG_ChildSlide"/>
1479     <xsd:attribute name="matchingName" type="xsd:string" use="optional" default=""/>
1480     <xsd:attribute name="type" type="ST_SlideLayoutType" use="optional" default="cust"/>
1481     <xsd:attribute name="preserve" type="xsd:boolean" use="optional" default="false"/>
1482     <xsd:attribute name="userDrawn" type="xsd:boolean" use="optional" default="false"/>
1483 </xsd:complexType>

```

```

1484 <xsd:element name="sldLayout" type="CT_SlideLayout"/>
1485 <xsd:complexType name="CT_SlideMasterTextStyles">
1486   <xsd:sequence>
1487     <xsd:element name="titleStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1488     <xsd:element name="bodyStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1489     <xsd:element name="otherStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1490     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1491   </xsd:sequence>
1492 </xsd:complexType>
1493 <xsd:simpleType name="ST_SlideLayoutId">
1494   <xsd:restriction base="xsd:unsignedInt">
1495     <xsd:minInclusive value="2147483648"/>
1496   </xsd:restriction>
1497 </xsd:simpleType>
1498 <xsd:complexType name="CT_SlideLayoutIdListEntry">
1499   <xsd:sequence>
1500     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1501   </xsd:sequence>
1502   <xsd:attribute name="id" type="ST_SlideLayoutId" use="optional"/>
1503   <xsd:attribute ref="r:id" use="required"/>
1504 </xsd:complexType>
1505 <xsd:complexType name="CT_SlideLayoutIdList">
1506   <xsd:sequence>
1507     <xsd:element name="sldLayoutId" type="CT_SlideLayoutIdListEntry" minOccurs="0"
1508       maxOccurs="unbounded"/>
1509   </xsd:sequence>
1510 </xsd:complexType>
1511 <xsd:complexType name="CT_SlideMaster">
1512   <xsd:sequence minOccurs="1" maxOccurs="1">
1513     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1514     <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1515     <xsd:element name="sldLayoutIdLst" type="CT_SlideLayoutIdList" minOccurs="0"
1516       maxOccurs="1"/>
1517     <xsd:element name="transition" type="CT_SlideTransition" minOccurs="0" maxOccurs="1"/>
1518     <xsd:element name="timing" type="CT_SlideTiming" minOccurs="0" maxOccurs="1"/>
1519     <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1520     <xsd:element name="txStyles" type="CT_SlideMasterTextStyles" minOccurs="0" maxOccurs="1"/>
1521     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1522   </xsd:sequence>
1523   <xsd:attribute name="preserve" type="xsd:boolean" use="optional" default="false"/>
1524 </xsd:complexType>
1525 <xsd:element name="sldMaster" type="CT_SlideMaster"/>
1526 <xsd:complexType name="CT_HandoutMaster">
1527   <xsd:sequence>
1528     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1529     <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1530     <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1531     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1532   </xsd:sequence>
1533 </xsd:complexType>
1534 <xsd:element name="handoutMaster" type="CT_HandoutMaster"/>
1535 <xsd:complexType name="CT_NotesMaster">
1536   <xsd:sequence>

```

```

1537     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1538     <xsd:group ref="EG_TopLevelSlide" minOccurs="1" maxOccurs="1"/>
1539     <xsd:element name="hf" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1540     <xsd:element name="notesStyle" type="a:CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
1541     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1542   </xsd:sequence>
1543 </xsd:complexType>
1544 <xsd:element name="notesMaster" type="CT_NotesMaster"/>
1545 <xsd:complexType name="CT_NotesSlide">
1546   <xsd:sequence minOccurs="1" maxOccurs="1">
1547     <xsd:element name="cSld" type="CT_CommonSlideData" minOccurs="1" maxOccurs="1"/>
1548     <xsd:group ref="EG_ChildSlide" minOccurs="0" maxOccurs="1"/>
1549     <xsd:element name="extLst" type="CT_ExtensionListModify" minOccurs="0" maxOccurs="1"/>
1550   </xsd:sequence>
1551   <xsd:attributeGroup ref="AG_ChildSlide"/>
1552 </xsd:complexType>
1553 <xsd:element name="notes" type="CT_NotesSlide"/>
1554 <xsd:complexType name="CT_SlideSyncProperties">
1555   <xsd:sequence>
1556     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1557   </xsd:sequence>
1558   <xsd:attribute name="serverSldId" type="xsd:string" use="required"/>
1559   <xsd:attribute name="serverSldModifiedTime" type="xsd:dateTime" use="required"/>
1560   <xsd:attribute name="clientInsertedTime" type="xsd:dateTime" use="required"/>
1561 </xsd:complexType>
1562 <xsd:element name="sldSyncPr" type="CT_SlideSyncProperties"/>
1563 <xsd:complexType name="CT_StringTag">
1564   <xsd:attribute name="name" type="xsd:string"/>
1565   <xsd:attribute name="val" type="xsd:string"/>
1566 </xsd:complexType>
1567 <xsd:complexType name="CT_TagList">
1568   <xsd:sequence>
1569     <xsd:element name="tag" type="CT_StringTag" minOccurs="0" maxOccurs="unbounded"/>
1570   </xsd:sequence>
1571 </xsd:complexType>
1572 <xsd:element name="tagLst" type="CT_TagList"/>
1573 <xsd:simpleType name="ST_SplitterBarState">
1574   <xsd:restriction base="xsd:token">
1575     <xsd:enumeration value="minimized"/>
1576     <xsd:enumeration value="restored"/>
1577     <xsd:enumeration value="maximized"/>
1578   </xsd:restriction>
1579 </xsd:simpleType>
1580 <xsd:simpleType name="ST_ViewType">
1581   <xsd:restriction base="xsd:token">
1582     <xsd:enumeration value="sldView"/>
1583     <xsd:enumeration value="sldMasterView"/>
1584     <xsd:enumeration value="notesView"/>
1585     <xsd:enumeration value="handoutView"/>
1586     <xsd:enumeration value="notesMasterView"/>
1587     <xsd:enumeration value="outlineView"/>
1588     <xsd:enumeration value="sldSorterView"/>
1589     <xsd:enumeration value="sldThumbnailView"/>

```

```

1590     </xsd:restriction>
1591 </xsd:simpleType>
1592 <xsd:complexType name="CT_NormalViewPortion">
1593     <xsd:attribute name="sz" type="a:ST_PositiveFixedPercentage" use="required"/>
1594     <xsd:attribute name="autoAdjust" type="xsd:boolean" use="optional" default="true"/>
1595 </xsd:complexType>
1596 <xsd:complexType name="CT_NormalViewProperties">
1597     <xsd:sequence>
1598         <xsd:element name="restoredLeft" type="CT_NormalViewPortion" minOccurs="1" maxOccurs="1"/>
1599         <xsd:element name="restoredTop" type="CT_NormalViewPortion" minOccurs="1" maxOccurs="1"/>
1600         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1601     </xsd:sequence>
1602     <xsd:attribute name="showOutlineIcons" type="xsd:boolean" use="optional" default="true"/>
1603     <xsd:attribute name="snapVertSplitter" type="xsd:boolean" use="optional" default="false"/>
1604     <xsd:attribute name="vertBarState" type="ST_SplitterBarState" use="optional"
1605         default="restored"/>
1606     <xsd:attribute name="horzBarState" type="ST_SplitterBarState" use="optional"
1607         default="restored"/>
1608     <xsd:attribute name="preferSingleView" type="xsd:boolean" use="optional" default="false"/>
1609 </xsd:complexType>
1610 <xsd:complexType name="CT_CommonViewProperties">
1611     <xsd:sequence>
1612         <xsd:element name="scale" type="a:CT_Scale2D" minOccurs="1" maxOccurs="1"/>
1613         <xsd:element name="origin" type="a:CT_Point2D" minOccurs="1" maxOccurs="1"/>
1614     </xsd:sequence>
1615     <xsd:attribute name="varScale" type="xsd:boolean" use="optional" default="false"/>
1616 </xsd:complexType>
1617 <xsd:complexType name="CT_NotesTextViewProperties">
1618     <xsd:sequence minOccurs="1" maxOccurs="1">
1619         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1620         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1621     </xsd:sequence>
1622 </xsd:complexType>
1623 <xsd:complexType name="CT_OutlineViewSlideEntry">
1624     <xsd:attribute ref="r:id" use="required"/>
1625     <xsd:attribute name="collapse" type="xsd:boolean" use="optional" default="false"/>
1626 </xsd:complexType>
1627 <xsd:complexType name="CT_OutlineViewSlideList">
1628     <xsd:sequence>
1629         <xsd:element name="sld" type="CT_OutlineViewSlideEntry" minOccurs="0"
1630             maxOccurs="unbounded"/>
1631     </xsd:sequence>
1632 </xsd:complexType>
1633 <xsd:complexType name="CT_OutlineViewProperties">
1634     <xsd:sequence minOccurs="1" maxOccurs="1">
1635         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1636         <xsd:element name="sldLst" type="CT_OutlineViewSlideList" minOccurs="0" maxOccurs="1"/>
1637         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1638     </xsd:sequence>
1639 </xsd:complexType>
1640 <xsd:complexType name="CT_SlideSorterViewProperties">
1641     <xsd:sequence minOccurs="1" maxOccurs="1">
1642         <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>

```



```

1643     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1644   </xsd:sequence>
1645   <xsd:attribute name="showFormatting" type="xsd:boolean" use="optional" default="true"/>
1646 </xsd:complexType>
1647 <xsd:complexType name="CT_Guide">
1648   <xsd:attribute name="orient" type="ST_Direction" use="optional" default="vert"/>
1649   <xsd:attribute name="pos" type="a:ST_Coordinate32" use="optional" default="0"/>
1650 </xsd:complexType>
1651 <xsd:complexType name="CT_GuideList">
1652   <xsd:sequence minOccurs="0" maxOccurs="1">
1653     <xsd:element name="guide" type="CT_Guide" minOccurs="0" maxOccurs="unbounded"/>
1654   </xsd:sequence>
1655 </xsd:complexType>
1656 <xsd:complexType name="CT_CommonSlideViewProperties">
1657   <xsd:sequence>
1658     <xsd:element name="cViewPr" type="CT_CommonViewProperties" minOccurs="1" maxOccurs="1"/>
1659     <xsd:element name="guideLst" type="CT_GuideList" minOccurs="0" maxOccurs="1"/>
1660   </xsd:sequence>
1661   <xsd:attribute name="snapToGrid" type="xsd:boolean" use="optional" default="true"/>
1662   <xsd:attribute name="snapToObjects" type="xsd:boolean" use="optional" default="false"/>
1663   <xsd:attribute name="showGuides" type="xsd:boolean" use="optional" default="false"/>
1664 </xsd:complexType>
1665 <xsd:complexType name="CT_SlideViewProperties">
1666   <xsd:sequence>
1667     <xsd:element name="cSldViewPr" type="CT_CommonSlideViewProperties" minOccurs="1"
1668       maxOccurs="1"/>
1669     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1670   </xsd:sequence>
1671 </xsd:complexType>
1672 <xsd:complexType name="CT_NotesViewProperties">
1673   <xsd:sequence>
1674     <xsd:element name="cSldViewPr" type="CT_CommonSlideViewProperties" minOccurs="1"
1675       maxOccurs="1"/>
1676     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1677   </xsd:sequence>
1678 </xsd:complexType>
1679 <xsd:complexType name="CT_ViewProperties">
1680   <xsd:sequence minOccurs="0" maxOccurs="1">
1681     <xsd:element name="normalViewPr" type="CT_NormalViewProperties" minOccurs="0"
1682       maxOccurs="1"/>
1683     <xsd:element name="slideViewPr" type="CT_SlideViewProperties" minOccurs="0"
1684       maxOccurs="1"/>
1685     <xsd:element name="outlineViewPr" type="CT_OutlineViewProperties" minOccurs="0"
1686       maxOccurs="1"/>
1687     <xsd:element name="notesTextViewPr" type="CT_NotesTextViewProperties" minOccurs="0"
1688       maxOccurs="1"/>
1689     <xsd:element name="sorterViewPr" type="CT_SlideSorterViewProperties" minOccurs="0"
1690       maxOccurs="1"/>
1691     <xsd:element name="notesViewPr" type="CT_NotesViewProperties" minOccurs="0"
1692       maxOccurs="1"/>
1693     <xsd:element name="gridSpacing" type="a:CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
1694     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1695   </xsd:sequence>

```

```

1696     <xsd:attribute name="lastView" type="ST ViewType" use="optional" default="sldView"/>
1697     <xsd:attribute name="showComments" type="xsd:boolean" use="optional" default="true"/>
1698   </xsd:complexType>
1699   <xsd:element name="viewPr" type="CT ViewProperties"/>
1700 </xsd:schema>

```

A.4 DrawingML - Framework

A.4.1 DrawingML - Main

```

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/compatibility"
18     schemaLocation="dml-compatibility.xsd"/>
19   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/lockedCanvas"
20     schemaLocation="dml-lockedCanvas.xsd"/>
21   <xsd:complexType name="CT_AudioFile">
22     <xsd:sequence>
23       <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
24     </xsd:sequence>
25     <xsd:attribute ref="r:link" use="required"/>
26     <xsd:attribute name="contentType" type="xsd:string" use="optional"/>
27   </xsd:complexType>
28   <xsd:complexType name="CT_VideoFile">
29     <xsd:sequence>
30       <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
31     </xsd:sequence>
32     <xsd:attribute ref="r:link" use="required"/>
33     <xsd:attribute name="contentType" type="xsd:string" use="optional"/>
34   </xsd:complexType>
35   <xsd:complexType name="CT_QuickTimeFile">
36     <xsd:sequence>
37       <xsd:element name="extLst" type="CT OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
38     </xsd:sequence>
39     <xsd:attribute ref="r:link" use="required"/>
40   </xsd:complexType>
41   <xsd:complexType name="CT_AudioCDTime">
42     <xsd:attribute name="track" type="xsd:unsignedByte" use="required"/>
43     <xsd:attribute name="time" type="xsd:unsignedInt" use="optional" default="0"/>

```

```

44 </xsd:complexType>
45 <xsd:complexType name="CT_AudioCD">
46   <xsd:sequence>
47     <xsd:element name="st" type="CT_AudioCDTime" minOccurs="1" maxOccurs="1"/>
48     <xsd:element name="end" type="CT_AudioCDTime" minOccurs="1" maxOccurs="1"/>
49     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
50   </xsd:sequence>
51 </xsd:complexType>
52 <xsd:group name="EG_Media">
53   <xsd:choice>
54     <xsd:element name="audioCd" type="CT_AudioCD"/>
55     <xsd:element name="wavAudioFile" type="CT_EmbeddedWAVAudioFile"/>
56     <xsd:element name="audioFile" type="CT_AudioFile"/>
57     <xsd:element name="videoFile" type="CT_VideoFile"/>
58     <xsd:element name="quickTimeFile" type="CT_QuickTimeFile"/>
59   </xsd:choice>
60 </xsd:group>
61 <xsd:element name="videoFile" type="CT_VideoFile"/>
62 <xsd:simpleType name="ST_StyleMatrixColumnIndex">
63   <xsd:restriction base="xsd:unsignedInt"/>
64 </xsd:simpleType>
65 <xsd:simpleType name="ST_FontCollectionIndex">
66   <xsd:restriction base="xsd:token">
67     <xsd:enumeration value="major"/>
68     <xsd:enumeration value="minor"/>
69     <xsd:enumeration value="none"/>
70   </xsd:restriction>
71 </xsd:simpleType>
72 <xsd:simpleType name="ST_ColorSchemeIndex">
73   <xsd:restriction base="xsd:token">
74     <xsd:enumeration value="dk1"/>
75     <xsd:enumeration value="lt1"/>
76     <xsd:enumeration value="dk2"/>
77     <xsd:enumeration value="lt2"/>
78     <xsd:enumeration value="accent1"/>
79     <xsd:enumeration value="accent2"/>
80     <xsd:enumeration value="accent3"/>
81     <xsd:enumeration value="accent4"/>
82     <xsd:enumeration value="accent5"/>
83     <xsd:enumeration value="accent6"/>
84     <xsd:enumeration value="hlink"/>
85     <xsd:enumeration value="folHlink"/>
86   </xsd:restriction>
87 </xsd:simpleType>
88 <xsd:complexType name="CT_ColorScheme">
89   <xsd:sequence>
90     <xsd:element name="dk1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
91     <xsd:element name="lt1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
92     <xsd:element name="dk2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
93     <xsd:element name="lt2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
94     <xsd:element name="accent1" type="CT_Color" minOccurs="1" maxOccurs="1"/>
95     <xsd:element name="accent2" type="CT_Color" minOccurs="1" maxOccurs="1"/>
96     <xsd:element name="accent3" type="CT_Color" minOccurs="1" maxOccurs="1"/>

```

```

97      <xsd:element name="accent4" type="CT_Color" minOccurs="1" maxOccurs="1"/>
98      <xsd:element name="accent5" type="CT_Color" minOccurs="1" maxOccurs="1"/>
99      <xsd:element name="accent6" type="CT_Color" minOccurs="1" maxOccurs="1"/>
100     <xsd:element name="hlink" type="CT_Color" minOccurs="1" maxOccurs="1"/>
101     <xsd:element name="folHlink" type="CT_Color" minOccurs="1" maxOccurs="1"/>
102     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
103   </xsd:sequence>
104   <xsd:attribute name="name" type="xsd:string" use="required"/>
105 </xsd:complexType>
106 <xsd:complexType name="CT_CustomColor">
107   <xsd:sequence>
108     <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
109   </xsd:sequence>
110   <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
111 </xsd:complexType>
112 <xsd:complexType name="CT_SupplementalFont">
113   <xsd:attribute name="script" type="xsd:string" use="required"/>
114   <xsd:attribute name="typeface" type="ST_TextTypeface" use="required"/>
115 </xsd:complexType>
116 <xsd:complexType name="CT_CustomColorList">
117   <xsd:sequence>
118     <xsd:element name="custClr" type="CT_CustomColor" minOccurs="0" maxOccurs="unbounded"/>
119   </xsd:sequence>
120 </xsd:complexType>
121 <xsd:complexType name="CT_FontCollection">
122   <xsd:sequence>
123     <xsd:element name="latin" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
124     <xsd:element name="ea" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
125     <xsd:element name="cs" type="CT_TextFont" minOccurs="1" maxOccurs="1"/>
126     <xsd:element name="font" type="CT_SupplementalFont" minOccurs="0" maxOccurs="unbounded"/>
127     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
128   </xsd:sequence>
129 </xsd:complexType>
130 <xsd:complexType name="CT_EffectStyleItem">
131   <xsd:sequence>
132     <xsd:group ref="EG_EffectProperties" minOccurs="1" maxOccurs="1"/>
133     <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
134     <xsd:element name="sp3d" type="CT_Shape3D" minOccurs="0" maxOccurs="1"/>
135   </xsd:sequence>
136 </xsd:complexType>
137 <xsd:complexType name="CT_FontScheme">
138   <xsd:sequence>
139     <xsd:element name="majorFont" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
140     <xsd:element name="minorFont" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
141     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
142   </xsd:sequence>
143   <xsd:attribute name="name" type="xsd:string" use="required"/>
144 </xsd:complexType>
145 <xsd:complexType name="CT_FillStyleList">
146   <xsd:sequence>
147     <xsd:group ref="EG_FillProperties" minOccurs="3" maxOccurs="unbounded"/>
148   </xsd:sequence>
149 </xsd:complexType>

```

```

150 <xsd:complexType name="CT_LineStyleList">
151   <xsd:sequence>
152     <xsd:element name="ln" type="CT_LineProperties" minOccurs="3" maxOccurs="unbounded"/>
153   </xsd:sequence>
154 </xsd:complexType>
155 <xsd:complexType name="CT_EffectStyleList">
156   <xsd:sequence>
157     <xsd:element name="effectStyle" type="CT_EffectStyleItem" minOccurs="3"
158       maxOccurs="unbounded"/>
159   </xsd:sequence>
160 </xsd:complexType>
161 <xsd:complexType name="CT_BackgroundFillStyleList">
162   <xsd:sequence>
163     <xsd:group ref="EG_FillProperties" minOccurs="3" maxOccurs="unbounded"/>
164   </xsd:sequence>
165 </xsd:complexType>
166 <xsd:complexType name="CT_StyleMatrix">
167   <xsd:sequence>
168     <xsd:element name="fillStyleLst" type="CT_FillStyleList" minOccurs="1" maxOccurs="1"/>
169     <xsd:element name="lnStyleLst" type="CT_LineStyleList" minOccurs="1" maxOccurs="1"/>
170     <xsd:element name="effectStyleLst" type="CT_EffectStyleList" minOccurs="1" maxOccurs="1"/>
171     <xsd:element name="bgFillStyleLst" type="CT_BackgroundFillStyleList" minOccurs="1"
172       maxOccurs="1"/>
173   </xsd:sequence>
174   <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
175 </xsd:complexType>
176 <xsd:complexType name="CT_BaseStyles">
177   <xsd:sequence>
178     <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="1" maxOccurs="1"/>
179     <xsd:element name="fontScheme" type="CT_FontScheme" minOccurs="1" maxOccurs="1"/>
180     <xsd:element name="fmtScheme" type="CT_StyleMatrix" minOccurs="1" maxOccurs="1"/>
181     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
182   </xsd:sequence>
183 </xsd:complexType>
184 <xsd:complexType name="CT_OfficeArtExtension">
185   <xsd:sequence>
186     <xsd:any processContents="lax"/>
187   </xsd:sequence>
188   <xsd:attribute name="uri" type="xsd:token"/>
189 </xsd:complexType>
190 <xsd:simpleType name="ST_Coordinate">
191   <xsd:union memberTypes="ST_CoordinateUnqualified s:ST_UniversalMeasure"/>
192 </xsd:simpleType>
193 <xsd:simpleType name="ST_CoordinateUnqualified">
194   <xsd:restriction base="xsd:long">
195     <xsd:minInclusive value="-27273042329600"/>
196     <xsd:maxInclusive value="27273042316900"/>
197   </xsd:restriction>
198 </xsd:simpleType>
199 <xsd:simpleType name="ST_Coordinate32">
200   <xsd:union memberTypes="ST_Coordinate32Unqualified s:ST_UniversalMeasure"/>
201 </xsd:simpleType>
202 <xsd:simpleType name="ST_Coordinate32Unqualified">

```

```

203     <xsd:restriction base="xsd:int"/>
204 </xsd:simpleType>
205 <xsd:simpleType name="ST_PositiveCoordinate">
206     <xsd:restriction base="xsd:long">
207         <xsd:minInclusive value="0"/>
208         <xsd:maxInclusive value="27273042316900"/>
209     </xsd:restriction>
210 </xsd:simpleType>
211 <xsd:simpleType name="ST_PositiveCoordinate32">
212     <xsd:restriction base="ST_Coordinate32Unqualified">
213         <xsd:minInclusive value="0"/>
214     </xsd:restriction>
215 </xsd:simpleType>
216 <xsd:simpleType name="ST_Angle">
217     <xsd:restriction base="xsd:int"/>
218 </xsd:simpleType>
219 <xsd:complexType name="CT_Angle">
220     <xsd:attribute name="val" type="ST_Angle" use="required"/>
221 </xsd:complexType>
222 <xsd:simpleType name="ST_FixedAngle">
223     <xsd:restriction base="ST_Angle">
224         <xsd:minExclusive value="-5400000"/>
225         <xsd:maxExclusive value="5400000"/>
226     </xsd:restriction>
227 </xsd:simpleType>
228 <xsd:simpleType name="ST_PositiveFixedAngle">
229     <xsd:restriction base="ST_Angle">
230         <xsd:minInclusive value="0"/>
231         <xsd:maxExclusive value="21600000"/>
232     </xsd:restriction>
233 </xsd:simpleType>
234 <xsd:complexType name="CT_PositiveFixedAngle">
235     <xsd:attribute name="val" type="ST_PositiveFixedAngle" use="required"/>
236 </xsd:complexType>
237 <xsd:simpleType name="ST_Percentage">
238     <xsd:union memberTypes="ST_PercentageDecimal s:ST_Percentage"/>
239 </xsd:simpleType>
240 <xsd:simpleType name="ST_PercentageDecimal">
241     <xsd:restriction base="xsd:int"/>
242 </xsd:simpleType>
243 <xsd:complexType name="CT_Percentage">
244     <xsd:attribute name="val" type="ST_Percentage" use="required"/>
245 </xsd:complexType>
246 <xsd:simpleType name="ST_PositivePercentage">
247     <xsd:union memberTypes="ST_PositivePercentageDecimal s:ST_PositivePercentage"/>
248 </xsd:simpleType>
249 <xsd:simpleType name="ST_PositivePercentageDecimal">
250     <xsd:restriction base="ST_PercentageDecimal">
251         <xsd:minInclusive value="0"/>
252     </xsd:restriction>
253 </xsd:simpleType>
254 <xsd:complexType name="CT_PositivePercentage">
255     <xsd:attribute name="val" type="ST_PositivePercentage" use="required"/>

```

```

256 </xsd:complexType>
257 <xsd:simpleType name="ST_FixedPercentage">
258   <xsd:union memberTypes="ST_FixedPercentageDecimal s:ST_FixedPercentage"/>
259 </xsd:simpleType>
260 <xsd:simpleType name="ST_FixedPercentageDecimal">
261   <xsd:restriction base="ST_PercentageDecimal">
262     <xsd:minInclusive value="-100000"/>
263     <xsd:maxInclusive value="100000"/>
264   </xsd:restriction>
265 </xsd:simpleType>
266 <xsd:complexType name="CT_FixedPercentage">
267   <xsd:attribute name="val" type="ST_FixedPercentage" use="required"/>
268 </xsd:complexType>
269 <xsd:simpleType name="ST_PositiveFixedPercentage">
270   <xsd:union memberTypes="ST_PositiveFixedPercentageDecimal s:ST_PositiveFixedPercentage"/>
271 </xsd:simpleType>
272 <xsd:simpleType name="ST_PositiveFixedPercentageDecimal">
273   <xsd:restriction base="ST_PercentageDecimal">
274     <xsd:minInclusive value="0"/>
275     <xsd:maxInclusive value="100000"/>
276   </xsd:restriction>
277 </xsd:simpleType>
278 <xsd:complexType name="CT_PositiveFixedPercentage">
279   <xsd:attribute name="val" type="ST_PositiveFixedPercentage" use="required"/>
280 </xsd:complexType>
281 <xsd:complexType name="CT_Ratio">
282   <xsd:attribute name="n" type="xsd:long" use="required"/>
283   <xsd:attribute name="d" type="xsd:long" use="required"/>
284 </xsd:complexType>
285 <xsd:complexType name="CT_Point2D">
286   <xsd:attribute name="x" type="ST_Coordinate" use="required"/>
287   <xsd:attribute name="y" type="ST_Coordinate" use="required"/>
288 </xsd:complexType>
289 <xsd:complexType name="CT_PositiveSize2D">
290   <xsd:attribute name="cx" type="ST_PositiveCoordinate" use="required"/>
291   <xsd:attribute name="cy" type="ST_PositiveCoordinate" use="required"/>
292 </xsd:complexType>
293 <xsd:complexType name="CT_ComplementTransform"/>
294 <xsd:complexType name="CT_InverseTransform"/>
295 <xsd:complexType name="CT_GrayscaleTransform"/>
296 <xsd:complexType name="CT_GammaTransform"/>
297 <xsd:complexType name="CT_InverseGammaTransform"/>
298 <xsd:group name="EG_ColorTransform">
299   <xsd:choice>
300     <xsd:element name="tint" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
301     <xsd:element name="shade" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
302     <xsd:element name="comp" type="CT_ComplementTransform" minOccurs="1" maxOccurs="1"/>
303     <xsd:element name="inv" type="CT_InverseTransform" minOccurs="1" maxOccurs="1"/>
304     <xsd:element name="gray" type="CT_GrayscaleTransform" minOccurs="1" maxOccurs="1"/>
305     <xsd:element name="alpha" type="CT_PositiveFixedPercentage" minOccurs="1" maxOccurs="1"/>
306     <xsd:element name="alphaOff" type="CT_FixedPercentage" minOccurs="1" maxOccurs="1"/>
307     <xsd:element name="alphaMod" type="CT_PositivePercentage" minOccurs="1" maxOccurs="1"/>
308     <xsd:element name="hue" type="CT_PositiveFixedAngle" minOccurs="1" maxOccurs="1"/>

```

```

309     <xsd:element name="hueOff" type="CT_Angle" minOccurs="1" maxOccurs="1"/>
310     <xsd:element name="hueMod" type="CT_PositivePercentage" minOccurs="1" maxOccurs="1"/>
311     <xsd:element name="sat" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
312     <xsd:element name="satOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
313     <xsd:element name="satMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
314     <xsd:element name="lum" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
315     <xsd:element name="lumOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
316     <xsd:element name="lumMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
317     <xsd:element name="red" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
318     <xsd:element name="redOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
319     <xsd:element name="redMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
320     <xsd:element name="green" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
321     <xsd:element name="greenOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
322     <xsd:element name="greenMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
323     <xsd:element name="blue" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
324     <xsd:element name="blueOff" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
325     <xsd:element name="blueMod" type="CT_Percentage" minOccurs="1" maxOccurs="1"/>
326     <xsd:element name="gamma" type="CT_GammaTransform" minOccurs="1" maxOccurs="1"/>
327     <xsd:element name="invGamma" type="CT_InverseGammaTransform" minOccurs="1" maxOccurs="1"/>
328   </xsd:choice>
329 </xsd:group>
330 <xsd:complexType name="CT_ScRgbColor">
331   <xsd:sequence>
332     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
333   </xsd:sequence>
334   <xsd:attribute name="r" type="ST_Percentage" use="required"/>
335   <xsd:attribute name="g" type="ST_Percentage" use="required"/>
336   <xsd:attribute name="b" type="ST_Percentage" use="required"/>
337 </xsd:complexType>
338 <xsd:complexType name="CT_SRgbColor">
339   <xsd:sequence>
340     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
341   </xsd:sequence>
342   <xsd:attribute name="val" type="s:ST_HexColorRGB" use="required"/>
343 </xsd:complexType>
344 <xsd:complexType name="CT_HslColor">
345   <xsd:sequence>
346     <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
347   </xsd:sequence>
348   <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="required"/>
349   <xsd:attribute name="sat" type="ST_Percentage" use="required"/>
350   <xsd:attribute name="lum" type="ST_Percentage" use="required"/>
351 </xsd:complexType>
352 <xsd:simpleType name="ST_SystemColorVal">
353   <xsd:restriction base="xsd:token">
354     <xsd:enumeration value="scrollBar"/>
355     <xsd:enumeration value="background"/>
356     <xsd:enumeration value="activeCaption"/>
357     <xsd:enumeration value="inactiveCaption"/>
358     <xsd:enumeration value="menu"/>
359     <xsd:enumeration value="window"/>
360     <xsd:enumeration value="windowFrame"/>
361     <xsd:enumeration value="menuText"/>

```



```

362     <xsd:enumeration value="windowText"/>
363     <xsd:enumeration value="captionText"/>
364     <xsd:enumeration value="activeBorder"/>
365     <xsd:enumeration value="inactiveBorder"/>
366     <xsd:enumeration value="appWorkspace"/>
367     <xsd:enumeration value="highlight"/>
368     <xsd:enumeration value="highlightText"/>
369     <xsd:enumeration value="btnFace"/>
370     <xsd:enumeration value="btnShadow"/>
371     <xsd:enumeration value="grayText"/>
372     <xsd:enumeration value="btnText"/>
373     <xsd:enumeration value="inactiveCaptionText"/>
374     <xsd:enumeration value="btnHighlight"/>
375     <xsd:enumeration value="3dDkShadow"/>
376     <xsd:enumeration value="3dLight"/>
377     <xsd:enumeration value="infoText"/>
378     <xsd:enumeration value="infoBk"/>
379     <xsd:enumeration value="hotLight"/>
380     <xsd:enumeration value="gradientActiveCaption"/>
381     <xsd:enumeration value="gradientInactiveCaption"/>
382     <xsd:enumeration value="menuHighlight"/>
383     <xsd:enumeration value="menuBar"/>
384 </xsd:restriction>
385 </xsd:simpleType>
386 <xsd:complexType name="CT_SystemColor">
387     <xsd:sequence>
388         <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
389     </xsd:sequence>
390     <xsd:attribute name="val" type="ST_SystemColorVal" use="required"/>
391     <xsd:attribute name="lastClr" type="s:ST_HexColorRGB" use="optional"/>
392 </xsd:complexType>
393 <xsd:simpleType name="ST_SchemeColorVal">
394     <xsd:restriction base="xsd:token">
395         <xsd:enumeration value="bg1"/>
396         <xsd:enumeration value="tx1"/>
397         <xsd:enumeration value="bg2"/>
398         <xsd:enumeration value="tx2"/>
399         <xsd:enumeration value="accent1"/>
400         <xsd:enumeration value="accent2"/>
401         <xsd:enumeration value="accent3"/>
402         <xsd:enumeration value="accent4"/>
403         <xsd:enumeration value="accent5"/>
404         <xsd:enumeration value="accent6"/>
405         <xsd:enumeration value="hlink"/>
406         <xsd:enumeration value="folHlink"/>
407         <xsd:enumeration value="phClr"/>
408         <xsd:enumeration value="dk1"/>
409         <xsd:enumeration value="lt1"/>
410         <xsd:enumeration value="dk2"/>
411         <xsd:enumeration value="lt2"/>
412     </xsd:restriction>
413 </xsd:simpleType>
414 <xsd:complexType name="CT_SchemeColor">

```

```

415     <xsd:sequence>
416         <xsd:group ref="EG ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
417     </xsd:sequence>
418     <xsd:attribute name="val" type="ST SchemeColorVal" use="required"/>
419 </xsd:complexType>
420 <xsd:simpleType name="ST_PresetColorVal">
421     <xsd:restriction base="xsd:token">
422         <xsd:enumeration value="aliceBlue"/>
423         <xsd:enumeration value="antiqueWhite"/>
424         <xsd:enumeration value="aqua"/>
425         <xsd:enumeration value="aquamarine"/>
426         <xsd:enumeration value="azure"/>
427         <xsd:enumeration value="beige"/>
428         <xsd:enumeration value="bisque"/>
429         <xsd:enumeration value="black"/>
430         <xsd:enumeration value="blanchedAlmond"/>
431         <xsd:enumeration value="blue"/>
432         <xsd:enumeration value="blueViolet"/>
433         <xsd:enumeration value="brown"/>
434         <xsd:enumeration value="burlyWood"/>
435         <xsd:enumeration value="cadetBlue"/>
436         <xsd:enumeration value="chartreuse"/>
437         <xsd:enumeration value="chocolate"/>
438         <xsd:enumeration value="coral"/>
439         <xsd:enumeration value="cornflowerBlue"/>
440         <xsd:enumeration value="cornsilk"/>
441         <xsd:enumeration value="crimson"/>
442         <xsd:enumeration value="cyan"/>
443         <xsd:enumeration value="darkBlue"/>
444         <xsd:enumeration value="darkCyan"/>
445         <xsd:enumeration value="darkGoldenrod"/>
446         <xsd:enumeration value="darkGray"/>
447         <xsd:enumeration value="darkGrey"/>
448         <xsd:enumeration value="darkGreen"/>
449         <xsd:enumeration value="darkKhaki"/>
450         <xsd:enumeration value="darkMagenta"/>
451         <xsd:enumeration value="darkOliveGreen"/>
452         <xsd:enumeration value="darkOrange"/>
453         <xsd:enumeration value="darkOrchid"/>
454         <xsd:enumeration value="darkRed"/>
455         <xsd:enumeration value="darkSalmon"/>
456         <xsd:enumeration value="darkSeaGreen"/>
457         <xsd:enumeration value="darkSlateBlue"/>
458         <xsd:enumeration value="darkSlateGray"/>
459         <xsd:enumeration value="darkSlateGrey"/>
460         <xsd:enumeration value="darkTurquoise"/>
461         <xsd:enumeration value="darkViolet"/>
462         <xsd:enumeration value="dkBlue"/>
463         <xsd:enumeration value="dkCyan"/>
464         <xsd:enumeration value="dkGoldenrod"/>
465         <xsd:enumeration value="dkGray"/>
466         <xsd:enumeration value="dkGrey"/>
467         <xsd:enumeration value="dkGreen"/>

```

```

468      <xsd:enumeration value="dkKhaki"/>
469      <xsd:enumeration value="dkMagenta"/>
470      <xsd:enumeration value="dkOliveGreen"/>
471      <xsd:enumeration value="dkOrange"/>
472      <xsd:enumeration value="dkOrchid"/>
473      <xsd:enumeration value="dkRed"/>
474      <xsd:enumeration value="dkSalmon"/>
475      <xsd:enumeration value="dkSeaGreen"/>
476      <xsd:enumeration value="dkSlateBlue"/>
477      <xsd:enumeration value="dkSlateGray"/>
478      <xsd:enumeration value="dkSlateGrey"/>
479      <xsd:enumeration value="dkTurquoise"/>
480      <xsd:enumeration value="dkViolet"/>
481      <xsd:enumeration value="deepPink"/>
482      <xsd:enumeration value="deepSkyBlue"/>
483      <xsd:enumeration value="dimGray"/>
484      <xsd:enumeration value="dimGrey"/>
485      <xsd:enumeration value="dodgerBlue"/>
486      <xsd:enumeration value="firebrick"/>
487      <xsd:enumeration value="floralWhite"/>
488      <xsd:enumeration value="forestGreen"/>
489      <xsd:enumeration value="fuchsia"/>
490      <xsd:enumeration value="gainsboro"/>
491      <xsd:enumeration value="ghostWhite"/>
492      <xsd:enumeration value="gold"/>
493      <xsd:enumeration value="goldenrod"/>
494      <xsd:enumeration value="gray"/>
495      <xsd:enumeration value="grey"/>
496      <xsd:enumeration value="green"/>
497      <xsd:enumeration value="greenYellow"/>
498      <xsd:enumeration value="honeydew"/>
499      <xsd:enumeration value="hotPink"/>
500      <xsd:enumeration value="indianRed"/>
501      <xsd:enumeration value="indigo"/>
502      <xsd:enumeration value="ivory"/>
503      <xsd:enumeration value="khaki"/>
504      <xsd:enumeration value="lavender"/>
505      <xsd:enumeration value="lavenderBlush"/>
506      <xsd:enumeration value="lawnGreen"/>
507      <xsd:enumeration value="lemonChiffon"/>
508      <xsd:enumeration value="lightBlue"/>
509      <xsd:enumeration value="lightCoral"/>
510      <xsd:enumeration value="lightCyan"/>
511      <xsd:enumeration value="lightGoldenrodYellow"/>
512      <xsd:enumeration value="lightGray"/>
513      <xsd:enumeration value="lightGrey"/>
514      <xsd:enumeration value="lightGreen"/>
515      <xsd:enumeration value="lightPink"/>
516      <xsd:enumeration value="lightSalmon"/>
517      <xsd:enumeration value="lightSeaGreen"/>
518      <xsd:enumeration value="lightSkyBlue"/>
519      <xsd:enumeration value="lightSlateGray"/>
520      <xsd:enumeration value="lightSlateGrey"/>

```

```

521     <xsd:enumeration value="lightSteelBlue"/>
522     <xsd:enumeration value="lightYellow"/>
523     <xsd:enumeration value="ltBlue"/>
524     <xsd:enumeration value="ltCoral"/>
525     <xsd:enumeration value="ltCyan"/>
526     <xsd:enumeration value="ltGoldenrodYellow"/>
527     <xsd:enumeration value="ltGray"/>
528     <xsd:enumeration value="ltGrey"/>
529     <xsd:enumeration value="ltGreen"/>
530     <xsd:enumeration value="ltPink"/>
531     <xsd:enumeration value="ltSalmon"/>
532     <xsd:enumeration value="ltSeaGreen"/>
533     <xsd:enumeration value="ltSkyBlue"/>
534     <xsd:enumeration value="ltSlateGray"/>
535     <xsd:enumeration value="ltSlateGrey"/>
536     <xsd:enumeration value="ltSteelBlue"/>
537     <xsd:enumeration value="ltYellow"/>
538     <xsd:enumeration value="lime"/>
539     <xsd:enumeration value="limeGreen"/>
540     <xsd:enumeration value="linen"/>
541     <xsd:enumeration value="magenta"/>
542     <xsd:enumeration value="maroon"/>
543     <xsd:enumeration value="medAquamarine"/>
544     <xsd:enumeration value="medBlue"/>
545     <xsd:enumeration value="medOrchid"/>
546     <xsd:enumeration value="medPurple"/>
547     <xsd:enumeration value="medSeaGreen"/>
548     <xsd:enumeration value="medSlateBlue"/>
549     <xsd:enumeration value="medSpringGreen"/>
550     <xsd:enumeration value="medTurquoise"/>
551     <xsd:enumeration value="medVioletRed"/>
552     <xsd:enumeration value="mediumAquamarine"/>
553     <xsd:enumeration value="mediumBlue"/>
554     <xsd:enumeration value="mediumOrchid"/>
555     <xsd:enumeration value="mediumPurple"/>
556     <xsd:enumeration value="mediumSeaGreen"/>
557     <xsd:enumeration value="mediumSlateBlue"/>
558     <xsd:enumeration value="mediumSpringGreen"/>
559     <xsd:enumeration value="mediumTurquoise"/>
560     <xsd:enumeration value="mediumVioletRed"/>
561     <xsd:enumeration value="midnightBlue"/>
562     <xsd:enumeration value="mintCream"/>
563     <xsd:enumeration value="mistyRose"/>
564     <xsd:enumeration value="moccasin"/>
565     <xsd:enumeration value="navajoWhite"/>
566     <xsd:enumeration value="navy"/>
567     <xsd:enumeration value="oldLace"/>
568     <xsd:enumeration value="olive"/>
569     <xsd:enumeration value="oliveDrab"/>
570     <xsd:enumeration value="orange"/>
571     <xsd:enumeration value="orangeRed"/>
572     <xsd:enumeration value="orchid"/>
573     <xsd:enumeration value="paleGoldenrod"/>

```

```

574      <xsd:enumeration value="paleGreen"/>
575      <xsd:enumeration value="paleTurquoise"/>
576      <xsd:enumeration value="paleVioletRed"/>
577      <xsd:enumeration value="papayaWhip"/>
578      <xsd:enumeration value="peachPuff"/>
579      <xsd:enumeration value="peru"/>
580      <xsd:enumeration value="pink"/>
581      <xsd:enumeration value="plum"/>
582      <xsd:enumeration value="powderBlue"/>
583      <xsd:enumeration value="purple"/>
584      <xsd:enumeration value="red"/>
585      <xsd:enumeration value="rosyBrown"/>
586      <xsd:enumeration value="royalBlue"/>
587      <xsd:enumeration value="saddleBrown"/>
588      <xsd:enumeration value="salmon"/>
589      <xsd:enumeration value="sandyBrown"/>
590      <xsd:enumeration value="seaGreen"/>
591      <xsd:enumeration value="seaShell"/>
592      <xsd:enumeration value="sienna"/>
593      <xsd:enumeration value="silver"/>
594      <xsd:enumeration value="skyBlue"/>
595      <xsd:enumeration value="slateBlue"/>
596      <xsd:enumeration value="slateGray"/>
597      <xsd:enumeration value="slateGrey"/>
598      <xsd:enumeration value="snow"/>
599      <xsd:enumeration value="springGreen"/>
600      <xsd:enumeration value="steelBlue"/>
601      <xsd:enumeration value="tan"/>
602      <xsd:enumeration value="teal"/>
603      <xsd:enumeration value="thistle"/>
604      <xsd:enumeration value="tomato"/>
605      <xsd:enumeration value="turquoise"/>
606      <xsd:enumeration value="violet"/>
607      <xsd:enumeration value="wheat"/>
608      <xsd:enumeration value="white"/>
609      <xsd:enumeration value="whiteSmoke"/>
610      <xsd:enumeration value="yellow"/>
611      <xsd:enumeration value="yellowGreen"/>
612    </xsd:restriction>
613  </xsd:simpleType>
614  <xsd:complexType name="CT_PresetColor">
615    <xsd:sequence>
616      <xsd:group ref="EG_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>
617    </xsd:sequence>
618    <xsd:attribute name="val" type="ST_PresetColorVal"/>
619  </xsd:complexType>
620  <xsd:group name="EG_OfficeArtExtensionList">
621    <xsd:sequence>
622      <xsd:element name="ext" type="CT_OfficeArtExtension" minOccurs="0" maxOccurs="unbounded"/>
623    </xsd:sequence>
624  </xsd:group>
625  <xsd:complexType name="CT_OfficeArtExtensionList">
626    <xsd:sequence>

```

```

627     <xsd:group ref="EG_OfficeArtExtensionList" minOccurs="1" maxOccurs="1"/>
628   </xsd:sequence>
629 </xsd:complexType>
630 <xsd:complexType name="CT_Scale2D">
631   <xsd:sequence>
632     <xsd:element name="sx" type="CT_Ratio" minOccurs="1" maxOccurs="1"/>
633     <xsd:element name="sy" type="CT_Ratio" minOccurs="1" maxOccurs="1"/>
634   </xsd:sequence>
635 </xsd:complexType>
636 <xsd:complexType name="CT_Transform2D">
637   <xsd:sequence>
638     <xsd:element name="off" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
639     <xsd:element name="ext" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
640   </xsd:sequence>
641   <xsd:attribute name="rot" type="ST_Angle" use="optional" default="0"/>
642   <xsd:attribute name="flipH" type="xsd:boolean" use="optional" default="false"/>
643   <xsd:attribute name="flipV" type="xsd:boolean" use="optional" default="false"/>
644 </xsd:complexType>
645 <xsd:complexType name="CT_GroupTransform2D">
646   <xsd:sequence>
647     <xsd:element name="off" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
648     <xsd:element name="ext" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
649     <xsd:element name="chOff" type="CT_Point2D" minOccurs="0" maxOccurs="1"/>
650     <xsd:element name="chExt" type="CT_PositiveSize2D" minOccurs="0" maxOccurs="1"/>
651   </xsd:sequence>
652   <xsd:attribute name="rot" type="ST_Angle" use="optional" default="0"/>
653   <xsd:attribute name="flipH" type="xsd:boolean" use="optional" default="false"/>
654   <xsd:attribute name="flipV" type="xsd:boolean" use="optional" default="false"/>
655 </xsd:complexType>
656 <xsd:complexType name="CT_Point3D">
657   <xsd:attribute name="x" type="ST_Coordinate" use="required"/>
658   <xsd:attribute name="y" type="ST_Coordinate" use="required"/>
659   <xsd:attribute name="z" type="ST_Coordinate" use="required"/>
660 </xsd:complexType>
661 <xsd:complexType name="CT_Vector3D">
662   <xsd:attribute name="dx" type="ST_Coordinate" use="required"/>
663   <xsd:attribute name="dy" type="ST_Coordinate" use="required"/>
664   <xsd:attribute name="dz" type="ST_Coordinate" use="required"/>
665 </xsd:complexType>
666 <xsd:complexType name="CT_SphereCoords">
667   <xsd:attribute name="lat" type="ST_PositiveFixedAngle" use="required"/>
668   <xsd:attribute name="lon" type="ST_PositiveFixedAngle" use="required"/>
669   <xsd:attribute name="rev" type="ST_PositiveFixedAngle" use="required"/>
670 </xsd:complexType>
671 <xsd:complexType name="CT_RelativeRect">
672   <xsd:attribute name="l" type="ST_Percentage" use="optional" default="0%"/>
673   <xsd:attribute name="t" type="ST_Percentage" use="optional" default="0%"/>
674   <xsd:attribute name="r" type="ST_Percentage" use="optional" default="0%"/>
675   <xsd:attribute name="b" type="ST_Percentage" use="optional" default="0%"/>
676 </xsd:complexType>
677 <xsd:simpleType name="ST_RectAlignment">
678   <xsd:restriction base="xsd:token">
679     <xsd:enumeration value="tl"/>

```

```

680         <xsd:enumeration value="t"/>
681         <xsd:enumeration value="tr"/>
682         <xsd:enumeration value="l"/>
683         <xsd:enumeration value="ctr"/>
684         <xsd:enumeration value="r"/>
685         <xsd:enumeration value="bl"/>
686         <xsd:enumeration value="b"/>
687         <xsd:enumeration value="br"/>
688     </xsd:restriction>
689 </xsd:simpleType>
690 <xsd:group name="EG_ColorChoice">
691     <xsd:choice>
692         <xsd:element name="scrgbClr" type="CT_ScRgbColor" minOccurs="1" maxOccurs="1"/>
693         <xsd:element name="srgbClr" type="CT_SRgbColor" minOccurs="1" maxOccurs="1"/>
694         <xsd:element name="hslClr" type="CT_HslColor" minOccurs="1" maxOccurs="1"/>
695         <xsd:element name="sysClr" type="CT_SystemColor" minOccurs="1" maxOccurs="1"/>
696         <xsd:element name="schemeClr" type="CT_SchemeColor" minOccurs="1" maxOccurs="1"/>
697         <xsd:element name="prstClr" type="CT_PresetColor" minOccurs="1" maxOccurs="1"/>
698     </xsd:choice>
699 </xsd:group>
700 <xsd:complexType name="CT_Color">
701     <xsd:sequence>
702         <xsd:group ref="EG_ColorChoice"/>
703     </xsd:sequence>
704 </xsd:complexType>
705 <xsd:complexType name="CT_ColorMRU">
706     <xsd:sequence>
707         <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="unbounded"/>
708     </xsd:sequence>
709 </xsd:complexType>
710 <xsd:simpleType name="ST_BlackWhiteMode">
711     <xsd:restriction base="xsd:token">
712         <xsd:enumeration value="clr"/>
713         <xsd:enumeration value="auto"/>
714         <xsd:enumeration value="gray"/>
715         <xsd:enumeration value="ltGray"/>
716         <xsd:enumeration value="invGray"/>
717         <xsd:enumeration value="grayWhite"/>
718         <xsd:enumeration value="blackGray"/>
719         <xsd:enumeration value="blackWhite"/>
720         <xsd:enumeration value="black"/>
721         <xsd:enumeration value="white"/>
722         <xsd:enumeration value="hidden"/>
723     </xsd:restriction>
724 </xsd:simpleType>
725 <xsd:attributeGroup name="AG_Blob">
726     <xsd:attribute ref="r:embed" use="optional" default=""/>
727     <xsd:attribute ref="r:link" use="optional" default=""/>
728 </xsd:attributeGroup>
729 <xsd:complexType name="CT_EmbeddedWAVAudioFile">
730     <xsd:attribute ref="r:embed" use="required"/>
731     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
732 </xsd:complexType>

```

```

733 <xsd:complexType name="CT_Hyperlink">
734   <xsd:sequence>
735     <xsd:element name="snd" type="CT_EmbeddedWAVAudioFile" minOccurs="0" maxOccurs="1"/>
736     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
737   </xsd:sequence>
738   <xsd:attribute ref="r:id" use="optional"/>
739   <xsd:attribute name="invalidUrl" type="xsd:string" use="optional" default=""/>
740   <xsd:attribute name="action" type="xsd:string" use="optional" default=""/>
741   <xsd:attribute name="tgtFrame" type="xsd:string" use="optional" default=""/>
742   <xsd:attribute name="tooltip" type="xsd:string" use="optional" default=""/>
743   <xsd:attribute name="history" type="xsd:boolean" use="optional" default="true"/>
744   <xsd:attribute name="highlightClick" type="xsd:boolean" use="optional" default="false"/>
745   <xsd:attribute name="endSnd" type="xsd:boolean" use="optional" default="false"/>
746 </xsd:complexType>
747 <xsd:simpleType name="ST_DrawingElementId">
748   <xsd:restriction base="xsd:unsignedInt"/>
749 </xsd:simpleType>
750 <xsd:attributeGroup name="AG_Locking">
751   <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
752   <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
753   <xsd:attribute name="noRot" type="xsd:boolean" use="optional" default="false"/>
754   <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
755   <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
756   <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
757   <xsd:attribute name="noEditPoints" type="xsd:boolean" use="optional" default="false"/>
758   <xsd:attribute name="noAdjustHandles" type="xsd:boolean" use="optional" default="false"/>
759   <xsd:attribute name="noChangeArrowheads" type="xsd:boolean" use="optional" default="false"/>
760   <xsd:attribute name="noChangeShapeType" type="xsd:boolean" use="optional" default="false"/>
761 </xsd:attributeGroup>
762 <xsd:complexType name="CT_ConnectorLocking">
763   <xsd:sequence>
764     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
765   </xsd:sequence>
766   <xsd:attributeGroup ref="AG_Locking"/>
767 </xsd:complexType>
768 <xsd:complexType name="CT_ShapeLocking">
769   <xsd:sequence>
770     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
771   </xsd:sequence>
772   <xsd:attributeGroup ref="AG_Locking"/>
773   <xsd:attribute name="noTextEdit" type="xsd:boolean" use="optional" default="false"/>
774 </xsd:complexType>
775 <xsd:complexType name="CT_PictureLocking">
776   <xsd:sequence>
777     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
778   </xsd:sequence>
779   <xsd:attributeGroup ref="AG_Locking"/>
780   <xsd:attribute name="noCrop" type="xsd:boolean" use="optional" default="false"/>
781 </xsd:complexType>
782 <xsd:complexType name="CT_GroupLocking">
783   <xsd:sequence>
784     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
785   </xsd:sequence>

```



```

786     <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
787     <xsd:attribute name="noUngrp" type="xsd:boolean" use="optional" default="false"/>
788     <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
789     <xsd:attribute name="noRot" type="xsd:boolean" use="optional" default="false"/>
790     <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
791     <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
792     <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
793 </xsd:complexType>
794 <xsd:complexType name="CT_GraphicalObjectFrameLocking">
795     <xsd:sequence>
796         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
797     </xsd:sequence>
798     <xsd:attribute name="noGrp" type="xsd:boolean" use="optional" default="false"/>
799     <xsd:attribute name="noDrilldown" type="xsd:boolean" use="optional" default="false"/>
800     <xsd:attribute name="noSelect" type="xsd:boolean" use="optional" default="false"/>
801     <xsd:attribute name="noChangeAspect" type="xsd:boolean" use="optional" default="false"/>
802     <xsd:attribute name="noMove" type="xsd:boolean" use="optional" default="false"/>
803     <xsd:attribute name="noResize" type="xsd:boolean" use="optional" default="false"/>
804 </xsd:complexType>
805 <xsd:complexType name="CT_NonVisualDrawingProps">
806     <xsd:sequence>
807         <xsd:element name="hlinkClick" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
808         <xsd:element name="hlinkHover" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
809         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
810     </xsd:sequence>
811     <xsd:attribute name="id" type="ST_DrawingElementId" use="required"/>
812     <xsd:attribute name="name" type="xsd:string" use="required"/>
813     <xsd:attribute name="descr" type="xsd:string" use="optional" default=""/>
814     <xsd:attribute name="hidden" type="xsd:boolean" use="optional" default="false"/>
815     <xsd:attribute name="title" type="xsd:string" use="optional" default=""/>
816 </xsd:complexType>
817 <xsd:complexType name="CT_NonVisualDrawingShapeProps">
818     <xsd:sequence>
819         <xsd:element name="spLocks" type="CT_ShapeLocking" minOccurs="0" maxOccurs="1"/>
820         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
821     </xsd:sequence>
822     <xsd:attribute name="txBox" type="xsd:boolean" use="optional" default="false"/>
823 </xsd:complexType>
824 <xsd:complexType name="CT_NonVisualConnectorProperties">
825     <xsd:sequence>
826         <xsd:element name="cxnSpLocks" type="CT_ConnectorLocking" minOccurs="0" maxOccurs="1"/>
827         <xsd:element name="stCxn" type="CT_Connection" minOccurs="0" maxOccurs="1"/>
828         <xsd:element name="endCxn" type="CT_Connection" minOccurs="0" maxOccurs="1"/>
829         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
830     </xsd:sequence>
831 </xsd:complexType>
832 <xsd:complexType name="CT_NonVisualPictureProperties">
833     <xsd:sequence>
834         <xsd:element name="picLocks" type="CT_PictureLocking" minOccurs="0" maxOccurs="1"/>
835         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
836     </xsd:sequence>
837     <xsd:attribute name="preferRelativeResize" type="xsd:boolean" use="optional" default="true"/>
838 </xsd:complexType>

```

```

839 <xsd:complexType name="CT_NonVisualGroupDrawingShapeProps">
840   <xsd:sequence>
841     <xsd:element name="grpSpLocks" type="CT_GroupLocking" minOccurs="0" maxOccurs="1"/>
842     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
843   </xsd:sequence>
844 </xsd:complexType>
845 <xsd:complexType name="CT_NonVisualGraphicFrameProperties">
846   <xsd:sequence>
847     <xsd:element name="graphicFrameLocks" type="CT_GraphicalObjectFrameLocking" minOccurs="0"
848       maxOccurs="1"/>
849     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
850   </xsd:sequence>
851 </xsd:complexType>
852 <xsd:complexType name="CT_GraphicalObjectData">
853   <xsd:sequence>
854     <xsd:any minOccurs="0" maxOccurs="unbounded" processContents="strict"/>
855   </xsd:sequence>
856   <xsd:attribute name="uri" type="xsd:token"/>
857 </xsd:complexType>
858 <xsd:complexType name="CT_GraphicalObject">
859   <xsd:sequence>
860     <xsd:element name="graphicData" type="CT_GraphicalObjectData"/>
861   </xsd:sequence>
862 </xsd:complexType>
863 <xsd:element name="graphic" type="CT_GraphicalObject"/>
864 <xsd:simpleType name="ST_ChartBuildStep">
865   <xsd:restriction base="xsd:token">
866     <xsd:enumeration value="category"/>
867     <xsd:enumeration value="ptInCategory"/>
868     <xsd:enumeration value="series"/>
869     <xsd:enumeration value="ptInSeries"/>
870     <xsd:enumeration value="allPts"/>
871     <xsd:enumeration value="gridLegend"/>
872   </xsd:restriction>
873 </xsd:simpleType>
874 <xsd:simpleType name="ST_DgmBuildStep">
875   <xsd:restriction base="xsd:token">
876     <xsd:enumeration value="sp"/>
877     <xsd:enumeration value="bg"/>
878   </xsd:restriction>
879 </xsd:simpleType>
880 <xsd:complexType name="CT_AnimationDgmElement">
881   <xsd:attribute name="id" type="s:ST_Guid" use="optional" default="{00000000-0000-0000-0000-
882     000000000000}"/>
883   <xsd:attribute name="bldStep" type="ST_DgmBuildStep" use="optional" default="sp"/>
884 </xsd:complexType>
885 <xsd:complexType name="CT_AnimationChartElement">
886   <xsd:attribute name="seriesIdx" type="xsd:int" use="optional" default="-1"/>
887   <xsd:attribute name="categoryIdx" type="xsd:int" use="optional" default="-1"/>
888   <xsd:attribute name="bldStep" type="ST_ChartBuildStep" use="required"/>
889 </xsd:complexType>
890 <xsd:complexType name="CT_AnimationElementChoice">
891   <xsd:choice minOccurs="1" maxOccurs="1">

```

```

892     <xsd:element name="dgm" type="CT_AnimationDgmElement"/>
893     <xsd:element name="chart" type="CT_AnimationChartElement"/>
894   </xsd:choice>
895 </xsd:complexType>
896 <xsd:simpleType name="ST_AnimationBuildType">
897   <xsd:restriction base="xsd:token">
898     <xsd:enumeration value="allAtOnce"/>
899   </xsd:restriction>
900 </xsd:simpleType>
901 <xsd:simpleType name="ST_AnimationDgmOnlyBuildType">
902   <xsd:restriction base="xsd:token">
903     <xsd:enumeration value="one"/>
904     <xsd:enumeration value="lvlOne"/>
905     <xsd:enumeration value="lvlAtOnce"/>
906   </xsd:restriction>
907 </xsd:simpleType>
908 <xsd:simpleType name="ST_AnimationDgmBuildType">
909   <xsd:union memberTypes="ST_AnimationBuildType ST_AnimationDgmOnlyBuildType"/>
910 </xsd:simpleType>
911 <xsd:complexType name="CT_AnimationDgmBuildProperties">
912   <xsd:attribute name="bld" type="ST_AnimationDgmBuildType" use="optional" default="allAtOnce"/>
913   <xsd:attribute name="rev" type="xsd:boolean" use="optional" default="false"/>
914 </xsd:complexType>
915 <xsd:simpleType name="ST_AnimationChartOnlyBuildType">
916   <xsd:restriction base="xsd:token">
917     <xsd:enumeration value="series"/>
918     <xsd:enumeration value="category"/>
919     <xsd:enumeration value="seriesEl"/>
920     <xsd:enumeration value="categoryEl"/>
921   </xsd:restriction>
922 </xsd:simpleType>
923 <xsd:simpleType name="ST_AnimationChartBuildType">
924   <xsd:union memberTypes="ST_AnimationBuildType ST_AnimationChartOnlyBuildType"/>
925 </xsd:simpleType>
926 <xsd:complexType name="CT_AnimationChartBuildProperties">
927   <xsd:attribute name="bld" type="ST_AnimationChartBuildType" use="optional"
928     default="allAtOnce"/>
929   <xsd:attribute name="animBg" type="xsd:boolean" use="optional" default="true"/>
930 </xsd:complexType>
931 <xsd:complexType name="CT_AnimationGraphicalObjectBuildProperties">
932   <xsd:choice>
933     <xsd:element name="bldDgm" type="CT_AnimationDgmBuildProperties"/>
934     <xsd:element name="bldChart" type="CT_AnimationChartBuildProperties"/>
935   </xsd:choice>
936 </xsd:complexType>
937 <xsd:complexType name="CT_BackgroundFormatting">
938   <xsd:sequence>
939     <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
940     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
941   </xsd:sequence>
942 </xsd:complexType>
943 <xsd:complexType name="CT_WholeE2oFormatting">
944   <xsd:sequence>

```

```

945     <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
946     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
947   </xsd:sequence>
948 </xsd:complexType>
949 <xsd:complexType name="CT_GvmlUseShapeRectangle"/>
950 <xsd:complexType name="CT_GvmlTextShape">
951   <xsd:sequence>
952     <xsd:element name="txBody" type="CT_TextBody" minOccurs="1" maxOccurs="1"/>
953     <xsd:choice>
954       <xsd:element name="useSpRect" type="CT_GvmlUseShapeRectangle" minOccurs="1"
955         maxOccurs="1"/>
956       <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="1" maxOccurs="1"/>
957     </xsd:choice>
958     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
959   </xsd:sequence>
960 </xsd:complexType>
961 <xsd:complexType name="CT_GvmlShapeNonVisual">
962   <xsd:sequence>
963     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
964     <xsd:element name="cNvSpPr" type="CT_NonVisualDrawingShapeProps" minOccurs="1"
965       maxOccurs="1"/>
966   </xsd:sequence>
967 </xsd:complexType>
968 <xsd:complexType name="CT_GvmlShape">
969   <xsd:sequence>
970     <xsd:element name="nvSpPr" type="CT_GvmlShapeNonVisual" minOccurs="1" maxOccurs="1"/>
971     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
972     <xsd:element name="txSp" type="CT_GvmlTextShape" minOccurs="0" maxOccurs="1"/>
973     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
974     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
975   </xsd:sequence>
976 </xsd:complexType>
977 <xsd:complexType name="CT_GvmlConnectorNonVisual">
978   <xsd:sequence>
979     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
980     <xsd:element name="cNvCxnSpPr" type="CT_NonVisualConnectorProperties" minOccurs="1"
981       maxOccurs="1"/>
982   </xsd:sequence>
983 </xsd:complexType>
984 <xsd:complexType name="CT_GvmlConnector">
985   <xsd:sequence>
986     <xsd:element name="nvCxnSpPr" type="CT_GvmlConnectorNonVisual" minOccurs="1"
987       maxOccurs="1"/>
988     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
989     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
990     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
991   </xsd:sequence>
992 </xsd:complexType>
993 <xsd:complexType name="CT_GvmlPictureNonVisual">
994   <xsd:sequence>
995     <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
996     <xsd:element name="cNvPicPr" type="CT_NonVisualPictureProperties" minOccurs="1"
997       maxOccurs="1"/>

```

```

998     </xsd:sequence>
999 </xsd:complexType>
1000 <xsd:complexType name="CT_GvmlPicture">
1001     <xsd:sequence>
1002         <xsd:element name="nvPicPr" type="CT_GvmlPictureNonVisual" minOccurs="1" maxOccurs="1"/>
1003         <xsd:element name="blipFill" type="CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
1004         <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
1005         <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
1006         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1007     </xsd:sequence>
1008 </xsd:complexType>
1009 <xsd:complexType name="CT_GvmlGraphicFrameNonVisual">
1010     <xsd:sequence>
1011         <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1012         <xsd:element name="cNvGraphicFramePr" type="CT_NonVisualGraphicFrameProperties"
1013             minOccurs="1" maxOccurs="1"/>
1014     </xsd:sequence>
1015 </xsd:complexType>
1016 <xsd:complexType name="CT_GvmlGraphicalObjectFrame">
1017     <xsd:sequence>
1018         <xsd:element name="nvGraphicFramePr" type="CT_GvmlGraphicFrameNonVisual" minOccurs="1"
1019             maxOccurs="1"/>
1020         <xsd:element ref="graphic" minOccurs="1" maxOccurs="1"/>
1021         <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="1" maxOccurs="1"/>
1022         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1023     </xsd:sequence>
1024 </xsd:complexType>
1025 <xsd:complexType name="CT_GvmlGroupShapeNonVisual">
1026     <xsd:sequence>
1027         <xsd:element name="cNvPr" type="CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
1028         <xsd:element name="cNvGrpSpPr" type="CT_NonVisualGroupDrawingShapeProps" minOccurs="1"
1029             maxOccurs="1"/>
1030     </xsd:sequence>
1031 </xsd:complexType>
1032 <xsd:complexType name="CT_GvmlGroupShape">
1033     <xsd:sequence>
1034         <xsd:element name="nvGrpSpPr" type="CT_GvmlGroupShapeNonVisual" minOccurs="1"
1035             maxOccurs="1"/>
1036         <xsd:element name="grpSpPr" type="CT_GroupShapeProperties" minOccurs="1" maxOccurs="1"/>
1037         <xsd:choice minOccurs="0" maxOccurs="unbounded">
1038             <xsd:element name="txSp" type="CT_GvmlTextShape"/>
1039             <xsd:element name="sp" type="CT_GvmlShape"/>
1040             <xsd:element name="cxnSp" type="CT_GvmlConnector"/>
1041             <xsd:element name="pic" type="CT_GvmlPicture"/>
1042             <xsd:element name="graphicFrame" type="CT_GvmlGraphicalObjectFrame"/>
1043             <xsd:element name="grpSp" type="CT_GvmlGroupShape"/>
1044         </xsd:choice>
1045         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1046     </xsd:sequence>
1047 </xsd:complexType>
1048 <xsd:simpleType name="ST_PresetCameraType">
1049     <xsd:restriction base="xsd:token">
1050         <xsd:enumeration value="legacyObliqueTopLeft"/>

```

```

1051     <xsd:enumeration value="legacyObliqueTop"/>
1052     <xsd:enumeration value="legacyObliqueTopRight"/>
1053     <xsd:enumeration value="legacyObliqueLeft"/>
1054     <xsd:enumeration value="legacyObliqueFront"/>
1055     <xsd:enumeration value="legacyObliqueRight"/>
1056     <xsd:enumeration value="legacyObliqueBottomLeft"/>
1057     <xsd:enumeration value="legacyObliqueBottom"/>
1058     <xsd:enumeration value="legacyObliqueBottomRight"/>
1059     <xsd:enumeration value="legacyPerspectiveTopLeft"/>
1060     <xsd:enumeration value="legacyPerspectiveTop"/>
1061     <xsd:enumeration value="legacyPerspectiveTopRight"/>
1062     <xsd:enumeration value="legacyPerspectiveLeft"/>
1063     <xsd:enumeration value="legacyPerspectiveFront"/>
1064     <xsd:enumeration value="legacyPerspectiveRight"/>
1065     <xsd:enumeration value="legacyPerspectiveBottomLeft"/>
1066     <xsd:enumeration value="legacyPerspectiveBottom"/>
1067     <xsd:enumeration value="legacyPerspectiveBottomRight"/>
1068     <xsd:enumeration value="orthographicFront"/>
1069     <xsd:enumeration value="isometricTopUp"/>
1070     <xsd:enumeration value="isometricTopDown"/>
1071     <xsd:enumeration value="isometricBottomUp"/>
1072     <xsd:enumeration value="isometricBottomDown"/>
1073     <xsd:enumeration value="isometricLeftUp"/>
1074     <xsd:enumeration value="isometricLeftDown"/>
1075     <xsd:enumeration value="isometricRightUp"/>
1076     <xsd:enumeration value="isometricRightDown"/>
1077     <xsd:enumeration value="isometricOffAxis1Left"/>
1078     <xsd:enumeration value="isometricOffAxis1Right"/>
1079     <xsd:enumeration value="isometricOffAxis1Top"/>
1080     <xsd:enumeration value="isometricOffAxis2Left"/>
1081     <xsd:enumeration value="isometricOffAxis2Right"/>
1082     <xsd:enumeration value="isometricOffAxis2Top"/>
1083     <xsd:enumeration value="isometricOffAxis3Left"/>
1084     <xsd:enumeration value="isometricOffAxis3Right"/>
1085     <xsd:enumeration value="isometricOffAxis3Bottom"/>
1086     <xsd:enumeration value="isometricOffAxis4Left"/>
1087     <xsd:enumeration value="isometricOffAxis4Right"/>
1088     <xsd:enumeration value="isometricOffAxis4Bottom"/>
1089     <xsd:enumeration value="obliqueTopLeft"/>
1090     <xsd:enumeration value="obliqueTop"/>
1091     <xsd:enumeration value="obliqueTopRight"/>
1092     <xsd:enumeration value="obliqueLeft"/>
1093     <xsd:enumeration value="obliqueRight"/>
1094     <xsd:enumeration value="obliqueBottomLeft"/>
1095     <xsd:enumeration value="obliqueBottom"/>
1096     <xsd:enumeration value="obliqueBottomRight"/>
1097     <xsd:enumeration value="perspectiveFront"/>
1098     <xsd:enumeration value="perspectiveLeft"/>
1099     <xsd:enumeration value="perspectiveRight"/>
1100     <xsd:enumeration value="perspectiveAbove"/>
1101     <xsd:enumeration value="perspectiveBelow"/>
1102     <xsd:enumeration value="perspectiveAboveLeftFacing"/>
1103     <xsd:enumeration value="perspectiveAboveRightFacing"/>

```

```

1104     <xsd:enumeration value="perspectiveContrastingLeftFacing"/>
1105     <xsd:enumeration value="perspectiveContrastingRightFacing"/>
1106     <xsd:enumeration value="perspectiveHeroicLeftFacing"/>
1107     <xsd:enumeration value="perspectiveHeroicRightFacing"/>
1108     <xsd:enumeration value="perspectiveHeroicExtremeLeftFacing"/>
1109     <xsd:enumeration value="perspectiveHeroicExtremeRightFacing"/>
1110     <xsd:enumeration value="perspectiveRelaxed"/>
1111     <xsd:enumeration value="perspectiveRelaxedModerately"/>
1112 </xsd:restriction>
1113 </xsd:simpleType>
1114 <xsd:simpleType name="ST_FOVAngle">
1115     <xsd:restriction base="ST_Angle">
1116         <xsd:minInclusive value="0"/>
1117         <xsd:maxInclusive value="10800000"/>
1118     </xsd:restriction>
1119 </xsd:simpleType>
1120 <xsd:complexType name="CT_Camera">
1121     <xsd:sequence>
1122         <xsd:element name="rot" type="CT_SphereCoords" minOccurs="0" maxOccurs="1"/>
1123     </xsd:sequence>
1124     <xsd:attribute name="prst" type="ST_PresetCameraType" use="required"/>
1125     <xsd:attribute name="fov" type="ST_FOVAngle" use="optional"/>
1126     <xsd:attribute name="zoom" type="ST_PositivePercentage" use="optional" default="100%"/>
1127 </xsd:complexType>
1128 <xsd:simpleType name="ST_LightRigDirection">
1129     <xsd:restriction base="xsd:token">
1130         <xsd:enumeration value="tl"/>
1131         <xsd:enumeration value="t"/>
1132         <xsd:enumeration value="tr"/>
1133         <xsd:enumeration value="l"/>
1134         <xsd:enumeration value="r"/>
1135         <xsd:enumeration value="bl"/>
1136         <xsd:enumeration value="b"/>
1137         <xsd:enumeration value="br"/>
1138     </xsd:restriction>
1139 </xsd:simpleType>
1140 <xsd:simpleType name="ST_LightRigType">
1141     <xsd:restriction base="xsd:token">
1142         <xsd:enumeration value="legacyFlat1"/>
1143         <xsd:enumeration value="legacyFlat2"/>
1144         <xsd:enumeration value="legacyFlat3"/>
1145         <xsd:enumeration value="legacyFlat4"/>
1146         <xsd:enumeration value="legacyNormal1"/>
1147         <xsd:enumeration value="legacyNormal2"/>
1148         <xsd:enumeration value="legacyNormal3"/>
1149         <xsd:enumeration value="legacyNormal4"/>
1150         <xsd:enumeration value="legacyHarsh1"/>
1151         <xsd:enumeration value="legacyHarsh2"/>
1152         <xsd:enumeration value="legacyHarsh3"/>
1153         <xsd:enumeration value="legacyHarsh4"/>
1154         <xsd:enumeration value="threePt"/>
1155         <xsd:enumeration value="balanced"/>
1156         <xsd:enumeration value="soft"/>

```

```

1157     <xsd:enumeration value="harsh"/>
1158     <xsd:enumeration value="flood"/>
1159     <xsd:enumeration value="contrasting"/>
1160     <xsd:enumeration value="morning"/>
1161     <xsd:enumeration value="sunrise"/>
1162     <xsd:enumeration value="sunset"/>
1163     <xsd:enumeration value="chilly"/>
1164     <xsd:enumeration value="freezing"/>
1165     <xsd:enumeration value="flat"/>
1166     <xsd:enumeration value="twoPt"/>
1167     <xsd:enumeration value="glow"/>
1168     <xsd:enumeration value="brightRoom"/>
1169   </xsd:restriction>
1170 </xsd:simpleType>
1171 <xsd:complexType name="CT_LightRig">
1172   <xsd:sequence>
1173     <xsd:element name="rot" type="CT_SphereCoords" minOccurs="0" maxOccurs="1"/>
1174   </xsd:sequence>
1175   <xsd:attribute name="rig" type="ST_LightRigType" use="required"/>
1176   <xsd:attribute name="dir" type="ST_LightRigDirection" use="required"/>
1177 </xsd:complexType>
1178 <xsd:complexType name="CT_Scene3D">
1179   <xsd:sequence>
1180     <xsd:element name="camera" type="CT_Camera" minOccurs="1" maxOccurs="1"/>
1181     <xsd:element name="lightRig" type="CT_LightRig" minOccurs="1" maxOccurs="1"/>
1182     <xsd:element name="backdrop" type="CT_Backdrop" minOccurs="0" maxOccurs="1"/>
1183     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1184   </xsd:sequence>
1185 </xsd:complexType>
1186 <xsd:complexType name="CT_Backdrop">
1187   <xsd:sequence>
1188     <xsd:element name="anchor" type="CT_Point3D" minOccurs="1" maxOccurs="1"/>
1189     <xsd:element name="norm" type="CT_Vector3D" minOccurs="1" maxOccurs="1"/>
1190     <xsd:element name="up" type="CT_Vector3D" minOccurs="1" maxOccurs="1"/>
1191     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1192   </xsd:sequence>
1193 </xsd:complexType>
1194 <xsd:simpleType name="ST_BevelPresetType">
1195   <xsd:restriction base="xsd:token">
1196     <xsd:enumeration value="relaxedInset"/>
1197     <xsd:enumeration value="circle"/>
1198     <xsd:enumeration value="slope"/>
1199     <xsd:enumeration value="cross"/>
1200     <xsd:enumeration value="angle"/>
1201     <xsd:enumeration value="softRound"/>
1202     <xsd:enumeration value="convex"/>
1203     <xsd:enumeration value="coolSlant"/>
1204     <xsd:enumeration value="divot"/>
1205     <xsd:enumeration value="ribblet"/>
1206     <xsd:enumeration value="hardEdge"/>
1207     <xsd:enumeration value="artDeco"/>
1208   </xsd:restriction>
1209 </xsd:simpleType>

```



```

1210 <xsd:complexType name="CT_Bevel">
1211   <xsd:attribute name="w" type="ST_PositiveCoordinate" use="optional" default="76200"/>
1212   <xsd:attribute name="h" type="ST_PositiveCoordinate" use="optional" default="76200"/>
1213   <xsd:attribute name="prst" type="ST_BevelPresetType" use="optional" default="circle"/>
1214 </xsd:complexType>
1215 <xsd:simpleType name="ST_PresetMaterialType">
1216   <xsd:restriction base="xsd:token">
1217     <xsd:enumeration value="legacyMatte"/>
1218     <xsd:enumeration value="legacyPlastic"/>
1219     <xsd:enumeration value="legacyMetal"/>
1220     <xsd:enumeration value="legacyWireframe"/>
1221     <xsd:enumeration value="matte"/>
1222     <xsd:enumeration value="plastic"/>
1223     <xsd:enumeration value="metal"/>
1224     <xsd:enumeration value="warmMatte"/>
1225     <xsd:enumeration value="translucentPowder"/>
1226     <xsd:enumeration value="powder"/>
1227     <xsd:enumeration value="dkEdge"/>
1228     <xsd:enumeration value="softEdge"/>
1229     <xsd:enumeration value="clear"/>
1230     <xsd:enumeration value="flat"/>
1231     <xsd:enumeration value="softmetal"/>
1232   </xsd:restriction>
1233 </xsd:simpleType>
1234 <xsd:complexType name="CT_Shape3D">
1235   <xsd:sequence>
1236     <xsd:element name="bevelT" type="CT_Bevel" minOccurs="0" maxOccurs="1"/>
1237     <xsd:element name="bevelB" type="CT_Bevel" minOccurs="0" maxOccurs="1"/>
1238     <xsd:element name="extrusionClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1239     <xsd:element name="contourClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1240     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1241   </xsd:sequence>
1242   <xsd:attribute name="z" type="ST_Coordinate" use="optional" default="0"/>
1243   <xsd:attribute name="extrusionH" type="ST_PositiveCoordinate" use="optional" default="0"/>
1244   <xsd:attribute name="contourW" type="ST_PositiveCoordinate" use="optional" default="0"/>
1245   <xsd:attribute name="prstMaterial" type="ST_PresetMaterialType" use="optional"
1246     default="warmMatte"/>
1247 </xsd:complexType>
1248 <xsd:complexType name="CT_FlatText">
1249   <xsd:attribute name="z" type="ST_Coordinate" use="optional" default="0"/>
1250 </xsd:complexType>
1251 <xsd:group name="EG_Text3D">
1252   <xsd:choice>
1253     <xsd:element name="sp3d" type="CT_Shape3D" minOccurs="1" maxOccurs="1"/>
1254     <xsd:element name="flatTx" type="CT_FlatText" minOccurs="1" maxOccurs="1"/>
1255   </xsd:choice>
1256 </xsd:group>
1257 <xsd:complexType name="CT_AlphaBiLevelEffect">
1258   <xsd:attribute name="thresh" type="ST_PositiveFixedPercentage" use="required"/>
1259 </xsd:complexType>
1260 <xsd:complexType name="CT_AlphaCeilingEffect"/>
1261 <xsd:complexType name="CT_AlphaFloorEffect"/>
1262 <xsd:complexType name="CT_AlphaInverseEffect">

```

```

1263     <xsd:sequence>
1264         <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
1265     </xsd:sequence>
1266 </xsd:complexType>
1267 <xsd:complexType name="CT_AlphaModulateFixedEffect">
1268     <xsd:attribute name="amt" type="ST_PositivePercentage" use="optional" default="100%"/>
1269 </xsd:complexType>
1270 <xsd:complexType name="CT_AlphaOutsetEffect">
1271     <xsd:attribute name="rad" type="ST_Coordinate" use="optional" default="0"/>
1272 </xsd:complexType>
1273 <xsd:complexType name="CT_AlphaReplaceEffect">
1274     <xsd:attribute name="a" type="ST_PositiveFixedPercentage" use="required"/>
1275 </xsd:complexType>
1276 <xsd:complexType name="CT_BiLevelEffect">
1277     <xsd:attribute name="thresh" type="ST_PositiveFixedPercentage" use="required"/>
1278 </xsd:complexType>
1279 <xsd:complexType name="CT_BlurEffect">
1280     <xsd:attribute name="rad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1281     <xsd:attribute name="grow" type="xsd:boolean" use="optional" default="true"/>
1282 </xsd:complexType>
1283 <xsd:complexType name="CT_ColorChangeEffect">
1284     <xsd:sequence>
1285         <xsd:element name="clrFrom" type="CT_Color" minOccurs="1" maxOccurs="1"/>
1286         <xsd:element name="clrTo" type="CT_Color" minOccurs="1" maxOccurs="1"/>
1287     </xsd:sequence>
1288     <xsd:attribute name="useA" type="xsd:boolean" use="optional" default="true"/>
1289 </xsd:complexType>
1290 <xsd:complexType name="CT_ColorReplaceEffect">
1291     <xsd:sequence>
1292         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1293     </xsd:sequence>
1294 </xsd:complexType>
1295 <xsd:complexType name="CT_DuotoneEffect">
1296     <xsd:sequence>
1297         <xsd:group ref="EG_ColorChoice" minOccurs="2" maxOccurs="2"/>
1298     </xsd:sequence>
1299 </xsd:complexType>
1300 <xsd:complexType name="CT_GlowEffect">
1301     <xsd:sequence>
1302         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1303     </xsd:sequence>
1304     <xsd:attribute name="rad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1305 </xsd:complexType>
1306 <xsd:complexType name="CT_GrayscaleEffect"/>
1307 <xsd:complexType name="CT_HSLEffect">
1308     <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1309     <xsd:attribute name="sat" type="ST_FixedPercentage" use="optional" default="0%"/>
1310     <xsd:attribute name="lum" type="ST_FixedPercentage" use="optional" default="0%"/>
1311 </xsd:complexType>
1312 <xsd:complexType name="CT_InnerShadowEffect">
1313     <xsd:sequence>
1314         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1315     </xsd:sequence>

```

```

1316     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1317     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1318     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1319 </xsd:complexType>
1320 <xsd:complexType name="CT_LuminanceEffect">
1321     <xsd:attribute name="bright" type="ST_FixedPercentage" use="optional" default="0%"/>
1322     <xsd:attribute name="contrast" type="ST_FixedPercentage" use="optional" default="0%"/>
1323 </xsd:complexType>
1324 <xsd:complexType name="CT_OuterShadowEffect">
1325     <xsd:sequence>
1326         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1327     </xsd:sequence>
1328     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1329     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1330     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1331     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1332     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1333     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1334     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1335     <xsd:attribute name="align" type="ST_RectAlignment" use="optional" default="b"/>
1336     <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional" default="true"/>
1337 </xsd:complexType>
1338 <xsd:simpleType name="ST_PresetShadowVal">
1339     <xsd:restriction base="xsd:token">
1340         <xsd:enumeration value="shdw1"/>
1341         <xsd:enumeration value="shdw2"/>
1342         <xsd:enumeration value="shdw3"/>
1343         <xsd:enumeration value="shdw4"/>
1344         <xsd:enumeration value="shdw5"/>
1345         <xsd:enumeration value="shdw6"/>
1346         <xsd:enumeration value="shdw7"/>
1347         <xsd:enumeration value="shdw8"/>
1348         <xsd:enumeration value="shdw9"/>
1349         <xsd:enumeration value="shdw10"/>
1350         <xsd:enumeration value="shdw11"/>
1351         <xsd:enumeration value="shdw12"/>
1352         <xsd:enumeration value="shdw13"/>
1353         <xsd:enumeration value="shdw14"/>
1354         <xsd:enumeration value="shdw15"/>
1355         <xsd:enumeration value="shdw16"/>
1356         <xsd:enumeration value="shdw17"/>
1357         <xsd:enumeration value="shdw18"/>
1358         <xsd:enumeration value="shdw19"/>
1359         <xsd:enumeration value="shdw20"/>
1360     </xsd:restriction>
1361 </xsd:simpleType>
1362 <xsd:complexType name="CT_PresetShadowEffect">
1363     <xsd:sequence>
1364         <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1365     </xsd:sequence>
1366     <xsd:attribute name="prst" type="ST_PresetShadowVal" use="required"/>
1367     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1368     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>

```

```

1369 </xsd:complexType>
1370 <xsd:complexType name="CT_ReflectionEffect">
1371     <xsd:attribute name="blurRad" type="ST_PositiveCoordinate" use="optional" default="0"/>
1372     <xsd:attribute name="stA" type="ST_PositiveFixedPercentage" use="optional" default="100%"/>
1373     <xsd:attribute name="stPos" type="ST_PositiveFixedPercentage" use="optional" default="0%"/>
1374     <xsd:attribute name="endA" type="ST_PositiveFixedPercentage" use="optional" default="0%"/>
1375     <xsd:attribute name="endPos" type="ST_PositiveFixedPercentage" use="optional" default="100%"/>
1376     <xsd:attribute name="dist" type="ST_PositiveCoordinate" use="optional" default="0"/>
1377     <xsd:attribute name="dir" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1378     <xsd:attribute name="fadeDir" type="ST_PositiveFixedAngle" use="optional" default="5400000"/>
1379     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1380     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1381     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1382     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1383     <xsd:attribute name="align" type="ST_RectAlignment" use="optional" default="b"/>
1384     <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional" default="true"/>
1385 </xsd:complexType>
1386 <xsd:complexType name="CT_RelativeOffsetEffect">
1387     <xsd:attribute name="tx" type="ST_Percentage" use="optional" default="0%"/>
1388     <xsd:attribute name="ty" type="ST_Percentage" use="optional" default="0%"/>
1389 </xsd:complexType>
1390 <xsd:complexType name="CT_SoftEdgesEffect">
1391     <xsd:attribute name="rad" type="ST_PositiveCoordinate" use="required"/>
1392 </xsd:complexType>
1393 <xsd:complexType name="CT_TintEffect">
1394     <xsd:attribute name="hue" type="ST_PositiveFixedAngle" use="optional" default="0"/>
1395     <xsd:attribute name="amt" type="ST_FixedPercentage" use="optional" default="0%"/>
1396 </xsd:complexType>
1397 <xsd:complexType name="CT_TransformEffect">
1398     <xsd:attribute name="sx" type="ST_Percentage" use="optional" default="100%"/>
1399     <xsd:attribute name="sy" type="ST_Percentage" use="optional" default="100%"/>
1400     <xsd:attribute name="kx" type="ST_FixedAngle" use="optional" default="0"/>
1401     <xsd:attribute name="ky" type="ST_FixedAngle" use="optional" default="0"/>
1402     <xsd:attribute name="tx" type="ST_Coordinate" use="optional" default="0"/>
1403     <xsd:attribute name="ty" type="ST_Coordinate" use="optional" default="0"/>
1404 </xsd:complexType>
1405 <xsd:complexType name="CT_NoFillProperties"/>
1406 <xsd:complexType name="CT_SolidColorFillProperties">
1407     <xsd:sequence>
1408         <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
1409     </xsd:sequence>
1410 </xsd:complexType>
1411 <xsd:complexType name="CT_LinearShadeProperties">
1412     <xsd:attribute name="ang" type="ST_PositiveFixedAngle" use="optional"/>
1413     <xsd:attribute name="scaled" type="xsd:boolean" use="optional"/>
1414 </xsd:complexType>
1415 <xsd:simpleType name="ST_PathShadeType">
1416     <xsd:restriction base="xsd:token">
1417         <xsd:enumeration value="shape"/>
1418         <xsd:enumeration value="circle"/>
1419         <xsd:enumeration value="rect"/>
1420     </xsd:restriction>
1421 </xsd:simpleType>

```

```

1422 <xsd:complexType name="CT_PathShadeProperties">
1423   <xsd:sequence>
1424     <xsd:element name="fillToRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1425   </xsd:sequence>
1426   <xsd:attribute name="path" type="ST_PathShadeType" use="optional"/>
1427 </xsd:complexType>
1428 <xsd:group name="EG_ShadeProperties">
1429   <xsd:choice>
1430     <xsd:element name="lin" type="CT_LinearShadeProperties" minOccurs="1" maxOccurs="1"/>
1431     <xsd:element name="path" type="CT_PathShadeProperties" minOccurs="1" maxOccurs="1"/>
1432   </xsd:choice>
1433 </xsd:group>
1434 <xsd:simpleType name="ST_TileFlipMode">
1435   <xsd:restriction base="xsd:token">
1436     <xsd:enumeration value="none"/>
1437     <xsd:enumeration value="x"/>
1438     <xsd:enumeration value="y"/>
1439     <xsd:enumeration value="xy"/>
1440   </xsd:restriction>
1441 </xsd:simpleType>
1442 <xsd:complexType name="CT_GradientStop">
1443   <xsd:sequence>
1444     <xsd:group ref="EG_ColorChoice" minOccurs="1" maxOccurs="1"/>
1445   </xsd:sequence>
1446   <xsd:attribute name="pos" type="ST_PositiveFixedPercentage" use="required"/>
1447 </xsd:complexType>
1448 <xsd:complexType name="CT_GradientStopList">
1449   <xsd:sequence>
1450     <xsd:element name="gs" type="CT_GradientStop" minOccurs="2" maxOccurs="unbounded"/>
1451   </xsd:sequence>
1452 </xsd:complexType>
1453 <xsd:complexType name="CT_GradientFillProperties">
1454   <xsd:sequence>
1455     <xsd:element name="gsList" type="CT_GradientStopList" minOccurs="0" maxOccurs="1"/>
1456     <xsd:group ref="EG_ShadeProperties" minOccurs="0" maxOccurs="1"/>
1457     <xsd:element name="tileRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1458   </xsd:sequence>
1459   <xsd:attribute name="flip" type="ST_TileFlipMode" use="optional"/>
1460   <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional"/>
1461 </xsd:complexType>
1462 <xsd:complexType name="CT_TileInfoProperties">
1463   <xsd:attribute name="tx" type="ST_Coordinate" use="optional"/>
1464   <xsd:attribute name="ty" type="ST_Coordinate" use="optional"/>
1465   <xsd:attribute name="sx" type="ST_Percentage" use="optional"/>
1466   <xsd:attribute name="sy" type="ST_Percentage" use="optional"/>
1467   <xsd:attribute name="flip" type="ST_TileFlipMode" use="optional"/>
1468   <xsd:attribute name="align" type="ST_RectAlignment" use="optional"/>
1469 </xsd:complexType>
1470 <xsd:complexType name="CT_StretchInfoProperties">
1471   <xsd:sequence>
1472     <xsd:element name="fillRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1473   </xsd:sequence>
1474 </xsd:complexType>

```

```

1475 <xsd:group name="EG_FillModeProperties">
1476   <xsd:choice>
1477     <xsd:element name="tile" type="CT_TileInfoProperties" minOccurs="1" maxOccurs="1"/>
1478     <xsd:element name="stretch" type="CT_StretchInfoProperties" minOccurs="1" maxOccurs="1"/>
1479   </xsd:choice>
1480 </xsd:group>
1481 <xsd:simpleType name="ST_BlipCompression">
1482   <xsd:restriction base="xsd:token">
1483     <xsd:enumeration value="email"/>
1484     <xsd:enumeration value="screen"/>
1485     <xsd:enumeration value="print"/>
1486     <xsd:enumeration value="hqprint"/>
1487     <xsd:enumeration value="none"/>
1488   </xsd:restriction>
1489 </xsd:simpleType>
1490 <xsd:complexType name="CT_Blip">
1491   <xsd:sequence>
1492     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1493       <xsd:element name="alphaBiLevel" type="CT_AlphaBiLevelEffect" minOccurs="1"
1494         maxOccurs="1"/>
1495       <xsd:element name="alphaCeiling" type="CT_AlphaCeilingEffect" minOccurs="1"
1496         maxOccurs="1"/>
1497       <xsd:element name="alphaFloor" type="CT_AlphaFloorEffect" minOccurs="1" maxOccurs="1"/>
1498       <xsd:element name="alphaInv" type="CT_AlphaInverseEffect" minOccurs="1" maxOccurs="1"/>
1499       <xsd:element name="alphaMod" type="CT_AlphaModulateEffect" minOccurs="1"
1500         maxOccurs="1"/>
1501       <xsd:element name="alphaModFix" type="CT_AlphaModulateFixedEffect" minOccurs="1"
1502         maxOccurs="1"/>
1503       <xsd:element name="alphaRepl" type="CT_AlphaReplaceEffect" minOccurs="1"
1504         maxOccurs="1"/>
1505       <xsd:element name="biLevel" type="CT_BiLevelEffect" minOccurs="1" maxOccurs="1"/>
1506       <xsd:element name="blur" type="CT_BlurEffect" minOccurs="1" maxOccurs="1"/>
1507       <xsd:element name="clrChange" type="CT_ColorChangeEffect" minOccurs="1" maxOccurs="1"/>
1508       <xsd:element name="clrRepl" type="CT_ColorReplaceEffect" minOccurs="1" maxOccurs="1"/>
1509       <xsd:element name="duotone" type="CT_DuotoneEffect" minOccurs="1" maxOccurs="1"/>
1510       <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="1"
1511         maxOccurs="1"/>
1512       <xsd:element name="grayscale" type="CT_GrayscaleEffect" minOccurs="1" maxOccurs="1"/>
1513       <xsd:element name="hsl" type="CT_HSLEffect" minOccurs="1" maxOccurs="1"/>
1514       <xsd:element name="lum" type="CT_LuminanceEffect" minOccurs="1" maxOccurs="1"/>
1515       <xsd:element name="tint" type="CT_TintEffect" minOccurs="1" maxOccurs="1"/>
1516     </xsd:choice>
1517     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
1518   </xsd:sequence>
1519   <xsd:attributeGroup ref="AG_Blob"/>
1520   <xsd:attribute name="cstate" type="ST_BlipCompression" use="optional" default="none"/>
1521 </xsd:complexType>
1522 <xsd:complexType name="CT_BlipFillProperties">
1523   <xsd:sequence>
1524     <xsd:element name="blip" type="CT_Blip" minOccurs="0" maxOccurs="1"/>
1525     <xsd:element name="srcRect" type="CT_RelativeRect" minOccurs="0" maxOccurs="1"/>
1526     <xsd:group ref="EG_FillModeProperties" minOccurs="0" maxOccurs="1"/>
1527   </xsd:sequence>

```

```

1528     <xsd:attribute name="dpi" type="xsd:unsignedInt" use="optional"/>
1529     <xsd:attribute name="rotWithShape" type="xsd:boolean" use="optional"/>
1530 </xsd:complexType>
1531 <xsd:simpleType name="ST_PresetPatternVal">
1532     <xsd:restriction base="xsd:token">
1533         <xsd:enumeration value="pct5"/>
1534         <xsd:enumeration value="pct10"/>
1535         <xsd:enumeration value="pct20"/>
1536         <xsd:enumeration value="pct25"/>
1537         <xsd:enumeration value="pct30"/>
1538         <xsd:enumeration value="pct40"/>
1539         <xsd:enumeration value="pct50"/>
1540         <xsd:enumeration value="pct60"/>
1541         <xsd:enumeration value="pct70"/>
1542         <xsd:enumeration value="pct75"/>
1543         <xsd:enumeration value="pct80"/>
1544         <xsd:enumeration value="pct90"/>
1545         <xsd:enumeration value="horz"/>
1546         <xsd:enumeration value="vert"/>
1547         <xsd:enumeration value="ltHorz"/>
1548         <xsd:enumeration value="ltVert"/>
1549         <xsd:enumeration value="dkHorz"/>
1550         <xsd:enumeration value="dkVert"/>
1551         <xsd:enumeration value="narHorz"/>
1552         <xsd:enumeration value="narVert"/>
1553         <xsd:enumeration value="dashHorz"/>
1554         <xsd:enumeration value="dashVert"/>
1555         <xsd:enumeration value="cross"/>
1556         <xsd:enumeration value="dnDiag"/>
1557         <xsd:enumeration value="upDiag"/>
1558         <xsd:enumeration value="ltDnDiag"/>
1559         <xsd:enumeration value="ltUpDiag"/>
1560         <xsd:enumeration value="dkDnDiag"/>
1561         <xsd:enumeration value="dkUpDiag"/>
1562         <xsd:enumeration value="wdDnDiag"/>
1563         <xsd:enumeration value="wdUpDiag"/>
1564         <xsd:enumeration value="dashDnDiag"/>
1565         <xsd:enumeration value="dashUpDiag"/>
1566         <xsd:enumeration value="diagCross"/>
1567         <xsd:enumeration value="smCheck"/>
1568         <xsd:enumeration value="lgCheck"/>
1569         <xsd:enumeration value="smGrid"/>
1570         <xsd:enumeration value="lgGrid"/>
1571         <xsd:enumeration value="dotGrid"/>
1572         <xsd:enumeration value="smConfetti"/>
1573         <xsd:enumeration value="lgConfetti"/>
1574         <xsd:enumeration value="horzBrick"/>
1575         <xsd:enumeration value="diagBrick"/>
1576         <xsd:enumeration value="solidDmnd"/>
1577         <xsd:enumeration value="openDmnd"/>
1578         <xsd:enumeration value="dotDmnd"/>
1579         <xsd:enumeration value="plaid"/>
1580         <xsd:enumeration value="sphere"/>

```

```

1581     <xsd:enumeration value="weave"/>
1582     <xsd:enumeration value="divot"/>
1583     <xsd:enumeration value="shingle"/>
1584     <xsd:enumeration value="wave"/>
1585     <xsd:enumeration value="trellis"/>
1586     <xsd:enumeration value="zigZag"/>
1587   </xsd:restriction>
1588 </xsd:simpleType>
1589 <xsd:complexType name="CT_PatternFillProperties">
1590   <xsd:sequence>
1591     <xsd:element name="fgClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1592     <xsd:element name="bgClr" type="CT_Color" minOccurs="0" maxOccurs="1"/>
1593   </xsd:sequence>
1594   <xsd:attribute name="prst" type="ST_PresetPatternVal" use="optional"/>
1595 </xsd:complexType>
1596 <xsd:complexType name="CT_GroupFillProperties"/>
1597 <xsd:group name="EG_FillProperties">
1598   <xsd:choice>
1599     <xsd:element name="noFill" type="CT_NoFillProperties" minOccurs="1" maxOccurs="1"/>
1600     <xsd:element name="solidFill" type="CT_SolidColorFillProperties" minOccurs="1"
1601       maxOccurs="1"/>
1602     <xsd:element name="gradFill" type="CT_GradientFillProperties" minOccurs="1"
1603       maxOccurs="1"/>
1604     <xsd:element name="blipFill" type="CT_BlipFillProperties" minOccurs="1" maxOccurs="1"/>
1605     <xsd:element name="pattFill" type="CT_PatternFillProperties" minOccurs="1" maxOccurs="1"/>
1606     <xsd:element name="grpFill" type="CT_GroupFillProperties" minOccurs="1" maxOccurs="1"/>
1607   </xsd:choice>
1608 </xsd:group>
1609 <xsd:complexType name="CT_FillProperties">
1610   <xsd:sequence>
1611     <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1612   </xsd:sequence>
1613 </xsd:complexType>
1614 <xsd:complexType name="CT_FillEffect">
1615   <xsd:sequence>
1616     <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1617   </xsd:sequence>
1618 </xsd:complexType>
1619 <xsd:simpleType name="ST_BlendMode">
1620   <xsd:restriction base="xsd:token">
1621     <xsd:enumeration value="over"/>
1622     <xsd:enumeration value="mult"/>
1623     <xsd:enumeration value="screen"/>
1624     <xsd:enumeration value="darken"/>
1625     <xsd:enumeration value="lighten"/>
1626   </xsd:restriction>
1627 </xsd:simpleType>
1628 <xsd:complexType name="CT_FillOverlayEffect">
1629   <xsd:sequence>
1630     <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
1631   </xsd:sequence>
1632   <xsd:attribute name="blend" type="ST_BlendMode" use="required"/>
1633 </xsd:complexType>

```



```

1634 <xsd:complexType name="CT_EffectReference">
1635   <xsd:attribute name="ref" type="xsd:token"/>
1636 </xsd:complexType>
1637 <xsd:group name="EG_Effect">
1638   <xsd:choice>
1639     <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1640     <xsd:element name="effect" type="CT_EffectReference" minOccurs="1" maxOccurs="1"/>
1641     <xsd:element name="alphaBiLevel" type="CT_AlphaBiLevelEffect" minOccurs="1"
1642       maxOccurs="1"/>
1643     <xsd:element name="alphaCeiling" type="CT_AlphaCeilingEffect" minOccurs="1"
1644       maxOccurs="1"/>
1645     <xsd:element name="alphaFloor" type="CT_AlphaFloorEffect" minOccurs="1" maxOccurs="1"/>
1646     <xsd:element name="alphaInv" type="CT_AlphaInverseEffect" minOccurs="1" maxOccurs="1"/>
1647     <xsd:element name="alphaMod" type="CT_AlphaModulateEffect" minOccurs="1" maxOccurs="1"/>
1648     <xsd:element name="alphaModFix" type="CT_AlphaModulateFixedEffect" minOccurs="1"
1649       maxOccurs="1"/>
1650     <xsd:element name="alphaOutset" type="CT_AlphaOutsetEffect" minOccurs="1" maxOccurs="1"/>
1651     <xsd:element name="alphaRepl" type="CT_AlphaReplaceEffect" minOccurs="1" maxOccurs="1"/>
1652     <xsd:element name="biLevel" type="CT_BiLevelEffect" minOccurs="1" maxOccurs="1"/>
1653     <xsd:element name="blend" type="CT_BlendEffect" minOccurs="1" maxOccurs="1"/>
1654     <xsd:element name="blur" type="CT_BlurEffect" minOccurs="1" maxOccurs="1"/>
1655     <xsd:element name="clrChange" type="CT_ColorChangeEffect" minOccurs="1" maxOccurs="1"/>
1656     <xsd:element name="clrRepl" type="CT_ColorReplaceEffect" minOccurs="1" maxOccurs="1"/>
1657     <xsd:element name="duotone" type="CT_DuotoneEffect" minOccurs="1" maxOccurs="1"/>
1658     <xsd:element name="fill" type="CT_FillEffect" minOccurs="1" maxOccurs="1"/>
1659     <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="1" maxOccurs="1"/>
1660     <xsd:element name="glow" type="CT_GlowEffect" minOccurs="1" maxOccurs="1"/>
1661     <xsd:element name="grayscale" type="CT_GrayscaleEffect" minOccurs="1" maxOccurs="1"/>
1662     <xsd:element name="hsl" type="CT_HSLEffect" minOccurs="1" maxOccurs="1"/>
1663     <xsd:element name="innerShdw" type="CT_InnerShadowEffect" minOccurs="1" maxOccurs="1"/>
1664     <xsd:element name="lum" type="CT_LuminanceEffect" minOccurs="1" maxOccurs="1"/>
1665     <xsd:element name="outerShdw" type="CT_OuterShadowEffect" minOccurs="1" maxOccurs="1"/>
1666     <xsd:element name="prstShdw" type="CT_PresetShadowEffect" minOccurs="1" maxOccurs="1"/>
1667     <xsd:element name="reflection" type="CT_ReflectionEffect" minOccurs="1" maxOccurs="1"/>
1668     <xsd:element name="relOff" type="CT_RelativeOffsetEffect" minOccurs="1" maxOccurs="1"/>
1669     <xsd:element name="softEdge" type="CT_SoftEdgesEffect" minOccurs="1" maxOccurs="1"/>
1670     <xsd:element name="tint" type="CT_TintEffect" minOccurs="1" maxOccurs="1"/>
1671     <xsd:element name="xfrm" type="CT_TransformEffect" minOccurs="1" maxOccurs="1"/>
1672   </xsd:choice>
1673 </xsd:group>
1674 <xsd:simpleType name="ST_EffectContainerType">
1675   <xsd:restriction base="xsd:token">
1676     <xsd:enumeration value="sib"/>
1677     <xsd:enumeration value="tree"/>
1678   </xsd:restriction>
1679 </xsd:simpleType>
1680 <xsd:complexType name="CT_EffectContainer">
1681   <xsd:group ref="EG_Effect" minOccurs="0" maxOccurs="unbounded"/>
1682   <xsd:attribute name="type" type="ST_EffectContainerType" use="optional" default="sib"/>
1683   <xsd:attribute name="name" type="xsd:token" use="optional"/>
1684 </xsd:complexType>
1685 <xsd:complexType name="CT_AlphaModulateEffect">
1686   <xsd:sequence>

```

```

1687     <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1688   </xsd:sequence>
1689 </xsd:complexType>
1690 <xsd:complexType name="CT_BlendEffect">
1691   <xsd:sequence>
1692     <xsd:element name="cont" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1693   </xsd:sequence>
1694   <xsd:attribute name="blend" type="ST_BlendMode" use="required"/>
1695 </xsd:complexType>
1696 <xsd:complexType name="CT_EffectList">
1697   <xsd:sequence>
1698     <xsd:element name="blur" type="CT_BlurEffect" minOccurs="0" maxOccurs="1"/>
1699     <xsd:element name="fillOverlay" type="CT_FillOverlayEffect" minOccurs="0" maxOccurs="1"/>
1700     <xsd:element name="glow" type="CT_GlowEffect" minOccurs="0" maxOccurs="1"/>
1701     <xsd:element name="innerShdw" type="CT_InnerShadowEffect" minOccurs="0" maxOccurs="1"/>
1702     <xsd:element name="outerShdw" type="CT_OuterShadowEffect" minOccurs="0" maxOccurs="1"/>
1703     <xsd:element name="prstShdw" type="CT_PresetShadowEffect" minOccurs="0" maxOccurs="1"/>
1704     <xsd:element name="reflection" type="CT_ReflectionEffect" minOccurs="0" maxOccurs="1"/>
1705     <xsd:element name="softEdge" type="CT_SoftEdgesEffect" minOccurs="0" maxOccurs="1"/>
1706   </xsd:sequence>
1707 </xsd:complexType>
1708 <xsd:group name="EG_EffectProperties">
1709   <xsd:choice>
1710     <xsd:element name="effectList" type="CT_EffectList" minOccurs="1" maxOccurs="1"/>
1711     <xsd:element name="effectDag" type="CT_EffectContainer" minOccurs="1" maxOccurs="1"/>
1712   </xsd:choice>
1713 </xsd:group>
1714 <xsd:complexType name="CT_EffectProperties">
1715   <xsd:sequence>
1716     <xsd:group ref="EG_EffectProperties" minOccurs="1" maxOccurs="1"/>
1717   </xsd:sequence>
1718 </xsd:complexType>
1719 <xsd:element name="blip" type="CT_Blip"/>
1720 <xsd:simpleType name="ST_ShapeType">
1721   <xsd:restriction base="xsd:token">
1722     <xsd:enumeration value="line"/>
1723     <xsd:enumeration value="lineInv"/>
1724     <xsd:enumeration value="triangle"/>
1725     <xsd:enumeration value="rtTriangle"/>
1726     <xsd:enumeration value="rect"/>
1727     <xsd:enumeration value="diamond"/>
1728     <xsd:enumeration value="parallelogram"/>
1729     <xsd:enumeration value="trapezoid"/>
1730     <xsd:enumeration value="nonIsoscelesTrapezoid"/>
1731     <xsd:enumeration value="pentagon"/>
1732     <xsd:enumeration value="hexagon"/>
1733     <xsd:enumeration value="heptagon"/>
1734     <xsd:enumeration value="octagon"/>
1735     <xsd:enumeration value="decagon"/>
1736     <xsd:enumeration value="dodecagon"/>
1737     <xsd:enumeration value="star4"/>
1738     <xsd:enumeration value="star5"/>
1739     <xsd:enumeration value="star6"/>

```

```

1740 <xsd:enumeration value="star7"/>
1741 <xsd:enumeration value="star8"/>
1742 <xsd:enumeration value="star10"/>
1743 <xsd:enumeration value="star12"/>
1744 <xsd:enumeration value="star16"/>
1745 <xsd:enumeration value="star24"/>
1746 <xsd:enumeration value="star32"/>
1747 <xsd:enumeration value="roundRect"/>
1748 <xsd:enumeration value="round1Rect"/>
1749 <xsd:enumeration value="round2SameRect"/>
1750 <xsd:enumeration value="round2DiagRect"/>
1751 <xsd:enumeration value="snipRoundRect"/>
1752 <xsd:enumeration value="snip1Rect"/>
1753 <xsd:enumeration value="snip2SameRect"/>
1754 <xsd:enumeration value="snip2DiagRect"/>
1755 <xsd:enumeration value="plaque"/>
1756 <xsd:enumeration value="ellipse"/>
1757 <xsd:enumeration value="teardrop"/>
1758 <xsd:enumeration value="homePlate"/>
1759 <xsd:enumeration value="chevron"/>
1760 <xsd:enumeration value="pieWedge"/>
1761 <xsd:enumeration value="pie"/>
1762 <xsd:enumeration value="blockArc"/>
1763 <xsd:enumeration value="donut"/>
1764 <xsd:enumeration value="noSmoking"/>
1765 <xsd:enumeration value="rightArrow"/>
1766 <xsd:enumeration value="leftArrow"/>
1767 <xsd:enumeration value="upArrow"/>
1768 <xsd:enumeration value="downArrow"/>
1769 <xsd:enumeration value="stripedRightArrow"/>
1770 <xsd:enumeration value="notchedRightArrow"/>
1771 <xsd:enumeration value="bentUpArrow"/>
1772 <xsd:enumeration value="leftRightArrow"/>
1773 <xsd:enumeration value="upDownArrow"/>
1774 <xsd:enumeration value="leftUpArrow"/>
1775 <xsd:enumeration value="leftRightUpArrow"/>
1776 <xsd:enumeration value="quadArrow"/>
1777 <xsd:enumeration value="leftArrowCallout"/>
1778 <xsd:enumeration value="rightArrowCallout"/>
1779 <xsd:enumeration value="upArrowCallout"/>
1780 <xsd:enumeration value="downArrowCallout"/>
1781 <xsd:enumeration value="leftRightArrowCallout"/>
1782 <xsd:enumeration value="upDownArrowCallout"/>
1783 <xsd:enumeration value="quadArrowCallout"/>
1784 <xsd:enumeration value="bentArrow"/>
1785 <xsd:enumeration value="uturnArrow"/>
1786 <xsd:enumeration value="circularArrow"/>
1787 <xsd:enumeration value="leftCircularArrow"/>
1788 <xsd:enumeration value="leftRightCircularArrow"/>
1789 <xsd:enumeration value="curvedRightArrow"/>
1790 <xsd:enumeration value="curvedLeftArrow"/>
1791 <xsd:enumeration value="curvedUpArrow"/>
1792 <xsd:enumeration value="curvedDownArrow"/>

```

```

1793     <xsd:enumeration value="swooshArrow"/>
1794     <xsd:enumeration value="cube"/>
1795     <xsd:enumeration value="can"/>
1796     <xsd:enumeration value="lightningBolt"/>
1797     <xsd:enumeration value="heart"/>
1798     <xsd:enumeration value="sun"/>
1799     <xsd:enumeration value="moon"/>
1800     <xsd:enumeration value="smileyFace"/>
1801     <xsd:enumeration value="irregularSeal1"/>
1802     <xsd:enumeration value="irregularSeal2"/>
1803     <xsd:enumeration value="foldedCorner"/>
1804     <xsd:enumeration value="bevel"/>
1805     <xsd:enumeration value="frame"/>
1806     <xsd:enumeration value="halfFrame"/>
1807     <xsd:enumeration value="corner"/>
1808     <xsd:enumeration value="diagStripe"/>
1809     <xsd:enumeration value="chord"/>
1810     <xsd:enumeration value="arc"/>
1811     <xsd:enumeration value="leftBracket"/>
1812     <xsd:enumeration value="rightBracket"/>
1813     <xsd:enumeration value="leftBrace"/>
1814     <xsd:enumeration value="rightBrace"/>
1815     <xsd:enumeration value="bracketPair"/>
1816     <xsd:enumeration value="bracePair"/>
1817     <xsd:enumeration value="straightConnector1"/>
1818     <xsd:enumeration value="bentConnector2"/>
1819     <xsd:enumeration value="bentConnector3"/>
1820     <xsd:enumeration value="bentConnector4"/>
1821     <xsd:enumeration value="bentConnector5"/>
1822     <xsd:enumeration value="curvedConnector2"/>
1823     <xsd:enumeration value="curvedConnector3"/>
1824     <xsd:enumeration value="curvedConnector4"/>
1825     <xsd:enumeration value="curvedConnector5"/>
1826     <xsd:enumeration value="callout1"/>
1827     <xsd:enumeration value="callout2"/>
1828     <xsd:enumeration value="callout3"/>
1829     <xsd:enumeration value="accentCallout1"/>
1830     <xsd:enumeration value="accentCallout2"/>
1831     <xsd:enumeration value="accentCallout3"/>
1832     <xsd:enumeration value="borderCallout1"/>
1833     <xsd:enumeration value="borderCallout2"/>
1834     <xsd:enumeration value="borderCallout3"/>
1835     <xsd:enumeration value="accentBorderCallout1"/>
1836     <xsd:enumeration value="accentBorderCallout2"/>
1837     <xsd:enumeration value="accentBorderCallout3"/>
1838     <xsd:enumeration value="wedgeRectCallout"/>
1839     <xsd:enumeration value="wedgeRoundRectCallout"/>
1840     <xsd:enumeration value="wedgeEllipseCallout"/>
1841     <xsd:enumeration value="cloudCallout"/>
1842     <xsd:enumeration value="cloud"/>
1843     <xsd:enumeration value="ribbon"/>
1844     <xsd:enumeration value="ribbon2"/>
1845     <xsd:enumeration value="ellipseRibbon"/>

```

```

1846 <xsd:enumeration value="ellipseRibbon2"/>
1847 <xsd:enumeration value="leftRightRibbon"/>
1848 <xsd:enumeration value="verticalScroll"/>
1849 <xsd:enumeration value="horizontalScroll"/>
1850 <xsd:enumeration value="wave"/>
1851 <xsd:enumeration value="doubleWave"/>
1852 <xsd:enumeration value="plus"/>
1853 <xsd:enumeration value="flowChartProcess"/>
1854 <xsd:enumeration value="flowChartDecision"/>
1855 <xsd:enumeration value="flowChartInputOutput"/>
1856 <xsd:enumeration value="flowChartPredefinedProcess"/>
1857 <xsd:enumeration value="flowChartInternalStorage"/>
1858 <xsd:enumeration value="flowChartDocument"/>
1859 <xsd:enumeration value="flowChartMultidocument"/>
1860 <xsd:enumeration value="flowChartTerminator"/>
1861 <xsd:enumeration value="flowChartPreparation"/>
1862 <xsd:enumeration value="flowChartManualInput"/>
1863 <xsd:enumeration value="flowChartManualOperation"/>
1864 <xsd:enumeration value="flowChartConnector"/>
1865 <xsd:enumeration value="flowChartPunchedCard"/>
1866 <xsd:enumeration value="flowChartPunchedTape"/>
1867 <xsd:enumeration value="flowChartSummingJunction"/>
1868 <xsd:enumeration value="flowChartOr"/>
1869 <xsd:enumeration value="flowChartCollate"/>
1870 <xsd:enumeration value="flowChartSort"/>
1871 <xsd:enumeration value="flowChartExtract"/>
1872 <xsd:enumeration value="flowChartMerge"/>
1873 <xsd:enumeration value="flowChartOfflineStorage"/>
1874 <xsd:enumeration value="flowChartOnlineStorage"/>
1875 <xsd:enumeration value="flowChartMagneticTape"/>
1876 <xsd:enumeration value="flowChartMagneticDisk"/>
1877 <xsd:enumeration value="flowChartMagneticDrum"/>
1878 <xsd:enumeration value="flowChartDisplay"/>
1879 <xsd:enumeration value="flowChartDelay"/>
1880 <xsd:enumeration value="flowChartAlternateProcess"/>
1881 <xsd:enumeration value="flowChartOffpageConnector"/>
1882 <xsd:enumeration value="actionButtonBlank"/>
1883 <xsd:enumeration value="actionButtonHome"/>
1884 <xsd:enumeration value="actionButtonHelp"/>
1885 <xsd:enumeration value="actionButtonInformation"/>
1886 <xsd:enumeration value="actionButtonForwardNext"/>
1887 <xsd:enumeration value="actionButtonBackPrevious"/>
1888 <xsd:enumeration value="actionButtonEnd"/>
1889 <xsd:enumeration value="actionButtonBeginning"/>
1890 <xsd:enumeration value="actionButtonReturn"/>
1891 <xsd:enumeration value="actionButtonDocument"/>
1892 <xsd:enumeration value="actionButtonSound"/>
1893 <xsd:enumeration value="actionButtonMovie"/>
1894 <xsd:enumeration value="gear6"/>
1895 <xsd:enumeration value="gear9"/>
1896 <xsd:enumeration value="funnel"/>
1897 <xsd:enumeration value="mathPlus"/>
1898 <xsd:enumeration value="mathMinus"/>

```

```

1899     <xsd:enumeration value="mathMultiply"/>
1900     <xsd:enumeration value="mathDivide"/>
1901     <xsd:enumeration value="mathEqual"/>
1902     <xsd:enumeration value="mathNotEqual"/>
1903     <xsd:enumeration value="cornerTabs"/>
1904     <xsd:enumeration value="squareTabs"/>
1905     <xsd:enumeration value="plaqueTabs"/>
1906     <xsd:enumeration value="chartX"/>
1907     <xsd:enumeration value="chartStar"/>
1908     <xsd:enumeration value="chartPlus"/>
1909 </xsd:restriction>
1910 </xsd:simpleType>
1911 <xsd:simpleType name="ST_TextShapeType">
1912     <xsd:restriction base="xsd:token">
1913         <xsd:enumeration value="textNoShape"/>
1914         <xsd:enumeration value="textPlain"/>
1915         <xsd:enumeration value="textStop"/>
1916         <xsd:enumeration value="textTriangle"/>
1917         <xsd:enumeration value="textTriangleInverted"/>
1918         <xsd:enumeration value="textChevron"/>
1919         <xsd:enumeration value="textChevronInverted"/>
1920         <xsd:enumeration value="textRingInside"/>
1921         <xsd:enumeration value="textRingOutside"/>
1922         <xsd:enumeration value="textArchUp"/>
1923         <xsd:enumeration value="textArchDown"/>
1924         <xsd:enumeration value="textCircle"/>
1925         <xsd:enumeration value="textButton"/>
1926         <xsd:enumeration value="textArchUpPour"/>
1927         <xsd:enumeration value="textArchDownPour"/>
1928         <xsd:enumeration value="textCirclePour"/>
1929         <xsd:enumeration value="textButtonPour"/>
1930         <xsd:enumeration value="textCurveUp"/>
1931         <xsd:enumeration value="textCurveDown"/>
1932         <xsd:enumeration value="textCanUp"/>
1933         <xsd:enumeration value="textCanDown"/>
1934         <xsd:enumeration value="textWave1"/>
1935         <xsd:enumeration value="textWave2"/>
1936         <xsd:enumeration value="textDoubleWave1"/>
1937         <xsd:enumeration value="textWave4"/>
1938         <xsd:enumeration value="textInflate"/>
1939         <xsd:enumeration value="textDeflate"/>
1940         <xsd:enumeration value="textInflateBottom"/>
1941         <xsd:enumeration value="textDeflateBottom"/>
1942         <xsd:enumeration value="textInflateTop"/>
1943         <xsd:enumeration value="textDeflateTop"/>
1944         <xsd:enumeration value="textDeflateInflate"/>
1945         <xsd:enumeration value="textDeflateInflateDeflate"/>
1946         <xsd:enumeration value="textFadeRight"/>
1947         <xsd:enumeration value="textFadeLeft"/>
1948         <xsd:enumeration value="textFadeUp"/>
1949         <xsd:enumeration value="textFadeDown"/>
1950         <xsd:enumeration value="textSlantUp"/>
1951         <xsd:enumeration value="textSlantDown"/>

```

```

1952         <xsd:enumeration value="textCascadeUp"/>
1953         <xsd:enumeration value="textCascadeDown"/>
1954     </xsd:restriction>
1955 </xsd:simpleType>
1956 <xsd:simpleType name="ST_GeomGuideName">
1957     <xsd:restriction base="xsd:token"/>
1958 </xsd:simpleType>
1959 <xsd:simpleType name="ST_GeomGuideFormula">
1960     <xsd:restriction base="xsd:string"/>
1961 </xsd:simpleType>
1962 <xsd:complexType name="CT_GeomGuide">
1963     <xsd:attribute name="name" type="ST_GeomGuideName" use="required"/>
1964     <xsd:attribute name="fmla" type="ST_GeomGuideFormula" use="required"/>
1965 </xsd:complexType>
1966 <xsd:complexType name="CT_GeomGuideList">
1967     <xsd:sequence>
1968         <xsd:element name="gd" type="CT_GeomGuide" minOccurs="0" maxOccurs="unbounded"/>
1969     </xsd:sequence>
1970 </xsd:complexType>
1971 <xsd:simpleType name="ST_AdjCoordinate">
1972     <xsd:union memberTypes="ST_Coordinate ST_GeomGuideName"/>
1973 </xsd:simpleType>
1974 <xsd:simpleType name="ST_AdjAngle">
1975     <xsd:union memberTypes="ST_Angle ST_GeomGuideName"/>
1976 </xsd:simpleType>
1977 <xsd:complexType name="CT_AdjPoint2D">
1978     <xsd:attribute name="x" type="ST_AdjCoordinate" use="required"/>
1979     <xsd:attribute name="y" type="ST_AdjCoordinate" use="required"/>
1980 </xsd:complexType>
1981 <xsd:complexType name="CT_GeomRect">
1982     <xsd:attribute name="l" type="ST_AdjCoordinate" use="required"/>
1983     <xsd:attribute name="t" type="ST_AdjCoordinate" use="required"/>
1984     <xsd:attribute name="r" type="ST_AdjCoordinate" use="required"/>
1985     <xsd:attribute name="b" type="ST_AdjCoordinate" use="required"/>
1986 </xsd:complexType>
1987 <xsd:complexType name="CT_XYAdjustHandle">
1988     <xsd:sequence>
1989         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
1990     </xsd:sequence>
1991     <xsd:attribute name="gdRefX" type="ST_GeomGuideName" use="optional"/>
1992     <xsd:attribute name="minX" type="ST_AdjCoordinate" use="optional"/>
1993     <xsd:attribute name="maxX" type="ST_AdjCoordinate" use="optional"/>
1994     <xsd:attribute name="gdRefY" type="ST_GeomGuideName" use="optional"/>
1995     <xsd:attribute name="minY" type="ST_AdjCoordinate" use="optional"/>
1996     <xsd:attribute name="maxY" type="ST_AdjCoordinate" use="optional"/>
1997 </xsd:complexType>
1998 <xsd:complexType name="CT_PolarAdjustHandle">
1999     <xsd:sequence>
2000         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2001     </xsd:sequence>
2002     <xsd:attribute name="gdRefR" type="ST_GeomGuideName" use="optional"/>
2003     <xsd:attribute name="minR" type="ST_AdjCoordinate" use="optional"/>
2004     <xsd:attribute name="maxR" type="ST_AdjCoordinate" use="optional"/>

```

```

2005     <xsd:attribute name="gdRefAng" type="ST_GeomGuideName" use="optional"/>
2006     <xsd:attribute name="minAng" type="ST_AdjAngle" use="optional"/>
2007     <xsd:attribute name="maxAng" type="ST_AdjAngle" use="optional"/>
2008 </xsd:complexType>
2009 <xsd:complexType name="CT_ConnectionSite">
2010     <xsd:sequence>
2011         <xsd:element name="pos" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2012     </xsd:sequence>
2013     <xsd:attribute name="ang" type="ST_AdjAngle" use="required"/>
2014 </xsd:complexType>
2015 <xsd:complexType name="CT_AdjustHandleList">
2016     <xsd:choice minOccurs="0" maxOccurs="unbounded">
2017         <xsd:element name="ahXY" type="CT_XYAdjustHandle" minOccurs="1" maxOccurs="1"/>
2018         <xsd:element name="ahPolar" type="CT_PolarAdjustHandle" minOccurs="1" maxOccurs="1"/>
2019     </xsd:choice>
2020 </xsd:complexType>
2021 <xsd:complexType name="CT_ConnectionSiteList">
2022     <xsd:sequence>
2023         <xsd:element name="cxn" type="CT_ConnectionSite" minOccurs="0" maxOccurs="unbounded"/>
2024     </xsd:sequence>
2025 </xsd:complexType>
2026 <xsd:complexType name="CT_Connection">
2027     <xsd:attribute name="id" type="ST_DrawingElementId" use="required"/>
2028     <xsd:attribute name="idx" type="xsd:unsignedInt" use="required"/>
2029 </xsd:complexType>
2030 <xsd:complexType name="CT_Path2DMoveTo">
2031     <xsd:sequence>
2032         <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2033     </xsd:sequence>
2034 </xsd:complexType>
2035 <xsd:complexType name="CT_Path2DLineTo">
2036     <xsd:sequence>
2037         <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="1" maxOccurs="1"/>
2038     </xsd:sequence>
2039 </xsd:complexType>
2040 <xsd:complexType name="CT_Path2DArcTo">
2041     <xsd:attribute name="wR" type="ST_AdjCoordinate" use="required"/>
2042     <xsd:attribute name="hR" type="ST_AdjCoordinate" use="required"/>
2043     <xsd:attribute name="stAng" type="ST_AdjAngle" use="required"/>
2044     <xsd:attribute name="swAng" type="ST_AdjAngle" use="required"/>
2045 </xsd:complexType>
2046 <xsd:complexType name="CT_Path2DQuadBezierTo">
2047     <xsd:sequence>
2048         <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="2" maxOccurs="2"/>
2049     </xsd:sequence>
2050 </xsd:complexType>
2051 <xsd:complexType name="CT_Path2DCubicBezierTo">
2052     <xsd:sequence>
2053         <xsd:element name="pt" type="CT_AdjPoint2D" minOccurs="3" maxOccurs="3"/>
2054     </xsd:sequence>
2055 </xsd:complexType>
2056 <xsd:complexType name="CT_Path2DClose"/>
2057 <xsd:simpleType name="ST_PathFillMode">

```



```

2058     <xsd:restriction base="xsd:token">
2059         <xsd:enumeration value="none"/>
2060         <xsd:enumeration value="norm"/>
2061         <xsd:enumeration value="lighten"/>
2062         <xsd:enumeration value="lightenLess"/>
2063         <xsd:enumeration value="darken"/>
2064         <xsd:enumeration value="darkenLess"/>
2065     </xsd:restriction>
2066 </xsd:simpleType>
2067 <xsd:complexType name="CT_Path2D">
2068     <xsd:choice minOccurs="0" maxOccurs="unbounded">
2069         <xsd:element name="close" type="CT_Path2DClose" minOccurs="1" maxOccurs="1"/>
2070         <xsd:element name="moveTo" type="CT_Path2DMoveTo" minOccurs="1" maxOccurs="1"/>
2071         <xsd:element name="lnTo" type="CT_Path2DLineTo" minOccurs="1" maxOccurs="1"/>
2072         <xsd:element name="arcTo" type="CT_Path2DArcTo" minOccurs="1" maxOccurs="1"/>
2073         <xsd:element name="quadBezTo" type="CT_Path2DQuadBezierTo" minOccurs="1" maxOccurs="1"/>
2074         <xsd:element name="cubicBezTo" type="CT_Path2DCubicBezierTo" minOccurs="1" maxOccurs="1"/>
2075     </xsd:choice>
2076     <xsd:attribute name="w" type="ST_PositiveCoordinate" use="optional" default="0"/>
2077     <xsd:attribute name="h" type="ST_PositiveCoordinate" use="optional" default="0"/>
2078     <xsd:attribute name="fill" type="ST_PathFillMode" use="optional" default="norm"/>
2079     <xsd:attribute name="stroke" type="xsd:boolean" use="optional" default="true"/>
2080     <xsd:attribute name="extrusionOk" type="xsd:boolean" use="optional" default="true"/>
2081 </xsd:complexType>
2082 <xsd:complexType name="CT_Path2DList">
2083     <xsd:sequence>
2084         <xsd:element name="path" type="CT_Path2D" minOccurs="0" maxOccurs="unbounded"/>
2085     </xsd:sequence>
2086 </xsd:complexType>
2087 <xsd:complexType name="CT_PresetGeometry2D">
2088     <xsd:sequence>
2089         <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2090     </xsd:sequence>
2091     <xsd:attribute name="prst" type="ST_ShapeType" use="required"/>
2092 </xsd:complexType>
2093 <xsd:complexType name="CT_PresetTextShape">
2094     <xsd:sequence>
2095         <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2096     </xsd:sequence>
2097     <xsd:attribute name="prst" type="ST_TextShapeType" use="required"/>
2098 </xsd:complexType>
2099 <xsd:complexType name="CT_CustomGeometry2D">
2100     <xsd:sequence>
2101         <xsd:element name="avLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2102         <xsd:element name="gdLst" type="CT_GeomGuideList" minOccurs="0" maxOccurs="1"/>
2103         <xsd:element name="ahLst" type="CT_AdjustHandleList" minOccurs="0" maxOccurs="1"/>
2104         <xsd:element name="cxnLst" type="CT_ConnectionSiteList" minOccurs="0" maxOccurs="1"/>
2105         <xsd:element name="rect" type="CT_GeomRect" minOccurs="0" maxOccurs="1"/>
2106         <xsd:element name="pathLst" type="CT_Path2DList" minOccurs="1" maxOccurs="1"/>
2107     </xsd:sequence>
2108 </xsd:complexType>
2109 <xsd:group name="EG_Geometry">
2110     <xsd:choice>

```

```

2111     <xsd:element name="custGeom" type="CT_CustomGeometry2D" minOccurs="1" maxOccurs="1"/>
2112     <xsd:element name="prstGeom" type="CT_PresetGeometry2D" minOccurs="1" maxOccurs="1"/>
2113   </xsd:choice>
2114 </xsd:group>
2115 <xsd:group name="EG_TextGeometry">
2116   <xsd:choice>
2117     <xsd:element name="custGeom" type="CT_CustomGeometry2D" minOccurs="1" maxOccurs="1"/>
2118     <xsd:element name="prstTxWarp" type="CT_PresetTextShape" minOccurs="1" maxOccurs="1"/>
2119   </xsd:choice>
2120 </xsd:group>
2121 <xsd:simpleType name="ST_LineEndType">
2122   <xsd:restriction base="xsd:token">
2123     <xsd:enumeration value="none"/>
2124     <xsd:enumeration value="triangle"/>
2125     <xsd:enumeration value="stealth"/>
2126     <xsd:enumeration value="diamond"/>
2127     <xsd:enumeration value="oval"/>
2128     <xsd:enumeration value="arrow"/>
2129   </xsd:restriction>
2130 </xsd:simpleType>
2131 <xsd:simpleType name="ST_LineEndWidth">
2132   <xsd:restriction base="xsd:token">
2133     <xsd:enumeration value="sm"/>
2134     <xsd:enumeration value="med"/>
2135     <xsd:enumeration value="lg"/>
2136   </xsd:restriction>
2137 </xsd:simpleType>
2138 <xsd:simpleType name="ST_LineEndLength">
2139   <xsd:restriction base="xsd:token">
2140     <xsd:enumeration value="sm"/>
2141     <xsd:enumeration value="med"/>
2142     <xsd:enumeration value="lg"/>
2143   </xsd:restriction>
2144 </xsd:simpleType>
2145 <xsd:complexType name="CT_LineEndProperties">
2146   <xsd:attribute name="type" type="ST_LineEndType" use="optional"/>
2147   <xsd:attribute name="w" type="ST_LineEndWidth" use="optional"/>
2148   <xsd:attribute name="len" type="ST_LineEndLength" use="optional"/>
2149 </xsd:complexType>
2150 <xsd:group name="EG_LineFillProperties">
2151   <xsd:choice>
2152     <xsd:element name="noFill" type="CT_NoFillProperties" minOccurs="1" maxOccurs="1"/>
2153     <xsd:element name="solidFill" type="CT_SolidColorFillProperties" minOccurs="1"
2154       maxOccurs="1"/>
2155     <xsd:element name="gradFill" type="CT_GradientFillProperties" minOccurs="1"
2156       maxOccurs="1"/>
2157     <xsd:element name="pattFill" type="CT_PatternFillProperties" minOccurs="1" maxOccurs="1"/>
2158   </xsd:choice>
2159 </xsd:group>
2160 <xsd:complexType name="CT_LineJoinBevel"/>
2161 <xsd:complexType name="CT_LineJoinRound"/>
2162 <xsd:complexType name="CT_LineJoinMiterProperties">
2163   <xsd:attribute name="lim" type="ST_PositivePercentage" use="optional"/>

```

```

2164 </xsd:complexType>
2165 <xsd:group name="EG_LineJoinProperties">
2166   <xsd:choice>
2167     <xsd:element name="round" type="CT_LineJoinRound" minOccurs="1" maxOccurs="1"/>
2168     <xsd:element name="bevel" type="CT_LineJoinBevel" minOccurs="1" maxOccurs="1"/>
2169     <xsd:element name="miter" type="CT_LineJoinMiterProperties" minOccurs="1" maxOccurs="1"/>
2170   </xsd:choice>
2171 </xsd:group>
2172 <xsd:simpleType name="ST_PresetLineDashVal">
2173   <xsd:restriction base="xsd:token">
2174     <xsd:enumeration value="solid"/>
2175     <xsd:enumeration value="dot"/>
2176     <xsd:enumeration value="dash"/>
2177     <xsd:enumeration value="lgDash"/>
2178     <xsd:enumeration value="dashDot"/>
2179     <xsd:enumeration value="lgDashDot"/>
2180     <xsd:enumeration value="lgDashDotDot"/>
2181     <xsd:enumeration value="sysDash"/>
2182     <xsd:enumeration value="sysDot"/>
2183     <xsd:enumeration value="sysDashDot"/>
2184     <xsd:enumeration value="sysDashDotDot"/>
2185   </xsd:restriction>
2186 </xsd:simpleType>
2187 <xsd:complexType name="CT_PresetLineDashProperties">
2188   <xsd:attribute name="val" type="ST_PresetLineDashVal" use="optional"/>
2189 </xsd:complexType>
2190 <xsd:complexType name="CT_DashStop">
2191   <xsd:attribute name="d" type="ST_PositivePercentage" use="required"/>
2192   <xsd:attribute name="sp" type="ST_PositivePercentage" use="required"/>
2193 </xsd:complexType>
2194 <xsd:complexType name="CT_DashStopList">
2195   <xsd:sequence>
2196     <xsd:element name="ds" type="CT_DashStop" minOccurs="0" maxOccurs="unbounded"/>
2197   </xsd:sequence>
2198 </xsd:complexType>
2199 <xsd:group name="EG_LineDashProperties">
2200   <xsd:choice>
2201     <xsd:element name="prstDash" type="CT_PresetLineDashProperties" minOccurs="1"
2202       maxOccurs="1"/>
2203     <xsd:element name="custDash" type="CT_DashStopList" minOccurs="1" maxOccurs="1"/>
2204   </xsd:choice>
2205 </xsd:group>
2206 <xsd:simpleType name="ST_LineCap">
2207   <xsd:restriction base="xsd:token">
2208     <xsd:enumeration value="rnd"/>
2209     <xsd:enumeration value="sq"/>
2210     <xsd:enumeration value="flat"/>
2211   </xsd:restriction>
2212 </xsd:simpleType>
2213 <xsd:simpleType name="ST_LineWidth">
2214   <xsd:restriction base="ST_Coordinate32Unqualified">
2215     <xsd:minInclusive value="0"/>
2216     <xsd:maxInclusive value="20116800"/>

```

```

2217     </xsd:restriction>
2218 </xsd:simpleType>
2219 <xsd:simpleType name="ST_PenAlignment">
2220     <xsd:restriction base="xsd:token">
2221         <xsd:enumeration value="ctr"/>
2222         <xsd:enumeration value="in"/>
2223     </xsd:restriction>
2224 </xsd:simpleType>
2225 <xsd:simpleType name="ST_CompoundLine">
2226     <xsd:restriction base="xsd:token">
2227         <xsd:enumeration value="sng"/>
2228         <xsd:enumeration value="dbl"/>
2229         <xsd:enumeration value="thickThin"/>
2230         <xsd:enumeration value="thinThick"/>
2231         <xsd:enumeration value="tri"/>
2232     </xsd:restriction>
2233 </xsd:simpleType>
2234 <xsd:complexType name="CT_LineProperties">
2235     <xsd:sequence>
2236         <xsd:group ref="EG_LineFillProperties" minOccurs="0" maxOccurs="1"/>
2237         <xsd:group ref="EG_LineDashProperties" minOccurs="0" maxOccurs="1"/>
2238         <xsd:group ref="EG_LineJoinProperties" minOccurs="0" maxOccurs="1"/>
2239         <xsd:element name="headEnd" type="CT_LineEndProperties" minOccurs="0" maxOccurs="1"/>
2240         <xsd:element name="tailEnd" type="CT_LineEndProperties" minOccurs="0" maxOccurs="1"/>
2241         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2242     </xsd:sequence>
2243     <xsd:attribute name="w" type="ST_LineWidth" use="optional"/>
2244     <xsd:attribute name="cap" type="ST_LineCap" use="optional"/>
2245     <xsd:attribute name="cmpd" type="ST_CompoundLine" use="optional"/>
2246     <xsd:attribute name="algn" type="ST_PenAlignment" use="optional"/>
2247 </xsd:complexType>
2248 <xsd:simpleType name="ST_ShapeID">
2249     <xsd:restriction base="xsd:token"/>
2250 </xsd:simpleType>
2251 <xsd:complexType name="CT_ShapeProperties">
2252     <xsd:sequence>
2253         <xsd:element name="xfrm" type="CT_Transform2D" minOccurs="0" maxOccurs="1"/>
2254         <xsd:group ref="EG_Geometry" minOccurs="0" maxOccurs="1"/>
2255         <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2256         <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2257         <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2258         <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
2259         <xsd:element name="sp3d" type="CT_Shape3D" minOccurs="0" maxOccurs="1"/>
2260         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2261     </xsd:sequence>
2262     <xsd:attribute name="bwMode" type="ST_BlackWhiteMode" use="optional"/>
2263 </xsd:complexType>
2264 <xsd:complexType name="CT_GroupShapeProperties">
2265     <xsd:sequence>
2266         <xsd:element name="xfrm" type="CT_GroupTransform2D" minOccurs="0" maxOccurs="1"/>
2267         <xsd:group ref="EG_FillProperties" 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="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2271   </xsd:sequence>
2272   <xsd:attribute name="bwMode" type="ST_BlackWhiteMode" use="optional"/>
2273 </xsd:complexType>
2274 <xsd:complexType name="CT_StyleMatrixReference">
2275   <xsd:sequence>
2276     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2277   </xsd:sequence>
2278   <xsd:attribute name="idx" type="ST_StyleMatrixColumnIndex" use="required"/>
2279 </xsd:complexType>
2280 <xsd:complexType name="CT_FontReference">
2281   <xsd:sequence>
2282     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2283   </xsd:sequence>
2284   <xsd:attribute name="idx" type="ST_FontCollectionIndex" use="required"/>
2285 </xsd:complexType>
2286 <xsd:complexType name="CT_ShapeStyle">
2287   <xsd:sequence>
2288     <xsd:element name="lnRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2289     <xsd:element name="fillRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2290     <xsd:element name="effectRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2291     <xsd:element name="fontRef" type="CT_FontReference" minOccurs="1" maxOccurs="1"/>
2292   </xsd:sequence>
2293 </xsd:complexType>
2294 <xsd:complexType name="CT_DefaultShapeDefinition">
2295   <xsd:sequence>
2296     <xsd:element name="spPr" type="CT_ShapeProperties" minOccurs="1" maxOccurs="1"/>
2297     <xsd:element name="bodyPr" type="CT_TextBodyProperties" minOccurs="1" maxOccurs="1"/>
2298     <xsd:element name="lstStyle" type="CT_TextListStyle" minOccurs="1" maxOccurs="1"/>
2299     <xsd:element name="style" type="CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
2300     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2301   </xsd:sequence>
2302 </xsd:complexType>
2303 <xsd:complexType name="CT_ObjectStyleDefaults">
2304   <xsd:sequence>
2305     <xsd:element name="spDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2306     <xsd:element name="lnDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2307     <xsd:element name="txDef" type="CT_DefaultShapeDefinition" minOccurs="0" maxOccurs="1"/>
2308     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2309   </xsd:sequence>
2310 </xsd:complexType>
2311 <xsd:complexType name="CT_EmptyElement"/>
2312 <xsd:complexType name="CT_ColorMapping">
2313   <xsd:sequence>
2314     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2315   </xsd:sequence>
2316   <xsd:attribute name="bg1" type="ST_ColorSchemeIndex" use="required"/>
2317   <xsd:attribute name="tx1" type="ST_ColorSchemeIndex" use="required"/>
2318   <xsd:attribute name="bg2" type="ST_ColorSchemeIndex" use="required"/>
2319   <xsd:attribute name="tx2" type="ST_ColorSchemeIndex" use="required"/>
2320   <xsd:attribute name="accent1" type="ST_ColorSchemeIndex" use="required"/>
2321   <xsd:attribute name="accent2" type="ST_ColorSchemeIndex" use="required"/>
2322   <xsd:attribute name="accent3" type="ST_ColorSchemeIndex" use="required"/>

```

```

2323     <xsd:attribute name="accent4" type="ST_ColorSchemeIndex" use="required"/>
2324     <xsd:attribute name="accent5" type="ST_ColorSchemeIndex" use="required"/>
2325     <xsd:attribute name="accent6" type="ST_ColorSchemeIndex" use="required"/>
2326     <xsd:attribute name="hlink" type="ST_ColorSchemeIndex" use="required"/>
2327     <xsd:attribute name="folHlink" type="ST_ColorSchemeIndex" use="required"/>
2328   </xsd:complexType>
2329   <xsd:complexType name="CT_ColorMappingOverride">
2330     <xsd:sequence>
2331       <xsd:choice minOccurs="1" maxOccurs="1">
2332         <xsd:element name="masterClrMapping" type="CT_EmptyElement"/>
2333         <xsd:element name="overrideClrMapping" type="CT_ColorMapping"/>
2334       </xsd:choice>
2335     </xsd:sequence>
2336   </xsd:complexType>
2337   <xsd:complexType name="CT_ColorSchemeAndMapping">
2338     <xsd:sequence>
2339       <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="1" maxOccurs="1"/>
2340       <xsd:element name="clrMap" type="CT_ColorMapping" minOccurs="0" maxOccurs="1"/>
2341     </xsd:sequence>
2342   </xsd:complexType>
2343   <xsd:complexType name="CT_ColorSchemeList">
2344     <xsd:sequence>
2345       <xsd:element name="extraClrScheme" type="CT_ColorSchemeAndMapping" minOccurs="0"
2346         maxOccurs="unbounded"/>
2347     </xsd:sequence>
2348   </xsd:complexType>
2349   <xsd:complexType name="CT_OfficeStyleSheet">
2350     <xsd:sequence>
2351       <xsd:element name="themeElements" type="CT_BaseStyles" minOccurs="1" maxOccurs="1"/>
2352       <xsd:element name="objectDefaults" type="CT_ObjectStyleDefaults" minOccurs="0"
2353         maxOccurs="1"/>
2354       <xsd:element name="extraClrSchemeLst" type="CT_ColorSchemeList" minOccurs="0"
2355         maxOccurs="1"/>
2356       <xsd:element name="custClrLst" type="CT_CustomColorList" minOccurs="0" maxOccurs="1"/>
2357       <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2358     </xsd:sequence>
2359     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
2360   </xsd:complexType>
2361   <xsd:complexType name="CT_BaseStylesOverride">
2362     <xsd:sequence>
2363       <xsd:element name="clrScheme" type="CT_ColorScheme" minOccurs="0" maxOccurs="1"/>
2364       <xsd:element name="fontScheme" type="CT_FontScheme" minOccurs="0" maxOccurs="1"/>
2365       <xsd:element name="fmtScheme" type="CT_StyleMatrix" minOccurs="0" maxOccurs="1"/>
2366     </xsd:sequence>
2367   </xsd:complexType>
2368   <xsd:complexType name="CT_ClipboardStyleSheet">
2369     <xsd:sequence>
2370       <xsd:element name="themeElements" type="CT_BaseStyles" minOccurs="1" maxOccurs="1"/>
2371       <xsd:element name="clrMap" type="CT_ColorMapping" minOccurs="1" maxOccurs="1"/>
2372     </xsd:sequence>
2373   </xsd:complexType>
2374   <xsd:element name="theme" type="CT_OfficeStyleSheet"/>
2375   <xsd:element name="themeOverride" type="CT_BaseStylesOverride"/>

```

```

2376 <xsd:element name="themeManager" type="CT_EmptyElement"/>
2377 <xsd:complexType name="CT_TableCellProperties">
2378   <xsd:sequence>
2379     <xsd:element name="lnL" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2380     <xsd:element name="lnR" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2381     <xsd:element name="lnT" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2382     <xsd:element name="lnB" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2383     <xsd:element name="lnTlToBr" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2384     <xsd:element name="lnBlToTr" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2385     <xsd:element name="cell3D" type="CT_Cell3D" minOccurs="0" maxOccurs="1"/>
2386     <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2387     <xsd:element name="headers" type="CT_Headers" minOccurs="0"/>
2388     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2389   </xsd:sequence>
2390   <xsd:attribute name="marL" type="ST_Coordinate32" use="optional" default="91440"/>
2391   <xsd:attribute name="marR" type="ST_Coordinate32" use="optional" default="91440"/>
2392   <xsd:attribute name="marT" type="ST_Coordinate32" use="optional" default="45720"/>
2393   <xsd:attribute name="marB" type="ST_Coordinate32" use="optional" default="45720"/>
2394   <xsd:attribute name="vert" type="ST_TextVerticalType" use="optional" default="horz"/>
2395   <xsd:attribute name="anchor" type="ST_TextAnchoringType" use="optional" default="t"/>
2396   <xsd:attribute name="anchorCtr" type="xsd:boolean" use="optional" default="false"/>
2397   <xsd:attribute name="horzOverflow" type="ST_TextHorzOverflowType" use="optional"
2398     default="clip"/>
2399 </xsd:complexType>
2400 <xsd:complexType name="CT_Headers">
2401   <xsd:sequence minOccurs="0" maxOccurs="unbounded">
2402     <xsd:element name="header" type="xsd:string"/>
2403   </xsd:sequence>
2404 </xsd:complexType>
2405 <xsd:complexType name="CT_TableCol">
2406   <xsd:sequence>
2407     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2408   </xsd:sequence>
2409   <xsd:attribute name="w" type="ST_Coordinate" use="required"/>
2410 </xsd:complexType>
2411 <xsd:complexType name="CT_TableGrid">
2412   <xsd:sequence>
2413     <xsd:element name="gridCol" type="CT_TableCol" minOccurs="0" maxOccurs="unbounded"/>
2414   </xsd:sequence>
2415 </xsd:complexType>
2416 <xsd:complexType name="CT_TableCell">
2417   <xsd:sequence>
2418     <xsd:element name="txBody" type="CT_TextBody" minOccurs="0" maxOccurs="1"/>
2419     <xsd:element name="tcPr" type="CT_TableCellProperties" minOccurs="0" maxOccurs="1"/>
2420     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2421   </xsd:sequence>
2422   <xsd:attribute name="rowSpan" type="xsd:int" use="optional" default="1"/>
2423   <xsd:attribute name="gridSpan" type="xsd:int" use="optional" default="1"/>
2424   <xsd:attribute name="hMerge" type="xsd:boolean" use="optional" default="false"/>
2425   <xsd:attribute name="vMerge" type="xsd:boolean" use="optional" default="false"/>
2426   <xsd:attribute name="id" type="xsd:string" use="optional"/>
2427 </xsd:complexType>
2428 <xsd:complexType name="CT_TableRow">

```

```

2429     <xsd:sequence>
2430         <xsd:element name="tc" type="CT_TableCell" minOccurs="0" maxOccurs="unbounded"/>
2431         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2432     </xsd:sequence>
2433     <xsd:attribute name="h" type="ST_Coordinate" use="required"/>
2434 </xsd:complexType>
2435 <xsd:complexType name="CT_TableProperties">
2436     <xsd:sequence>
2437         <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2438         <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2439         <xsd:choice minOccurs="0" maxOccurs="1">
2440             <xsd:element name="tableStyle" type="CT_TableStyle"/>
2441             <xsd:element name="tableStyleId" type="s:ST_Guid"/>
2442         </xsd:choice>
2443         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2444     </xsd:sequence>
2445     <xsd:attribute name="rtl" type="xsd:boolean" use="optional" default="false"/>
2446     <xsd:attribute name="firstRow" type="xsd:boolean" use="optional" default="false"/>
2447     <xsd:attribute name="firstCol" type="xsd:boolean" use="optional" default="false"/>
2448     <xsd:attribute name="lastRow" type="xsd:boolean" use="optional" default="false"/>
2449     <xsd:attribute name="lastCol" type="xsd:boolean" use="optional" default="false"/>
2450     <xsd:attribute name="bandRow" type="xsd:boolean" use="optional" default="false"/>
2451     <xsd:attribute name="bandCol" type="xsd:boolean" use="optional" default="false"/>
2452 </xsd:complexType>
2453 <xsd:complexType name="CT_Table">
2454     <xsd:sequence>
2455         <xsd:element name="tblPr" type="CT_TableProperties" minOccurs="0" maxOccurs="1"/>
2456         <xsd:element name="tblGrid" type="CT_TableGrid" minOccurs="1" maxOccurs="1"/>
2457         <xsd:element name="tr" type="CT_TableRow" minOccurs="0" maxOccurs="unbounded"/>
2458     </xsd:sequence>
2459 </xsd:complexType>
2460 <xsd:element name="tbl" type="CT_Table"/>
2461 <xsd:complexType name="CT_Cell3D">
2462     <xsd:sequence>
2463         <xsd:element name="bevel" type="CT_Bevel" minOccurs="1" maxOccurs="1"/>
2464         <xsd:element name="lightRig" type="CT_LightRig" minOccurs="0" maxOccurs="1"/>
2465         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2466     </xsd:sequence>
2467     <xsd:attribute name="prstMaterial" type="ST_PresetMaterialType" use="optional"
2468         default="plastic"/>
2469 </xsd:complexType>
2470 <xsd:group name="EG_ThemeableFillStyle">
2471     <xsd:choice>
2472         <xsd:element name="fill" type="CT_FillProperties" minOccurs="1" maxOccurs="1"/>
2473         <xsd:element name="fillRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2474     </xsd:choice>
2475 </xsd:group>
2476 <xsd:complexType name="CT_ThemeableLineStyle">
2477     <xsd:choice>
2478         <xsd:element name="ln" type="CT_LineProperties" minOccurs="1" maxOccurs="1"/>
2479         <xsd:element name="lnRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2480     </xsd:choice>
2481 </xsd:complexType>

```



```

2482 <xsd:group name="EG_ThemeableEffectStyle">
2483   <xsd:choice>
2484     <xsd:element name="effect" type="CT_EffectProperties" minOccurs="1" maxOccurs="1"/>
2485     <xsd:element name="effectRef" type="CT_StyleMatrixReference" minOccurs="1" maxOccurs="1"/>
2486   </xsd:choice>
2487 </xsd:group>
2488 <xsd:group name="EG_ThemeableFontStyles">
2489   <xsd:choice>
2490     <xsd:element name="font" type="CT_FontCollection" minOccurs="1" maxOccurs="1"/>
2491     <xsd:element name="fontRef" type="CT_FontReference" minOccurs="1" maxOccurs="1"/>
2492   </xsd:choice>
2493 </xsd:group>
2494 <xsd:simpleType name="ST_OnOffStyleType">
2495   <xsd:restriction base="xsd:token">
2496     <xsd:enumeration value="on"/>
2497     <xsd:enumeration value="off"/>
2498     <xsd:enumeration value="def"/>
2499   </xsd:restriction>
2500 </xsd:simpleType>
2501 <xsd:complexType name="CT_TableStyleTextStyle">
2502   <xsd:sequence>
2503     <xsd:group ref="EG_ThemeableFontStyles" minOccurs="0" maxOccurs="1"/>
2504     <xsd:group ref="EG_ColorChoice" minOccurs="0" maxOccurs="1"/>
2505     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2506   </xsd:sequence>
2507   <xsd:attribute name="b" type="ST_OnOffStyleType" use="optional" default="def"/>
2508   <xsd:attribute name="i" type="ST_OnOffStyleType" use="optional" default="def"/>
2509 </xsd:complexType>
2510 <xsd:complexType name="CT_TableCellBorderStyle">
2511   <xsd:sequence>
2512     <xsd:element name="left" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2513     <xsd:element name="right" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2514     <xsd:element name="top" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2515     <xsd:element name="bottom" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2516     <xsd:element name="insideH" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2517     <xsd:element name="insideV" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2518     <xsd:element name="tl2br" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2519     <xsd:element name="tr2bl" type="CT_ThemeableLineStyle" minOccurs="0" maxOccurs="1"/>
2520     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2521   </xsd:sequence>
2522 </xsd:complexType>
2523 <xsd:complexType name="CT_TableBackgroundStyle">
2524   <xsd:sequence>
2525     <xsd:group ref="EG_ThemeableFillStyle" minOccurs="0" maxOccurs="1"/>
2526     <xsd:group ref="EG_ThemeableEffectStyle" minOccurs="0" maxOccurs="1"/>
2527   </xsd:sequence>
2528 </xsd:complexType>
2529 <xsd:complexType name="CT_TableStyleCellStyle">
2530   <xsd:sequence>
2531     <xsd:element name="tcBdr" type="CT_TableCellBorderStyle" minOccurs="0" maxOccurs="1"/>
2532     <xsd:group ref="EG_ThemeableFillStyle" minOccurs="0" maxOccurs="1"/>
2533     <xsd:element name="cell3D" type="CT_Cell3D" minOccurs="0" maxOccurs="1"/>
2534   </xsd:sequence>

```

```

2535 </xsd:complexType>
2536 <xsd:complexType name="CT_TablePartStyle">
2537   <xsd:sequence>
2538     <xsd:element name="tcTxStyle" type="CT_TableStyleTextStyle" minOccurs="0" maxOccurs="1"/>
2539     <xsd:element name="tcStyle" type="CT_TableStyleCellStyle" minOccurs="0" maxOccurs="1"/>
2540   </xsd:sequence>
2541 </xsd:complexType>
2542 <xsd:complexType name="CT_TableStyle">
2543   <xsd:sequence>
2544     <xsd:element name="tblBg" type="CT_TableBackgroundStyle" minOccurs="0" maxOccurs="1"/>
2545     <xsd:element name="wholeTbl" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2546     <xsd:element name="band1H" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2547     <xsd:element name="band2H" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2548     <xsd:element name="band1V" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2549     <xsd:element name="band2V" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2550     <xsd:element name="lastCol" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2551     <xsd:element name="firstCol" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2552     <xsd:element name="lastRow" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2553     <xsd:element name="seCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2554     <xsd:element name="swCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2555     <xsd:element name="firstRow" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2556     <xsd:element name="neCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2557     <xsd:element name="nwCell" type="CT_TablePartStyle" minOccurs="0" maxOccurs="1"/>
2558     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2559   </xsd:sequence>
2560   <xsd:attribute name="styleId" type="s:ST_Guid" use="required"/>
2561   <xsd:attribute name="styleName" type="xsd:string" use="required"/>
2562 </xsd:complexType>
2563 <xsd:complexType name="CT_TableStyleList">
2564   <xsd:sequence>
2565     <xsd:element name="tblStyle" type="CT_TableStyle" minOccurs="0" maxOccurs="unbounded"/>
2566   </xsd:sequence>
2567   <xsd:attribute name="def" type="s:ST_Guid" use="required"/>
2568 </xsd:complexType>
2569 <xsd:element name="tblStyleLst" type="CT_TableStyleList"/>
2570 <xsd:complexType name="CT_TextParagraph">
2571   <xsd:sequence>
2572     <xsd:element name="pPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
2573     <xsd:group ref="EG_TextRun" minOccurs="0" maxOccurs="unbounded"/>
2574     <xsd:element name="endParaRPr" type="CT_TextCharacterProperties" minOccurs="0"
2575       maxOccurs="1"/>
2576   </xsd:sequence>
2577 </xsd:complexType>
2578 <xsd:simpleType name="ST_TextAnchoringType">
2579   <xsd:restriction base="xsd:token">
2580     <xsd:enumeration value="t"/>
2581     <xsd:enumeration value="ctr"/>
2582     <xsd:enumeration value="b"/>
2583     <xsd:enumeration value="just"/>
2584     <xsd:enumeration value="dist"/>
2585   </xsd:restriction>
2586 </xsd:simpleType>
2587 <xsd:simpleType name="ST_TextVertOverflowType">

```

```

2588     <xsd:restriction base="xsd:token">
2589         <xsd:enumeration value="overflow"/>
2590         <xsd:enumeration value="ellipsis"/>
2591         <xsd:enumeration value="clip"/>
2592     </xsd:restriction>
2593 </xsd:simpleType>
2594 <xsd:simpleType name="ST_TextHorzOverflowType">
2595     <xsd:restriction base="xsd:token">
2596         <xsd:enumeration value="overflow"/>
2597         <xsd:enumeration value="clip"/>
2598     </xsd:restriction>
2599 </xsd:simpleType>
2600 <xsd:simpleType name="ST_TextVerticalType">
2601     <xsd:restriction base="xsd:token">
2602         <xsd:enumeration value="horz"/>
2603         <xsd:enumeration value="vert"/>
2604         <xsd:enumeration value="vert270"/>
2605         <xsd:enumeration value="wordArtVert"/>
2606         <xsd:enumeration value="eaVert"/>
2607         <xsd:enumeration value="mongolianVert"/>
2608         <xsd:enumeration value="wordArtVertRtl"/>
2609     </xsd:restriction>
2610 </xsd:simpleType>
2611 <xsd:simpleType name="ST_TextWrappingType">
2612     <xsd:restriction base="xsd:token">
2613         <xsd:enumeration value="none"/>
2614         <xsd:enumeration value="square"/>
2615     </xsd:restriction>
2616 </xsd:simpleType>
2617 <xsd:simpleType name="ST_TextColumnCount">
2618     <xsd:restriction base="xsd:int">
2619         <xsd:minInclusive value="1"/>
2620         <xsd:maxInclusive value="16"/>
2621     </xsd:restriction>
2622 </xsd:simpleType>
2623 <xsd:complexType name="CT_TextListStyle">
2624     <xsd:sequence>
2625         <xsd:element name="defPPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
2626         <xsd:element name="lv11pPr" type="CT_TextParagraphProperties" minOccurs="0"
2627             maxOccurs="1"/>
2628         <xsd:element name="lv12pPr" type="CT_TextParagraphProperties" minOccurs="0"
2629             maxOccurs="1"/>
2630         <xsd:element name="lv13pPr" type="CT_TextParagraphProperties" minOccurs="0"
2631             maxOccurs="1"/>
2632         <xsd:element name="lv14pPr" type="CT_TextParagraphProperties" minOccurs="0"
2633             maxOccurs="1"/>
2634         <xsd:element name="lv15pPr" type="CT_TextParagraphProperties" minOccurs="0"
2635             maxOccurs="1"/>
2636         <xsd:element name="lv16pPr" type="CT_TextParagraphProperties" minOccurs="0"
2637             maxOccurs="1"/>
2638         <xsd:element name="lv17pPr" type="CT_TextParagraphProperties" minOccurs="0"
2639             maxOccurs="1"/>

```

```

2640     <xsd:element name="lv18pPr" type="CT_TextParagraphProperties" minOccurs="0"
2641         maxOccurs="1"/>
2642     <xsd:element name="lv19pPr" type="CT_TextParagraphProperties" minOccurs="0"
2643         maxOccurs="1"/>
2644     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2645 </xsd:sequence>
2646 </xsd:complexType>
2647 <xsd:simpleType name="ST_TextFontScalePercentOrPercentString">
2648     <xsd:union memberTypes="ST_TextFontScalePercent s:ST_Percentage"/>
2649 </xsd:simpleType>
2650 <xsd:simpleType name="ST_TextFontScalePercent">
2651     <xsd:restriction base="ST_PercentageDecimal">
2652         <xsd:minInclusive value="1000"/>
2653         <xsd:maxInclusive value="100000"/>
2654     </xsd:restriction>
2655 </xsd:simpleType>
2656 <xsd:complexType name="CT_TextNormalAutofit">
2657     <xsd:attribute name="fontScale" type="ST_TextFontScalePercentOrPercentString" use="optional"
2658         default="100%"/>
2659     <xsd:attribute name="lnSpcReduction" type="ST_TextSpacingPercentOrPercentString"
2660         use="optional" default="0%"/>
2661 </xsd:complexType>
2662 <xsd:complexType name="CT_TextShapeAutofit"/>
2663 <xsd:complexType name="CT_TextNoAutofit"/>
2664 <xsd:group name="EG_TextAutofit">
2665     <xsd:choice>
2666         <xsd:element name="noAutofit" type="CT_TextNoAutofit"/>
2667         <xsd:element name="normAutofit" type="CT_TextNormalAutofit"/>
2668         <xsd:element name="spAutoFit" type="CT_TextShapeAutofit"/>
2669     </xsd:choice>
2670 </xsd:group>
2671 <xsd:complexType name="CT_TextBodyProperties">
2672     <xsd:sequence>
2673         <xsd:element name="prstTxWarp" type="CT_PresetTextShape" minOccurs="0" maxOccurs="1"/>
2674         <xsd:group ref="EG_TextAutofit" minOccurs="0" maxOccurs="1"/>
2675         <xsd:element name="scene3d" type="CT_Scene3D" minOccurs="0" maxOccurs="1"/>
2676         <xsd:group ref="EG_Text3D" minOccurs="0" maxOccurs="1"/>
2677         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2678     </xsd:sequence>
2679     <xsd:attribute name="rot" type="ST_Angle" use="optional"/>
2680     <xsd:attribute name="spcFirstLastPara" type="xsd:boolean" use="optional"/>
2681     <xsd:attribute name="vertOverflow" type="ST_TextVertOverflowType" use="optional"/>
2682     <xsd:attribute name="horzOverflow" type="ST_TextHorzOverflowType" use="optional"/>
2683     <xsd:attribute name="vert" type="ST_TextVerticalType" use="optional"/>
2684     <xsd:attribute name="wrap" type="ST_TextWrappingType" use="optional"/>
2685     <xsd:attribute name="lIns" type="ST_Coordinate32" use="optional"/>
2686     <xsd:attribute name="tIns" type="ST_Coordinate32" use="optional"/>
2687     <xsd:attribute name="rIns" type="ST_Coordinate32" use="optional"/>
2688     <xsd:attribute name="bIns" type="ST_Coordinate32" use="optional"/>
2689     <xsd:attribute name="numCol" type="ST_TextColumnCount" use="optional"/>
2690     <xsd:attribute name="spcCol" type="ST_PositiveCoordinate32" use="optional"/>
2691     <xsd:attribute name="rtlCol" type="xsd:boolean" use="optional"/>
2692     <xsd:attribute name="fromWordArt" type="xsd:boolean" use="optional"/>

```

```

2693     <xsd:attribute name="anchor" type="ST_TextAnchoringType" use="optional"/>
2694     <xsd:attribute name="anchorCtr" type="xsd:boolean" use="optional"/>
2695     <xsd:attribute name="forceAA" type="xsd:boolean" use="optional"/>
2696     <xsd:attribute name="upright" type="xsd:boolean" use="optional" default="false"/>
2697     <xsd:attribute name="compatLnSpc" type="xsd:boolean" use="optional"/>
2698 </xsd:complexType>
2699 <xsd:complexType name="CT_TextBody">
2700     <xsd:sequence>
2701         <xsd:element name="bodyPr" type="CT_TextBodyProperties" minOccurs="1" maxOccurs="1"/>
2702         <xsd:element name="lstStyle" type="CT_TextListStyle" minOccurs="0" maxOccurs="1"/>
2703         <xsd:element name="p" type="CT_TextParagraph" minOccurs="1" maxOccurs="unbounded"/>
2704     </xsd:sequence>
2705 </xsd:complexType>
2706 <xsd:simpleType name="ST_TextBulletStartAtNum">
2707     <xsd:restriction base="xsd:int">
2708         <xsd:minInclusive value="1"/>
2709         <xsd:maxInclusive value="32767"/>
2710     </xsd:restriction>
2711 </xsd:simpleType>
2712 <xsd:simpleType name="ST_TextAutonumberScheme">
2713     <xsd:restriction base="xsd:token">
2714         <xsd:enumeration value="alphaLcParenBoth"/>
2715         <xsd:enumeration value="alphaUcParenBoth"/>
2716         <xsd:enumeration value="alphaLcParenR"/>
2717         <xsd:enumeration value="alphaUcParenR"/>
2718         <xsd:enumeration value="alphaLcPeriod"/>
2719         <xsd:enumeration value="alphaUcPeriod"/>
2720         <xsd:enumeration value="arabicParenBoth"/>
2721         <xsd:enumeration value="arabicParenR"/>
2722         <xsd:enumeration value="arabicPeriod"/>
2723         <xsd:enumeration value="arabicPlain"/>
2724         <xsd:enumeration value="romanLcParenBoth"/>
2725         <xsd:enumeration value="romanUcParenBoth"/>
2726         <xsd:enumeration value="romanLcParenR"/>
2727         <xsd:enumeration value="romanUcParenR"/>
2728         <xsd:enumeration value="romanLcPeriod"/>
2729         <xsd:enumeration value="romanUcPeriod"/>
2730         <xsd:enumeration value="circleNumDbPlain"/>
2731         <xsd:enumeration value="circleNumWdBlackPlain"/>
2732         <xsd:enumeration value="circleNumWdWhitePlain"/>
2733         <xsd:enumeration value="arabicDbPeriod"/>
2734         <xsd:enumeration value="arabicDbPlain"/>
2735         <xsd:enumeration value="ea1ChsPeriod"/>
2736         <xsd:enumeration value="ea1ChsPlain"/>
2737         <xsd:enumeration value="ea1ChtPeriod"/>
2738         <xsd:enumeration value="ea1ChtPlain"/>
2739         <xsd:enumeration value="ea1JpnChsDbPeriod"/>
2740         <xsd:enumeration value="ea1JpnKorPlain"/>
2741         <xsd:enumeration value="ea1JpnKorPeriod"/>
2742         <xsd:enumeration value="arabic1Minus"/>
2743         <xsd:enumeration value="arabic2Minus"/>
2744         <xsd:enumeration value="hebrew2Minus"/>
2745         <xsd:enumeration value="thaiAlphaPeriod"/>

```

```

2746     <xsd:enumeration value="thaiAlphaParenR"/>
2747     <xsd:enumeration value="thaiAlphaParenBoth"/>
2748     <xsd:enumeration value="thaiNumPeriod"/>
2749     <xsd:enumeration value="thaiNumParenR"/>
2750     <xsd:enumeration value="thaiNumParenBoth"/>
2751     <xsd:enumeration value="hindiAlphaPeriod"/>
2752     <xsd:enumeration value="hindiNumPeriod"/>
2753     <xsd:enumeration value="hindiNumParenR"/>
2754     <xsd:enumeration value="hindiAlpha1Period"/>
2755   </xsd:restriction>
2756 </xsd:simpleType>
2757 <xsd:complexType name="CT_TextBulletColorFollowText"/>
2758 <xsd:group name="EG_TextBulletColor">
2759   <xsd:choice>
2760     <xsd:element name="buClrTx" type="CT_TextBulletColorFollowText" minOccurs="1"
2761       maxOccurs="1"/>
2762     <xsd:element name="buClr" type="CT_Color" minOccurs="1" maxOccurs="1"/>
2763   </xsd:choice>
2764 </xsd:group>
2765 <xsd:simpleType name="ST_TextBulletSizePercent">
2766   <xsd:restriction base="ST_PercentageDecimal">
2767     <xsd:minInclusive value="25000"/>
2768     <xsd:maxInclusive value="400000"/>
2769   </xsd:restriction>
2770 </xsd:simpleType>
2771 <xsd:complexType name="CT_TextBulletSizeFollowText"/>
2772 <xsd:complexType name="CT_TextBulletSizePercent">
2773   <xsd:attribute name="val" type="ST_TextBulletSizePercent"/>
2774 </xsd:complexType>
2775 <xsd:complexType name="CT_TextBulletSizePoint">
2776   <xsd:attribute name="val" type="ST_TextFontSize"/>
2777 </xsd:complexType>
2778 <xsd:group name="EG_TextBulletSize">
2779   <xsd:choice>
2780     <xsd:element name="buSzTx" type="CT_TextBulletSizeFollowText"/>
2781     <xsd:element name="buSzPct" type="CT_TextBulletSizePercent"/>
2782     <xsd:element name="buSzPts" type="CT_TextBulletSizePoint"/>
2783   </xsd:choice>
2784 </xsd:group>
2785 <xsd:complexType name="CT_TextBulletTypefaceFollowText"/>
2786 <xsd:group name="EG_TextBulletTypeface">
2787   <xsd:choice>
2788     <xsd:element name="buFontTx" type="CT_TextBulletTypefaceFollowText"/>
2789     <xsd:element name="buFont" type="CT_TextFont"/>
2790   </xsd:choice>
2791 </xsd:group>
2792 <xsd:complexType name="CT_TextAutonumberBullet">
2793   <xsd:attribute name="type" type="ST_TextAutonumberScheme" use="required"/>
2794   <xsd:attribute name="startAt" type="ST_TextBulletStartAtNum" use="optional" default="1"/>
2795 </xsd:complexType>
2796 <xsd:complexType name="CT_TextCharBullet">
2797   <xsd:attribute name="char" type="xsd:string" use="required"/>
2798 </xsd:complexType>

```

```

2799 <xsd:complexType name="CT_TextBlipBullet">
2800   <xsd:sequence>
2801     <xsd:element name="blip" type="CT_Blip" minOccurs="1" maxOccurs="1"/>
2802   </xsd:sequence>
2803 </xsd:complexType>
2804 <xsd:complexType name="CT_TextNoBullet"/>
2805 <xsd:group name="EG_TextBullet">
2806   <xsd:choice>
2807     <xsd:element name="buNone" type="CT_TextNoBullet"/>
2808     <xsd:element name="buAutoNum" type="CT_TextAutonumberBullet"/>
2809     <xsd:element name="buChar" type="CT_TextCharBullet"/>
2810     <xsd:element name="buBlip" type="CT_TextBlipBullet"/>
2811   </xsd:choice>
2812 </xsd:group>
2813 <xsd:simpleType name="ST_TextPoint">
2814   <xsd:union memberTypes="ST_TextPointUnqualified s:ST_UniversalMeasure"/>
2815 </xsd:simpleType>
2816 <xsd:simpleType name="ST_TextPointUnqualified">
2817   <xsd:restriction base="xsd:int">
2818     <xsd:minInclusive value="-400000"/>
2819     <xsd:maxInclusive value="400000"/>
2820   </xsd:restriction>
2821 </xsd:simpleType>
2822 <xsd:simpleType name="ST_TextNonNegativePoint">
2823   <xsd:restriction base="xsd:int">
2824     <xsd:minInclusive value="0"/>
2825     <xsd:maxInclusive value="400000"/>
2826   </xsd:restriction>
2827 </xsd:simpleType>
2828 <xsd:simpleType name="ST_TextFontSize">
2829   <xsd:restriction base="xsd:int">
2830     <xsd:minInclusive value="100"/>
2831     <xsd:maxInclusive value="400000"/>
2832   </xsd:restriction>
2833 </xsd:simpleType>
2834 <xsd:simpleType name="ST_TextTypeface">
2835   <xsd:restriction base="xsd:string"/>
2836 </xsd:simpleType>
2837 <xsd:complexType name="CT_TextFont">
2838   <xsd:attribute name="typeface" type="ST_TextTypeface"/>
2839   <xsd:attribute name="panose" type="s:ST_Panose" use="optional"/>
2840   <xsd:attribute name="pitchFamily" type="xsd:byte" use="optional" default="0"/>
2841   <xsd:attribute name="charset" type="xsd:byte" use="optional" default="1"/>
2842 </xsd:complexType>
2843 <xsd:simpleType name="ST_TextUnderlineType">
2844   <xsd:restriction base="xsd:token">
2845     <xsd:enumeration value="none"/>
2846     <xsd:enumeration value="words"/>
2847     <xsd:enumeration value="sng"/>
2848     <xsd:enumeration value="dbl"/>
2849     <xsd:enumeration value="heavy"/>
2850     <xsd:enumeration value="dotted"/>
2851     <xsd:enumeration value="dottedHeavy"/>

```

```

2852     <xsd:enumeration value="dash"/>
2853     <xsd:enumeration value="dashHeavy"/>
2854     <xsd:enumeration value="dashLong"/>
2855     <xsd:enumeration value="dashLongHeavy"/>
2856     <xsd:enumeration value="dotDash"/>
2857     <xsd:enumeration value="dotDashHeavy"/>
2858     <xsd:enumeration value="dotDotDash"/>
2859     <xsd:enumeration value="dotDotDashHeavy"/>
2860     <xsd:enumeration value="wavy"/>
2861     <xsd:enumeration value="wavyHeavy"/>
2862     <xsd:enumeration value="wavyDbl"/>
2863   </xsd:restriction>
2864 </xsd:simpleType>
2865 <xsd:complexType name="CT_TextUnderlineLineFollowText"/>
2866 <xsd:complexType name="CT_TextUnderlineFillFollowText"/>
2867 <xsd:complexType name="CT_TextUnderlineFillGroupWrapper">
2868   <xsd:group ref="EG_FillProperties" minOccurs="1" maxOccurs="1"/>
2869 </xsd:complexType>
2870 <xsd:group name="EG_TextUnderlineLine">
2871   <xsd:choice>
2872     <xsd:element name="uLnTx" type="CT_TextUnderlineLineFollowText"/>
2873     <xsd:element name="uLn" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2874   </xsd:choice>
2875 </xsd:group>
2876 <xsd:group name="EG_TextUnderlineFill">
2877   <xsd:choice>
2878     <xsd:element name="uFillTx" type="CT_TextUnderlineFillFollowText"/>
2879     <xsd:element name="uFill" type="CT_TextUnderlineFillGroupWrapper"/>
2880   </xsd:choice>
2881 </xsd:group>
2882 <xsd:simpleType name="ST_TextStrikeType">
2883   <xsd:restriction base="xsd:token">
2884     <xsd:enumeration value="noStrike"/>
2885     <xsd:enumeration value="sngStrike"/>
2886     <xsd:enumeration value="dblStrike"/>
2887   </xsd:restriction>
2888 </xsd:simpleType>
2889 <xsd:simpleType name="ST_TextCapsType">
2890   <xsd:restriction base="xsd:token">
2891     <xsd:enumeration value="none"/>
2892     <xsd:enumeration value="small"/>
2893     <xsd:enumeration value="all"/>
2894   </xsd:restriction>
2895 </xsd:simpleType>
2896 <xsd:complexType name="CT_TextCharacterProperties">
2897   <xsd:sequence>
2898     <xsd:element name="ln" type="CT_LineProperties" minOccurs="0" maxOccurs="1"/>
2899     <xsd:group ref="EG_FillProperties" minOccurs="0" maxOccurs="1"/>
2900     <xsd:group ref="EG_EffectProperties" minOccurs="0" maxOccurs="1"/>
2901     <xsd:element name="highlight" type="CT_Color" minOccurs="0" maxOccurs="1"/>
2902     <xsd:group ref="EG_TextUnderlineLine" minOccurs="0" maxOccurs="1"/>
2903     <xsd:group ref="EG_TextUnderlineFill" minOccurs="0" maxOccurs="1"/>
2904     <xsd:element name="latin" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>

```



```

2905     <xsd:element name="ea" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2906     <xsd:element name="cs" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2907     <xsd:element name="sym" type="CT_TextFont" minOccurs="0" maxOccurs="1"/>
2908     <xsd:element name="hlinkClick" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
2909     <xsd:element name="hlinkMouseOver" type="CT_Hyperlink" minOccurs="0" maxOccurs="1"/>
2910     <xsd:element name="rtl" type="CT_Boolean" minOccurs="0"/>
2911     <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
2912 </xsd:sequence>
2913 <xsd:attribute name="kumimoji" type="xsd:boolean" use="optional"/>
2914 <xsd:attribute name="lang" type="s:ST_Lang" use="optional"/>
2915 <xsd:attribute name="altLang" type="s:ST_Lang" use="optional"/>
2916 <xsd:attribute name="sz" type="ST_TextFontSize" use="optional"/>
2917 <xsd:attribute name="b" type="xsd:boolean" use="optional"/>
2918 <xsd:attribute name="i" type="xsd:boolean" use="optional"/>
2919 <xsd:attribute name="u" type="ST_TextUnderlineType" use="optional"/>
2920 <xsd:attribute name="strike" type="ST_TextStrikeType" use="optional"/>
2921 <xsd:attribute name="kern" type="ST_TextNonNegativePoint" use="optional"/>
2922 <xsd:attribute name="cap" type="ST_TextCapsType" use="optional"/>
2923 <xsd:attribute name="spc" type="ST_TextPoint" use="optional"/>
2924 <xsd:attribute name="normalizeH" type="xsd:boolean" use="optional"/>
2925 <xsd:attribute name="baseline" type="ST_Percentage" use="optional"/>
2926 <xsd:attribute name="noProof" type="xsd:boolean" use="optional"/>
2927 <xsd:attribute name="dirty" type="xsd:boolean" use="optional" default="true"/>
2928 <xsd:attribute name="err" type="xsd:boolean" use="optional" default="false"/>
2929 <xsd:attribute name="smtClean" type="xsd:boolean" use="optional" default="true"/>
2930 <xsd:attribute name="smtId" type="xsd:unsignedInt" use="optional" default="0"/>
2931 <xsd:attribute name="bmk" type="xsd:string" use="optional"/>
2932 </xsd:complexType>
2933 <xsd:complexType name="CT_Boolean">
2934   <xsd:attribute name="val" type="s:ST_OnOff" default="0"/>
2935 </xsd:complexType>
2936 <xsd:simpleType name="ST_TextSpacingPoint">
2937   <xsd:restriction base="xsd:int">
2938     <xsd:minInclusive value="0"/>
2939     <xsd:maxInclusive value="158400"/>
2940   </xsd:restriction>
2941 </xsd:simpleType>
2942 <xsd:simpleType name="ST_TextSpacingPercentOrPercentString">
2943   <xsd:union memberTypes="ST_TextSpacingPercent s:ST_Percentage"/>
2944 </xsd:simpleType>
2945 <xsd:simpleType name="ST_TextSpacingPercent">
2946   <xsd:restriction base="ST_PercentageDecimal">
2947     <xsd:minInclusive value="0"/>
2948     <xsd:maxInclusive value="13200000"/>
2949   </xsd:restriction>
2950 </xsd:simpleType>
2951 <xsd:complexType name="CT_TextSpacingPercent">
2952   <xsd:attribute name="val" type="ST_TextSpacingPercentOrPercentString" use="required"/>
2953 </xsd:complexType>
2954 <xsd:complexType name="CT_TextSpacingPoint">
2955   <xsd:attribute name="val" type="ST_TextSpacingPoint" use="required"/>
2956 </xsd:complexType>
2957 <xsd:simpleType name="ST_TextMargin">

```

```

2958     <xsd:restriction base="ST_Coordinate32Unqualified">
2959         <xsd:minInclusive value="0"/>
2960         <xsd:maxInclusive value="51206400"/>
2961     </xsd:restriction>
2962 </xsd:simpleType>
2963 <xsd:simpleType name="ST_TextIndent">
2964     <xsd:restriction base="ST_Coordinate32Unqualified">
2965         <xsd:minInclusive value="-51206400"/>
2966         <xsd:maxInclusive value="51206400"/>
2967     </xsd:restriction>
2968 </xsd:simpleType>
2969 <xsd:simpleType name="ST_TextTabAlignType">
2970     <xsd:restriction base="xsd:token">
2971         <xsd:enumeration value="l"/>
2972         <xsd:enumeration value="ctr"/>
2973         <xsd:enumeration value="r"/>
2974         <xsd:enumeration value="dec"/>
2975     </xsd:restriction>
2976 </xsd:simpleType>
2977 <xsd:complexType name="CT_TextTabStop">
2978     <xsd:attribute name="pos" type="ST_Coordinate32" use="optional"/>
2979     <xsd:attribute name="align" type="ST_TextTabAlignType" use="optional"/>
2980 </xsd:complexType>
2981 <xsd:complexType name="CT_TextTabStopList">
2982     <xsd:sequence>
2983         <xsd:element name="tab" type="CT_TextTabStop" minOccurs="0" maxOccurs="32"/>
2984     </xsd:sequence>
2985 </xsd:complexType>
2986 <xsd:complexType name="CT_TextLineBreak">
2987     <xsd:sequence>
2988         <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
2989     </xsd:sequence>
2990 </xsd:complexType>
2991 <xsd:complexType name="CT_TextSpacing">
2992     <xsd:choice>
2993         <xsd:element name="spcPct" type="CT_TextSpacingPercent"/>
2994         <xsd:element name="spcPts" type="CT_TextSpacingPoint"/>
2995     </xsd:choice>
2996 </xsd:complexType>
2997 <xsd:simpleType name="ST_TextAlignType">
2998     <xsd:restriction base="xsd:token">
2999         <xsd:enumeration value="l"/>
3000         <xsd:enumeration value="ctr"/>
3001         <xsd:enumeration value="r"/>
3002         <xsd:enumeration value="just"/>
3003         <xsd:enumeration value="justLow"/>
3004         <xsd:enumeration value="dist"/>
3005         <xsd:enumeration value="thaiDist"/>
3006     </xsd:restriction>
3007 </xsd:simpleType>
3008 <xsd:simpleType name="ST_TextFontAlignType">
3009     <xsd:restriction base="xsd:token">
3010         <xsd:enumeration value="auto"/>

```

```

3011         <xsd:enumeration value="t"/>
3012         <xsd:enumeration value="ctr"/>
3013         <xsd:enumeration value="base"/>
3014         <xsd:enumeration value="b"/>
3015     </xsd:restriction>
3016 </xsd:simpleType>
3017 <xsd:simpleType name="ST_TextIndentLevelType">
3018     <xsd:restriction base="xsd:int">
3019         <xsd:minInclusive value="0"/>
3020         <xsd:maxInclusive value="8"/>
3021     </xsd:restriction>
3022 </xsd:simpleType>
3023 <xsd:complexType name="CT_TextParagraphProperties">
3024     <xsd:sequence>
3025         <xsd:element name="lnSp" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3026         <xsd:element name="spcBef" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3027         <xsd:element name="spcAft" type="CT_TextSpacing" minOccurs="0" maxOccurs="1"/>
3028         <xsd:group ref="EG_TextBulletColor" minOccurs="0" maxOccurs="1"/>
3029         <xsd:group ref="EG_TextBulletSize" minOccurs="0" maxOccurs="1"/>
3030         <xsd:group ref="EG_TextBulletTypeface" minOccurs="0" maxOccurs="1"/>
3031         <xsd:group ref="EG_TextBullet" minOccurs="0" maxOccurs="1"/>
3032         <xsd:element name="tabLst" type="CT_TextTabStopList" minOccurs="0" maxOccurs="1"/>
3033         <xsd:element name="defRPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3034         <xsd:element name="extLst" type="CT_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>
3035     </xsd:sequence>
3036     <xsd:attribute name="marL" type="ST_TextMargin" use="optional"/>
3037     <xsd:attribute name="marR" type="ST_TextMargin" use="optional"/>
3038     <xsd:attribute name="lvl" type="ST_TextIndentLevelType" use="optional"/>
3039     <xsd:attribute name="indent" type="ST_TextIndent" use="optional"/>
3040     <xsd:attribute name="algn" type="ST_TextAlignType" use="optional"/>
3041     <xsd:attribute name="defTabSz" type="ST_Coordinate32" use="optional"/>
3042     <xsd:attribute name="rtl" type="xsd:boolean" use="optional"/>
3043     <xsd:attribute name="eaLnBrk" type="xsd:boolean" use="optional"/>
3044     <xsd:attribute name="fontAlgn" type="ST_TextFontAlignType" use="optional"/>
3045     <xsd:attribute name="latinLnBrk" type="xsd:boolean" use="optional"/>
3046     <xsd:attribute name="hangingPunct" type="xsd:boolean" use="optional"/>
3047 </xsd:complexType>
3048 <xsd:complexType name="CT_TextField">
3049     <xsd:sequence>
3050         <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3051         <xsd:element name="pPr" type="CT_TextParagraphProperties" minOccurs="0" maxOccurs="1"/>
3052         <xsd:element name="t" type="xsd:string" minOccurs="0" maxOccurs="1"/>
3053     </xsd:sequence>
3054     <xsd:attribute name="id" type="s:ST_Guid" use="required"/>
3055     <xsd:attribute name="type" type="xsd:string" use="optional"/>
3056 </xsd:complexType>
3057 <xsd:group name="EG_TextRun">
3058     <xsd:choice>
3059         <xsd:element name="r" type="CT-RegularTextRun"/>
3060         <xsd:element name="br" type="CT_TextLineBreak"/>
3061         <xsd:element name="fld" type="CT_TextField"/>
3062     </xsd:choice>
3063 </xsd:group>

```

```

3064 <xsd:complexType name="CT_RegularTextRun">
3065   <xsd:sequence>
3066     <xsd:element name="rPr" type="CT_TextCharacterProperties" minOccurs="0" maxOccurs="1"/>
3067     <xsd:element name="t" type="xsd:string" minOccurs="1" maxOccurs="1"/>
3068   </xsd:sequence>
3069 </xsd:complexType>
3070 </xsd:schema>

```

A.4.2 DrawingML - Picture

```

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 - Legacy Compatibility

```

1 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
2   xmlns="http://schemas.openxmlformats.org/drawingml/2006/compatibility"
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/compatibility">
7   <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
8     main.xsd"/>
9   <xsd:complexType name="CT_Compat">
10     <xsd:attribute name="spid" type="a:ST_ShapeID" use="required"/>
11   </xsd:complexType>
12   <xsd:element name="legacyDrawing" type="CT_Compat"/>
13 </xsd:schema>

```

A.4.4 DrawingML - Locked Canvas

```

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"/>
10 </xsd:schema>

```

A.4.5 DrawingML - WordprocessingML Drawing

```

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="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
5    targetNamespace="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
6    elementFormDefault="qualified">
7    <xsd:import namespace="http://schemas.openxmlformats.org/drawingml/2006/main" schemaLocation="dml-
8      main.xsd"/>
9    <xsd:import schemaLocation="wml.xsd"
10      namespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main"/>
11    <xsd:complexType name="CT_EffectExtent">
12      <xsd:attribute name="l" type="a:ST_Coordinate" use="required"/>
13      <xsd:attribute name="t" type="a:ST_Coordinate" use="required"/>
14      <xsd:attribute name="r" type="a:ST_Coordinate" use="required"/>
15      <xsd:attribute name="b" type="a:ST_Coordinate" use="required"/>
16    </xsd:complexType>
17    <xsd:simpleType name="ST_WrapDistance">
18      <xsd:restriction base="xsd:unsignedInt"/>
19    </xsd:simpleType>
20    <xsd:complexType name="CT_Inline">
21      <xsd:sequence>
22        <xsd:element name="extent" type="a:CT_PositiveSize2D"/>
23        <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
24        <xsd:element name="docPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
25        <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
26          minOccurs="0" maxOccurs="1"/>
27        <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
28      </xsd:sequence>
29      <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
30      <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
31      <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
32      <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
33    </xsd:complexType>
34    <xsd:simpleType name="ST_WrapText">
35      <xsd:restriction base="xsd:token">
36        <xsd:enumeration value="bothSides"/>
37        <xsd:enumeration value="left"/>
38        <xsd:enumeration value="right"/>
39        <xsd:enumeration value="largest"/>
40      </xsd:restriction>
41    </xsd:simpleType>
42    <xsd:complexType name="CT_WrapPath">

```

```

43     <xsd:sequence>
44         <xsd:element name="start" type="a:CT_Point2D" minOccurs="1" maxOccurs="1"/>
45         <xsd:element name="lineTo" type="a:CT_Point2D" minOccurs="2" maxOccurs="unbounded"/>
46     </xsd:sequence>
47     <xsd:attribute name="edited" type="xsd:boolean" use="optional"/>
48 </xsd:complexType>
49 <xsd:complexType name="CT_WrapNone"/>
50 <xsd:complexType name="CT_WrapSquare">
51     <xsd:sequence>
52         <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
53     </xsd:sequence>
54     <xsd:attribute name="wrapText" type="ST_WrapText" use="required"/>
55     <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
56     <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
57     <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
58     <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
59 </xsd:complexType>
60 <xsd:complexType name="CT_WrapTight">
61     <xsd:sequence>
62         <xsd:element name="wrapPolygon" type="CT_WrapPath" minOccurs="1" maxOccurs="1"/>
63     </xsd:sequence>
64     <xsd:attribute name="wrapText" type="ST_WrapText" use="required"/>
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_WrapThrough">
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_WrapTopBottom">
77     <xsd:sequence>
78         <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
79     </xsd:sequence>
80     <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
81     <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
82 </xsd:complexType>
83 <xsd:group name="EG_WrapType">
84     <xsd:sequence>
85         <xsd:choice minOccurs="1" maxOccurs="1">
86             <xsd:element name="wrapNone" type="CT_WrapNone" minOccurs="1" maxOccurs="1"/>
87             <xsd:element name="wrapSquare" type="CT_WrapSquare" minOccurs="1" maxOccurs="1"/>
88             <xsd:element name="wrapTight" type="CT_WrapTight" minOccurs="1" maxOccurs="1"/>
89             <xsd:element name="wrapThrough" type="CT_WrapThrough" minOccurs="1" maxOccurs="1"/>
90             <xsd:element name="wrapTopAndBottom" type="CT_WrapTopBottom" minOccurs="1"
91                 maxOccurs="1"/>
92         </xsd:choice>
93     </xsd:sequence>
94 </xsd:group>
95 <xsd:simpleType name="ST_PositionOffset">

```

```

96     <xsd:restriction base="xsd:int"/>
97 </xsd:simpleType>
98 <xsd:simpleType name="ST_AlignH">
99     <xsd:restriction base="xsd:token">
100         <xsd:enumeration value="left"/>
101         <xsd:enumeration value="right"/>
102         <xsd:enumeration value="center"/>
103         <xsd:enumeration value="inside"/>
104         <xsd:enumeration value="outside"/>
105     </xsd:restriction>
106 </xsd:simpleType>
107 <xsd:simpleType name="ST_RelFromH">
108     <xsd:restriction base="xsd:token">
109         <xsd:enumeration value="margin"/>
110         <xsd:enumeration value="page"/>
111         <xsd:enumeration value="column"/>
112         <xsd:enumeration value="character"/>
113         <xsd:enumeration value="leftMargin"/>
114         <xsd:enumeration value="rightMargin"/>
115         <xsd:enumeration value="insideMargin"/>
116         <xsd:enumeration value="outsideMargin"/>
117     </xsd:restriction>
118 </xsd:simpleType>
119 <xsd:complexType name="CT_PosH">
120     <xsd:sequence>
121         <xsd:choice minOccurs="1" maxOccurs="1">
122             <xsd:element name="align" type="ST_AlignH" minOccurs="1" maxOccurs="1"/>
123             <xsd:element name="posOffset" type="ST_PositionOffset" minOccurs="1" maxOccurs="1"/>
124         </xsd:choice>
125     </xsd:sequence>
126     <xsd:attribute name="relativeFrom" type="ST_RelFromH" use="required"/>
127 </xsd:complexType>
128 <xsd:simpleType name="ST_AlignV">
129     <xsd:restriction base="xsd:token">
130         <xsd:enumeration value="top"/>
131         <xsd:enumeration value="bottom"/>
132         <xsd:enumeration value="center"/>
133         <xsd:enumeration value="inside"/>
134         <xsd:enumeration value="outside"/>
135     </xsd:restriction>
136 </xsd:simpleType>
137 <xsd:simpleType name="ST_RelFromV">
138     <xsd:restriction base="xsd:token">
139         <xsd:enumeration value="margin"/>
140         <xsd:enumeration value="page"/>
141         <xsd:enumeration value="paragraph"/>
142         <xsd:enumeration value="line"/>
143         <xsd:enumeration value="topMargin"/>
144         <xsd:enumeration value="bottomMargin"/>
145         <xsd:enumeration value="insideMargin"/>
146         <xsd:enumeration value="outsideMargin"/>
147     </xsd:restriction>
148 </xsd:simpleType>

```

```

149 <xsd:complexType name="CT_PosV">
150   <xsd:sequence>
151     <xsd:choice minOccurs="1" maxOccurs="1">
152       <xsd:element name="align" type="ST_AlignV" minOccurs="1" maxOccurs="1"/>
153       <xsd:element name="posOffset" type="ST_PositionOffset" minOccurs="1" maxOccurs="1"/>
154     </xsd:choice>
155   </xsd:sequence>
156   <xsd:attribute name="relativeFrom" type="ST_RelFromV" use="required"/>
157 </xsd:complexType>
158 <xsd:complexType name="CT_Anchor">
159   <xsd:sequence>
160     <xsd:element name="simplePos" type="a:CT_Point2D"/>
161     <xsd:element name="positionH" type="CT_PosH"/>
162     <xsd:element name="positionV" type="CT_PosV"/>
163     <xsd:element name="extent" type="a:CT_PositiveSize2D"/>
164     <xsd:element name="effectExtent" type="CT_EffectExtent" minOccurs="0"/>
165     <xsd:group ref="EG_WrapType"/>
166     <xsd:element name="docPr" type="a:CT_NonVisualDrawingProps" minOccurs="1" maxOccurs="1"/>
167     <xsd:element name="cNvGraphicFramePr" type="a:CT_NonVisualGraphicFrameProperties"
168       minOccurs="0" maxOccurs="1"/>
169     <xsd:element ref="a:graphic" minOccurs="1" maxOccurs="1"/>
170   </xsd:sequence>
171   <xsd:attribute name="distT" type="ST_WrapDistance" use="optional"/>
172   <xsd:attribute name="distB" type="ST_WrapDistance" use="optional"/>
173   <xsd:attribute name="distL" type="ST_WrapDistance" use="optional"/>
174   <xsd:attribute name="distR" type="ST_WrapDistance" use="optional"/>
175   <xsd:attribute name="simplePos" type="xsd:boolean"/>
176   <xsd:attribute name="relativeHeight" type="xsd:unsignedInt" use="required"/>
177   <xsd:attribute name="behindDoc" type="xsd:boolean" use="required"/>
178   <xsd:attribute name="locked" type="xsd:boolean" use="required"/>
179   <xsd:attribute name="layoutInCell" type="xsd:boolean" use="required"/>
180   <xsd:attribute name="hidden" type="xsd:boolean" use="optional"/>
181   <xsd:attribute name="allowOverlap" type="xsd:boolean" use="required"/>
182 </xsd:complexType>
183 <xsd:element name="inline" type="CT_Inline"/>
184 <xsd:element name="anchor" type="CT_Anchor"/>
185 </xsd:schema>

```

A.4.6 DrawingML - SpreadsheetML Drawing

```

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

```

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:simpleType>
204 <xsd:complexType name="CT_HPercent">
205     <xsd:attribute name="val" type="ST_HPercent" default="100"/>
206 </xsd:complexType>
207 <xsd:simpleType name="ST_RotY">
208     <xsd:restriction base="xsd:unsignedShort">
209         <xsd:minInclusive value="0"/>
210         <xsd:maxInclusive value="360"/>
211     </xsd:restriction>
212 </xsd:simpleType>
213 <xsd:complexType name="CT_RotY">
214     <xsd:attribute name="val" type="ST_RotY" default="0"/>
215 </xsd:complexType>
216 <xsd:simpleType name="ST_DepthPercent">
217     <xsd:restriction base="xsd:unsignedShort">
218         <xsd:minInclusive value="20"/>
219         <xsd:maxInclusive value="2000"/>
220     </xsd:restriction>
221 </xsd:simpleType>
222 <xsd:complexType name="CT_DepthPercent">
223     <xsd:attribute name="val" type="ST_DepthPercent" default="100"/>
224 </xsd:complexType>
225 <xsd:simpleType name="ST_Perspective">
226     <xsd:restriction base="xsd:unsignedByte">
227         <xsd:minInclusive value="0"/>
228         <xsd:maxInclusive value="240"/>
229     </xsd:restriction>
230 </xsd:simpleType>
231 <xsd:complexType name="CT_Perspective">
232     <xsd:attribute name="val" type="ST_Perspective" default="30"/>
233 </xsd:complexType>
234 <xsd:complexType name="CT_View3D">
235     <xsd:sequence>
236         <xsd:element name="rotX" type="CT_RotX" minOccurs="0" maxOccurs="1"/>
237         <xsd:element name="hPercent" type="CT_HPercent" minOccurs="0" maxOccurs="1"/>
238         <xsd:element name="rotY" type="CT_RotY" minOccurs="0" maxOccurs="1"/>
239         <xsd:element name="depthPercent" type="CT_DepthPercent" minOccurs="0" maxOccurs="1"/>
240         <xsd:element name="rAngAx" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
241         <xsd:element name="perspective" type="CT_Perspective" minOccurs="0" maxOccurs="1"/>
242         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
243     </xsd:sequence>
244 </xsd:complexType>
245 <xsd:complexType name="CT_Surface">
246     <xsd:sequence>
247         <xsd:element name="thickness" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
248         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>

```



```

249     <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
250     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
251   </xsd:sequence>
252 </xsd:complexType>
253 <xsd:complexType name="CT_DTable">
254   <xsd:sequence>
255     <xsd:element name="showHorzBorder" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
256     <xsd:element name="showVertBorder" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
257     <xsd:element name="showOutline" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
258     <xsd:element name="showKeys" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
259     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
260     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
261     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
262   </xsd:sequence>
263 </xsd:complexType>
264 <xsd:simpleType name="ST_GapAmount">
265   <xsd:restriction base="xsd:unsignedShort">
266     <xsd:minInclusive value="0"/>
267     <xsd:maxInclusive value="500"/>
268   </xsd:restriction>
269 </xsd:simpleType>
270 <xsd:complexType name="CT_GapAmount">
271   <xsd:attribute name="val" type="ST_GapAmount" default="150"/>
272 </xsd:complexType>
273 <xsd:simpleType name="ST_Overlap">
274   <xsd:restriction base="xsd:byte">
275     <xsd:minInclusive value="-100"/>
276     <xsd:maxInclusive value="100"/>
277   </xsd:restriction>
278 </xsd:simpleType>
279 <xsd:complexType name="CT_Overlap">
280   <xsd:attribute name="val" type="ST_Overlap" default="0"/>
281 </xsd:complexType>
282 <xsd:simpleType name="ST_BubbleScale">
283   <xsd:restriction base="xsd:unsignedInt">
284     <xsd:minInclusive value="0"/>
285     <xsd:maxInclusive value="300"/>
286   </xsd:restriction>
287 </xsd:simpleType>
288 <xsd:complexType name="CT_BubbleScale">
289   <xsd:attribute name="val" type="ST_BubbleScale" default="100"/>
290 </xsd:complexType>
291 <xsd:simpleType name="ST_SizeRepresents">
292   <xsd:restriction base="xsd:string">
293     <xsd:enumeration value="area"/>
294     <xsd:enumeration value="w"/>
295   </xsd:restriction>
296 </xsd:simpleType>
297 <xsd:complexType name="CT_SizeRepresents">
298   <xsd:attribute name="val" type="ST_SizeRepresents" default="area"/>
299 </xsd:complexType>
300 <xsd:simpleType name="ST_FirstSliceAng">
301   <xsd:restriction base="xsd:unsignedShort">

```

```

302         <xsd:minInclusive value="0"/>
303         <xsd:maxInclusive value="360"/>
304     </xsd:restriction>
305 </xsd:simpleType>
306 <xsd:complexType name="CT_FirstSliceAng">
307     <xsd:attribute name="val" type="ST_FirstSliceAng" default="0"/>
308 </xsd:complexType>
309 <xsd:simpleType name="ST_HoleSize">
310     <xsd:restriction base="xsd:unsignedByte">
311         <xsd:minInclusive value="10"/>
312         <xsd:maxInclusive value="90"/>
313     </xsd:restriction>
314 </xsd:simpleType>
315 <xsd:complexType name="CT_HoleSize">
316     <xsd:attribute name="val" type="ST_HoleSize" default="10"/>
317 </xsd:complexType>
318 <xsd:simpleType name="ST_SplitType">
319     <xsd:restriction base="xsd:string">
320         <xsd:enumeration value="auto"/>
321         <xsd:enumeration value="cust"/>
322         <xsd:enumeration value="percent"/>
323         <xsd:enumeration value="pos"/>
324         <xsd:enumeration value="val"/>
325     </xsd:restriction>
326 </xsd:simpleType>
327 <xsd:complexType name="CT_SplitType">
328     <xsd:attribute name="val" type="ST_SplitType" default="auto"/>
329 </xsd:complexType>
330 <xsd:complexType name="CT_CustSplit">
331     <xsd:sequence>
332         <xsd:element name="secondPiePt" type="CT_UnsignedInt" minOccurs="0"
333             maxOccurs="unbounded"/>
334     </xsd:sequence>
335 </xsd:complexType>
336 <xsd:simpleType name="ST_SecondPieSize">
337     <xsd:restriction base="xsd:unsignedShort">
338         <xsd:minInclusive value="5"/>
339         <xsd:maxInclusive value="200"/>
340     </xsd:restriction>
341 </xsd:simpleType>
342 <xsd:complexType name="CT_SecondPieSize">
343     <xsd:attribute name="val" type="ST_SecondPieSize" default="75"/>
344 </xsd:complexType>
345 <xsd:complexType name="CT_NumFmt">
346     <xsd:attribute name="formatCode" type="s:ST_Xstring" use="required"/>
347     <xsd:attribute name="sourceLinked" type="xsd:boolean"/>
348 </xsd:complexType>
349 <xsd:simpleType name="ST_LblAlign">
350     <xsd:restriction base="xsd:string">
351         <xsd:enumeration value="ctr"/>
352         <xsd:enumeration value="l"/>
353         <xsd:enumeration value="r"/>
354     </xsd:restriction>

```

```

355 </xsd:simpleType>
356 <xsd:complexType name="CT_LblAlgn">
357   <xsd:attribute name="val" type="ST_LblAlgn" use="required"/>
358 </xsd:complexType>
359 <xsd:simpleType name="ST_DLblPos">
360   <xsd:restriction base="xsd:string">
361     <xsd:enumeration value="bestFit"/>
362     <xsd:enumeration value="b"/>
363     <xsd:enumeration value="ctr"/>
364     <xsd:enumeration value="inBase"/>
365     <xsd:enumeration value="inEnd"/>
366     <xsd:enumeration value="l"/>
367     <xsd:enumeration value="outEnd"/>
368     <xsd:enumeration value="r"/>
369     <xsd:enumeration value="t"/>
370   </xsd:restriction>
371 </xsd:simpleType>
372 <xsd:complexType name="CT_DLblPos">
373   <xsd:attribute name="val" type="ST_DLblPos" use="required"/>
374 </xsd:complexType>
375 <xsd:group name="EG_DLblShared">
376   <xsd:sequence>
377     <xsd:element name="numFmt" type="CT_NumFmt" minOccurs="0" maxOccurs="1"/>
378     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
379     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
380     <xsd:element name="dLblPos" type="CT_DLblPos" minOccurs="0" maxOccurs="1"/>
381     <xsd:element name="showLegendKey" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
382     <xsd:element name="showVal" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
383     <xsd:element name="showCatName" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
384     <xsd:element name="showSerName" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
385     <xsd:element name="showPercent" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
386     <xsd:element name="showBubbleSize" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
387     <xsd:element name="separator" type="xsd:string" minOccurs="0" maxOccurs="1"/>
388   </xsd:sequence>
389 </xsd:group>
390 <xsd:group name="Group_DLbl">
391   <xsd:sequence>
392     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
393     <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
394     <xsd:group ref="EG_DLblShared" minOccurs="1" maxOccurs="1"/>
395   </xsd:sequence>
396 </xsd:group>
397 <xsd:complexType name="CT_DLbl">
398   <xsd:sequence>
399     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
400     <xsd:choice>
401       <xsd:element name="delete" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
402       <xsd:group ref="Group_DLbl" minOccurs="1" maxOccurs="1"/>
403     </xsd:choice>
404     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
405   </xsd:sequence>
406 </xsd:complexType>
407 <xsd:group name="Group_DLbls">

```

```

408     <xsd:sequence>
409         <xsd:group ref="EG_DLblShared" minOccurs="1" maxOccurs="1"/>
410         <xsd:element name="showLeaderLines" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
411         <xsd:element name="leaderLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
412     </xsd:sequence>
413 </xsd:group>
414 <xsd:complexType name="CT_DLbls">
415     <xsd:sequence>
416         <xsd:element name="dLbl" type="CT_DLbl" minOccurs="0" maxOccurs="unbounded"/>
417         <xsd:choice>
418             <xsd:element name="delete" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
419             <xsd:group ref="Group_DLbls" minOccurs="1" maxOccurs="1"/>
420         </xsd:choice>
421         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
422     </xsd:sequence>
423 </xsd:complexType>
424 <xsd:simpleType name="ST_MarkerStyle">
425     <xsd:restriction base="xsd:string">
426         <xsd:enumeration value="circle"/>
427         <xsd:enumeration value="dash"/>
428         <xsd:enumeration value="diamond"/>
429         <xsd:enumeration value="dot"/>
430         <xsd:enumeration value="none"/>
431         <xsd:enumeration value="picture"/>
432         <xsd:enumeration value="plus"/>
433         <xsd:enumeration value="square"/>
434         <xsd:enumeration value="star"/>
435         <xsd:enumeration value="triangle"/>
436         <xsd:enumeration value="x"/>
437     </xsd:restriction>
438 </xsd:simpleType>
439 <xsd:complexType name="CT_MarkerStyle">
440     <xsd:attribute name="val" type="ST_MarkerStyle" use="required"/>
441 </xsd:complexType>
442 <xsd:simpleType name="ST_MarkerSize">
443     <xsd:restriction base="xsd:unsignedByte">
444         <xsd:minInclusive value="2"/>
445         <xsd:maxInclusive value="72"/>
446     </xsd:restriction>
447 </xsd:simpleType>
448 <xsd:complexType name="CT_MarkerSize">
449     <xsd:attribute name="val" type="ST_MarkerSize" default="5"/>
450 </xsd:complexType>
451 <xsd:complexType name="CT_Marker">
452     <xsd:sequence>
453         <xsd:element name="symbol" type="CT_MarkerStyle" minOccurs="0" maxOccurs="1"/>
454         <xsd:element name="size" type="CT_MarkerSize" minOccurs="0" maxOccurs="1"/>
455         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
456         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
457     </xsd:sequence>
458 </xsd:complexType>
459 <xsd:complexType name="CT_DPt">
460     <xsd:sequence>

```

```

461     <xsd:element name="idx" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
462     <xsd:element name="invertIfNegative" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
463     <xsd:element name="marker" type="CT Marker" minOccurs="0" maxOccurs="1"/>
464     <xsd:element name="bubble3D" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
465     <xsd:element name="explosion" type="CT UnsignedInt" minOccurs="0" maxOccurs="1"/>
466     <xsd:element name="spPr" type="a:CT ShapeProperties" minOccurs="0" maxOccurs="1"/>
467     <xsd:element name="pictureOptions" type="CT PictureOptions" minOccurs="0" maxOccurs="1"/>
468     <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
469   </xsd:sequence>
470 </xsd:complexType>
471 <xsd:simpleType name="ST_TrendlineType">
472   <xsd:restriction base="xsd:string">
473     <xsd:enumeration value="exp"/>
474     <xsd:enumeration value="linear"/>
475     <xsd:enumeration value="log"/>
476     <xsd:enumeration value="movingAvg"/>
477     <xsd:enumeration value="poly"/>
478     <xsd:enumeration value="power"/>
479   </xsd:restriction>
480 </xsd:simpleType>
481 <xsd:complexType name="CT_TrendlineType">
482   <xsd:attribute name="val" type="ST TrendlineType" default="linear"/>
483 </xsd:complexType>
484 <xsd:simpleType name="ST_Order">
485   <xsd:restriction base="xsd:unsignedByte">
486     <xsd:minInclusive value="2"/>
487     <xsd:maxInclusive value="6"/>
488   </xsd:restriction>
489 </xsd:simpleType>
490 <xsd:complexType name="CT_Order">
491   <xsd:attribute name="val" type="ST Order" default="2"/>
492 </xsd:complexType>
493 <xsd:simpleType name="ST_Period">
494   <xsd:restriction base="xsd:unsignedByte">
495     <xsd:minInclusive value="2"/>
496     <xsd:maxInclusive value="255"/>
497   </xsd:restriction>
498 </xsd:simpleType>
499 <xsd:complexType name="CT_Period">
500   <xsd:attribute name="val" type="ST Period" default="2"/>
501 </xsd:complexType>
502 <xsd:complexType name="CT_TrendlineLb1">
503   <xsd:sequence>
504     <xsd:element name="layout" type="CT Layout" minOccurs="0" maxOccurs="1"/>
505     <xsd:element name="tx" type="CT Tx" minOccurs="0" maxOccurs="1"/>
506     <xsd:element name="numFmt" type="CT NumFmt" minOccurs="0" maxOccurs="1"/>
507     <xsd:element name="spPr" type="a:CT ShapeProperties" minOccurs="0" maxOccurs="1"/>
508     <xsd:element name="txPr" type="a:CT TextBody" minOccurs="0" maxOccurs="1"/>
509     <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
510   </xsd:sequence>
511 </xsd:complexType>
512 <xsd:complexType name="CT_Trendline">
513   <xsd:sequence>

```

```

514     <xsd:element name="name" type="xsd:string" minOccurs="0" maxOccurs="1"/>
515     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
516     <xsd:element name="trendlineType" type="CT_TrendlineType" minOccurs="1" maxOccurs="1"/>
517     <xsd:element name="order" type="CT_Order" minOccurs="0" maxOccurs="1"/>
518     <xsd:element name="period" type="CT_Period" minOccurs="0" maxOccurs="1"/>
519     <xsd:element name="forward" type="CT_Double" minOccurs="0" maxOccurs="1"/>
520     <xsd:element name="backward" type="CT_Double" minOccurs="0" maxOccurs="1"/>
521     <xsd:element name="intercept" type="CT_Double" minOccurs="0" maxOccurs="1"/>
522     <xsd:element name="dispRSqr" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
523     <xsd:element name="dispEq" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
524     <xsd:element name="trendlineLbl" type="CT_TrendlineLbl" minOccurs="0" maxOccurs="1"/>
525     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
526   </xsd:sequence>
527 </xsd:complexType>
528 <xsd:simpleType name="ST_ErrDir">
529   <xsd:restriction base="xsd:string">
530     <xsd:enumeration value="x"/>
531     <xsd:enumeration value="y"/>
532   </xsd:restriction>
533 </xsd:simpleType>
534 <xsd:complexType name="CT_ErrDir">
535   <xsd:attribute name="val" type="ST_ErrDir" use="required"/>
536 </xsd:complexType>
537 <xsd:simpleType name="ST_ErrBarType">
538   <xsd:restriction base="xsd:string">
539     <xsd:enumeration value="both"/>
540     <xsd:enumeration value="minus"/>
541     <xsd:enumeration value="plus"/>
542   </xsd:restriction>
543 </xsd:simpleType>
544 <xsd:complexType name="CT_ErrBarType">
545   <xsd:attribute name="val" type="ST_ErrBarType" default="both"/>
546 </xsd:complexType>
547 <xsd:simpleType name="ST_ErrValType">
548   <xsd:restriction base="xsd:string">
549     <xsd:enumeration value="cust"/>
550     <xsd:enumeration value="fixedVal"/>
551     <xsd:enumeration value="percentage"/>
552     <xsd:enumeration value="stdDev"/>
553     <xsd:enumeration value="stdErr"/>
554   </xsd:restriction>
555 </xsd:simpleType>
556 <xsd:complexType name="CT_ErrValType">
557   <xsd:attribute name="val" type="ST_ErrValType" default="fixedVal"/>
558 </xsd:complexType>
559 <xsd:complexType name="CT_ErrBars">
560   <xsd:sequence>
561     <xsd:element name="errDir" type="CT_ErrDir" minOccurs="0" maxOccurs="1"/>
562     <xsd:element name="errBarType" type="CT_ErrBarType" minOccurs="1" maxOccurs="1"/>
563     <xsd:element name="errValType" type="CT_ErrValType" minOccurs="1" maxOccurs="1"/>
564     <xsd:element name="noEndCap" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
565     <xsd:element name="plus" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
566     <xsd:element name="minus" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>

```

```

567     <xsd:element name="val" type="CT_Double" minOccurs="0" maxOccurs="1"/>
568     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
569     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
570   </xsd:sequence>
571 </xsd:complexType>
572 <xsd:complexType name="CT_UpDownBar">
573   <xsd:sequence>
574     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
575   </xsd:sequence>
576 </xsd:complexType>
577 <xsd:complexType name="CT_UpDownBars">
578   <xsd:sequence>
579     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
580     <xsd:element name="upBars" type="CT_UpDownBar" minOccurs="0" maxOccurs="1"/>
581     <xsd:element name="downBars" type="CT_UpDownBar" minOccurs="0" maxOccurs="1"/>
582     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
583   </xsd:sequence>
584 </xsd:complexType>
585 <xsd:group name="EG_SerShared">
586   <xsd:sequence>
587     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
588     <xsd:element name="order" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
589     <xsd:element name="tx" type="CT_SerTx" minOccurs="0" maxOccurs="1"/>
590     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
591   </xsd:sequence>
592 </xsd:group>
593 <xsd:complexType name="CT_LineSer">
594   <xsd:sequence>
595     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
596     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
597     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
598     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
599     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
600     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="1"/>
601     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
602     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
603     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
604     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
605   </xsd:sequence>
606 </xsd:complexType>
607 <xsd:complexType name="CT_ScatterSer">
608   <xsd:sequence>
609     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
610     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
611     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
612     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
613     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
614     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
615     <xsd:element name="xVal" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
616     <xsd:element name="yVal" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
617     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
618     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
619   </xsd:sequence>

```

```

620 </xsd:complexType>
621 <xsd:complexType name="CT_RadarSer">
622   <xsd:sequence>
623     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
624     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
625     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
626     <xsd:element name="dLbls" type="CT_DLbIs" minOccurs="0" maxOccurs="1"/>
627     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
628     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
629     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
630   </xsd:sequence>
631 </xsd:complexType>
632 <xsd:complexType name="CT_BarSer">
633   <xsd:sequence>
634     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
635     <xsd:element name="invertIfNegative" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
636     <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
637     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
638     <xsd:element name="dLbls" type="CT_DLbIs" minOccurs="0" maxOccurs="1"/>
639     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
640     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="1"/>
641     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
642     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
643     <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
644     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
645   </xsd:sequence>
646 </xsd:complexType>
647 <xsd:complexType name="CT_AreaSer">
648   <xsd:sequence>
649     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
650     <xsd:element name="pictureOptions" type="CT_PictureOptions" minOccurs="0" maxOccurs="1"/>
651     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
652     <xsd:element name="dLbls" type="CT_DLbIs" minOccurs="0" maxOccurs="1"/>
653     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
654     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
655     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
656     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
657     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
658   </xsd:sequence>
659 </xsd:complexType>
660 <xsd:complexType name="CT_PieSer">
661   <xsd:sequence>
662     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
663     <xsd:element name="explosion" type="CT_UnsignedInt" minOccurs="0" maxOccurs="1"/>
664     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
665     <xsd:element name="dLbls" type="CT_DLbIs" minOccurs="0" maxOccurs="1"/>
666     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
667     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
668     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
669   </xsd:sequence>
670 </xsd:complexType>
671 <xsd:complexType name="CT_BubbleSer">
672   <xsd:sequence>

```



```

673     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
674     <xsd:element name="invertIfNegative" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
675     <xsd:element name="dPt" type="CT_DPt" minOccurs="0" maxOccurs="unbounded"/>
676     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
677     <xsd:element name="trendline" type="CT_Trendline" minOccurs="0" maxOccurs="unbounded"/>
678     <xsd:element name="errBars" type="CT_ErrBars" minOccurs="0" maxOccurs="2"/>
679     <xsd:element name="xVal" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
680     <xsd:element name="yVal" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
681     <xsd:element name="bubbleSize" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
682     <xsd:element name="bubble3D" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
683     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
684   </xsd:sequence>
685 </xsd:complexType>
686 <xsd:complexType name="CT_SurfaceSer">
687   <xsd:sequence>
688     <xsd:group ref="EG_SerShared" minOccurs="1" maxOccurs="1"/>
689     <xsd:element name="cat" type="CT_AxDataSource" minOccurs="0" maxOccurs="1"/>
690     <xsd:element name="val" type="CT_NumDataSource" minOccurs="0" maxOccurs="1"/>
691     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
692   </xsd:sequence>
693 </xsd:complexType>
694 <xsd:simpleType name="ST_Grouping">
695   <xsd:restriction base="xsd:string">
696     <xsd:enumeration value="percentStacked"/>
697     <xsd:enumeration value="standard"/>
698     <xsd:enumeration value="stacked"/>
699   </xsd:restriction>
700 </xsd:simpleType>
701 <xsd:complexType name="CT_Grouping">
702   <xsd:attribute name="val" type="ST_Grouping" default="standard"/>
703 </xsd:complexType>
704 <xsd:complexType name="CT_ChartLines">
705   <xsd:sequence>
706     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
707   </xsd:sequence>
708 </xsd:complexType>
709 <xsd:group name="EG_LineChartShared">
710   <xsd:sequence>
711     <xsd:element name="grouping" type="CT_Grouping" minOccurs="1" maxOccurs="1"/>
712     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
713     <xsd:element name="ser" type="CT_LineSer" minOccurs="0" maxOccurs="unbounded"/>
714     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
715     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
716   </xsd:sequence>
717 </xsd:group>
718 <xsd:complexType name="CT_LineChart">
719   <xsd:sequence>
720     <xsd:group ref="EG_LineChartShared" minOccurs="1" maxOccurs="1"/>
721     <xsd:element name="hiLowLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
722     <xsd:element name="upDownBars" type="CT_UpDownBars" minOccurs="0" maxOccurs="1"/>
723     <xsd:element name="marker" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
724     <xsd:element name="smooth" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
725     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>

```

```

726     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
727   </xsd:sequence>
728 </xsd:complexType>
729 <xsd:complexType name="CT_Line3DChart">
730   <xsd:sequence>
731     <xsd:group ref="EG_LineChartShared" minOccurs="1" maxOccurs="1"/>
732     <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
733     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="3" maxOccurs="3"/>
734     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
735   </xsd:sequence>
736 </xsd:complexType>
737 <xsd:complexType name="CT_StockChart">
738   <xsd:sequence>
739     <xsd:element name="ser" type="CT_LineSer" minOccurs="3" maxOccurs="4"/>
740     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
741     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
742     <xsd:element name="hiLowLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
743     <xsd:element name="upDownBars" type="CT_UpDownBars" minOccurs="0" maxOccurs="1"/>
744     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
745     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
746   </xsd:sequence>
747 </xsd:complexType>
748 <xsd:simpleType name="ST_ScatterStyle">
749   <xsd:restriction base="xsd:string">
750     <xsd:enumeration value="none"/>
751     <xsd:enumeration value="line"/>
752     <xsd:enumeration value="lineMarker"/>
753     <xsd:enumeration value="marker"/>
754     <xsd:enumeration value="smooth"/>
755     <xsd:enumeration value="smoothMarker"/>
756   </xsd:restriction>
757 </xsd:simpleType>
758 <xsd:complexType name="CT_ScatterStyle">
759   <xsd:attribute name="val" type="ST_ScatterStyle" default="marker"/>
760 </xsd:complexType>
761 <xsd:complexType name="CT_ScatterChart">
762   <xsd:sequence>
763     <xsd:element name="scatterStyle" type="CT_ScatterStyle" minOccurs="1" maxOccurs="1"/>
764     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
765     <xsd:element name="ser" type="CT_ScatterSer" minOccurs="0" maxOccurs="unbounded"/>
766     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
767     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
768     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
769   </xsd:sequence>
770 </xsd:complexType>
771 <xsd:simpleType name="ST_RadarStyle">
772   <xsd:restriction base="xsd:string">
773     <xsd:enumeration value="standard"/>
774     <xsd:enumeration value="marker"/>
775     <xsd:enumeration value="filled"/>
776   </xsd:restriction>
777 </xsd:simpleType>
778 <xsd:complexType name="CT_RadarStyle">

```

```

779     <xsd:attribute name="val" type="ST_RadarStyle" default="standard"/>
780 </xsd:complexType>
781 <xsd:complexType name="CT_RadarChart">
782     <xsd:sequence>
783         <xsd:element name="radarStyle" type="CT_RadarStyle" minOccurs="1" maxOccurs="1"/>
784         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
785         <xsd:element name="ser" type="CT_RadarSer" minOccurs="0" maxOccurs="unbounded"/>
786         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
787         <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
788         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
789     </xsd:sequence>
790 </xsd:complexType>
791 <xsd:simpleType name="ST_BarGrouping">
792     <xsd:restriction base="xsd:string">
793         <xsd:enumeration value="percentStacked"/>
794         <xsd:enumeration value="clustered"/>
795         <xsd:enumeration value="standard"/>
796         <xsd:enumeration value="stacked"/>
797     </xsd:restriction>
798 </xsd:simpleType>
799 <xsd:complexType name="CT_BarGrouping">
800     <xsd:attribute name="val" type="ST_BarGrouping" default="clustered"/>
801 </xsd:complexType>
802 <xsd:simpleType name="ST_BarDir">
803     <xsd:restriction base="xsd:string">
804         <xsd:enumeration value="bar"/>
805         <xsd:enumeration value="col"/>
806     </xsd:restriction>
807 </xsd:simpleType>
808 <xsd:complexType name="CT_BarDir">
809     <xsd:attribute name="val" type="ST_BarDir" default="col"/>
810 </xsd:complexType>
811 <xsd:simpleType name="ST_Shape">
812     <xsd:restriction base="xsd:string">
813         <xsd:enumeration value="cone"/>
814         <xsd:enumeration value="coneToMax"/>
815         <xsd:enumeration value="box"/>
816         <xsd:enumeration value="cylinder"/>
817         <xsd:enumeration value="pyramid"/>
818         <xsd:enumeration value="pyramidToMax"/>
819     </xsd:restriction>
820 </xsd:simpleType>
821 <xsd:complexType name="CT_Shape">
822     <xsd:attribute name="val" type="ST_Shape" default="box"/>
823 </xsd:complexType>
824 <xsd:group name="EG_BarChartShared">
825     <xsd:sequence>
826         <xsd:element name="barDir" type="CT_BarDir" minOccurs="1" maxOccurs="1"/>
827         <xsd:element name="grouping" type="CT_BarGrouping" minOccurs="0" maxOccurs="1"/>
828         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
829         <xsd:element name="ser" type="CT_BarSer" minOccurs="0" maxOccurs="unbounded"/>
830         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
831     </xsd:sequence>

```

```

832 </xsd:group>
833 <xsd:complexType name="CT_BarChart">
834   <xsd:sequence>
835     <xsd:group ref="EG_BarChartShared" minOccurs="1" maxOccurs="1"/>
836     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
837     <xsd:element name="overlap" type="CT_Overlap" minOccurs="0" maxOccurs="1"/>
838     <xsd:element name="serLines" type="CT_ChartLines" minOccurs="0" maxOccurs="unbounded"/>
839     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
840     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
841   </xsd:sequence>
842 </xsd:complexType>
843 <xsd:complexType name="CT_Bar3DChart">
844   <xsd:sequence>
845     <xsd:group ref="EG_BarChartShared" minOccurs="1" maxOccurs="1"/>
846     <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
847     <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
848     <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
849     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="3"/>
850     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
851   </xsd:sequence>
852 </xsd:complexType>
853 <xsd:group name="EG_AreaChartShared">
854   <xsd:sequence>
855     <xsd:element name="grouping" type="CT_Grouping" minOccurs="0" maxOccurs="1"/>
856     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
857     <xsd:element name="ser" type="CT_AreaSer" minOccurs="0" maxOccurs="unbounded"/>
858     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
859     <xsd:element name="dropLines" type="CT_ChartLines" minOccurs="0" maxOccurs="1"/>
860   </xsd:sequence>
861 </xsd:group>
862 <xsd:complexType name="CT_AreaChart">
863   <xsd:sequence>
864     <xsd:group ref="EG_AreaChartShared" minOccurs="1" maxOccurs="1"/>
865     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="2"/>
866     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
867   </xsd:sequence>
868 </xsd:complexType>
869 <xsd:complexType name="CT_Area3DChart">
870   <xsd:sequence>
871     <xsd:group ref="EG_AreaChartShared" minOccurs="1" maxOccurs="1"/>
872     <xsd:element name="gapDepth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
873     <xsd:element name="axId" type="CT_UnsignedInt" minOccurs="2" maxOccurs="3"/>
874     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
875   </xsd:sequence>
876 </xsd:complexType>
877 <xsd:group name="EG_PieChartShared">
878   <xsd:sequence>
879     <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
880     <xsd:element name="ser" type="CT_PieSer" minOccurs="0" maxOccurs="unbounded"/>
881     <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
882   </xsd:sequence>
883 </xsd:group>
884 <xsd:complexType name="CT_PieChart">

```

```

885     <xsd:sequence>
886         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
887         <xsd:element name="firstSliceAng" type="CT_FirstSliceAng" minOccurs="0" maxOccurs="1"/>
888         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
889     </xsd:sequence>
890 </xsd:complexType>
891 <xsd:complexType name="CT_Pie3DChart">
892     <xsd:sequence>
893         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
894         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
895     </xsd:sequence>
896 </xsd:complexType>
897 <xsd:complexType name="CT_DoughnutChart">
898     <xsd:sequence>
899         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
900         <xsd:element name="firstSliceAng" type="CT_FirstSliceAng" minOccurs="0" maxOccurs="1"/>
901         <xsd:element name="holeSize" type="CT_HoleSize" minOccurs="0" maxOccurs="1"/>
902         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
903     </xsd:sequence>
904 </xsd:complexType>
905 <xsd:simpleType name="ST_OfPieType">
906     <xsd:restriction base="xsd:string">
907         <xsd:enumeration value="pie"/>
908         <xsd:enumeration value="bar"/>
909     </xsd:restriction>
910 </xsd:simpleType>
911 <xsd:complexType name="CT_OfPieType">
912     <xsd:attribute name="val" type="ST_OfPieType" default="pie"/>
913 </xsd:complexType>
914 <xsd:complexType name="CT_OfPieChart">
915     <xsd:sequence>
916         <xsd:element name="ofPieType" type="CT_OfPieType" minOccurs="1" maxOccurs="1"/>
917         <xsd:group ref="EG_PieChartShared" minOccurs="1" maxOccurs="1"/>
918         <xsd:element name="gapWidth" type="CT_GapAmount" minOccurs="0" maxOccurs="1"/>
919         <xsd:element name="splitType" type="CT_SplitType" minOccurs="0" maxOccurs="1"/>
920         <xsd:element name="splitPos" type="CT_Double" minOccurs="0" maxOccurs="1"/>
921         <xsd:element name="custSplit" type="CT_CustSplit" minOccurs="0" maxOccurs="1"/>
922         <xsd:element name="secondPieSize" type="CT_SecondPieSize" minOccurs="0" maxOccurs="1"/>
923         <xsd:element name="serLines" type="CT_ChartLines" minOccurs="0" maxOccurs="unbounded"/>
924         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
925     </xsd:sequence>
926 </xsd:complexType>
927 <xsd:complexType name="CT_BubbleChart">
928     <xsd:sequence>
929         <xsd:element name="varyColors" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
930         <xsd:element name="ser" type="CT_BubbleSer" minOccurs="0" maxOccurs="unbounded"/>
931         <xsd:element name="dLbls" type="CT_DLbls" minOccurs="0" maxOccurs="1"/>
932         <xsd:element name="bubble3D" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
933         <xsd:element name="bubbleScale" type="CT_BubbleScale" minOccurs="0" maxOccurs="1"/>
934         <xsd:element name="showNegBubbles" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
935         <xsd:element name="sizeRepresents" type="CT_SizeRepresents" minOccurs="0" 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_BandFmt">
941     <xsd:sequence>
942         <xsd:element name="idx" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
943         <xsd:element name="spPr" type="a:CT ShapeProperties" minOccurs="0" maxOccurs="1"/>
944     </xsd:sequence>
945 </xsd:complexType>
946 <xsd:complexType name="CT_BandFmts">
947     <xsd:sequence>
948         <xsd:element name="bandFmt" type="CT_BandFmt" minOccurs="0" maxOccurs="unbounded"/>
949     </xsd:sequence>
950 </xsd:complexType>
951 <xsd:group name="EG_SurfaceChartShared">
952     <xsd:sequence>
953         <xsd:element name="wireframe" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
954         <xsd:element name="ser" type="CT_SurfaceSer" minOccurs="0" maxOccurs="unbounded"/>
955         <xsd:element name="bandFmts" type="CT_BandFmts" minOccurs="0" maxOccurs="1"/>
956     </xsd:sequence>
957 </xsd:group>
958 <xsd:complexType name="CT_SurfaceChart">
959     <xsd:sequence>
960         <xsd:group ref="EG_SurfaceChartShared" minOccurs="1" maxOccurs="1"/>
961         <xsd:element name="axId" type="CT UnsignedInt" minOccurs="2" maxOccurs="3"/>
962         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
963     </xsd:sequence>
964 </xsd:complexType>
965 <xsd:complexType name="CT_Surface3DChart">
966     <xsd:sequence>
967         <xsd:group ref="EG_SurfaceChartShared" minOccurs="1" maxOccurs="1"/>
968         <xsd:element name="axId" type="CT UnsignedInt" minOccurs="3" maxOccurs="3"/>
969         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
970     </xsd:sequence>
971 </xsd:complexType>
972 <xsd:simpleType name="ST_AxPos">
973     <xsd:restriction base="xsd:string">
974         <xsd:enumeration value="b"/>
975         <xsd:enumeration value="l"/>
976         <xsd:enumeration value="r"/>
977         <xsd:enumeration value="t"/>
978     </xsd:restriction>
979 </xsd:simpleType>
980 <xsd:complexType name="CT_AxPos">
981     <xsd:attribute name="val" type="ST_AxPos" use="required"/>
982 </xsd:complexType>
983 <xsd:simpleType name="ST_Crosses">
984     <xsd:restriction base="xsd:string">
985         <xsd:enumeration value="autoZero"/>
986         <xsd:enumeration value="max"/>
987         <xsd:enumeration value="min"/>
988     </xsd:restriction>
989 </xsd:simpleType>
990 <xsd:complexType name="CT_Crosses">

```

```

991     <xsd:attribute name="val" type="ST_Crosses" use="required"/>
992 </xsd:complexType>
993 <xsd:simpleType name="ST_CrossBetween">
994     <xsd:restriction base="xsd:string">
995         <xsd:enumeration value="between"/>
996         <xsd:enumeration value="midCat"/>
997     </xsd:restriction>
998 </xsd:simpleType>
999 <xsd:complexType name="CT_CrossBetween">
1000     <xsd:attribute name="val" type="ST_CrossBetween" use="required"/>
1001 </xsd:complexType>
1002 <xsd:simpleType name="ST_TickMark">
1003     <xsd:restriction base="xsd:string">
1004         <xsd:enumeration value="cross"/>
1005         <xsd:enumeration value="in"/>
1006         <xsd:enumeration value="none"/>
1007         <xsd:enumeration value="out"/>
1008     </xsd:restriction>
1009 </xsd:simpleType>
1010 <xsd:complexType name="CT_TickMark">
1011     <xsd:attribute name="val" type="ST_TickMark" default="cross"/>
1012 </xsd:complexType>
1013 <xsd:simpleType name="ST_TickLblPos">
1014     <xsd:restriction base="xsd:string">
1015         <xsd:enumeration value="high"/>
1016         <xsd:enumeration value="low"/>
1017         <xsd:enumeration value="nextTo"/>
1018         <xsd:enumeration value="none"/>
1019     </xsd:restriction>
1020 </xsd:simpleType>
1021 <xsd:complexType name="CT_TickLblPos">
1022     <xsd:attribute name="val" type="ST_TickLblPos" default="nextTo"/>
1023 </xsd:complexType>
1024 <xsd:simpleType name="ST_Skip">
1025     <xsd:restriction base="xsd:unsignedShort">
1026         <xsd:minInclusive value="1"/>
1027     </xsd:restriction>
1028 </xsd:simpleType>
1029 <xsd:complexType name="CT_Skip">
1030     <xsd:attribute name="val" type="ST_Skip" use="required"/>
1031 </xsd:complexType>
1032 <xsd:simpleType name="ST_TimeUnit">
1033     <xsd:restriction base="xsd:string">
1034         <xsd:enumeration value="days"/>
1035         <xsd:enumeration value="months"/>
1036         <xsd:enumeration value="years"/>
1037     </xsd:restriction>
1038 </xsd:simpleType>
1039 <xsd:complexType name="CT_TimeUnit">
1040     <xsd:attribute name="val" type="ST_TimeUnit" default="days"/>
1041 </xsd:complexType>
1042 <xsd:simpleType name="ST_AxisUnit">
1043     <xsd:restriction base="xsd:double">

```

```

1044     <xsd:minExclusive value="0"/>
1045   </xsd:restriction>
1046 </xsd:simpleType>
1047 <xsd:complexType name="CT_AxisUnit">
1048   <xsd:attribute name="val" type="ST_AxisUnit" use="required"/>
1049 </xsd:complexType>
1050 <xsd:simpleType name="ST_BuiltInUnit">
1051   <xsd:restriction base="xsd:string">
1052     <xsd:enumeration value="hundreds"/>
1053     <xsd:enumeration value="thousands"/>
1054     <xsd:enumeration value="tenThousands"/>
1055     <xsd:enumeration value="hundredThousands"/>
1056     <xsd:enumeration value="millions"/>
1057     <xsd:enumeration value="tenMillions"/>
1058     <xsd:enumeration value="hundredMillions"/>
1059     <xsd:enumeration value="billions"/>
1060     <xsd:enumeration value="trillions"/>
1061   </xsd:restriction>
1062 </xsd:simpleType>
1063 <xsd:complexType name="CT_BuiltInUnit">
1064   <xsd:attribute name="val" type="ST_BuiltInUnit" default="thousands"/>
1065 </xsd:complexType>
1066 <xsd:simpleType name="ST_PictureFormat">
1067   <xsd:restriction base="xsd:string">
1068     <xsd:enumeration value="stretch"/>
1069     <xsd:enumeration value="stack"/>
1070     <xsd:enumeration value="stackScale"/>
1071   </xsd:restriction>
1072 </xsd:simpleType>
1073 <xsd:complexType name="CT_PictureFormat">
1074   <xsd:attribute name="val" type="ST_PictureFormat" use="required"/>
1075 </xsd:complexType>
1076 <xsd:simpleType name="ST_PictureStackUnit">
1077   <xsd:restriction base="xsd:double">
1078     <xsd:minExclusive value="0"/>
1079   </xsd:restriction>
1080 </xsd:simpleType>
1081 <xsd:complexType name="CT_PictureStackUnit">
1082   <xsd:attribute name="val" type="ST_PictureStackUnit" use="required"/>
1083 </xsd:complexType>
1084 <xsd:complexType name="CT_PictureOptions">
1085   <xsd:sequence>
1086     <xsd:element name="applyToFront" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1087     <xsd:element name="applyToSides" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1088     <xsd:element name="applyToEnd" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1089     <xsd:element name="pictureFormat" type="CT_PictureFormat" minOccurs="0" maxOccurs="1"/>
1090     <xsd:element name="pictureStackUnit" type="CT_PictureStackUnit" minOccurs="0"
1091       maxOccurs="1"/>
1092   </xsd:sequence>
1093 </xsd:complexType>
1094 <xsd:complexType name="CT_DispUnitsLbl">
1095   <xsd:sequence>
1096     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>

```



```

1097     <xsd:element name="tx" type="CT_Tx" minOccurs="0" maxOccurs="1"/>
1098     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1099     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1100 </xsd:sequence>
1101 </xsd:complexType>
1102 <xsd:complexType name="CT_DispUnits">
1103     <xsd:sequence>
1104         <xsd:choice>
1105             <xsd:element name="custUnit" type="CT_Double" minOccurs="1" maxOccurs="1"/>
1106             <xsd:element name="builtInUnit" type="CT_BuiltInUnit" minOccurs="1" maxOccurs="1"/>
1107         </xsd:choice>
1108         <xsd:element name="dispUnitsLbl" type="CT_DispUnitsLbl" minOccurs="0" maxOccurs="1"/>
1109         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1110     </xsd:sequence>
1111 </xsd:complexType>
1112 <xsd:simpleType name="ST_Orientation">
1113     <xsd:restriction base="xsd:string">
1114         <xsd:enumeration value="maxMin"/>
1115         <xsd:enumeration value="minMax"/>
1116     </xsd:restriction>
1117 </xsd:simpleType>
1118 <xsd:complexType name="CT_Orientation">
1119     <xsd:attribute name="val" type="ST_Orientation" default="minMax"/>
1120 </xsd:complexType>
1121 <xsd:simpleType name="ST_LogBase">
1122     <xsd:restriction base="xsd:double">
1123         <xsd:minInclusive value="2"/>
1124         <xsd:maxInclusive value="1000"/>
1125     </xsd:restriction>
1126 </xsd:simpleType>
1127 <xsd:complexType name="CT_LogBase">
1128     <xsd:attribute name="val" type="ST_LogBase" use="required"/>
1129 </xsd:complexType>
1130 <xsd:complexType name="CT_Scaling">
1131     <xsd:sequence>
1132         <xsd:element name="logBase" type="CT_LogBase" minOccurs="0" maxOccurs="1"/>
1133         <xsd:element name="orientation" type="CT_Orientation" minOccurs="0" maxOccurs="1"/>
1134         <xsd:element name="max" type="CT_Double" minOccurs="0" maxOccurs="1"/>
1135         <xsd:element name="min" type="CT_Double" minOccurs="0" maxOccurs="1"/>
1136         <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1137     </xsd:sequence>
1138 </xsd:complexType>
1139 <xsd:simpleType name="ST_LblOffset">
1140     <xsd:restriction base="xsd:unsignedShort">
1141         <xsd:minInclusive value="0"/>
1142         <xsd:maxInclusive value="1000"/>
1143     </xsd:restriction>
1144 </xsd:simpleType>
1145 <xsd:complexType name="CT_LblOffset">
1146     <xsd:attribute name="val" type="ST_LblOffset" default="100"/>
1147 </xsd:complexType>
1148 <xsd:group name="EG_AxShared">
1149     <xsd:sequence>

```

```

1150     <xsd:element name="axId" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
1151     <xsd:element name="scaling" type="CT Scaling" minOccurs="1" maxOccurs="1"/>
1152     <xsd:element name="delete" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1153     <xsd:element name="axPos" type="CT AxPos" minOccurs="1" maxOccurs="1"/>
1154     <xsd:element name="majorGridlines" type="CT ChartLines" minOccurs="0" maxOccurs="1"/>
1155     <xsd:element name="minorGridlines" type="CT ChartLines" minOccurs="0" maxOccurs="1"/>
1156     <xsd:element name="title" type="CT Title" minOccurs="0" maxOccurs="1"/>
1157     <xsd:element name="numFmt" type="CT NumFmt" minOccurs="0" maxOccurs="1"/>
1158     <xsd:element name="majorTickMark" type="CT TickMark" minOccurs="0" maxOccurs="1"/>
1159     <xsd:element name="minorTickMark" type="CT TickMark" minOccurs="0" maxOccurs="1"/>
1160     <xsd:element name="tickLblPos" type="CT TickLblPos" minOccurs="0" maxOccurs="1"/>
1161     <xsd:element name="spPr" type="a:CT ShapeProperties" minOccurs="0" maxOccurs="1"/>
1162     <xsd:element name="txPr" type="a:CT TextBody" minOccurs="0" maxOccurs="1"/>
1163     <xsd:element name="crossAx" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
1164     <xsd:choice minOccurs="0" maxOccurs="1">
1165         <xsd:element name="crosses" type="CT Crosses" minOccurs="1" maxOccurs="1"/>
1166         <xsd:element name="crossesAt" type="CT Double" minOccurs="1" maxOccurs="1"/>
1167     </xsd:choice>
1168 </xsd:sequence>
1169 </xsd:group>
1170 <xsd:complexType name="CT_CatAx">
1171     <xsd:sequence>
1172         <xsd:group ref="EG AxShared" minOccurs="1" maxOccurs="1"/>
1173         <xsd:element name="auto" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1174         <xsd:element name="lblAlgn" type="CT LblAlgn" minOccurs="0" maxOccurs="1"/>
1175         <xsd:element name="lblOffset" type="CT LblOffset" minOccurs="0" maxOccurs="1"/>
1176         <xsd:element name="tickLblSkip" type="CT Skip" minOccurs="0" maxOccurs="1"/>
1177         <xsd:element name="tickMarkSkip" type="CT Skip" minOccurs="0" maxOccurs="1"/>
1178         <xsd:element name="noMultiLvlLbl" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1179         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1180     </xsd:sequence>
1181 </xsd:complexType>
1182 <xsd:complexType name="CT_DateAx">
1183     <xsd:sequence>
1184         <xsd:group ref="EG AxShared" minOccurs="1" maxOccurs="1"/>
1185         <xsd:element name="auto" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1186         <xsd:element name="lblOffset" type="CT LblOffset" minOccurs="0" maxOccurs="1"/>
1187         <xsd:element name="baseTimeUnit" type="CT TimeUnit" minOccurs="0" maxOccurs="1"/>
1188         <xsd:element name="majorUnit" type="CT AxisUnit" minOccurs="0" maxOccurs="1"/>
1189         <xsd:element name="majorTimeUnit" type="CT TimeUnit" minOccurs="0" maxOccurs="1"/>
1190         <xsd:element name="minorUnit" type="CT AxisUnit" minOccurs="0" maxOccurs="1"/>
1191         <xsd:element name="minorTimeUnit" type="CT TimeUnit" minOccurs="0" maxOccurs="1"/>
1192         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1193     </xsd:sequence>
1194 </xsd:complexType>
1195 <xsd:complexType name="CT_SerAx">
1196     <xsd:sequence>
1197         <xsd:group ref="EG AxShared" minOccurs="1" maxOccurs="1"/>
1198         <xsd:element name="tickLblSkip" type="CT Skip" minOccurs="0" maxOccurs="1"/>
1199         <xsd:element name="tickMarkSkip" type="CT Skip" minOccurs="0" maxOccurs="1"/>
1200         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1201     </xsd:sequence>
1202 </xsd:complexType>

```

```

1203 <xsd:complexType name="CT_ValAx">
1204   <xsd:sequence>
1205     <xsd:group ref="EG_AxShared" minOccurs="1" maxOccurs="1"/>
1206     <xsd:element name="crossBetween" type="CT_CrossBetween" minOccurs="0" maxOccurs="1"/>
1207     <xsd:element name="majorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1208     <xsd:element name="minorUnit" type="CT_AxisUnit" minOccurs="0" maxOccurs="1"/>
1209     <xsd:element name="dispUnits" type="CT_DispUnits" minOccurs="0" maxOccurs="1"/>
1210     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1211   </xsd:sequence>
1212 </xsd:complexType>
1213 <xsd:complexType name="CT_PlotArea">
1214   <xsd:sequence>
1215     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
1216     <xsd:choice minOccurs="1" maxOccurs="unbounded">
1217       <xsd:element name="areaChart" type="CT_AreaChart" minOccurs="1" maxOccurs="1"/>
1218       <xsd:element name="area3DChart" type="CT_Area3DChart" minOccurs="1" maxOccurs="1"/>
1219       <xsd:element name="lineChart" type="CT_LineChart" minOccurs="1" maxOccurs="1"/>
1220       <xsd:element name="line3DChart" type="CT_Line3DChart" minOccurs="1" maxOccurs="1"/>
1221       <xsd:element name="stockChart" type="CT_StockChart" minOccurs="1" maxOccurs="1"/>
1222       <xsd:element name="radarChart" type="CT_RadarChart" minOccurs="1" maxOccurs="1"/>
1223       <xsd:element name="scatterChart" type="CT_ScatterChart" minOccurs="1" maxOccurs="1"/>
1224       <xsd:element name="pieChart" type="CT_PieChart" minOccurs="1" maxOccurs="1"/>
1225       <xsd:element name="pie3DChart" type="CT_Pie3DChart" minOccurs="1" maxOccurs="1"/>
1226       <xsd:element name="doughnutChart" type="CT_DoughnutChart" minOccurs="1" maxOccurs="1"/>
1227       <xsd:element name="barChart" type="CT_BarChart" minOccurs="1" maxOccurs="1"/>
1228       <xsd:element name="bar3DChart" type="CT_Bar3DChart" minOccurs="1" maxOccurs="1"/>
1229       <xsd:element name="ofPieChart" type="CT_OfPieChart" minOccurs="1" maxOccurs="1"/>
1230       <xsd:element name="surfaceChart" type="CT_SurfaceChart" minOccurs="1" maxOccurs="1"/>
1231       <xsd:element name="surface3DChart" type="CT_Surface3DChart" minOccurs="1"
1232         maxOccurs="1"/>
1233       <xsd:element name="bubbleChart" type="CT_BubbleChart" minOccurs="1" maxOccurs="1"/>
1234     </xsd:choice>
1235     <xsd:choice minOccurs="0" maxOccurs="unbounded">
1236       <xsd:element name="valAx" type="CT_ValAx" minOccurs="1" maxOccurs="1"/>
1237       <xsd:element name="catAx" type="CT_CatAx" minOccurs="1" maxOccurs="1"/>
1238       <xsd:element name="dateAx" type="CT_DateAx" minOccurs="1" maxOccurs="1"/>
1239       <xsd:element name="serAx" type="CT_SerAx" minOccurs="1" maxOccurs="1"/>
1240     </xsd:choice>
1241     <xsd:element name="dTable" type="CT_DTable" minOccurs="0" maxOccurs="1"/>
1242     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1243     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1244   </xsd:sequence>
1245 </xsd:complexType>
1246 <xsd:complexType name="CT_PivotFmt">
1247   <xsd:sequence>
1248     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1249     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1250     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1251     <xsd:element name="marker" type="CT_Marker" minOccurs="0" maxOccurs="1"/>
1252     <xsd:element name="dLbl" type="CT_DLbl" minOccurs="0" maxOccurs="1"/>
1253     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1254   </xsd:sequence>
1255 </xsd:complexType>

```

```

1256 <xsd:complexType name="CT_PivotFmts">
1257   <xsd:sequence>
1258     <xsd:element name="pivotFmt" type="CT_PivotFmt" minOccurs="0" maxOccurs="unbounded"/>
1259   </xsd:sequence>
1260 </xsd:complexType>
1261 <xsd:simpleType name="ST_LegendPos">
1262   <xsd:restriction base="xsd:string">
1263     <xsd:enumeration value="b"/>
1264     <xsd:enumeration value="tr"/>
1265     <xsd:enumeration value="l"/>
1266     <xsd:enumeration value="r"/>
1267     <xsd:enumeration value="t"/>
1268   </xsd:restriction>
1269 </xsd:simpleType>
1270 <xsd:complexType name="CT_LegendPos">
1271   <xsd:attribute name="val" type="ST_LegendPos" default="r"/>
1272 </xsd:complexType>
1273 <xsd:group name="EG_LegendEntryData">
1274   <xsd:sequence>
1275     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1276   </xsd:sequence>
1277 </xsd:group>
1278 <xsd:complexType name="CT_LegendEntry">
1279   <xsd:sequence>
1280     <xsd:element name="idx" type="CT_UnsignedInt" minOccurs="1" maxOccurs="1"/>
1281     <xsd:choice>
1282       <xsd:element name="delete" type="CT_Boolean" minOccurs="1" maxOccurs="1"/>
1283       <xsd:group ref="EG_LegendEntryData" minOccurs="1" maxOccurs="1"/>
1284     </xsd:choice>
1285     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1286   </xsd:sequence>
1287 </xsd:complexType>
1288 <xsd:complexType name="CT_Legend">
1289   <xsd:sequence>
1290     <xsd:element name="legendPos" type="CT_LegendPos" minOccurs="0" maxOccurs="1"/>
1291     <xsd:element name="legendEntry" type="CT_LegendEntry" minOccurs="0"
1292       maxOccurs="unbounded"/>
1293     <xsd:element name="layout" type="CT_Layout" minOccurs="0" maxOccurs="1"/>
1294     <xsd:element name="overlay" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1295     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1296     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1297     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1298   </xsd:sequence>
1299 </xsd:complexType>
1300 <xsd:simpleType name="ST_DisbBlanksAs">
1301   <xsd:restriction base="xsd:string">
1302     <xsd:enumeration value="span"/>
1303     <xsd:enumeration value="gap"/>
1304     <xsd:enumeration value="zero"/>
1305   </xsd:restriction>
1306 </xsd:simpleType>
1307 <xsd:complexType name="CT_DisbBlanksAs">
1308   <xsd:attribute name="val" type="ST_DisbBlanksAs" default="zero"/>

```

```

1309 </xsd:complexType>
1310 <xsd:complexType name="CT_Chart">
1311     <xsd:sequence>
1312         <xsd:element name="title" type="CT Title" minOccurs="0" maxOccurs="1"/>
1313         <xsd:element name="autoTitleDeleted" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1314         <xsd:element name="pivotFmts" type="CT PivotFmts" minOccurs="0" maxOccurs="1"/>
1315         <xsd:element name="view3D" type="CT View3D" minOccurs="0" maxOccurs="1"/>
1316         <xsd:element name="floor" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1317         <xsd:element name="sideWall" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1318         <xsd:element name="backWall" type="CT Surface" minOccurs="0" maxOccurs="1"/>
1319         <xsd:element name="plotArea" type="CT PlotArea" minOccurs="1" maxOccurs="1"/>
1320         <xsd:element name="legend" type="CT Legend" minOccurs="0" maxOccurs="1"/>
1321         <xsd:element name="plotVisOnly" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1322         <xsd:element name="dispBlanksAs" type="CT DispBlanksAs" minOccurs="0" maxOccurs="1"/>
1323         <xsd:element name="showDLblsOverMax" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1324         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="1"/>
1325     </xsd:sequence>
1326 </xsd:complexType>
1327 <xsd:simpleType name="ST_Style">
1328     <xsd:restriction base="xsd:unsignedByte">
1329         <xsd:minInclusive value="1"/>
1330         <xsd:maxInclusive value="48"/>
1331     </xsd:restriction>
1332 </xsd:simpleType>
1333 <xsd:complexType name="CT_Style">
1334     <xsd:attribute name="val" type="ST Style" use="required"/>
1335 </xsd:complexType>
1336 <xsd:complexType name="CT_PivotSource">
1337     <xsd:sequence>
1338         <xsd:element name="name" type="s:ST Xstring" minOccurs="1" maxOccurs="1"/>
1339         <xsd:element name="fmtId" type="CT UnsignedInt" minOccurs="1" maxOccurs="1"/>
1340         <xsd:element name="extLst" type="CT ExtensionList" minOccurs="0" maxOccurs="unbounded"/>
1341     </xsd:sequence>
1342 </xsd:complexType>
1343 <xsd:complexType name="CT_Protection">
1344     <xsd:sequence>
1345         <xsd:element name="chartObject" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1346         <xsd:element name="data" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1347         <xsd:element name="formatting" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1348         <xsd:element name="selection" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1349         <xsd:element name="userInterface" type="CT Boolean" minOccurs="0" maxOccurs="1"/>
1350     </xsd:sequence>
1351 </xsd:complexType>
1352 <xsd:complexType name="CT_HeaderFooter">
1353     <xsd:sequence>
1354         <xsd:element name="oddHeader" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1355         <xsd:element name="oddFooter" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1356         <xsd:element name="evenHeader" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1357         <xsd:element name="evenFooter" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1358         <xsd:element name="firstHeader" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1359         <xsd:element name="firstFooter" type="s:ST Xstring" minOccurs="0" maxOccurs="1"/>
1360     </xsd:sequence>
1361     <xsd:attribute name="alignWithMargins" type="xsd:boolean" default="true"/>

```

```

1362     <xsd:attribute name="differentOddEven" type="xsd:boolean" default="false"/>
1363     <xsd:attribute name="differentFirst" type="xsd:boolean" default="false"/>
1364 </xsd:complexType>
1365 <xsd:complexType name="CT_PageMargins">
1366     <xsd:attribute name="l" type="xsd:double" use="required"/>
1367     <xsd:attribute name="r" type="xsd:double" use="required"/>
1368     <xsd:attribute name="t" type="xsd:double" use="required"/>
1369     <xsd:attribute name="b" type="xsd:double" use="required"/>
1370     <xsd:attribute name="header" type="xsd:double" use="required"/>
1371     <xsd:attribute name="footer" type="xsd:double" use="required"/>
1372 </xsd:complexType>
1373 <xsd:simpleType name="ST_PageSetupOrientation">
1374     <xsd:restriction base="xsd:string">
1375         <xsd:enumeration value="default"/>
1376         <xsd:enumeration value="portrait"/>
1377         <xsd:enumeration value="landscape"/>
1378     </xsd:restriction>
1379 </xsd:simpleType>
1380 <xsd:complexType name="CT_ExternalData">
1381     <xsd:sequence>
1382         <xsd:element name="autoUpdate" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1383     </xsd:sequence>
1384     <xsd:attribute ref="r:id" use="required"/>
1385 </xsd:complexType>
1386 <xsd:complexType name="CT_PageSetup">
1387     <xsd:attribute name="paperSize" type="xsd:unsignedInt" use="optional" default="1"/>
1388     <xsd:attribute name="paperHeight" type="s:ST_PositiveUniversalMeasure" use="optional"/>
1389     <xsd:attribute name="paperWidth" type="s:ST_PositiveUniversalMeasure" use="optional"/>
1390     <xsd:attribute name="firstPageNumber" type="xsd:unsignedInt" use="optional" default="1"/>
1391     <xsd:attribute name="orientation" type="ST_PageSetupOrientation" use="optional"
1392         default="default"/>
1393     <xsd:attribute name="blackAndWhite" type="xsd:boolean" use="optional" default="false"/>
1394     <xsd:attribute name="draft" type="xsd:boolean" use="optional" default="false"/>
1395     <xsd:attribute name="useFirstPageNumber" type="xsd:boolean" use="optional" default="false"/>
1396     <xsd:attribute name="horizontalDpi" type="xsd:int" use="optional" default="600"/>
1397     <xsd:attribute name="verticalDpi" type="xsd:int" use="optional" default="600"/>
1398     <xsd:attribute name="copies" type="xsd:unsignedInt" use="optional" default="1"/>
1399 </xsd:complexType>
1400 <xsd:complexType name="CT_PrintSettings">
1401     <xsd:sequence>
1402         <xsd:element name="headerFooter" type="CT_HeaderFooter" minOccurs="0" maxOccurs="1"/>
1403         <xsd:element name="pageMargins" type="CT_PageMargins" minOccurs="0" maxOccurs="1"/>
1404         <xsd:element name="pageSetup" type="CT_PageSetup" minOccurs="0" maxOccurs="1"/>
1405         <xsd:element name="legacyDrawingHF" type="CT_RelId" minOccurs="0" maxOccurs="1"/>
1406     </xsd:sequence>
1407 </xsd:complexType>
1408 <xsd:complexType name="CT_ChartSpace">
1409     <xsd:sequence>
1410         <xsd:element name="date1904" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1411         <xsd:element name="lang" type="CT_TextLanguageID" minOccurs="0" maxOccurs="1"/>
1412         <xsd:element name="roundedCorners" type="CT_Boolean" minOccurs="0" maxOccurs="1"/>
1413         <xsd:element name="style" type="CT_Style" minOccurs="0" maxOccurs="1"/>
1414         <xsd:element name="clrMapOvr" type="a:CT_ColorMapping" minOccurs="0" maxOccurs="1"/>

```

```

1415     <xsd:element name="pivotSource" type="CT_PivotSource" minOccurs="0" maxOccurs="1"/>
1416     <xsd:element name="protection" type="CT_Protection" minOccurs="0" maxOccurs="1"/>
1417     <xsd:element name="chart" type="CT_Chart" minOccurs="1" maxOccurs="1"/>
1418     <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
1419     <xsd:element name="txPr" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
1420     <xsd:element name="externalData" type="CT_ExternalData" minOccurs="0" maxOccurs="1"/>
1421     <xsd:element name="printSettings" type="CT_PrintSettings" minOccurs="0" maxOccurs="1"/>
1422     <xsd:element name="userShapes" type="CT_RelId" minOccurs="0" maxOccurs="1"/>
1423     <xsd:element name="extLst" type="CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
1424   </xsd:sequence>
1425 </xsd:complexType>
1426 <xsd:element name="chartSpace" type="CT_ChartSpace"/>
1427 <xsd:element name="userShapes" type="cdr:CT_Drawing"/>
1428 <xsd:element name="chart" type="CT_RelId"/>
1429 </xsd:schema>

```

A.5.2 DrawingML - Chart Drawings

```

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

```

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 id="s" 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         default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
76 </xsd:complexType>
77 <xsd:element name="colorsDef" type="CT_ColorTransform"/>
78 <xsd:complexType name="CT_ColorTransformHeader">
79     <xsd:sequence>
80         <xsd:element name="title" type="CT_CTName" minOccurs="1" maxOccurs="unbounded"/>
81         <xsd:element name="desc" type="CT_CTDescription" minOccurs="1" maxOccurs="unbounded"/>
82         <xsd:element name="catLst" type="CT_CTCategories" minOccurs="0"/>
83         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
84             maxOccurs="1"/>
85     </xsd:sequence>
86     <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
87     <xsd:attribute name="minVer" type="xsd:string" use="optional"
88         default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
89     <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
90 </xsd:complexType>
91 <xsd:element name="colorsDefHdr" type="CT_ColorTransformHeader"/>
92 <xsd:complexType name="CT_ColorTransformHeaderLst">
93     <xsd:sequence>
94         <xsd:element name="colorsDefHdr" type="CT_ColorTransformHeader" minOccurs="0"
95             maxOccurs="unbounded"/>
96     </xsd:sequence>
97 </xsd:complexType>
98 <xsd:element name="colorsDefHdrLst" type="CT_ColorTransformHeaderLst"/>
99 <xsd:simpleType name="ST_PtType">

```

```

100     <xsd:restriction base="xsd:token">
101         <xsd:enumeration value="node"/>
102         <xsd:enumeration value="asst"/>
103         <xsd:enumeration value="doc"/>
104         <xsd:enumeration value="pres"/>
105         <xsd:enumeration value="parTrans"/>
106         <xsd:enumeration value="sibTrans"/>
107     </xsd:restriction>
108 </xsd:simpleType>
109 <xsd:complexType name="CT_Pt">
110     <xsd:sequence>
111         <xsd:element name="prSet" type="CT_ElemPropSet" minOccurs="0" maxOccurs="1"/>
112         <xsd:element name="spPr" type="a:CT_ShapeProperties" minOccurs="0" maxOccurs="1"/>
113         <xsd:element name="t" type="a:CT_TextBody" minOccurs="0" maxOccurs="1"/>
114         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
115             maxOccurs="1"/>
116     </xsd:sequence>
117     <xsd:attribute name="modelId" type="ST_ModelId" use="required"/>
118     <xsd:attribute name="type" type="ST_PtType" use="optional" default="node"/>
119     <xsd:attribute name="cxnId" type="ST_ModelId" use="optional" default="0"/>
120 </xsd:complexType>
121 <xsd:complexType name="CT_PtList">
122     <xsd:sequence>
123         <xsd:element name="pt" type="CT_Pt" minOccurs="0" maxOccurs="unbounded"/>
124     </xsd:sequence>
125 </xsd:complexType>
126 <xsd:simpleType name="ST_CxnType">
127     <xsd:restriction base="xsd:token">
128         <xsd:enumeration value="parOf"/>
129         <xsd:enumeration value="presOf"/>
130         <xsd:enumeration value="presParOf"/>
131         <xsd:enumeration value="unknownRelationship"/>
132     </xsd:restriction>
133 </xsd:simpleType>
134 <xsd:complexType name="CT_Cxn">
135     <xsd:sequence>
136         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
137             maxOccurs="1"/>
138     </xsd:sequence>
139     <xsd:attribute name="modelId" type="ST_ModelId" use="required"/>
140     <xsd:attribute name="type" type="ST_CxnType" use="optional" default="parOf"/>
141     <xsd:attribute name="srcId" type="ST_ModelId" use="required"/>
142     <xsd:attribute name="destId" type="ST_ModelId" use="required"/>
143     <xsd:attribute name="srcOrd" type="xsd:unsignedInt" use="required"/>
144     <xsd:attribute name="destOrd" type="xsd:unsignedInt" use="required"/>
145     <xsd:attribute name="parTransId" type="ST_ModelId" use="optional" default="0"/>
146     <xsd:attribute name="sibTransId" type="ST_ModelId" use="optional" default="0"/>
147     <xsd:attribute name="presId" type="xsd:string" use="optional" default=""/>
148 </xsd:complexType>
149 <xsd:complexType name="CT_CxnList">
150     <xsd:sequence>
151         <xsd:element name="cxn" type="CT_Cxn" minOccurs="0" maxOccurs="unbounded"/>
152     </xsd:sequence>

```

```

153 </xsd:complexType>
154 <xsd:complexType name="CT_DataModel">
155     <xsd:sequence>
156         <xsd:element name="ptLst" type="CT_PtList"/>
157         <xsd:element name="cxnLst" type="CT_CxnList" minOccurs="0" maxOccurs="1"/>
158         <xsd:element name="bg" type="a:CT_BackgroundFormatting" minOccurs="0"/>
159         <xsd:element name="whole" type="a:CT_WholeE2oFormatting" minOccurs="0"/>
160         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
161             maxOccurs="1"/>
162     </xsd:sequence>
163 </xsd:complexType>
164 <xsd:element name="dataModel" type="CT_DataModel"/>
165 <xsd:attributeGroup name="AG_IteratorAttributes">
166     <xsd:attribute name="axis" type="ST_AxisTypes" use="optional" default="none"/>
167     <xsd:attribute name="ptType" type="ST_ElementTypes" use="optional" default="all"/>
168     <xsd:attribute name="hideLastTrans" type="ST_Booleans" use="optional" default="true"/>
169     <xsd:attribute name="st" type="ST_Ints" use="optional" default="1"/>
170     <xsd:attribute name="cnt" type="ST_UnsignedInts" use="optional" default="0"/>
171     <xsd:attribute name="step" type="ST_Ints" use="optional" default="1"/>
172 </xsd:attributeGroup>
173 <xsd:attributeGroup name="AG_ConstraintAttributes">
174     <xsd:attribute name="type" type="ST_ConstraintType" use="required"/>
175     <xsd:attribute name="for" type="ST_ConstraintRelationship" use="optional" default="self"/>
176     <xsd:attribute name="forName" type="xsd:string" use="optional" default=""/>
177     <xsd:attribute name="ptType" type="ST_ElementType" use="optional" default="all"/>
178 </xsd:attributeGroup>
179 <xsd:attributeGroup name="AG_ConstraintRefAttributes">
180     <xsd:attribute name="refType" type="ST_ConstraintType" use="optional" default="none"/>
181     <xsd:attribute name="refFor" type="ST_ConstraintRelationship" use="optional" default="self"/>
182     <xsd:attribute name="refForName" type="xsd:string" use="optional" default=""/>
183     <xsd:attribute name="refPtType" type="ST_ElementType" use="optional" default="all"/>
184 </xsd:attributeGroup>
185 <xsd:complexType name="CT_Constraint">
186     <xsd:sequence>
187         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
188             maxOccurs="1"/>
189     </xsd:sequence>
190     <xsd:attributeGroup ref="AG_ConstraintAttributes"/>
191     <xsd:attributeGroup ref="AG_ConstraintRefAttributes"/>
192     <xsd:attribute name="op" type="ST_BoolOperator" use="optional" default="none"/>
193     <xsd:attribute name="val" type="xsd:double" use="optional" default="0"/>
194     <xsd:attribute name="fact" type="xsd:double" use="optional" default="1"/>
195 </xsd:complexType>
196 <xsd:complexType name="CT_Constraints">
197     <xsd:sequence>
198         <xsd:element name="constr" type="CT_Constraint" minOccurs="0" maxOccurs="unbounded"/>
199     </xsd:sequence>
200 </xsd:complexType>
201 <xsd:complexType name="CT_NumericRule">
202     <xsd:sequence>
203         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
204             maxOccurs="1"/>
205     </xsd:sequence>

```

```

206     <xsd:attributeGroup ref="AG_ConstraintAttributes"/>
207     <xsd:attribute name="val" type="xsd:double" use="optional" default="NaN"/>
208     <xsd:attribute name="fact" type="xsd:double" use="optional" default="NaN"/>
209     <xsd:attribute name="max" type="xsd:double" use="optional" default="NaN"/>
210 </xsd:complexType>
211 <xsd:complexType name="CT_Rules">
212     <xsd:sequence>
213         <xsd:element name="rule" type="CT_NumericRule" minOccurs="0" maxOccurs="unbounded"/>
214     </xsd:sequence>
215 </xsd:complexType>
216 <xsd:complexType name="CT_PresentationOf">
217     <xsd:sequence>
218         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
219             maxOccurs="1"/>
220     </xsd:sequence>
221     <xsd:attributeGroup ref="AG_IteratorAttributes"/>
222 </xsd:complexType>
223 <xsd:simpleType name="ST_LayoutShapeType" final="restriction">
224     <xsd:union memberTypes="a:ST_ShapeType ST_OutputShapeType"/>
225 </xsd:simpleType>
226 <xsd:simpleType name="ST_Index1">
227     <xsd:restriction base="xsd:unsignedInt">
228         <xsd:minInclusive value="1"/>
229     </xsd:restriction>
230 </xsd:simpleType>
231 <xsd:complexType name="CT_Adj">
232     <xsd:attribute name="idx" type="ST_Index1" use="required"/>
233     <xsd:attribute name="val" type="xsd:double" use="required"/>
234 </xsd:complexType>
235 <xsd:complexType name="CT_AdjLst">
236     <xsd:sequence>
237         <xsd:element name="adj" type="CT_Adj" minOccurs="0" maxOccurs="unbounded"/>
238     </xsd:sequence>
239 </xsd:complexType>
240 <xsd:complexType name="CT_Shape">
241     <xsd:sequence>
242         <xsd:element name="adjLst" type="CT_AdjLst" minOccurs="0" maxOccurs="1"/>
243         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
244             maxOccurs="1"/>
245     </xsd:sequence>
246     <xsd:attribute name="rot" type="xsd:double" use="optional" default="0"/>
247     <xsd:attribute name="type" type="ST_LayoutShapeType" use="optional" default="none"/>
248     <xsd:attribute ref="r:blip" use="optional"/>
249     <xsd:attribute name="zOrderOff" type="xsd:int" use="optional" default="0"/>
250     <xsd:attribute name="hideGeom" type="xsd:boolean" use="optional" default="false"/>
251     <xsd:attribute name="lktXEntry" type="xsd:boolean" use="optional" default="false"/>
252     <xsd:attribute name="blipPhldr" type="xsd:boolean" use="optional" default="false"/>
253 </xsd:complexType>
254 <xsd:complexType name="CT_Parameter">
255     <xsd:attribute name="type" type="ST_ParameterId" use="required"/>
256     <xsd:attribute name="val" type="ST_ParameterVal" use="required"/>
257 </xsd:complexType>
258 <xsd:complexType name="CT_Algorithm">

```

```

259     <xsd:sequence>
260         <xsd:element name="param" type="CT_Parameter" minOccurs="0" maxOccurs="unbounded"/>
261         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
262             maxOccurs="1"/>
263     </xsd:sequence>
264     <xsd:attribute name="type" type="ST_AlgorithmType" use="required"/>
265     <xsd:attribute name="rev" type="xsd:unsignedInt" use="optional" default="0"/>
266 </xsd:complexType>
267 <xsd:complexType name="CT_LayoutNode">
268     <xsd:choice minOccurs="0" maxOccurs="unbounded">
269         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
270         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
271         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
272         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
273         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
274         <xsd:element name="varLst" type="CT_LayoutVariablePropertySet" minOccurs="0"
275             maxOccurs="1"/>
276         <xsd:element name="forEach" type="CT_ForEach"/>
277         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
278         <xsd:element name="choose" type="CT_Choose"/>
279         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
280             maxOccurs="1"/>
281     </xsd:choice>
282     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
283     <xsd:attribute name="styleLbl" type="xsd:string" use="optional" default=""/>
284     <xsd:attribute name="chOrder" type="ST_ChildOrderType" use="optional" default="b"/>
285     <xsd:attribute name="moveWith" type="xsd:string" use="optional" default=""/>
286 </xsd:complexType>
287 <xsd:complexType name="CT_ForEach">
288     <xsd:choice minOccurs="0" maxOccurs="unbounded">
289         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
290         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
291         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
292         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
293         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
294         <xsd:element name="forEach" type="CT_ForEach"/>
295         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
296         <xsd:element name="choose" type="CT_Choose"/>
297         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
298             maxOccurs="1"/>
299     </xsd:choice>
300     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
301     <xsd:attribute name="ref" type="xsd:string" use="optional" default=""/>
302     <xsd:attributeGroup ref="AG_IteratorAttributes"/>
303 </xsd:complexType>
304 <xsd:complexType name="CT_When">
305     <xsd:choice minOccurs="0" maxOccurs="unbounded">
306         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
307         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
308         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
309         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
310         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
311         <xsd:element name="forEach" type="CT_ForEach"/>

```

```

312     <xsd:element name="layoutNode" type="CT_LayoutNode"/>
313     <xsd:element name="choose" type="CT_Choose"/>
314     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
315         maxOccurs="1"/>
316 </xsd:choice>
317 <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
318 <xsd:attributeGroup ref="AG_IteratorAttributes"/>
319 <xsd:attribute name="func" type="ST_FunctionType" use="required"/>
320 <xsd:attribute name="arg" type="ST_FunctionArgument" use="optional" default="none"/>
321 <xsd:attribute name="op" type="ST_FunctionOperator" use="required"/>
322 <xsd:attribute name="val" type="ST_FunctionValue" use="required"/>
323 </xsd:complexType>
324 <xsd:complexType name="CT_Otherwise">
325     <xsd:choice minOccurs="0" maxOccurs="unbounded">
326         <xsd:element name="alg" type="CT_Algorithm" minOccurs="0" maxOccurs="1"/>
327         <xsd:element name="shape" type="CT_Shape" minOccurs="0" maxOccurs="1"/>
328         <xsd:element name="presOf" type="CT_PresentationOf" minOccurs="0" maxOccurs="1"/>
329         <xsd:element name="constrLst" type="CT_Constraints" minOccurs="0" maxOccurs="1"/>
330         <xsd:element name="ruleLst" type="CT_Rules" minOccurs="0" maxOccurs="1"/>
331         <xsd:element name="forEach" type="CT_ForEach"/>
332         <xsd:element name="layoutNode" type="CT_LayoutNode"/>
333         <xsd:element name="choose" type="CT_Choose"/>
334         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
335             maxOccurs="1"/>
336     </xsd:choice>
337     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
338 </xsd:complexType>
339 <xsd:complexType name="CT_Choose">
340     <xsd:sequence>
341         <xsd:element name="if" type="CT_When" maxOccurs="unbounded"/>
342         <xsd:element name="else" type="CT_Otherwise" minOccurs="0"/>
343     </xsd:sequence>
344     <xsd:attribute name="name" type="xsd:string" use="optional" default=""/>
345 </xsd:complexType>
346 <xsd:complexType name="CT_SampleData">
347     <xsd:sequence>
348         <xsd:element name="dataModel" type="CT_DataModel" minOccurs="0"/>
349     </xsd:sequence>
350     <xsd:attribute name="useDef" type="xsd:boolean" use="optional" default="false"/>
351 </xsd:complexType>
352 <xsd:complexType name="CT_Category">
353     <xsd:attribute name="type" type="xsd:anyURI" use="required"/>
354     <xsd:attribute name="pri" type="xsd:unsignedInt" use="required"/>
355 </xsd:complexType>
356 <xsd:complexType name="CT_Categories">
357     <xsd:sequence>
358         <xsd:element name="cat" type="CT_Category" minOccurs="0" maxOccurs="unbounded"/>
359     </xsd:sequence>
360 </xsd:complexType>
361 <xsd:complexType name="CT_Name">
362     <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
363     <xsd:attribute name="val" type="xsd:string" use="required"/>
364 </xsd:complexType>

```



```

365 <xsd:complexType name="CT_Description">
366   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
367   <xsd:attribute name="val" type="xsd:string" use="required"/>
368 </xsd:complexType>
369 <xsd:complexType name="CT_DiagramDefinition">
370   <xsd:sequence>
371     <xsd:element name="title" type="CT_Name" minOccurs="0" maxOccurs="unbounded"/>
372     <xsd:element name="desc" type="CT_Description" minOccurs="0" maxOccurs="unbounded"/>
373     <xsd:element name="catLst" type="CT_Categories" minOccurs="0"/>
374     <xsd:element name="sampData" type="CT_SampleData" minOccurs="0"/>
375     <xsd:element name="styleData" type="CT_SampleData" minOccurs="0"/>
376     <xsd:element name="clrData" type="CT_SampleData" minOccurs="0"/>
377     <xsd:element name="layoutNode" type="CT_LayoutNode"/>
378     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
379       maxOccurs="1"/>
380   </xsd:sequence>
381   <xsd:attribute name="uniqueId" type="xsd:string" use="optional" default=""/>
382   <xsd:attribute name="minVer" type="xsd:string" use="optional"
383     default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
384   <xsd:attribute name="defStyle" type="xsd:string" use="optional" default=""/>
385 </xsd:complexType>
386 <xsd:element name="layoutDef" type="CT_DiagramDefinition"/>
387 <xsd:complexType name="CT_DiagramDefinitionHeader">
388   <xsd:sequence>
389     <xsd:element name="title" type="CT_Name" minOccurs="1" maxOccurs="unbounded"/>
390     <xsd:element name="desc" type="CT_Description" minOccurs="1" maxOccurs="unbounded"/>
391     <xsd:element name="catLst" type="CT_Categories" minOccurs="0"/>
392     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
393       maxOccurs="1"/>
394   </xsd:sequence>
395   <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
396   <xsd:attribute name="minVer" type="xsd:string" use="optional"
397     default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
398   <xsd:attribute name="defStyle" type="xsd:string" use="optional" default=""/>
399   <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
400 </xsd:complexType>
401 <xsd:element name="layoutDefHdr" type="CT_DiagramDefinitionHeader"/>
402 <xsd:complexType name="CT_DiagramDefinitionHeaderLst">
403   <xsd:sequence>
404     <xsd:element name="layoutDefHdr" type="CT_DiagramDefinitionHeader" minOccurs="0"
405       maxOccurs="unbounded"/>
406   </xsd:sequence>
407 </xsd:complexType>
408 <xsd:element name="layoutDefHdrLst" type="CT_DiagramDefinitionHeaderLst"/>
409 <xsd:complexType name="CT_RelIds">
410   <xsd:attribute ref="r:dm" use="required"/>
411   <xsd:attribute ref="r:lo" use="required"/>
412   <xsd:attribute ref="r:qs" use="required"/>
413   <xsd:attribute ref="r:cs" use="required"/>
414 </xsd:complexType>
415 <xsd:element name="relIds" type="CT_RelIds"/>
416 <xsd:simpleType name="ST_ParameterVal">

```

```

417     <xsd:union memberTypes="ST_DiagramHorizontalAlignment ST_VerticalAlignment ST_ChildDirection
418       ST_ChildAlignment ST_SecondaryChildAlignment ST_LinearDirection ST_SecondaryLinearDirection
419       ST_StartingElement ST_BendPoint ST_ConnectorRouting ST_ArrowheadStyle ST_ConnectorDimension
420       ST_RotationPath ST_CenterShapeMapping ST_NodeHorizontalAlignment ST_NodeVerticalAlignment
421       ST_FallbackDimension ST_TextDirection ST_PyramidAccentPosition ST_PyramidAccentTextMargin
422       ST_TextBlockDirection ST_TextAnchorHorizontal ST_TextAnchorVertical ST_DiagramTextAlignment
423       ST_AutoTextRotation ST_GrowDirection ST_FlowDirection ST_ContinueDirection ST_Breakpoint
424       ST_Offset ST_HierarchyAlignment xsd:int xsd:double xsd:boolean xsd:string
425       ST_ConnectorPoint"/>
426   </xsd:simpleType>
427   <xsd:simpleType name="ST_ModelId">
428     <xsd:union memberTypes="xsd:int s:ST_Guid"/>
429   </xsd:simpleType>
430   <xsd:complexType name="CT_ElemPropSet">
431     <xsd:sequence>
432       <xsd:element name="presLayoutVars" type="CT_LayoutVariablePropertySet" minOccurs="0"
433         maxOccurs="1"/>
434       <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
435     </xsd:sequence>
436     <xsd:attribute name="presAssocID" type="ST_ModelId" use="optional"/>
437     <xsd:attribute name="presName" type="xsd:string" use="optional"/>
438     <xsd:attribute name="presStyleLbl" type="xsd:string" use="optional"/>
439     <xsd:attribute name="presStyleIdx" type="xsd:int" use="optional"/>
440     <xsd:attribute name="presStyleCnt" type="xsd:int" use="optional"/>
441     <xsd:attribute name="loTypeId" type="xsd:string" use="optional"/>
442     <xsd:attribute name="loCatId" type="xsd:string" use="optional"/>
443     <xsd:attribute name="qsTypeId" type="xsd:string" use="optional"/>
444     <xsd:attribute name="qsCatId" type="xsd:string" use="optional"/>
445     <xsd:attribute name="csTypeId" type="xsd:string" use="optional"/>
446     <xsd:attribute name="csCatId" type="xsd:string" use="optional"/>
447     <xsd:attribute name="coherent3DOff" type="xsd:boolean" use="optional"/>
448     <xsd:attribute name="phldrT" type="xsd:string" use="optional"/>
449     <xsd:attribute name="phldr" type="xsd:boolean" use="optional"/>
450     <xsd:attribute name="custAng" type="xsd:int" use="optional"/>
451     <xsd:attribute name="custFlipVert" type="xsd:boolean" use="optional"/>
452     <xsd:attribute name="custFlipHor" type="xsd:boolean" use="optional"/>
453     <xsd:attribute name="custSzX" type="xsd:int" use="optional"/>
454     <xsd:attribute name="custSzY" type="xsd:int" use="optional"/>
455     <xsd:attribute name="custScaleX" type="xsd:int" use="optional"/>
456     <xsd:attribute name="custScaleY" type="xsd:int" use="optional"/>
457     <xsd:attribute name="custT" type="xsd:boolean" use="optional"/>
458     <xsd:attribute name="custLinFactX" type="xsd:int" use="optional"/>
459     <xsd:attribute name="custLinFactY" type="xsd:int" use="optional"/>
460     <xsd:attribute name="custLinFactNeighborX" type="xsd:int" use="optional"/>
461     <xsd:attribute name="custLinFactNeighborY" type="xsd:int" use="optional"/>
462     <xsd:attribute name="custRadScaleRad" type="xsd:int" use="optional"/>
463     <xsd:attribute name="custRadScaleInc" type="xsd:int" use="optional"/>
464   </xsd:complexType>
465   <xsd:simpleType name="ST_Direction" final="restriction">
466     <xsd:restriction base="xsd:token">
467       <xsd:enumeration value="norm"/>
468       <xsd:enumeration value="rev"/>
469     </xsd:restriction>

```

```

470 </xsd:simpleType>
471 <xsd:simpleType name="ST_HierBranchStyle" final="restriction">
472   <xsd:restriction base="xsd:token">
473     <xsd:enumeration value="l"/>
474     <xsd:enumeration value="r"/>
475     <xsd:enumeration value="hang"/>
476     <xsd:enumeration value="std"/>
477     <xsd:enumeration value="init"/>
478   </xsd:restriction>
479 </xsd:simpleType>
480 <xsd:simpleType name="ST_AnimOneStr" final="restriction">
481   <xsd:restriction base="xsd:token">
482     <xsd:enumeration value="none"/>
483     <xsd:enumeration value="one"/>
484     <xsd:enumeration value="branch"/>
485   </xsd:restriction>
486 </xsd:simpleType>
487 <xsd:simpleType name="ST_AnimLvlStr" final="restriction">
488   <xsd:restriction base="xsd:token">
489     <xsd:enumeration value="none"/>
490     <xsd:enumeration value="lvl"/>
491     <xsd:enumeration value="ctr"/>
492   </xsd:restriction>
493 </xsd:simpleType>
494 <xsd:complexType name="CT_OrgChart">
495   <xsd:attribute name="val" type="xsd:boolean" default="false" use="optional"/>
496 </xsd:complexType>
497 <xsd:simpleType name="ST_NodeCount">
498   <xsd:restriction base="xsd:int">
499     <xsd:minInclusive value="-1"/>
500   </xsd:restriction>
501 </xsd:simpleType>
502 <xsd:complexType name="CT_ChildMax">
503   <xsd:attribute name="val" type="ST_NodeCount" default="-1" use="optional"/>
504 </xsd:complexType>
505 <xsd:complexType name="CT_ChildPref">
506   <xsd:attribute name="val" type="ST_NodeCount" default="-1" use="optional"/>
507 </xsd:complexType>
508 <xsd:complexType name="CT_BulletEnabled">
509   <xsd:attribute name="val" type="xsd:boolean" default="false" use="optional"/>
510 </xsd:complexType>
511 <xsd:complexType name="CT_Direction">
512   <xsd:attribute name="val" type="ST_Direction" default="norm" use="optional"/>
513 </xsd:complexType>
514 <xsd:complexType name="CT_HierBranchStyle">
515   <xsd:attribute name="val" type="ST_HierBranchStyle" default="std" use="optional"/>
516 </xsd:complexType>
517 <xsd:complexType name="CT_AnimOne">
518   <xsd:attribute name="val" type="ST_AnimOneStr" default="one" use="optional"/>
519 </xsd:complexType>
520 <xsd:complexType name="CT_AnimLvl">
521   <xsd:attribute name="val" type="ST_AnimLvlStr" default="none" use="optional"/>
522 </xsd:complexType>

```

```

523 <xsd:simpleType name="ST_ResizeHandlesStr" final="restriction">
524   <xsd:restriction base="xsd:token">
525     <xsd:enumeration value="exact"/>
526     <xsd:enumeration value="rel"/>
527   </xsd:restriction>
528 </xsd:simpleType>
529 <xsd:complexType name="CT_ResizeHandles">
530   <xsd:attribute name="val" type="ST_ResizeHandlesStr" default="rel" use="optional"/>
531 </xsd:complexType>
532 <xsd:complexType name="CT_LayoutVariablePropertySet">
533   <xsd:sequence>
534     <xsd:element name="orgChart" type="CT_OrgChart" minOccurs="0" maxOccurs="1"/>
535     <xsd:element name="chMax" type="CT_ChildMax" minOccurs="0" maxOccurs="1"/>
536     <xsd:element name="chPref" type="CT_ChildPref" minOccurs="0" maxOccurs="1"/>
537     <xsd:element name="bulletEnabled" type="CT_BulletEnabled" minOccurs="0" maxOccurs="1"/>
538     <xsd:element name="dir" type="CT_Direction" minOccurs="0" maxOccurs="1"/>
539     <xsd:element name="hierBranch" type="CT_HierBranchStyle" minOccurs="0" maxOccurs="1"/>
540     <xsd:element name="animOne" type="CT_AnimOne" minOccurs="0" maxOccurs="1"/>
541     <xsd:element name="animLvl" type="CT_AnimLvl" minOccurs="0" maxOccurs="1"/>
542     <xsd:element name="resizeHandles" type="CT_ResizeHandles" minOccurs="0" maxOccurs="1"/>
543   </xsd:sequence>
544 </xsd:complexType>
545 <xsd:complexType name="CT_SDName">
546   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
547   <xsd:attribute name="val" type="xsd:string" use="required"/>
548 </xsd:complexType>
549 <xsd:complexType name="CT_SDDescription">
550   <xsd:attribute name="lang" type="xsd:string" use="optional" default=""/>
551   <xsd:attribute name="val" type="xsd:string" use="required"/>
552 </xsd:complexType>
553 <xsd:complexType name="CT_SDCategory">
554   <xsd:attribute name="type" type="xsd:anyURI" use="required"/>
555   <xsd:attribute name="pri" type="xsd:unsignedInt" use="required"/>
556 </xsd:complexType>
557 <xsd:complexType name="CT_SDCategories">
558   <xsd:sequence minOccurs="0" maxOccurs="unbounded">
559     <xsd:element name="cat" type="CT_SDCategory" minOccurs="0" maxOccurs="unbounded"/>
560   </xsd:sequence>
561 </xsd:complexType>
562 <xsd:complexType name="CT_TextProps">
563   <xsd:sequence>
564     <xsd:group ref="a:EG_Text3D" minOccurs="0" maxOccurs="1"/>
565   </xsd:sequence>
566 </xsd:complexType>
567 <xsd:complexType name="CT_StyleLabel">
568   <xsd:sequence>
569     <xsd:element name="scene3d" type="a:CT_Scene3D" minOccurs="0" maxOccurs="1"/>
570     <xsd:element name="sp3d" type="a:CT_Shape3D" minOccurs="0" maxOccurs="1"/>
571     <xsd:element name="txPr" type="CT_TextProps" minOccurs="0" maxOccurs="1"/>
572     <xsd:element name="style" type="a:CT_ShapeStyle" minOccurs="0" maxOccurs="1"/>
573     <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
574       maxOccurs="1"/>
575   </xsd:sequence>

```

```

576     <xsd:attribute name="name" type="xsd:string" use="required"/>
577 </xsd:complexType>
578 <xsd:complexType name="CT_StyleDefinition">
579     <xsd:sequence>
580         <xsd:element name="title" type="CT_SDName" minOccurs="0" maxOccurs="unbounded"/>
581         <xsd:element name="desc" type="CT_SDDescription" minOccurs="0" maxOccurs="unbounded"/>
582         <xsd:element name="catLst" type="CT_SDCategories" minOccurs="0"/>
583         <xsd:element name="scene3d" type="a:CT_Scene3D" minOccurs="0" maxOccurs="1"/>
584         <xsd:element name="styleLbl" type="CT_StyleLabel" minOccurs="1" maxOccurs="unbounded"/>
585         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
586             maxOccurs="1"/>
587     </xsd:sequence>
588     <xsd:attribute name="uniqueId" type="xsd:string" use="optional" default=""/>
589     <xsd:attribute name="minVer" type="xsd:string" use="optional"
590         default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
591 </xsd:complexType>
592 <xsd:element name="styleDef" type="CT_StyleDefinition"/>
593 <xsd:complexType name="CT_StyleDefinitionHeader">
594     <xsd:sequence>
595         <xsd:element name="title" type="CT_SDName" minOccurs="1" maxOccurs="unbounded"/>
596         <xsd:element name="desc" type="CT_SDDescription" minOccurs="1" maxOccurs="unbounded"/>
597         <xsd:element name="catLst" type="CT_SDCategories" minOccurs="0"/>
598         <xsd:element name="extLst" type="a:CT_OfficeArtExtensionList" minOccurs="0"
599             maxOccurs="1"/>
600     </xsd:sequence>
601     <xsd:attribute name="uniqueId" type="xsd:string" use="required"/>
602     <xsd:attribute name="minVer" type="xsd:string" use="optional"
603         default="http://schemas.openxmlformats.org/drawingml/2006/diagram"/>
604     <xsd:attribute name="resId" type="xsd:int" use="optional" default="0"/>
605 </xsd:complexType>
606 <xsd:element name="styleDefHdr" type="CT_StyleDefinitionHeader"/>
607 <xsd:complexType name="CT_StyleDefinitionHeaderLst">
608     <xsd:sequence>
609         <xsd:element name="styleDefHdr" type="CT_StyleDefinitionHeader" minOccurs="0"
610             maxOccurs="unbounded"/>
611     </xsd:sequence>
612 </xsd:complexType>
613 <xsd:element name="styleDefHdrLst" type="CT_StyleDefinitionHeaderLst"/>
614 <xsd:simpleType name="ST_AlgorithmType" final="restriction">
615     <xsd:restriction base="xsd:token">
616         <xsd:enumeration value="composite"/>
617         <xsd:enumeration value="conn"/>
618         <xsd:enumeration value="cycle"/>
619         <xsd:enumeration value="hierChild"/>
620         <xsd:enumeration value="hierRoot"/>
621         <xsd:enumeration value="pyra"/>
622         <xsd:enumeration value="lin"/>
623         <xsd:enumeration value="sp"/>
624         <xsd:enumeration value="tx"/>
625         <xsd:enumeration value="snake"/>
626     </xsd:restriction>
627 </xsd:simpleType>
628 <xsd:simpleType name="ST_AxisType" final="restriction">

```

```

629     <xsd:restriction base="xsd:token">
630         <xsd:enumeration value="self"/>
631         <xsd:enumeration value="ch"/>
632         <xsd:enumeration value="des"/>
633         <xsd:enumeration value="desOrSelf"/>
634         <xsd:enumeration value="par"/>
635         <xsd:enumeration value="ancst"/>
636         <xsd:enumeration value="ancstOrSelf"/>
637         <xsd:enumeration value="followSib"/>
638         <xsd:enumeration value="precedSib"/>
639         <xsd:enumeration value="follow"/>
640         <xsd:enumeration value="preced"/>
641         <xsd:enumeration value="root"/>
642         <xsd:enumeration value="none"/>
643     </xsd:restriction>
644 </xsd:simpleType>
645 <xsd:simpleType name="ST_AxisTypes">
646     <xsd:list itemType="ST_AxisType"/>
647 </xsd:simpleType>
648 <xsd:simpleType name="ST_BoolOperator" final="restriction">
649     <xsd:restriction base="xsd:token">
650         <xsd:enumeration value="none"/>
651         <xsd:enumeration value="equ"/>
652         <xsd:enumeration value="gte"/>
653         <xsd:enumeration value="lte"/>
654     </xsd:restriction>
655 </xsd:simpleType>
656 <xsd:simpleType name="ST_ChildOrderType" final="restriction">
657     <xsd:restriction base="xsd:token">
658         <xsd:enumeration value="b"/>
659         <xsd:enumeration value="t"/>
660     </xsd:restriction>
661 </xsd:simpleType>
662 <xsd:simpleType name="ST_ConstraintType" final="restriction">
663     <xsd:restriction base="xsd:token">
664         <xsd:enumeration value="none"/>
665         <xsd:enumeration value="alignOff"/>
666         <xsd:enumeration value="begMarg"/>
667         <xsd:enumeration value="bendDist"/>
668         <xsd:enumeration value="begPad"/>
669         <xsd:enumeration value="b"/>
670         <xsd:enumeration value="bMarg"/>
671         <xsd:enumeration value="bOff"/>
672         <xsd:enumeration value="ctrX"/>
673         <xsd:enumeration value="ctrXOff"/>
674         <xsd:enumeration value="ctrY"/>
675         <xsd:enumeration value="ctrYOff"/>
676         <xsd:enumeration value="connDist"/>
677         <xsd:enumeration value="diam"/>
678         <xsd:enumeration value="endMarg"/>
679         <xsd:enumeration value="endPad"/>
680         <xsd:enumeration value="h"/>
681         <xsd:enumeration value="hArH"/>

```

```

682      <xsd:enumeration value="hOff"/>
683      <xsd:enumeration value="l"/>
684      <xsd:enumeration value="lMarg"/>
685      <xsd:enumeration value="lOff"/>
686      <xsd:enumeration value="r"/>
687      <xsd:enumeration value="rMarg"/>
688      <xsd:enumeration value="rOff"/>
689      <xsd:enumeration value="primFontSz"/>
690      <xsd:enumeration value="pyraAcctRatio"/>
691      <xsd:enumeration value="secFontSz"/>
692      <xsd:enumeration value="sibSp"/>
693      <xsd:enumeration value="secSibSp"/>
694      <xsd:enumeration value="sp"/>
695      <xsd:enumeration value="stemThick"/>
696      <xsd:enumeration value="t"/>
697      <xsd:enumeration value="tMarg"/>
698      <xsd:enumeration value="tOff"/>
699      <xsd:enumeration value="userA"/>
700      <xsd:enumeration value="userB"/>
701      <xsd:enumeration value="userC"/>
702      <xsd:enumeration value="userD"/>
703      <xsd:enumeration value="userE"/>
704      <xsd:enumeration value="userF"/>
705      <xsd:enumeration value="userG"/>
706      <xsd:enumeration value="userH"/>
707      <xsd:enumeration value="userI"/>
708      <xsd:enumeration value="userJ"/>
709      <xsd:enumeration value="userK"/>
710      <xsd:enumeration value="userL"/>
711      <xsd:enumeration value="userM"/>
712      <xsd:enumeration value="userN"/>
713      <xsd:enumeration value="userO"/>
714      <xsd:enumeration value="userP"/>
715      <xsd:enumeration value="userQ"/>
716      <xsd:enumeration value="userR"/>
717      <xsd:enumeration value="userS"/>
718      <xsd:enumeration value="userT"/>
719      <xsd:enumeration value="userU"/>
720      <xsd:enumeration value="userV"/>
721      <xsd:enumeration value="userW"/>
722      <xsd:enumeration value="userX"/>
723      <xsd:enumeration value="userY"/>
724      <xsd:enumeration value="userZ"/>
725      <xsd:enumeration value="w"/>
726      <xsd:enumeration value="wArH"/>
727      <xsd:enumeration value="wOff"/>
728    </xsd:restriction>
729  </xsd:simpleType>
730  <xsd:simpleType name="ST_ConstraintRelationship" final="restriction">
731    <xsd:restriction base="xsd:token">
732      <xsd:enumeration value="self"/>
733      <xsd:enumeration value="ch"/>
734      <xsd:enumeration value="des"/>

```

```

735     </xsd:restriction>
736 </xsd:simpleType>
737 <xsd:simpleType name="ST_ElementType" final="restriction">
738     <xsd:restriction base="xsd:token">
739         <xsd:enumeration value="all"/>
740         <xsd:enumeration value="doc"/>
741         <xsd:enumeration value="node"/>
742         <xsd:enumeration value="norm"/>
743         <xsd:enumeration value="nonNorm"/>
744         <xsd:enumeration value="asst"/>
745         <xsd:enumeration value="nonAsst"/>
746         <xsd:enumeration value="parTrans"/>
747         <xsd:enumeration value="pres"/>
748         <xsd:enumeration value="sibTrans"/>
749     </xsd:restriction>
750 </xsd:simpleType>
751 <xsd:simpleType name="ST_ElementTypes">
752     <xsd:list itemType="ST_ElementType"/>
753 </xsd:simpleType>
754 <xsd:simpleType name="ST_ParameterId" final="restriction">
755     <xsd:restriction base="xsd:token">
756         <xsd:enumeration value="horzAlign"/>
757         <xsd:enumeration value="vertAlign"/>
758         <xsd:enumeration value="chDir"/>
759         <xsd:enumeration value="chAlign"/>
760         <xsd:enumeration value="secChAlign"/>
761         <xsd:enumeration value="linDir"/>
762         <xsd:enumeration value="secLinDir"/>
763         <xsd:enumeration value="stElem"/>
764         <xsd:enumeration value="bendPt"/>
765         <xsd:enumeration value="connRout"/>
766         <xsd:enumeration value="begSty"/>
767         <xsd:enumeration value="endSty"/>
768         <xsd:enumeration value="dim"/>
769         <xsd:enumeration value="rotPath"/>
770         <xsd:enumeration value="ctrShpMap"/>
771         <xsd:enumeration value="nodeHorzAlign"/>
772         <xsd:enumeration value="nodeVertAlign"/>
773         <xsd:enumeration value="fallback"/>
774         <xsd:enumeration value="txDir"/>
775         <xsd:enumeration value="pyraAcctPos"/>
776         <xsd:enumeration value="pyraAcctTxMar"/>
777         <xsd:enumeration value="txBldir"/>
778         <xsd:enumeration value="txAnchorHorz"/>
779         <xsd:enumeration value="txAnchorVert"/>
780         <xsd:enumeration value="txAnchorHorzCh"/>
781         <xsd:enumeration value="txAnchorVertCh"/>
782         <xsd:enumeration value="parTxLTRAlign"/>
783         <xsd:enumeration value="parTxRTLAlign"/>
784         <xsd:enumeration value="shpTxLTRAlignCh"/>
785         <xsd:enumeration value="shpTxRTLAlignCh"/>
786         <xsd:enumeration value="autoTxRot"/>
787         <xsd:enumeration value="grDir"/>

```



```

788         <xsd:enumeration value="flowDir"/>
789         <xsd:enumeration value="contDir"/>
790         <xsd:enumeration value="bkpt"/>
791         <xsd:enumeration value="off"/>
792         <xsd:enumeration value="hierAlign"/>
793         <xsd:enumeration value="bkPtFixedVal"/>
794         <xsd:enumeration value="stBulletLvl"/>
795         <xsd:enumeration value="stAng"/>
796         <xsd:enumeration value="spanAng"/>
797         <xsd:enumeration value="ar"/>
798         <xsd:enumeration value="lnSpPar"/>
799         <xsd:enumeration value="lnSpAfParP"/>
800         <xsd:enumeration value="lnSpCh"/>
801         <xsd:enumeration value="lnSpAfChP"/>
802         <xsd:enumeration value="rtShortDist"/>
803         <xsd:enumeration value="alignTx"/>
804         <xsd:enumeration value="pyraLvlNode"/>
805         <xsd:enumeration value="pyraAcctBkgdNode"/>
806         <xsd:enumeration value="pyraAcctTxNode"/>
807         <xsd:enumeration value="srcNode"/>
808         <xsd:enumeration value="dstNode"/>
809         <xsd:enumeration value="begPts"/>
810         <xsd:enumeration value="endPts"/>
811     </xsd:restriction>
812 </xsd:simpleType>
813 <xsd:simpleType name="ST_Ints">
814     <xsd:list itemType="xsd:int"/>
815 </xsd:simpleType>
816 <xsd:simpleType name="ST_UnsignedInts">
817     <xsd:list itemType="xsd:unsignedInt"/>
818 </xsd:simpleType>
819 <xsd:simpleType name="ST_Booleans">
820     <xsd:list itemType="xsd:boolean"/>
821 </xsd:simpleType>
822 <xsd:simpleType name="ST_FunctionType" final="restriction">
823     <xsd:restriction base="xsd:token">
824         <xsd:enumeration value="cnt"/>
825         <xsd:enumeration value="pos"/>
826         <xsd:enumeration value="revPos"/>
827         <xsd:enumeration value="posEven"/>
828         <xsd:enumeration value="posOdd"/>
829         <xsd:enumeration value="var"/>
830         <xsd:enumeration value="depth"/>
831         <xsd:enumeration value="maxDepth"/>
832     </xsd:restriction>
833 </xsd:simpleType>
834 <xsd:simpleType name="ST_FunctionOperator" final="restriction">
835     <xsd:restriction base="xsd:token">
836         <xsd:enumeration value="equ"/>
837         <xsd:enumeration value="neq"/>
838         <xsd:enumeration value="gt"/>
839         <xsd:enumeration value="lt"/>
840         <xsd:enumeration value="gte"/>

```

```

841     <xsd:enumeration value="lte"/>
842   </xsd:restriction>
843 </xsd:simpleType>
844 <xsd:simpleType name="ST_DiagramHorizontalAlignment" final="restriction">
845   <xsd:restriction base="xsd:token">
846     <xsd:enumeration value="l"/>
847     <xsd:enumeration value="ctr"/>
848     <xsd:enumeration value="r"/>
849     <xsd:enumeration value="none"/>
850   </xsd:restriction>
851 </xsd:simpleType>
852 <xsd:simpleType name="ST_VerticalAlignment" final="restriction">
853   <xsd:restriction base="xsd:token">
854     <xsd:enumeration value="t"/>
855     <xsd:enumeration value="mid"/>
856     <xsd:enumeration value="b"/>
857     <xsd:enumeration value="none"/>
858   </xsd:restriction>
859 </xsd:simpleType>
860 <xsd:simpleType name="ST_ChildDirection" final="restriction">
861   <xsd:restriction base="xsd:token">
862     <xsd:enumeration value="horz"/>
863     <xsd:enumeration value="vert"/>
864   </xsd:restriction>
865 </xsd:simpleType>
866 <xsd:simpleType name="ST_ChildAlignment" final="restriction">
867   <xsd:restriction base="xsd:token">
868     <xsd:enumeration value="t"/>
869     <xsd:enumeration value="b"/>
870     <xsd:enumeration value="l"/>
871     <xsd:enumeration value="r"/>
872   </xsd:restriction>
873 </xsd:simpleType>
874 <xsd:simpleType name="ST_SecondaryChildAlignment" final="restriction">
875   <xsd:restriction base="xsd:token">
876     <xsd:enumeration value="none"/>
877     <xsd:enumeration value="t"/>
878     <xsd:enumeration value="b"/>
879     <xsd:enumeration value="l"/>
880     <xsd:enumeration value="r"/>
881   </xsd:restriction>
882 </xsd:simpleType>
883 <xsd:simpleType name="ST_LinearDirection" final="restriction">
884   <xsd:restriction base="xsd:token">
885     <xsd:enumeration value="fromL"/>
886     <xsd:enumeration value="fromR"/>
887     <xsd:enumeration value="fromT"/>
888     <xsd:enumeration value="fromB"/>
889   </xsd:restriction>
890 </xsd:simpleType>
891 <xsd:simpleType name="ST_SecondaryLinearDirection" final="restriction">
892   <xsd:restriction base="xsd:token">
893     <xsd:enumeration value="none"/>

```

```

894         <xsd:enumeration value="fromL"/>
895         <xsd:enumeration value="fromR"/>
896         <xsd:enumeration value="fromT"/>
897         <xsd:enumeration value="fromB"/>
898     </xsd:restriction>
899 </xsd:simpleType>
900 <xsd:simpleType name="ST_StartingElement" final="restriction">
901     <xsd:restriction base="xsd:token">
902         <xsd:enumeration value="node"/>
903         <xsd:enumeration value="trans"/>
904     </xsd:restriction>
905 </xsd:simpleType>
906 <xsd:simpleType name="ST_RotationPath" final="restriction">
907     <xsd:restriction base="xsd:token">
908         <xsd:enumeration value="none"/>
909         <xsd:enumeration value="alongPath"/>
910     </xsd:restriction>
911 </xsd:simpleType>
912 <xsd:simpleType name="ST_CenterShapeMapping" final="restriction">
913     <xsd:restriction base="xsd:token">
914         <xsd:enumeration value="none"/>
915         <xsd:enumeration value="fNode"/>
916     </xsd:restriction>
917 </xsd:simpleType>
918 <xsd:simpleType name="ST_BendPoint" final="restriction">
919     <xsd:restriction base="xsd:token">
920         <xsd:enumeration value="beg"/>
921         <xsd:enumeration value="def"/>
922         <xsd:enumeration value="end"/>
923     </xsd:restriction>
924 </xsd:simpleType>
925 <xsd:simpleType name="ST_ConnectorRouting" final="restriction">
926     <xsd:restriction base="xsd:token">
927         <xsd:enumeration value="stra"/>
928         <xsd:enumeration value="bend"/>
929         <xsd:enumeration value="curve"/>
930         <xsd:enumeration value="longCurve"/>
931     </xsd:restriction>
932 </xsd:simpleType>
933 <xsd:simpleType name="ST_ArrowheadStyle" final="restriction">
934     <xsd:restriction base="xsd:token">
935         <xsd:enumeration value="auto"/>
936         <xsd:enumeration value="arr"/>
937         <xsd:enumeration value="noArr"/>
938     </xsd:restriction>
939 </xsd:simpleType>
940 <xsd:simpleType name="ST_ConnectorDimension" final="restriction">
941     <xsd:restriction base="xsd:token">
942         <xsd:enumeration value="1D"/>
943         <xsd:enumeration value="2D"/>
944         <xsd:enumeration value="cust"/>
945     </xsd:restriction>
946 </xsd:simpleType>

```

```

947 <xsd:simpleType name="ST_ConnectorPoint" final="restriction">
948   <xsd:restriction base="xsd:token">
949     <xsd:enumeration value="auto"/>
950     <xsd:enumeration value="bCtr"/>
951     <xsd:enumeration value="ctr"/>
952     <xsd:enumeration value="midL"/>
953     <xsd:enumeration value="midR"/>
954     <xsd:enumeration value="tCtr"/>
955     <xsd:enumeration value="bL"/>
956     <xsd:enumeration value="bR"/>
957     <xsd:enumeration value="tL"/>
958     <xsd:enumeration value="tR"/>
959     <xsd:enumeration value="radial"/>
960   </xsd:restriction>
961 </xsd:simpleType>
962 <xsd:simpleType name="ST_NodeHorizontalAlignment" final="restriction">
963   <xsd:restriction base="xsd:token">
964     <xsd:enumeration value="l"/>
965     <xsd:enumeration value="ctr"/>
966     <xsd:enumeration value="r"/>
967   </xsd:restriction>
968 </xsd:simpleType>
969 <xsd:simpleType name="ST_NodeVerticalAlignment" final="restriction">
970   <xsd:restriction base="xsd:token">
971     <xsd:enumeration value="t"/>
972     <xsd:enumeration value="mid"/>
973     <xsd:enumeration value="b"/>
974   </xsd:restriction>
975 </xsd:simpleType>
976 <xsd:simpleType name="ST_FallbackDimension" final="restriction">
977   <xsd:restriction base="xsd:token">
978     <xsd:enumeration value="1D"/>
979     <xsd:enumeration value="2D"/>
980   </xsd:restriction>
981 </xsd:simpleType>
982 <xsd:simpleType name="ST_TextDirection" final="restriction">
983   <xsd:restriction base="xsd:token">
984     <xsd:enumeration value="fromT"/>
985     <xsd:enumeration value="fromB"/>
986   </xsd:restriction>
987 </xsd:simpleType>
988 <xsd:simpleType name="ST_PyramidAccentPosition" final="restriction">
989   <xsd:restriction base="xsd:token">
990     <xsd:enumeration value="bef"/>
991     <xsd:enumeration value="aft"/>
992   </xsd:restriction>
993 </xsd:simpleType>
994 <xsd:simpleType name="ST_PyramidAccentTextMargin" final="restriction">
995   <xsd:restriction base="xsd:token">
996     <xsd:enumeration value="step"/>
997     <xsd:enumeration value="stack"/>
998   </xsd:restriction>
999 </xsd:simpleType>

```

```

1000 <xsd:simpleType name="ST_TextBlockDirection" final="restriction">
1001   <xsd:restriction base="xsd:token">
1002     <xsd:enumeration value="horz"/>
1003     <xsd:enumeration value="vert"/>
1004   </xsd:restriction>
1005 </xsd:simpleType>
1006 <xsd:simpleType name="ST_TextAnchorHorizontal" final="restriction">
1007   <xsd:restriction base="xsd:token">
1008     <xsd:enumeration value="none"/>
1009     <xsd:enumeration value="ctr"/>
1010   </xsd:restriction>
1011 </xsd:simpleType>
1012 <xsd:simpleType name="ST_TextAnchorVertical" final="restriction">
1013   <xsd:restriction base="xsd:token">
1014     <xsd:enumeration value="t"/>
1015     <xsd:enumeration value="mid"/>
1016     <xsd:enumeration value="b"/>
1017   </xsd:restriction>
1018 </xsd:simpleType>
1019 <xsd:simpleType name="ST_DiagramTextAlignment" final="restriction">
1020   <xsd:restriction base="xsd:token">
1021     <xsd:enumeration value="l"/>
1022     <xsd:enumeration value="ctr"/>
1023     <xsd:enumeration value="r"/>
1024   </xsd:restriction>
1025 </xsd:simpleType>
1026 <xsd:simpleType name="ST_AutoTextRotation" final="restriction">
1027   <xsd:restriction base="xsd:token">
1028     <xsd:enumeration value="none"/>
1029     <xsd:enumeration value="upr"/>
1030     <xsd:enumeration value="grav"/>
1031   </xsd:restriction>
1032 </xsd:simpleType>
1033 <xsd:simpleType name="ST_GrowDirection" final="restriction">
1034   <xsd:restriction base="xsd:token">
1035     <xsd:enumeration value="tL"/>
1036     <xsd:enumeration value="tR"/>
1037     <xsd:enumeration value="bL"/>
1038     <xsd:enumeration value="bR"/>
1039   </xsd:restriction>
1040 </xsd:simpleType>
1041 <xsd:simpleType name="ST_FlowDirection" final="restriction">
1042   <xsd:restriction base="xsd:token">
1043     <xsd:enumeration value="row"/>
1044     <xsd:enumeration value="col"/>
1045   </xsd:restriction>
1046 </xsd:simpleType>
1047 <xsd:simpleType name="ST_ContinueDirection" final="restriction">
1048   <xsd:restriction base="xsd:token">
1049     <xsd:enumeration value="revDir"/>
1050     <xsd:enumeration value="sameDir"/>
1051   </xsd:restriction>
1052 </xsd:simpleType>

```

```

1053 <xsd:simpleType name="ST_Breakpoint" final="restriction">
1054   <xsd:restriction base="xsd:token">
1055     <xsd:enumeration value="endCnv"/>
1056     <xsd:enumeration value="bal"/>
1057     <xsd:enumeration value="fixed"/>
1058   </xsd:restriction>
1059 </xsd:simpleType>
1060 <xsd:simpleType name="ST_Offset" final="restriction">
1061   <xsd:restriction base="xsd:token">
1062     <xsd:enumeration value="ctr"/>
1063     <xsd:enumeration value="off"/>
1064   </xsd:restriction>
1065 </xsd:simpleType>
1066 <xsd:simpleType name="ST_HierarchyAlignment" final="restriction">
1067   <xsd:restriction base="xsd:token">
1068     <xsd:enumeration value="tL"/>
1069     <xsd:enumeration value="tR"/>
1070     <xsd:enumeration value="tCtrCh"/>
1071     <xsd:enumeration value="tCtrDes"/>
1072     <xsd:enumeration value="bL"/>
1073     <xsd:enumeration value="bR"/>
1074     <xsd:enumeration value="bCtrCh"/>
1075     <xsd:enumeration value="bCtrDes"/>
1076     <xsd:enumeration value="lT"/>
1077     <xsd:enumeration value="lB"/>
1078     <xsd:enumeration value="lCtrCh"/>
1079     <xsd:enumeration value="lCtrDes"/>
1080     <xsd:enumeration value="rT"/>
1081     <xsd:enumeration value="rB"/>
1082     <xsd:enumeration value="rCtrCh"/>
1083     <xsd:enumeration value="rCtrDes"/>
1084   </xsd:restriction>
1085 </xsd:simpleType>
1086 <xsd:simpleType name="ST_FunctionValue" final="restriction">
1087   <xsd:union memberTypes="xsd:int xsd:boolean ST_Direction ST_HierBranchStyle ST_AnimOneStr
1088     ST_AnimLvlStr ST_ResizeHandlesStr"/>
1089 </xsd:simpleType>
1090 <xsd:simpleType name="ST_VariableType" final="restriction">
1091   <xsd:restriction base="xsd:token">
1092     <xsd:enumeration value="none"/>
1093     <xsd:enumeration value="orgChart"/>
1094     <xsd:enumeration value="chMax"/>
1095     <xsd:enumeration value="chPref"/>
1096     <xsd:enumeration value="bulEnabled"/>
1097     <xsd:enumeration value="dir"/>
1098     <xsd:enumeration value="hierBranch"/>
1099     <xsd:enumeration value="animOne"/>
1100     <xsd:enumeration value="animLvl"/>
1101     <xsd:enumeration value="resizeHandles"/>
1102   </xsd:restriction>
1103 </xsd:simpleType>
1104 <xsd:simpleType name="ST_FunctionArgument" final="restriction">
1105   <xsd:union memberTypes="ST_VariableType"/>

```

```

1106 </xsd:simpleType>
1107 <xsd:simpleType name="ST_OutputShapeType" final="restriction">
1108   <xsd:restriction base="xsd:token">
1109     <xsd:enumeration value="none"/>
1110     <xsd:enumeration value="conn"/>
1111   </xsd:restriction>
1112 </xsd:simpleType>
1113 </xsd:schema>

```

A.6 VML

A.6.1 VML

```

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:vml" elementFormDefault="qualified"
10 attributeFormDefault="unqualified">
11   <xsd:import namespace="urn:schemas-microsoft-com:office:office" schemaLocation="vml-
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="vml-
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="vml-
20     spreadsheetDrawing.xsd"/>
21   <xsd:import namespace="urn:schemas-microsoft-com:office:powerpoint" schemaLocation="vml-
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:hrstd"/>
82     <xsd:attribute ref="o:hrnoshade"/>
83     <xsd:attribute ref="o:hrpct"/>
84     <xsd:attribute ref="o:hralign"/>
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:reldid"/>
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="pvml: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="pvml: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:choice>
377     <xsd:attributeGroup ref="AG_Id"/>
378     <xsd:attributeGroup ref="AG_Style"/>
379     <xsd:attribute name="inset" type="xsd:string" use="optional"/>
380     <xsd:attribute ref="o:singleclick"/>
381     <xsd:attribute ref="o:insetmode"/>
382 </xsd:complexType>
383 <xsd:complexType name="CT_TextPath">
384     <xsd:attributeGroup ref="AG_Id"/>
385     <xsd:attributeGroup ref="AG_Style"/>
386     <xsd:attribute name="on" type="s:ST TrueFalse" use="optional"/>
387     <xsd:attribute name="fitshape" type="s:ST TrueFalse" use="optional"/>
388     <xsd:attribute name="fitpath" type="s:ST TrueFalse" use="optional"/>
389     <xsd:attribute name="trim" type="s:ST TrueFalse" use="optional"/>
390     <xsd:attribute name="xscale" type="s:ST TrueFalse" use="optional"/>
391     <xsd:attribute name="string" type="xsd:string" use="optional"/>
392 </xsd:complexType>
393 <xsd:element name="arc" type="CT_Arc"/>
394 <xsd:element name="curve" type="CT_Curve"/>
395 <xsd:element name="image" type="CT_Image"/>
396 <xsd:element name="line" type="CT_Line"/>
397 <xsd:element name="oval" type="CT_Oval"/>
398 <xsd:element name="polyline" type="CT_PolyLine"/>
399 <xsd:element name="rect" type="CT_Rect"/>
400 <xsd:element name="roundrect" type="CT_RoundRect"/>
401 <xsd:complexType name="CT_Arc">
402     <xsd:sequence>
403         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
404     </xsd:sequence>
405     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
406     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
407     <xsd:attribute name="startAngle" type="xsd:decimal" use="optional"/>
408     <xsd:attribute name="endAngle" type="xsd:decimal" use="optional"/>
409 </xsd:complexType>
410 <xsd:complexType name="CT_Curve">
411     <xsd:sequence>

```

```

412     <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
413 </xsd:sequence>
414 <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
415 <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
416 <xsd:attribute name="from" type="xsd:string" use="optional"/>
417 <xsd:attribute name="control1" type="xsd:string" use="optional"/>
418 <xsd:attribute name="control2" type="xsd:string" use="optional"/>
419 <xsd:attribute name="to" type="xsd:string" use="optional"/>
420 </xsd:complexType>
421 <xsd:complexType name="CT_Image">
422     <xsd:sequence>
423         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
424     </xsd:sequence>
425     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
426     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
427     <xsd:attributeGroup ref="AG_ImageAttributes"/>
428 </xsd:complexType>
429 <xsd:complexType name="CT_Line">
430     <xsd:sequence>
431         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
432     </xsd:sequence>
433     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
434     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
435     <xsd:attribute name="from" type="xsd:string" use="optional"/>
436     <xsd:attribute name="to" type="xsd:string" use="optional"/>
437 </xsd:complexType>
438 <xsd:complexType name="CT_Oval">
439     <xsd:choice maxOccurs="unbounded">
440         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
441     </xsd:choice>
442     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
443     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
444 </xsd:complexType>
445 <xsd:complexType name="CT_PolyLine">
446     <xsd:choice minOccurs="0" maxOccurs="unbounded">
447         <xsd:group ref="EG_ShapeElements"/>
448         <xsd:element ref="o:ink"/>
449     </xsd:choice>
450     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
451     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
452     <xsd:attribute name="points" type="xsd:string" use="optional"/>
453 </xsd:complexType>
454 <xsd:complexType name="CT_Rect">
455     <xsd:choice maxOccurs="unbounded">
456         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
457     </xsd:choice>
458     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
459     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
460 </xsd:complexType>
461 <xsd:complexType name="CT_RoundRect">
462     <xsd:choice maxOccurs="unbounded">
463         <xsd:group ref="EG_ShapeElements" minOccurs="0" maxOccurs="unbounded"/>
464     </xsd:choice>

```

```

465     <xsd:attributeGroup ref="AG_AllCoreAttributes"/>
466     <xsd:attributeGroup ref="AG_AllShapeAttributes"/>
467     <xsd:attribute name="arcsize" type="xsd:string" use="optional"/>
468 </xsd:complexType>
469 <xsd:simpleType name="ST_Ext">
470     <xsd:restriction base="xsd:string">
471         <xsd:enumeration value="view"/>
472         <xsd:enumeration value="edit"/>
473         <xsd:enumeration value="backwardCompatible"/>
474     </xsd:restriction>
475 </xsd:simpleType>
476 <xsd:simpleType name="ST_FillType">
477     <xsd:restriction base="xsd:string">
478         <xsd:enumeration value="solid"/>
479         <xsd:enumeration value="gradient"/>
480         <xsd:enumeration value="gradientRadial"/>
481         <xsd:enumeration value="tile"/>
482         <xsd:enumeration value="pattern"/>
483         <xsd:enumeration value="frame"/>
484     </xsd:restriction>
485 </xsd:simpleType>
486 <xsd:simpleType name="ST_FillMethod">
487     <xsd:restriction base="xsd:string">
488         <xsd:enumeration value="none"/>
489         <xsd:enumeration value="linear"/>
490         <xsd:enumeration value="sigma"/>
491         <xsd:enumeration value="any"/>
492         <xsd:enumeration value="linear sigma"/>
493     </xsd:restriction>
494 </xsd:simpleType>
495 <xsd:simpleType name="ST_ShadowType">
496     <xsd:restriction base="xsd:string">
497         <xsd:enumeration value="single"/>
498         <xsd:enumeration value="double"/>
499         <xsd:enumeration value="emboss"/>
500         <xsd:enumeration value="perspective"/>
501     </xsd:restriction>
502 </xsd:simpleType>
503 <xsd:simpleType name="ST_StrokeLineStyle">
504     <xsd:restriction base="xsd:string">
505         <xsd:enumeration value="single"/>
506         <xsd:enumeration value="thinThin"/>
507         <xsd:enumeration value="thinThick"/>
508         <xsd:enumeration value="thickThin"/>
509         <xsd:enumeration value="thickBetweenThin"/>
510     </xsd:restriction>
511 </xsd:simpleType>
512 <xsd:simpleType name="ST_StrokeJoinStyle">
513     <xsd:restriction base="xsd:string">
514         <xsd:enumeration value="round"/>
515         <xsd:enumeration value="bevel"/>
516         <xsd:enumeration value="miter"/>
517     </xsd:restriction>

```



```

518 </xsd:simpleType>
519 <xsd:simpleType name="ST_StrokeEndCap">
520   <xsd:restriction base="xsd:string">
521     <xsd:enumeration value="flat"/>
522     <xsd:enumeration value="square"/>
523     <xsd:enumeration value="round"/>
524   </xsd:restriction>
525 </xsd:simpleType>
526 <xsd:simpleType name="ST_StrokeArrowLength">
527   <xsd:restriction base="xsd:string">
528     <xsd:enumeration value="short"/>
529     <xsd:enumeration value="medium"/>
530     <xsd:enumeration value="long"/>
531   </xsd:restriction>
532 </xsd:simpleType>
533 <xsd:simpleType name="ST_StrokeArrowWidth">
534   <xsd:restriction base="xsd:string">
535     <xsd:enumeration value="narrow"/>
536     <xsd:enumeration value="medium"/>
537     <xsd:enumeration value="wide"/>
538   </xsd:restriction>
539 </xsd:simpleType>
540 <xsd:simpleType name="ST_StrokeArrowType">
541   <xsd:restriction base="xsd:string">
542     <xsd:enumeration value="none"/>
543     <xsd:enumeration value="block"/>
544     <xsd:enumeration value="classic"/>
545     <xsd:enumeration value="oval"/>
546     <xsd:enumeration value="diamond"/>
547     <xsd:enumeration value="open"/>
548   </xsd:restriction>
549 </xsd:simpleType>
550 <xsd:simpleType name="ST_ImageAspect">
551   <xsd:restriction base="xsd:string">
552     <xsd:enumeration value="ignore"/>
553     <xsd:enumeration value="atMost"/>
554     <xsd:enumeration value="atLeast"/>
555   </xsd:restriction>
556 </xsd:simpleType>
557 <xsd:simpleType name="ST_EditAs">
558   <xsd:restriction base="xsd:string">
559     <xsd:enumeration value="canvas"/>
560     <xsd:enumeration value="orgchart"/>
561     <xsd:enumeration value="radial"/>
562     <xsd:enumeration value="cycle"/>
563     <xsd:enumeration value="stacked"/>
564     <xsd:enumeration value="venn"/>
565     <xsd:enumeration value="bullseye"/>
566   </xsd:restriction>
567 </xsd:simpleType>
568 </xsd:schema>

```

A.6.2 VML - Office Drawing

```

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="suggestedesigner" type="xsd:string" form="qualified"/>
108     <xsd:attribute name="suggestedesigner2" type="xsd:string" form="qualified"/>
109     <xsd:attribute name="suggestedesigneremail" 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="diffusity" 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

```

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

```

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

```

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

```

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 id="w" namespace="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
9      schemaLocation="wml.xsd"/>

```

```

10 <xsd:import id="s" namespace="http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
11     schemaLocation="shared-commonSimpleTypes.xsd"/>
12 <xsd:import id="xml" 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_YAlign" 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_RunLevelElts"/>
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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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_AlignType">
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:restriction base="xsd:boolean"/>
65 </xsd:simpleType>
66 <xsd:simpleType name="ST_String">
67     <xsd:restriction base="xsd:string"/>
68 </xsd:simpleType>
69 <xsd:simpleType name="ST_TrueFalse">
70     <xsd:restriction base="xsd:string">
71         <xsd:enumeration value="t"/>
72         <xsd:enumeration value="f"/>
73         <xsd:enumeration value="true"/>
74         <xsd:enumeration value="false"/>
75     </xsd:restriction>
76 </xsd:simpleType>
77 <xsd:simpleType name="ST_TrueFalseBlank">
78     <xsd:restriction base="xsd:string">
79         <xsd:enumeration value="t"/>
80         <xsd:enumeration value="f"/>
81         <xsd:enumeration value="true"/>
82         <xsd:enumeration value="false"/>
83         <xsd:enumeration value=""/>
84         <xsd:enumeration value="True"/>
85         <xsd:enumeration value="False"/>
86     </xsd:restriction>
87 </xsd:simpleType>
88 <xsd:simpleType name="ST_UnsignedDecimalNumber">
89     <xsd:restriction base="xsd:unsignedLong"/>
90 </xsd:simpleType>
91 <xsd:simpleType name="ST_TwipsMeasure">
92     <xsd:union memberTypes="ST_UnsignedDecimalNumber ST_PositiveUniversalMeasure"/>
93 </xsd:simpleType>
94 <xsd:simpleType name="ST_VerticalAlignRun">
95     <xsd:restriction base="xsd:string">
96         <xsd:enumeration value="baseline"/>
97         <xsd:enumeration value="superscript"/>
98         <xsd:enumeration value="subscript"/>
99     </xsd:restriction>

```

```

100 </xsd:simpleType>
101 <xsd:simpleType name="ST_Xstring">
102   <xsd:restriction base="xsd:string"/>
103 </xsd:simpleType>
104 <xsd:simpleType name="ST_XAlign">
105   <xsd:restriction base="xsd:string">
106     <xsd:enumeration value="left"/>
107     <xsd:enumeration value="center"/>
108     <xsd:enumeration value="right"/>
109     <xsd:enumeration value="inside"/>
110     <xsd:enumeration value="outside"/>
111   </xsd:restriction>
112 </xsd:simpleType>
113 <xsd:simpleType name="ST_YAlign">
114   <xsd:restriction base="xsd:string">
115     <xsd:enumeration value="inline"/>
116     <xsd:enumeration value="top"/>
117     <xsd:enumeration value="center"/>
118     <xsd:enumeration value="bottom"/>
119     <xsd:enumeration value="inside"/>
120     <xsd:enumeration value="outside"/>
121   </xsd:restriction>
122 </xsd:simpleType>
123 <xsd:simpleType name="ST_ConformanceClass">
124   <xsd:restriction base="xsd:string">
125     <xsd:enumeration value="strict"/>
126     <xsd:enumeration value="transitional"/>
127   </xsd:restriction>
128 </xsd:simpleType>
129 <xsd:simpleType name="ST_UniversalMeasure">
130   <xsd:restriction base="xsd:string">
131     <xsd:pattern value="-?[0-9]+(\.[0-9]+)?(mm|cm|in|pt|pc|pi)"/>
132   </xsd:restriction>
133 </xsd:simpleType>
134 <xsd:simpleType name="ST_PositiveUniversalMeasure">
135   <xsd:restriction base="ST_UniversalMeasure">
136     <xsd:pattern value="[0-9]+(\.[0-9]+)?(mm|cm|in|pt|pc|pi)"/>
137   </xsd:restriction>
138 </xsd:simpleType>
139 <xsd:simpleType name="ST_Percentage">
140   <xsd:restriction base="xsd:string">
141     <xsd:pattern value="-?[0-9]+(\.[0-9]+)?%/>
142   </xsd:restriction>
143 </xsd:simpleType>
144 <xsd:simpleType name="ST_FixedPercentage">
145   <xsd:restriction base="ST_Percentage">
146     <xsd:pattern value="-?((100)|([0-9][0-9]?))(\.[0-9][0-9]?)?%/>
147   </xsd:restriction>
148 </xsd:simpleType>
149 <xsd:simpleType name="ST_PositivePercentage">
150   <xsd:restriction base="ST_Percentage">
151     <xsd:pattern value="[0-9]+(\.[0-9]+)?%/>
152   </xsd:restriction>

```

```

153     </xsd:simpleType>
154     <xsd:simpleType name="ST_PositiveFixedPercentage">
155         <xsd:restriction base="ST_Percentage">
156             <xsd:pattern value="((100)|([0-9][0-9]?))(\.[0-9][0-9]?)?%"/>
157         </xsd:restriction>
158     </xsd:simpleType>
159 </xsd:schema>

```

A.8 Custom XML Schema References

```

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.

B.1 WordprocessingML

```

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:vm1"
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 = xsd:integer { minInclusive = "0" maxInclusive = "600" }
53 w_CT_TextScale = attribute w:val { w_ST_TextScale }?
54 w_ST_HighlightColor =
55     string "black"
56     | string "blue"
57     | string "cyan"
58     | string "green"
59     | string "magenta"
60     | string "red"
61     | string "yellow"
62     | string "white"
63     | string "darkBlue"
64     | string "darkCyan"
65     | string "darkGreen"
66     | string "darkMagenta"
67     | string "darkRed"
68     | string "darkYellow"
69     | string "darkGray"
70     | string "lightGray"
71     | string "none"
72 w_CT_Highlight = attribute w:val { w_ST_HighlightColor }
73 w_ST_HexColorAuto = string "auto"
74 w_ST_HexColor = w_ST_HexColorAuto | s_ST_HexColorRGB
75 w_CT_Color =
76     attribute w:val { w_ST_HexColor },
77     attribute w:themeColor { w_ST_ThemeColor }?,
78     attribute w:themeTint { w_ST_UcharHexNumber }?,
79     attribute w:themeShade { w_ST_UcharHexNumber }?
80 w_CT_Lang = attribute w:val { s_ST_Lang }
81 w_CT_Guid = attribute w:val { s_ST_Guid }?
82 w_ST_Underline =
83     string "single"
84     | string "words"
85     | string "double"
86     | string "thick"

```

```

87 | string "dotted"
88 | string "dottedHeavy"
89 | string "dash"
90 | string "dashedHeavy"
91 | string "dashLong"
92 | string "dashLongHeavy"
93 | string "dotDash"
94 | string "dashDotHeavy"
95 | string "dotDotDash"
96 | string "dashDotDotHeavy"
97 | string "wave"
98 | string "wavyHeavy"
99 | string "wavyDouble"
100 | string "none"
101 w_CT_Underline =
102   attribute w:val { w_ST_Underline }?,
103   attribute w:color { w_ST_HexColor }?,
104   attribute w:themeColor { w_ST_ThemeColor }?,
105   attribute w:themeTint { w_ST_UcharHexNumber }?,
106   attribute w:themeShade { w_ST_UcharHexNumber }?
107 w_ST_TextEffect =
108   string "blinkBackground"
109   | string "lights"
110   | string "antsBlack"
111   | string "antsRed"
112   | string "shimmer"
113   | string "sparkle"
114   | string "none"
115 w_CT_TextEffect = attribute w:val { w_ST_TextEffect }
116 w_ST_Border =
117   string "nil"
118   | string "none"
119   | string "single"
120   | string "thick"
121   | string "double"
122   | string "dotted"
123   | string "dashed"
124   | string "dotDash"
125   | string "dotDotDash"
126   | string "triple"
127   | string "thinThickSmallGap"
128   | string "thickThinSmallGap"
129   | string "thinThickThinSmallGap"
130   | string "thinThickMediumGap"
131   | string "thickThinMediumGap"
132   | string "thinThickThinMediumGap"
133   | string "thinThickLargeGap"
134   | string "thickThinLargeGap"
135   | string "thinThickThinLargeGap"
136   | string "wave"
137   | string "doubleWave"
138   | string "dashSmallGap"
139   | string "dashDotStroked"

```

```

140 | string "threeDEmboss"
141 | string "threeDEngrave"
142 | string "outset"
143 | string "inset"
144 | string "apples"
145 | string "archedScallops"
146 | string "babyPacifier"
147 | string "babyRattle"
148 | string "balloons3Colors"
149 | string "balloonsHotAir"
150 | string "basicBlackDashes"
151 | string "basicBlackDots"
152 | string "basicBlackSquares"
153 | string "basicThinLines"
154 | string "basicWhiteDashes"
155 | string "basicWhiteDots"
156 | string "basicWhiteSquares"
157 | string "basicWideInline"
158 | string "basicWideMidline"
159 | string "basicWideOutline"
160 | string "bats"
161 | string "birds"
162 | string "birdsFlight"
163 | string "cabins"
164 | string "cakeSlice"
165 | string "candyCorn"
166 | string "celticKnotwork"
167 | string "certificateBanner"
168 | string "chainLink"
169 | string "champagneBottle"
170 | string "checkedBarBlack"
171 | string "checkedBarColor"
172 | string "checkered"
173 | string "christmasTree"
174 | string "circlesLines"
175 | string "circlesRectangles"
176 | string "classicalWave"
177 | string "clocks"
178 | string "compass"
179 | string "confetti"
180 | string "confettiGrays"
181 | string "confettiOutline"
182 | string "confettiStreamers"
183 | string "confettiWhite"
184 | string "cornerTriangles"
185 | string "couponCutoutDashes"
186 | string "couponCutoutDots"
187 | string "crazyMaze"
188 | string "creaturesButterfly"
189 | string "creaturesFish"
190 | string "creaturesInsects"
191 | string "creaturesLadyBug"
192 | string "crossStitch"

```

```

193 | string "cup"
194 | string "decoArch"
195 | string "decoArchColor"
196 | string "decoBlocks"
197 | string "diamondsGray"
198 | string "doubleD"
199 | string "doubleDiamonds"
200 | string "earth1"
201 | string "earth2"
202 | string "earth3"
203 | string "eclipsingSquares1"
204 | string "eclipsingSquares2"
205 | string "eggsBlack"
206 | string "fans"
207 | string "film"
208 | string "firecrackers"
209 | string "flowersBlockPrint"
210 | string "flowersDaisies"
211 | string "flowersModern1"
212 | string "flowersModern2"
213 | string "flowersPansy"
214 | string "flowersRedRose"
215 | string "flowersRoses"
216 | string "flowersTeacup"
217 | string "flowersTiny"
218 | string "gems"
219 | string "gingerbreadMan"
220 | string "gradient"
221 | string "handmade1"
222 | string "handmade2"
223 | string "heartBalloon"
224 | string "heartGray"
225 | string "hearts"
226 | string "heebieJeebies"
227 | string "holly"
228 | string "houseFunky"
229 | string "hypnotic"
230 | string "iceCreamCones"
231 | string "lightBulb"
232 | string "lightning1"
233 | string "lightning2"
234 | string "mapPins"
235 | string "mapleLeaf"
236 | string "mapleMuffins"
237 | string "marquee"
238 | string "marqueeToothed"
239 | string "moons"
240 | string "mosaic"
241 | string "musicNotes"
242 | string "northwest"
243 | string "ovals"
244 | string "packages"
245 | string "palmsBlack"

```

246	string "palmsColor"
247	string "paperClips"
248	string "papyrus"
249	string "partyFavor"
250	string "partyGlass"
251	string "pencils"
252	string "people"
253	string "peopleWaving"
254	string "peopleHats"
255	string "poinsettias"
256	string "postageStamp"
257	string "pumpkin1"
258	string "pushPinNote2"
259	string "pushPinNote1"
260	string "pyramids"
261	string "pyramidsAbove"
262	string "quadrants"
263	string "rings"
264	string "safari"
265	string "sawtooth"
266	string "sawtoothGray"
267	string "scaredCat"
268	string "seattle"
269	string "shadowedSquares"
270	string "sharksTeeth"
271	string "shorebirdTracks"
272	string "skyrocket"
273	string "snowflakeFancy"
274	string "snowflakes"
275	string "sombrero"
276	string "southwest"
277	string "stars"
278	string "starsTop"
279	string "stars3d"
280	string "starsBlack"
281	string "starsShadowed"
282	string "sun"
283	string "swirligig"
284	string "tornPaper"
285	string "tornPaperBlack"
286	string "trees"
287	string "triangleParty"
288	string "triangles"
289	string "triangle1"
290	string "triangle2"
291	string "triangleCircle1"
292	string "triangleCircle2"
293	string "shapes1"
294	string "shapes2"
295	string "twistedLines1"
296	string "twistedLines2"
297	string "vine"
298	string "waveline"

```

299 | string "weavingAngles"
300 | string "weavingBraid"
301 | string "weavingRibbon"
302 | string "weavingStrips"
303 | string "whiteFlowers"
304 | string "woodwork"
305 | string "xIllusions"
306 | string "zanyTriangles"
307 | string "zigZag"
308 | string "zigZagStitch"
309 | string "custom"
310 w_CT_Border =
311   attribute w:val { w_ST_Border },
312   attribute w:color { w_ST_HexColor }?,
313   attribute w:themeColor { w_ST_ThemeColor }?,
314   attribute w:themeTint { w_ST_UcharHexNumber }?,
315   attribute w:themeShade { w_ST_UcharHexNumber }?,
316   attribute w:sz { w_ST_EighthPointMeasure }?,
317   attribute w:space { w_ST_PointMeasure }?,
318   attribute w:shadow { s_ST_OnOff }?,
319   attribute w:frame { s_ST_OnOff }?
320 w_ST_Shd =
321   string "nil"
322   | string "clear"
323   | string "solid"
324   | string "horzStripe"
325   | string "vertStripe"
326   | string "reverseDiagStripe"
327   | string "diagStripe"
328   | string "horzCross"
329   | string "diagCross"
330   | string "thinHorzStripe"
331   | string "thinVertStripe"
332   | string "thinReverseDiagStripe"
333   | string "thinDiagStripe"
334   | string "thinHorzCross"
335   | string "thinDiagCross"
336   | string "pct5"
337   | string "pct10"
338   | string "pct12"
339   | string "pct15"
340   | string "pct20"
341   | string "pct25"
342   | string "pct30"
343   | string "pct35"
344   | string "pct37"
345   | string "pct40"
346   | string "pct45"
347   | string "pct50"
348   | string "pct55"
349   | string "pct60"
350   | string "pct62"
351   | string "pct65"

```

```

352 | string "pct70"
353 | string "pct75"
354 | string "pct80"
355 | string "pct85"
356 | string "pct87"
357 | string "pct90"
358 | string "pct95"
359 w_CT_Shd =
360   attribute w:val { w_ST_Shd },
361   attribute w:color { w_ST_HexColor }?,
362   attribute w:themeColor { w_ST_ThemeColor }?,
363   attribute w:themeTint { w_ST_UcharHexNumber }?,
364   attribute w:themeShade { w_ST_UcharHexNumber }?,
365   attribute w:fill { w_ST_HexColor }?,
366   attribute w:themeFill { w_ST_ThemeColor }?,
367   attribute w:themeFillTint { w_ST_UcharHexNumber }?,
368   attribute w:themeFillShade { w_ST_UcharHexNumber }?
369 w_CT_VerticalAlignRun = attribute w:val { s_ST_VerticalAlignRun }
370 w_CT_FitText =
371   attribute w:val { s_ST_TwipsMeasure },
372   attribute w:id { w_ST_DecimalNumber }?
373 w_ST_Em =
374   string "none"
375   | string "dot"
376   | string "comma"
377   | string "circle"
378   | string "underDot"
379 w_CT_Em = attribute w:val { w_ST_Em }
380 w_CT_Language =
381   attribute w:val { s_ST_Lang }?,
382   attribute w:eastAsia { s_ST_Lang }?,
383   attribute w:bidirectional { s_ST_Lang }?
384 w_ST_CombineBrackets =
385   string "none"
386   | string "round"
387   | string "square"
388   | string "angle"
389   | string "curly"
390 w_CT_EastAsianLayout =
391   attribute w:id { w_ST_DecimalNumber }?,
392   attribute w:combine { s_ST_OnOff }?,
393   attribute w:combineBrackets { w_ST_CombineBrackets }?,
394   attribute w:vertical { s_ST_OnOff }?,
395   attribute w:verticalCompress { s_ST_OnOff }?
396 w_ST_HeightRule = string "auto" | string "exact" | string "atLeast"
397 w_ST_Wrap =
398   string "auto"
399   | string "notBeside"
400   | string "around"
401   | string "tight"
402   | string "through"
403   | string "none"
404 w_ST_VAnchor = string "text" | string "margin" | string "page"

```



```

405 w_ST_HAnchor = string "text" | string "margin" | string "page"
406 w_ST_DropCap = string "none" | string "drop" | string "margin"
407 w_CT_FramePr =
408     attribute w:dropCap { w_ST_DropCap }?,
409     attribute w:lines { w_ST_DecimalNumber }?,
410     attribute w:w { s_ST_TwipsMeasure }?,
411     attribute w:h { s_ST_TwipsMeasure }?,
412     attribute w:vSpace { s_ST_TwipsMeasure }?,
413     attribute w:hSpace { s_ST_TwipsMeasure }?,
414     attribute w:wrap { w_ST_Wrap }?,
415     attribute w:hAnchor { w_ST_HAnchor }?,
416     attribute w:vAnchor { w_ST_VAnchor }?,
417     attribute w:x { w_ST_SignedTwipsMeasure }?,
418     attribute w:xAlign { s_ST_XAlign }?,
419     attribute w:y { w_ST_SignedTwipsMeasure }?,
420     attribute w:yAlign { s_ST_YAlign }?,
421     attribute w:hRule { w_ST_HeightRule }?,
422     attribute w:anchorLock { s_ST_OnOff }?
423 w_ST_TabJc =
424     string "clear"
425     | string "start"
426     | string "center"
427     | string "end"
428     | string "decimal"
429     | string "bar"
430     | string "num"
431     | string "left"
432     | string "right"
433 w_ST_TabTlc =
434     string "none"
435     | string "dot"
436     | string "hyphen"
437     | string "underscore"
438     | string "heavy"
439     | string "middleDot"
440 w_CT_TabStop =
441     attribute w:val { w_ST_TabJc },
442     attribute w:leader { w_ST_TabTlc }?,
443     attribute w:pos { w_ST_SignedTwipsMeasure }
444 w_ST_LineSpacingRule = string "auto" | string "exact" | string "atLeast"
445 w_CT_Spacing =
446     attribute w:before { s_ST_TwipsMeasure }?,
447     attribute w:beforeLines { w_ST_DecimalNumber }?,
448     attribute w:beforeAutospacing { s_ST_OnOff }?,
449     attribute w:after { s_ST_TwipsMeasure }?,
450     attribute w:afterLines { w_ST_DecimalNumber }?,
451     attribute w:afterAutospacing { s_ST_OnOff }?,
452     attribute w:line { w_ST_SignedTwipsMeasure }?,
453     attribute w:lineRule { w_ST_LineSpacingRule }?
454 w_CT_Ind =
455     attribute w:start { w_ST_SignedTwipsMeasure }?,
456     attribute w:startChars { w_ST_DecimalNumber }?,
457     attribute w:end { w_ST_SignedTwipsMeasure }?,

```

```

458     attribute w:endChars { w_ST_DecimalNumber }?,
459     attribute w:left { w_ST_SignedTwipsMeasure }?,
460     attribute w:leftChars { w_ST_DecimalNumber }?,
461     attribute w:right { w_ST_SignedTwipsMeasure }?,
462     attribute w:rightChars { w_ST_DecimalNumber }?,
463     attribute w:hanging { s_ST_TwipsMeasure }?,
464     attribute w:hangingChars { w_ST_DecimalNumber }?,
465     attribute w:firstLine { s_ST_TwipsMeasure }?,
466     attribute w:firstLineChars { w_ST_DecimalNumber }?
467 w_ST_Jc =
468     string "start"
469     | string "center"
470     | string "end"
471     | string "both"
472     | string "mediumKashida"
473     | string "distribute"
474     | string "numTab"
475     | string "highKashida"
476     | string "lowKashida"
477     | string "thaiDistribute"
478     | string "left"
479     | string "right"
480 w_ST_JcTable =
481     string "center"
482     | string "end"
483     | string "left"
484     | string "right"
485     | string "start"
486 w_CT_Jc = attribute w:val { w_ST_Jc }
487 w_CT_JcTable = attribute w:val { w_ST_JcTable }
488 w_ST_View =
489     string "none"
490     | string "print"
491     | string "outline"
492     | string "masterPages"
493     | string "normal"
494     | string "web"
495 w_CT_View = attribute w:val { w_ST_View }
496 w_ST_Zoom =
497     string "none"
498     | string "fullPage"
499     | string "bestFit"
500     | string "textFit"
501 w_CT_Zoom =
502     attribute w:val { w_ST_Zoom }?,
503     attribute w:percent { w_ST_DecimalNumberOrPercent }
504 w_CT_WritingStyle =
505     attribute w:lang { s_ST_Lang },
506     attribute w:vendorID { s_ST_String },
507     attribute w:dllVersion { s_ST_String },
508     attribute w:nlCheck { s_ST_OnOff }?,
509     attribute w:checkStyle { s_ST_OnOff },
510     attribute w:appName { s_ST_String }

```

```

511 w_ST_Proof = string "clean" | string "dirty"
512 w_CT_Proof =
513     attribute w:spelling { w_ST_Proof }?,
514     attribute w:grammar { w_ST_Proof }?
515 w_ST_DocType = xsd:string
516 w_CT_DocType = attribute w:val { w_ST_DocType }
517 w_ST_DocProtect =
518     string "none"
519     | string "readOnly"
520     | string "comments"
521     | string "trackedChanges"
522     | string "forms"
523 w_AG_Password =
524     attribute w:algorithmName { s_ST_String }?,
525     attribute w:hashValue { xsd:base64Binary }?,
526     attribute w:saltValue { xsd:base64Binary }?,
527     attribute w:spinCount { w_ST_DecimalNumber }?
528 w_AG_TransitionalPassword =
529     attribute w:cryptProviderType { s_ST_CryptProv }?,
530     attribute w:cryptAlgorithmClass { s_ST_AlgClass }?,
531     attribute w:cryptAlgorithmType { s_ST_AlgType }?,
532     attribute w:cryptAlgorithmSid { w_ST_DecimalNumber }?,
533     attribute w:cryptSpinCount { w_ST_DecimalNumber }?,
534     attribute w:cryptProvider { s_ST_String }?,
535     attribute w:algIdExt { w_ST_LongHexNumber }?,
536     attribute w:algIdExtSource { s_ST_String }?,
537     attribute w:cryptProviderTypeExt { w_ST_LongHexNumber }?,
538     attribute w:cryptProviderTypeExtSource { s_ST_String }?,
539     attribute w:hash { xsd:base64Binary }?,
540     attribute w:salt { xsd:base64Binary }?
541 w_CT_DocProtect =
542     attribute w:edit { w_ST_DocProtect }?,
543     attribute w:formatting { s_ST_OnOff }?,
544     attribute w:enforcement { s_ST_OnOff }?,
545     w_AG_Password,
546     w_AG_TransitionalPassword
547 w_ST_MailMergeDocType =
548     string "catalog"
549     | string "envelopes"
550     | string "mailingLabels"
551     | string "formLetters"
552     | string "email"
553     | string "fax"
554 w_CT_MailMergeDocType = attribute w:val { w_ST_MailMergeDocType }
555 w_ST_MailMergeDataType = xsd:string
556 w_CT_MailMergeDataType = attribute w:val { w_ST_MailMergeDataType }
557 w_ST_MailMergeDest =
558     string "newDocument"
559     | string "printer"
560     | string "email"
561     | string "fax"
562 w_CT_MailMergeDest = attribute w:val { w_ST_MailMergeDest }
563 w_ST_MailMergeOdsoFMDFieldType = string "null" | string "dbColumn"

```

```

564 w_CT_MailMergeOdsoFMDFieldType =
565     attribute w:val { w_ST_MailMergeOdsoFMDFieldType }
566 w_CT_TrackChangesView =
567     attribute w:markup { s_ST_OnOff }?,
568     attribute w:comments { s_ST_OnOff }?,
569     attribute w:insDel { s_ST_OnOff }?,
570     attribute w:formatting { s_ST_OnOff }?,
571     attribute w:inkAnnotations { s_ST_OnOff }?
572 w_CT_Kinsoku =
573     attribute w:lang { s_ST_Lang },
574     attribute w:val { s_ST_String }
575 w_ST_TextDirection =
576     string "tb"
577     | string "rl"
578     | string "lr"
579     | string "tbV"
580     | string "rlV"
581     | string "lrV"
582     | string "btLr"
583     | string "lrTb"
584     | string "lrTbV"
585     | string "tbLrV"
586     | string "tbRl"
587     | string "tbRlV"
588 w_CT_TextDirection = attribute w:val { w_ST_TextDirection }
589 w_ST_TextAlignment =
590     string "top"
591     | string "center"
592     | string "baseline"
593     | string "bottom"
594     | string "auto"
595 w_CT_TextAlignment = attribute w:val { w_ST_TextAlignment }
596 w_ST_DisplacedByCustomXml = string "next" | string "prev"
597 w_ST_AnnotationVMerge = string "cont" | string "rest"
598 w_CT_Markup = attribute w:id { w_ST_DecimalNumber }
599 w_CT_TrackChange =
600     w_CT_Markup,
601     attribute w:author { s_ST_String },
602     attribute w:date { w_ST_DateTime }?
603 w_CT_CellMergeTrackChange =
604     w_CT_TrackChange,
605     attribute w:vMerge { w_ST_AnnotationVMerge }?,
606     attribute w:vMergeOrig { w_ST_AnnotationVMerge }?
607 w_CT_TrackChangeRange =
608     w_CT_TrackChange,
609     attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
610 w_CT_MarkupRange =
611     w_CT_Markup,
612     attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
613 w_CT_BookmarkRange =
614     w_CT_MarkupRange,
615     attribute w:colFirst { w_ST_DecimalNumber }?,
616     attribute w:colLast { w_ST_DecimalNumber }?

```

```

617 w_CT_Bookmark =
618     w_CT_BookmarkRange,
619     attribute w:name { s_ST_String }
620 w_CT_MoveBookmark =
621     w_CT_Bookmark,
622     attribute w:author { s_ST_String },
623     attribute w:date { w_ST_DateTime }
624 w_CT_Comment =
625     w_CT_TrackChange,
626     w_EG_BlockLevelElts*,
627     attribute w:initials { s_ST_String }?
628 w_CT_TrackChangeNumbering =
629     w_CT_TrackChange,
630     attribute w:original { s_ST_String }?
631 w_CT_TblPrExChange =
632     w_CT_TrackChange,
633     element tblPrEx { w_CT_TblPrExBase }
634 w_CT_TcPrChange =
635     w_CT_TrackChange,
636     element tcPr { w_CT_TcPrInner }
637 w_CT_TrPrChange =
638     w_CT_TrackChange,
639     element trPr { w_CT_TrPrBase }
640 w_CT_TblGridChange =
641     w_CT_Markup,
642     element tblGrid { w_CT_TblGridBase }
643 w_CT_TblPrChange =
644     w_CT_TrackChange,
645     element tblPr { w_CT_TblPrBase }
646 w_CT_SectPrChange =
647     w_CT_TrackChange,
648     element sectPr { w_CT_SectPrBase }?
649 w_CT_PPrChange =
650     w_CT_TrackChange,
651     element pPr { w_CT_PPrBase }
652 w_CT_RPrChange =
653     w_CT_TrackChange,
654     element rPr { w_CT_RPrOriginal }
655 w_CT_ParaRPrChange =
656     w_CT_TrackChange,
657     element rPr { w_CT_ParaRPrOriginal }
658 w_CT_RunTrackChange =
659     w_CT_TrackChange, (w_EG_ContentRunContent | m_EG_OMathMathElements)*
660 w_EG_CellMarkupElements =
661     element cellIns { w_CT_TrackChange }?
662     | element cellDel { w_CT_TrackChange }?
663     | element cellMerge { w_CT_CellMergeTrackChange }?
664 w_EG_RangeMarkupElements =
665     element bookmarkStart { w_CT_Bookmark }
666     | element bookmarkEnd { w_CT_MarkupRange }
667     | element moveFromRangeStart { w_CT_MoveBookmark }
668     | element moveFromRangeEnd { w_CT_MarkupRange }
669     | element moveToRangeStart { w_CT_MoveBookmark }

```

```

670 | element moveToRangeEnd { w_CT_MarkupRange }
671 | element commentRangeStart { w_CT_MarkupRange }
672 | element commentRangeEnd { w_CT_MarkupRange }
673 | element customXmlInsRangeStart { w_CT_TrackChange }
674 | element customXmlInsRangeEnd { w_CT_Markup }
675 | element customXmlDelRangeStart { w_CT_TrackChange }
676 | element customXmlDelRangeEnd { w_CT_Markup }
677 | element customXmlMoveFromRangeStart { w_CT_TrackChange }
678 | element customXmlMoveFromRangeEnd { w_CT_Markup }
679 | element customXmlMoveToRangeStart { w_CT_TrackChange }
680 | element customXmlMoveToRangeEnd { w_CT_Markup }
681 w_CT_NumPr =
682   element ilvl { w_CT_DecimalNumber }?,
683   element numId { w_CT_DecimalNumber }?,
684   element numberingChange { w_CT_TrackChangeNumbering }?,
685   element ins { w_CT_TrackChange }?
686 w_CT_PBdr =
687   element top { w_CT_Border }?,
688   element left { w_CT_Border }?,
689   element bottom { w_CT_Border }?,
690   element right { w_CT_Border }?,
691   element between { w_CT_Border }?,
692   element bar { w_CT_Border }?
693 w_CT_Tabs = element tab { w_CT_TabStop }+
694 w_ST_TextboxTightWrap =
695   string "none"
696   | string "allLines"
697   | string "firstAndLastLine"
698   | string "firstLineOnly"
699   | string "lastLineOnly"
700 w_CT_TextboxTightWrap = attribute w:val { w_ST_TextboxTightWrap }
701 w_CT_PPr =
702   w_CT_PPrBase,
703   element rPr { w_CT_ParaRPr }?,
704   element sectPr { w_CT_SectPr }?,
705   element pPrChange { w_CT_PPrChange }?
706 w_CT_PPrBase =
707   element pStyle { w_CT_String }?,
708   element keepNext { w_CT_OnOff }?,
709   element keepLines { w_CT_OnOff }?,
710   element pageBreakBefore { w_CT_OnOff }?,
711   element framePr { w_CT_FramePr }?,
712   element widowControl { w_CT_OnOff }?,
713   element numPr { w_CT_NumPr }?,
714   element suppressLineNumbers { w_CT_OnOff }?,
715   element pBdr { w_CT_PBdr }?,
716   element shd { w_CT_Shd }?,
717   element tabs { w_CT_Tabs }?,
718   element suppressAutoHyphens { w_CT_OnOff }?,
719   element kinsoku { w_CT_OnOff }?,
720   element wordWrap { w_CT_OnOff }?,
721   element overflowPunct { w_CT_OnOff }?,
722   element toplinePunct { w_CT_OnOff }?,

```

```

723 element autoSpaceDE { w_CT_OnOff }?,
724 element autoSpaceDN { w_CT_OnOff }?,
725 element bidi { w_CT_OnOff }?,
726 element adjustRightInd { w_CT_OnOff }?,
727 element snapToGrid { w_CT_OnOff }?,
728 element spacing { w_CT_Spacing }?,
729 element ind { w_CT_Ind }?,
730 element contextualSpacing { w_CT_OnOff }?,
731 element mirrorIndents { w_CT_OnOff }?,
732 element suppressOverlap { w_CT_OnOff }?,
733 element jc { w_CT_Jc }?,
734 element textDirection { w_CT_TextDirection }?,
735 element textAlignment { w_CT_TextAlignment }?,
736 element textboxTightWrap { w_CT_TextboxTightWrap }?,
737 element outlineLvl { w_CT_DecimalNumber }?,
738 element divId { w_CT_DecimalNumber }?,
739 element cnfStyle { w_CT_Cnf }?
740 w_CT_PPrGeneral =
741   w_CT_PPrBase,
742   element pPrChange { w_CT_PPrChange }?
743 w_CT_Control =
744   attribute w:name { s_ST_String }?,
745   attribute w:shapeid { s_ST_String }?,
746   r_id?
747 w_CT_Background =
748   attribute w:color { w_ST_HexColor }?,
749   attribute w:themeColor { w_ST_ThemeColor }?,
750   attribute w:themeTint { w_ST_UcharHexNumber }?,
751   attribute w:themeShade { w_ST_UcharHexNumber }?,
752   (w_any_vml_vml*, w_any_vml_office*)+,
753   element drawing { w_CT_Drawing }?
754 w_CT_Rel = r_id
755 w_CT_Object =
756   attribute w:dxaOrig { s_ST_TwipsMeasure }?,
757   attribute w:dyaOrig { s_ST_TwipsMeasure }?,
758   (w_any_vml_vml*, w_any_vml_office*)+,
759   element drawing { w_CT_Drawing }?,
760   (element control { w_CT_Control }
761    | element objectLink { w_CT_ObjectLink }
762    | element objectEmbed { w_CT_ObjectEmbed }
763    | element movie { w_CT_Rel })?
764 w_CT_Picture =
765   (w_any_vml_vml*, w_any_vml_office*)+,
766   element movie { w_CT_Rel }?,
767   element control { w_CT_Control }?
768 w_CT_ObjectEmbed =
769   attribute w:drawAspect { w_ST_ObjectDrawAspect }?,
770   r_id,
771   attribute w:progId { s_ST_String }?,
772   attribute w:shapeId { s_ST_String }?,
773   attribute w:fieldCodes { s_ST_String }?
774 w_ST_ObjectDrawAspect = string "content" | string "icon"
775 w_CT_ObjectLink =

```

```

776 w_CT_ObjectEmbed,
777 attribute w:updateMode { w_ST_ObjectUpdateMode },
778 attribute w:lockedField { s_ST_OnOff }?
779 w_ST_ObjectUpdateMode = string "always" | string "onCall"
780 w_CT_Drawing = (wp_anchor? | wp_inline?)+
781 w_CT_SimpleField =
782 attribute w:instr { s_ST_String },
783 attribute w:fldLock { s_ST_OnOff }?,
784 attribute w:dirty { s_ST_OnOff }?,
785 element fldData { w_CT_Text }?,
786 w_EG_PContent*
787 w_ST_FldCharType = string "begin" | string "separate" | string "end"
788 w_ST_InfoTextType = string "text" | string "autoText"
789 w_ST_FFHelpTextVal = xsd:string { maxLength = "256" }
790 w_ST_FFStatusTextVal = xsd:string { maxLength = "140" }
791 w_ST_FFName = xsd:string { maxLength = "65" }
792 w_ST_FFTextType =
793 string "regular"
794 | string "number"
795 | string "date"
796 | string "currentTime"
797 | string "currentDate"
798 | string "calculated"
799 w_CT_FFTextType = attribute w:val { w_ST_FFTextType }
800 w_CT_FFName = attribute w:val { w_ST_FFName }?
801 w_CT_FldChar =
802 attribute w:fldCharType { w_ST_FldCharType },
803 attribute w:fldLock { s_ST_OnOff }?,
804 attribute w:dirty { s_ST_OnOff }?,
805 (element fldData { w_CT_Text }?
806 | element ffData { w_CT_FFData }?
807 | element numberingChange { w_CT_TrackChangeNumbering }?)
808 w_CT_Hyperlink =
809 attribute w:tgtFrame { s_ST_String }?,
810 attribute w:tooltip { s_ST_String }?,
811 attribute w:docLocation { s_ST_String }?,
812 attribute w:history { s_ST_OnOff }?,
813 attribute w:anchor { s_ST_String }?,
814 r_id?,
815 w_EG_PContent*
816 w_CT_FFData =
817 (element name { w_CT_FFName }
818 | element label { w_CT_DecimalNumber }?
819 | element tabIndex { w_CT_UnsignedDecimalNumber }?
820 | element enabled { w_CT_OnOff }
821 | element calcOnExit { w_CT_OnOff }
822 | element entryMacro { w_CT_MacroName }?
823 | element exitMacro { w_CT_MacroName }?
824 | element helpText { w_CT_FFHelpText }?
825 | element statusText { w_CT_FFStatusText }?
826 | (element checkBox { w_CT_FFCheckBox }
827 | element ddList { w_CT_FFDDLList }
828 | element textInput { w_CT_FFTextInput })))

```



```

829 w_CT_FFHelpText =
830     attribute w:type { w_ST_InfoTextType }?,
831     attribute w:val { w_ST_FFHelpTextVal }?
832 w_CT_FFStatusText =
833     attribute w:type { w_ST_InfoTextType }?,
834     attribute w:val { w_ST_FFStatusTextVal }?
835 w_CT_FFCheckBox =
836     (element size { w_CT_HpsMeasure }
837     | element sizeAuto { w_CT_OnOff }?),
838     element default { w_CT_OnOff }?,
839     element checked { w_CT_OnOff }?
840 w_CT_FFDDLList =
841     element result { w_CT_DecimalNumber }?,
842     element default { w_CT_DecimalNumber }?,
843     element listEntry { w_CT_String }*
844 w_CT_FFTextInput =
845     element type { w_CT_FFTextType }?,
846     element default { w_CT_String }?,
847     element maxLength { w_CT_DecimalNumber }?,
848     element format { w_CT_String }?
849 w_ST_SectionMark =
850     string "nextPage"
851     | string "nextColumn"
852     | string "continuous"
853     | string "evenPage"
854     | string "oddPage"
855 w_CT_SectType = attribute w:val { w_ST_SectionMark }?
856 w_CT_PaperSource =
857     attribute w:first { w_ST_DecimalNumber }?,
858     attribute w:other { w_ST_DecimalNumber }?
859 w_ST_NumberFormat =
860     string "decimal"
861     | string "upperRoman"
862     | string "lowerRoman"
863     | string "upperLetter"
864     | string "lowerLetter"
865     | string "ordinal"
866     | string "cardinalText"
867     | string "ordinalText"
868     | string "hex"
869     | string "chicago"
870     | string "ideographDigital"
871     | string "japaneseCounting"
872     | string "aiueo"
873     | string "iroha"
874     | string "decimalFullWidth"
875     | string "decimalHalfWidth"
876     | string "japaneseLegal"
877     | string "japaneseDigitalTenThousand"
878     | string "decimalEnclosedCircle"
879     | string "decimalFullWidth2"
880     | string "aiueoFullWidth"
881     | string "irohaFullWidth"

```

```

882 | string "decimalZero"
883 | string "bullet"
884 | string "ganada"
885 | string "chosung"
886 | string "decimalEnclosedFullstop"
887 | string "decimalEnclosedParen"
888 | string "decimalEnclosedCircleChinese"
889 | string "ideographEnclosedCircle"
890 | string "ideographTraditional"
891 | string "ideographZodiac"
892 | string "ideographZodiacTraditional"
893 | string "taiwaneseCounting"
894 | string "ideographLegalTraditional"
895 | string "taiwaneseCountingThousand"
896 | string "taiwaneseDigital"
897 | string "chineseCounting"
898 | string "chineseLegalSimplified"
899 | string "chineseCountingThousand"
900 | string "koreanDigital"
901 | string "koreanCounting"
902 | string "koreanLegal"
903 | string "koreanDigital2"
904 | string "vietnameseCounting"
905 | string "russianLower"
906 | string "russianUpper"
907 | string "none"
908 | string "numberInDash"
909 | string "hebrew1"
910 | string "hebrew2"
911 | string "arabicAlpha"
912 | string "arabicAbjad"
913 | string "hindiVowels"
914 | string "hindiConsonants"
915 | string "hindiNumbers"
916 | string "hindiCounting"
917 | string "thaiLetters"
918 | string "thaiNumbers"
919 | string "thaiCounting"
920 | string "bahtText"
921 | string "dollarText"
922 | string "custom"
923 w_ST_PageOrientation = string "portrait" | string "landscape"
924 w_CT_PageSz =
925     attribute w:w { s_ST_TwipsMeasure }?,
926     attribute w:h { s_ST_TwipsMeasure }?,
927     attribute w:orient { w_ST_PageOrientation }?,
928     attribute w:code { w_ST_DecimalNumber }?
929 w_CT_PageMar =
930     attribute w:top { w_ST_SignedTwipsMeasure },
931     attribute w:right { s_ST_TwipsMeasure },
932     attribute w:bottom { w_ST_SignedTwipsMeasure },
933     attribute w:left { s_ST_TwipsMeasure },
934     attribute w:header { s_ST_TwipsMeasure },

```

```

935     attribute w:footer { s_ST_TwipsMeasure },
936     attribute w:gutter { s_ST_TwipsMeasure }
937 w_ST_PageBorderZOrder = string "front" | string "back"
938 w_ST_PageBorderDisplay =
939     string "allPages" | string "firstPage" | string "notFirstPage"
940 w_ST_PageBorderOffset = string "page" | string "text"
941 w_CT_PageBorders =
942     attribute w:zOrder { w_ST_PageBorderZOrder }?,
943     attribute w:display { w_ST_PageBorderDisplay }?,
944     attribute w:offsetFrom { w_ST_PageBorderOffset }?,
945     element top { w_CT_TopPageBorder }?,
946     element left { w_CT_PageBorder }?,
947     element bottom { w_CT_BottomPageBorder }?,
948     element right { w_CT_PageBorder }?
949 w_CT_PageBorder = w_CT_Border, r_id?
950 w_CT_BottomPageBorder = w_CT_PageBorder, r_bottomLeft?, r_bottomRight?
951 w_CT_TopPageBorder = w_CT_PageBorder, r_topLeft?, r_topRight?
952 w_ST_ChapterSep =
953     string "hyphen"
954     | string "period"
955     | string "colon"
956     | string "emDash"
957     | string "enDash"
958 w_ST_LineNumberRestart =
959     string "newPage" | string "newSection" | string "continuous"
960 w_CT_LineNumber =
961     attribute w:countBy { w_ST_DecimalNumber }?,
962     attribute w:start { w_ST_DecimalNumber }?,
963     attribute w:distance { s_ST_TwipsMeasure }?,
964     attribute w:restart { w_ST_LineNumberRestart }?
965 w_CT_PageNumber =
966     attribute w:fmt { w_ST_NumberFormat }?,
967     attribute w:start { w_ST_DecimalNumber }?,
968     attribute w:chapStyle { w_ST_DecimalNumber }?,
969     attribute w:chapSep { w_ST_ChapterSep }?
970 w_CT_Column =
971     attribute w:w { s_ST_TwipsMeasure }?,
972     attribute w:space { s_ST_TwipsMeasure }?
973 w_CT_Columns =
974     attribute w:equalWidth { s_ST_OnOff }?,
975     attribute w:space { s_ST_TwipsMeasure }?,
976     attribute w:num { w_ST_DecimalNumber }?,
977     attribute w:sep { s_ST_OnOff }?,
978     element col { w_CT_Column }*
979 w_ST_VerticalJc =
980     string "top" | string "center" | string "both" | string "bottom"
981 w_CT_VerticalJc = attribute w:val { w_ST_VerticalJc }
982 w_ST_DocGrid =
983     string "default"
984     | string "lines"
985     | string "linesAndChars"
986     | string "snapTOCars"
987 w_CT_DocGrid =

```

```

988     attribute w:type { w_ST_DocGrid }?,
989     attribute w:linePitch { w_ST_DecimalNumber }?,
990     attribute w:charSpace { w_ST_DecimalNumber }?
991 w_ST_HdrFtr = string "even" | string "default" | string "first"
992 w_ST_FtnEdn =
993     string "normal"
994     | string "separator"
995     | string "continuationSeparator"
996     | string "continuationNotice"
997 w_CT_HdrFtrRef =
998     w_CT_Rel,
999     attribute w:type { w_ST_HdrFtr }
1000 w_EG_HdrFtrReferences =
1001     element headerReference { w_CT_HdrFtrRef }?
1002     | element footerReference { w_CT_HdrFtrRef }?
1003 w_CT_HdrFtr = w_EG_BlockLevelElts+
1004 w_EG_SectPrContents =
1005     element footnotePr { w_CT_FtnProps }?,
1006     element endnotePr { w_CT_EdnProps }?,
1007     element type { w_CT_SectType }?,
1008     element pgSz { w_CT_PageSz }?,
1009     element pgMar { w_CT_PageMar }?,
1010     element paperSrc { w_CT_PaperSource }?,
1011     element pgBorders { w_CT_PageBorders }?,
1012     element lnNumType { w_CT_LineNumber }?,
1013     element pgNumType { w_CT_PageNumber }?,
1014     element cols { w_CT_Columns }?,
1015     element formProt { w_CT_OnOff }?,
1016     element vAlign { w_CT_VerticalJc }?,
1017     element noEndnote { w_CT_OnOff }?,
1018     element titlePg { w_CT_OnOff }?,
1019     element textDirection { w_CT_TextDirection }?,
1020     element bidi { w_CT_OnOff }?,
1021     element rtlGutter { w_CT_OnOff }?,
1022     element docGrid { w_CT_DocGrid }?,
1023     element printerSettings { w_CT_Rel }?
1024 w_AG_SectPrAttributes =
1025     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1026     attribute w:rsidDel { w_ST_LongHexNumber }?,
1027     attribute w:rsidR { w_ST_LongHexNumber }?,
1028     attribute w:rsidSect { w_ST_LongHexNumber }?
1029 w_CT_SectPrBase = w_AG_SectPrAttributes, w_EG_SectPrContents?
1030 w_CT_SectPr =
1031     w_AG_SectPrAttributes,
1032     w_EG_HdrFtrReferences*,
1033     w_EG_SectPrContents?,
1034     element sectPrChange { w_CT_SectPrChange }?
1035 w_ST_BrType = string "page" | string "column" | string "textWrapping"
1036 w_ST_BrClear =
1037     string "none" | string "left" | string "right" | string "all"
1038 w_CT_Br =
1039     attribute w:type { w_ST_BrType }?,
1040     attribute w:clear { w_ST_BrClear }?

```

```

1041 w_ST_PTabAlignment = string "left" | string "center" | string "right"
1042 w_ST_PTabRelativeTo = string "margin" | string "indent"
1043 w_ST_PTabLeader =
1044     string "none"
1045     | string "dot"
1046     | string "hyphen"
1047     | string "underscore"
1048     | string "middleDot"
1049 w_CT_PTab =
1050     attribute w:alignment { w_ST_PTabAlignment },
1051     attribute w:relativeTo { w_ST_PTabRelativeTo },
1052     attribute w:leader { w_ST_PTabLeader }
1053 w_CT_Sym =
1054     attribute w:font { s_ST_String }?,
1055     attribute w:char { w_ST_ShortHexNumber }?
1056 w_ST_ProofErr =
1057     string "spellStart"
1058     | string "spellEnd"
1059     | string "gramStart"
1060     | string "gramEnd"
1061 w_CT_ProofErr = attribute w:type { w_ST_ProofErr }
1062 w_ST_EdGrp =
1063     string "none"
1064     | string "everyone"
1065     | string "administrators"
1066     | string "contributors"
1067     | string "editors"
1068     | string "owners"
1069     | string "current"
1070 w_CT_Perm =
1071     attribute w:id { s_ST_String },
1072     attribute w:displacedByCustomXml { w_ST_DisplacedByCustomXml }?
1073 w_CT_PermStart =
1074     w_CT_Perm,
1075     attribute w:edGrp { w_ST_EdGrp }?,
1076     attribute w:ed { s_ST_String }?,
1077     attribute w:colFirst { w_ST_DecimalNumber }?,
1078     attribute w:colLast { w_ST_DecimalNumber }?
1079 w_CT_Text = s_ST_String, xml_space?
1080 w_EG_RunInnerContent =
1081     element br { w_CT_Br }
1082     | element t { w_CT_Text }
1083     | element contentPart { w_CT_Rel }
1084     | element delText { w_CT_Text }
1085     | element instrText { w_CT_Text }
1086     | element delInstrText { w_CT_Text }
1087     | element noBreakHyphen { w_CT_Empty }
1088     | element softHyphen { w_CT_Empty }?
1089     | element dayShort { w_CT_Empty }?
1090     | element monthShort { w_CT_Empty }?
1091     | element yearShort { w_CT_Empty }?
1092     | element dayLong { w_CT_Empty }?
1093     | element monthLong { w_CT_Empty }?

```

```

1094 | element yearLong { w_CT_Empty }?
1095 | element annotationRef { w_CT_Empty }?
1096 | element footnoteRef { w_CT_Empty }?
1097 | element endnoteRef { w_CT_Empty }?
1098 | element separator { w_CT_Empty }?
1099 | element continuationSeparator { w_CT_Empty }?
1100 | element sym { w_CT_Sym }?
1101 | element pgNum { w_CT_Empty }?
1102 | element cr { w_CT_Empty }?
1103 | element tab { w_CT_Empty }?
1104 | element object { w_CT_Object }
1105 | element pict { w_CT_Picture }
1106 | element fldChar { w_CT_FldChar }
1107 | element ruby { w_CT_Ruby }
1108 | element footnoteReference { w_CT_FtnEdnRef }
1109 | element endnoteReference { w_CT_FtnEdnRef }
1110 | element commentReference { w_CT_Markup }
1111 | element drawing { w_CT_Drawing }
1112 | element ptab { w_CT_PTab }?
1113 | element lastRenderedPageBreak { w_CT_Empty }?
1114 w_CT_R =
1115     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1116     attribute w:rsidDel { w_ST_LongHexNumber }?,
1117     attribute w:rsidR { w_ST_LongHexNumber }?,
1118     w_EG_RPr?,
1119     w_EG_RunInnerContent*
1120 w_ST_Hint = string "default" | string "eastAsia" | string "cs"
1121 w_ST_Theme =
1122     string "majorEastAsia"
1123     | string "majorBidi"
1124     | string "majorAscii"
1125     | string "majorHAnsi"
1126     | string "minorEastAsia"
1127     | string "minorBidi"
1128     | string "minorAscii"
1129     | string "minorHAnsi"
1130 w_CT_Fonts =
1131     attribute w:hint { w_ST_Hint }?,
1132     attribute w:ascii { s_ST_String }?,
1133     attribute w:hAnsi { s_ST_String }?,
1134     attribute w:eastAsia { s_ST_String }?,
1135     attribute w:cs { s_ST_String }?,
1136     attribute w:asciiTheme { w_ST_Theme }?,
1137     attribute w:hAnsiTheme { w_ST_Theme }?,
1138     attribute w:eastAsiaTheme { w_ST_Theme }?,
1139     attribute w:cstheme { w_ST_Theme }?
1140 w_EG_RPrBase =
1141     element rStyle { w_CT_String }?,
1142     element rFonts { w_CT_Fonts }?,
1143     element b { w_CT_OnOff }?,
1144     element bCs { w_CT_OnOff }?,
1145     element i { w_CT_OnOff }?,
1146     element iCs { w_CT_OnOff }?,

```

```

1147 element caps { w_CT_OnOff }?,
1148 element smallCaps { w_CT_OnOff }?,
1149 element strike { w_CT_OnOff }?,
1150 element dstrike { w_CT_OnOff }?,
1151 element outline { w_CT_OnOff }?,
1152 element shadow { w_CT_OnOff }?,
1153 element emboss { w_CT_OnOff }?,
1154 element imprint { w_CT_OnOff }?,
1155 element noProof { w_CT_OnOff }?,
1156 element snapToGrid { w_CT_OnOff }?,
1157 element vanish { w_CT_OnOff }?,
1158 element webHidden { w_CT_OnOff }?,
1159 element color { w_CT_Color }?,
1160 element spacing { w_CT_SignedTwipsMeasure }?,
1161 element w { w_CT_TextScale }?,
1162 element kern { w_CT_HpsMeasure }?,
1163 element position { w_CT_SignedHpsMeasure }?,
1164 element sz { w_CT_HpsMeasure }?,
1165 element szCs { w_CT_HpsMeasure }?,
1166 element highlight { w_CT_Highlight }?,
1167 element u { w_CT_Underline }?,
1168 element effect { w_CT_TextEffect }?,
1169 element bdr { w_CT_Border }?,
1170 element shd { w_CT_Shadow }?,
1171 element fitText { w_CT_FitText }?,
1172 element vertAlign { w_CT_VerticalAlignRun }?,
1173 element rtl { w_CT_OnOff }?,
1174 element cs { w_CT_OnOff }?,
1175 element em { w_CT_Em }?,
1176 element lang { w_CT_Language }?,
1177 element eastAsianLayout { w_CT_EastAsianLayout }?,
1178 element specVanish { w_CT_OnOff }?,
1179 element oMath { w_CT_OnOff }?
1180 w_EG_RPrContent =
1181     w_EG_RPrBase?,
1182     element rPrChange { w_CT_RPrChange }?
1183 w_CT_RPr = w_EG_RPrContent?
1184 w_EG_RPr = element rPr { w_CT_RPr }?
1185 w_EG_RPrMath =
1186     w_EG_RPr
1187     | element ins { w_CT_RPrChange }
1188     | element del { w_CT_RPrChange }
1189 w_CT_RPrOriginal = w_EG_RPrBase*
1190 w_CT_ParaRPrOriginal = w_EG_ParaRPrTrackChanges?, w_EG_RPrBase*
1191 w_CT_ParaRPr =
1192     w_EG_ParaRPrTrackChanges?,
1193     w_EG_RPrBase?,
1194     element rPrChange { w_CT_ParaRPrChange }?
1195 w_EG_ParaRPrTrackChanges =
1196     element ins { w_CT_TrackChange }?,
1197     element del { w_CT_TrackChange }?,
1198     element moveFrom { w_CT_TrackChange }?,
1199     element moveTo { w_CT_TrackChange }?

```

```

1200 w_CT_AltChunk =
1201     r_id?,
1202     element altChunkPr { w_CT_AltChunkPr }?
1203 w_CT_AltChunkPr = element matchSrc { w_CT_OnOff }?
1204 w_ST_RubyAlign =
1205     string "center"
1206     | string "distributeLetter"
1207     | string "distributeSpace"
1208     | string "left"
1209     | string "right"
1210     | string "rightVertical"
1211 w_CT_RubyAlign = attribute w:val { w_ST_RubyAlign }
1212 w_CT_RubyPr =
1213     element rubyAlign { w_CT_RubyAlign },
1214     element hps { w_CT_HpsMeasure },
1215     element hpsRaise { w_CT_HpsMeasure },
1216     element hpsBaseText { w_CT_HpsMeasure },
1217     element lid { w_CT_Lang },
1218     element dirty { w_CT_OnOff }?
1219 w_EG_RubyContent =
1220     element r { w_CT_R }
1221     | w_EG_RunLevelElts*
1222 w_CT_RubyContent = w_EG_RubyContent*
1223 w_CT_Ruby =
1224     element rubyPr { w_CT_RubyPr },
1225     element rt { w_CT_RubyContent },
1226     element rubyBase { w_CT_RubyContent }
1227 w_ST_Lock =
1228     string "sdtLocked"
1229     | string "contentLocked"
1230     | string "unlocked"
1231     | string "sdtContentLocked"
1232 w_CT_Lock = attribute w:val { w_ST_Lock }?
1233 w_CT_SdtListItem =
1234     attribute w:displayText { s_ST_String }?,
1235     attribute w:value { s_ST_String }?
1236 w_ST_SdtDateMappingType =
1237     string "text" | string "date" | string "dateTime"
1238 w_CT_SdtDateMappingType = attribute w:val { w_ST_SdtDateMappingType }?
1239 w_CT_CalendarType = attribute w:val { s_ST_CalendarType }?
1240 w_CT_SdtDate =
1241     attribute w:fullDate { w_ST_DateTime }?,
1242     element dateFormat { w_CT_String }?,
1243     element lid { w_CT_Lang }?,
1244     element storeMappedDataAs { w_CT_SdtDateMappingType }?,
1245     element calendar { w_CT_CalendarType }?
1246 w_CT_SdtComboBox =
1247     attribute w:lastValue { s_ST_String }?,
1248     element listItem { w_CT_SdtListItem }*
1249 w_CT_SdtDocPart =
1250     element docPartGallery { w_CT_String }?,
1251     element docPartCategory { w_CT_String }?,
1252     element docPartUnique { w_CT_OnOff }?

```



```

1253 w_CT_SdtDropDownList =
1254     attribute w:lastValue { s_ST_String }?,
1255     element listItem { w_CT_SdtListItem }*
1256 w_CT_Placeholder = element docPart { w_CT_String }
1257 w_CT_SdtText = attribute w:multiline { s_ST_OnOff }?
1258 w_CT_DataBinding =
1259     attribute w:prefixMappings { s_ST_String }?,
1260     attribute w:xpath { s_ST_String },
1261     attribute w:storeItemID { s_ST_String }
1262 w_CT_SdtPr =
1263     (element rPr { w_CT_RPr }?)
1264     | element alias { w_CT_String }?
1265     | element label { w_CT_DecimalNumber }?
1266     | element tabIndex { w_CT_UnsignedDecimalNumber }?
1267     | element lock { w_CT_Lock }?
1268     | element placeholder { w_CT_Placeholder }?
1269     | element showingPlcHdr { w_CT_OnOff }?
1270     | element dataBinding { w_CT_DataBinding }?
1271     | element temporary { w_CT_OnOff }?
1272     | element id { w_CT_DecimalNumber }?
1273     | element tag { w_CT_String }?
1274     | (element equation { w_CT_Empty }
1275         | element comboBox { w_CT_SdtComboBox }
1276         | element date { w_CT_SdtDate }
1277         | element docPartObj { w_CT_SdtDocPart }
1278         | element docPartList { w_CT_SdtDocPart }
1279         | element dropDownList { w_CT_SdtDropDownList }
1280         | element picture { w_CT_Empty }
1281         | element richText { w_CT_Empty }
1282         | element text { w_CT_SdtText }
1283         | element citation { w_CT_Empty }
1284         | element group { w_CT_Empty }
1285         | element bibliography { w_CT_Empty }?)?)
1286 w_CT_SdtEndPr = (element rPr { w_CT_RPr }?)
1287 w_EG_ContentRunContent =
1288     element customXml { w_CT_CustomXmlRun }
1289     | element smartTag { w_CT_SmartTagRun }
1290     | element sdt { w_CT_SdtRun }
1291     | element dir { w_CT_DirContentRun }
1292     | element bdo { w_CT_BdoContentRun }
1293     | element r { w_CT_R }
1294     | w_EG_RunLevelElts*
1295 w_CT_DirContentRun =
1296     attribute w:val { w_ST_Direction }?,
1297     w_EG_PContent*
1298 w_CT_BdoContentRun =
1299     attribute w:val { w_ST_Direction }?,
1300     w_EG_PContent*
1301 w_ST_Direction = string "ltr" | string "rtl"
1302 w_CT_SdtContentRun = w_EG_PContent*
1303 w_EG_ContentBlockContent =
1304     element customXml { w_CT_CustomXmlBlock }
1305     | element sdt { w_CT_SdtBlock }

```

```

1306 | element p { w_CT_P }*
1307 | element tbl { w_CT_Tbl }*
1308 | w_EG_RunLevelElts*
1309 w_CT_SdtContentBlock = w_EG_ContentBlockContent*
1310 w_EG_ContentRowContent =
1311   element tr { w_CT_Row }*
1312   | element customXml { w_CT_CustomXmlRow }
1313   | element sdt { w_CT_SdtRow }
1314   | w_EG_RunLevelElts*
1315 w_CT_SdtContentRow = w_EG_ContentRowContent*
1316 w_EG_ContentCellContent =
1317   element tc { w_CT_Tc }*
1318   | element customXml { w_CT_CustomXmlCell }
1319   | element sdt { w_CT_SdtCell }
1320   | w_EG_RunLevelElts*
1321 w_CT_SdtContentCell = w_EG_ContentCellContent*
1322 w_CT_SdtBlock =
1323   element sdtPr { w_CT_SdtPr }?,
1324   element sdtEndPr { w_CT_SdtEndPr }?,
1325   element sdtContent { w_CT_SdtContentBlock }?
1326 w_CT_SdtRun =
1327   element sdtPr { w_CT_SdtPr }?,
1328   element sdtEndPr { w_CT_SdtEndPr }?,
1329   element sdtContent { w_CT_SdtContentRun }?
1330 w_CT_SdtCell =
1331   element sdtPr { w_CT_SdtPr }?,
1332   element sdtEndPr { w_CT_SdtEndPr }?,
1333   element sdtContent { w_CT_SdtContentCell }?
1334 w_CT_SdtRow =
1335   element sdtPr { w_CT_SdtPr }?,
1336   element sdtEndPr { w_CT_SdtEndPr }?,
1337   element sdtContent { w_CT_SdtContentRow }?
1338 w_CT_Attr =
1339   attribute w:uri { s_ST_String }?,
1340   attribute w:name { s_ST_String },
1341   attribute w:val { s_ST_String }
1342 w_CT_CustomXmlRun =
1343   attribute w:uri { s_ST_String }?,
1344   attribute w:element { s_ST_String },
1345   element customXmlPr { w_CT_CustomXmlPr }?,
1346   w_EG_PContent*
1347 w_CT_SmartTagRun =
1348   attribute w:uri { s_ST_String }?,
1349   attribute w:element { s_ST_String },
1350   element smartTagPr { w_CT_SmartTagPr }?,
1351   w_EG_PContent*
1352 w_CT_CustomXmlBlock =
1353   attribute w:uri { s_ST_String }?,
1354   attribute w:element { s_ST_String },
1355   element customXmlPr { w_CT_CustomXmlPr }?,
1356   w_EG_ContentBlockContent*
1357 w_CT_CustomXmlPr =
1358   element placeholder { w_CT_String }?,

```

```

1359     element attr { w_CT_Attr }*
1360 w_CT_CustomXmlRow =
1361     attribute w:uri { s_ST_String }?,
1362     attribute w:element { s_ST_String },
1363     element customXmlPr { w_CT_CustomXmlPr }?,
1364     w_EG_ContentRowContent*
1365 w_CT_CustomXmlCell =
1366     attribute w:uri { s_ST_String }?,
1367     attribute w:element { s_ST_String },
1368     element customXmlPr { w_CT_CustomXmlPr }?,
1369     w_EG_ContentCellContent*
1370 w_CT_SmartTagPr = element attr { w_CT_Attr }*
1371 w_EG_PContent =
1372     w_EG_ContentRunContent*
1373     | element fldSimple { w_CT_SimpleField }*
1374     | element hyperlink { w_CT_Hyperlink }
1375     | element subDoc { w_CT_Rel }
1376 w_CT_P =
1377     attribute w:rsidRPr { w_ST_LongHexNumber }?,
1378     attribute w:rsidR { w_ST_LongHexNumber }?,
1379     attribute w:rsidDel { w_ST_LongHexNumber }?,
1380     attribute w:rsidP { w_ST_LongHexNumber }?,
1381     attribute w:rsidRDefault { w_ST_LongHexNumber }?,
1382     element pPr { w_CT_PPr }?,
1383     w_EG_PContent*
1384 w_ST_TblWidth =
1385     string "nil" | string "pct" | string "dxa" | string "auto"
1386 w_CT_Height =
1387     attribute w:val { s_ST_TwipsMeasure }?,
1388     attribute w:hRule { w_ST_HeightRule }?
1389 w_CT_TblWidth =
1390     attribute w:w { w_ST_DecimalNumberOrPercent }?,
1391     attribute w:type { w_ST_TblWidth }?
1392 w_CT_TblGridCol = attribute w:w { s_ST_TwipsMeasure }?
1393 w_CT_TblGridBase = element gridCol { w_CT_TblGridCol }*
1394 w_CT_TblGrid =
1395     w_CT_TblGridBase,
1396     element tblGridChange { w_CT_TblGridChange }?
1397 w_CT_TcBorders =
1398     element top { w_CT_Border }?,
1399     element start { w_CT_Border }?,
1400     element left { w_CT_Border }?,
1401     element bottom { w_CT_Border }?,
1402     element end { w_CT_Border }?,
1403     element right { w_CT_Border }?,
1404     element insideH { w_CT_Border }?,
1405     element insideV { w_CT_Border }?,
1406     element tl2br { w_CT_Border }?,
1407     element tr2bl { w_CT_Border }?
1408 w_CT_TcMar =
1409     element top { w_CT_TblWidth }?,
1410     element start { w_CT_TblWidth }?,
1411     element left { w_CT_TblWidth }?,

```

```

1412     element bottom { w_CT_TblWidth }?,
1413     element end { w_CT_TblWidth }?,
1414     element right { w_CT_TblWidth }?
1415 w_ST_Merge = string "continue" | string "restart"
1416 w_CT_VMerge = attribute w:val { w_ST_Merge }?
1417 w_CT_HMerge = attribute w:val { w_ST_Merge }?
1418 w_CT_TcPrBase =
1419     element cnfStyle { w_CT_Cnf }?,
1420     element tcW { w_CT_TblWidth }?,
1421     element gridSpan { w_CT_DecimalNumber }?,
1422     element hMerge { w_CT_HMerge }?,
1423     element vMerge { w_CT_VMerge }?,
1424     element tcBorders { w_CT_TcBorders }?,
1425     element shd { w_CT_Shdc }?,
1426     element nowrap { w_CT_OnOff }?,
1427     element tcMar { w_CT_TcMar }?,
1428     element textDirection { w_CT_TextDirection }?,
1429     element tcFitText { w_CT_OnOff }?,
1430     element vAlign { w_CT_VerticalJc }?,
1431     element hideMark { w_CT_OnOff }?,
1432     element headers { w_CT_Headers }?
1433 w_CT_TcPr =
1434     w_CT_TcPrInner,
1435     element tcPrChange { w_CT_TcPrChange }?
1436 w_CT_TcPrInner = w_CT_TcPrBase, w_EG_CellMarkupElements?
1437 w_CT_Tc =
1438     attribute w:id { s_ST_String }?,
1439     element tcPr { w_CT_TcPr }?,
1440     w_EG_BlockLevelElts+
1441 w_ST_Cnf = xsd:string { length = "12" pattern = "[01]*" }
1442 w_CT_Cnf =
1443     attribute w:val { w_ST_Cnf }?,
1444     attribute w:firstRow { s_ST_OnOff }?,
1445     attribute w:lastRow { s_ST_OnOff }?,
1446     attribute w:firstColumn { s_ST_OnOff }?,
1447     attribute w:lastColumn { s_ST_OnOff }?,
1448     attribute w:oddVBand { s_ST_OnOff }?,
1449     attribute w:evenVBand { s_ST_OnOff }?,
1450     attribute w:oddHBand { s_ST_OnOff }?,
1451     attribute w:evenHBand { s_ST_OnOff }?,
1452     attribute w:firstRowFirstColumn { s_ST_OnOff }?,
1453     attribute w:firstRowLastColumn { s_ST_OnOff }?,
1454     attribute w:lastRowFirstColumn { s_ST_OnOff }?,
1455     attribute w:lastRowLastColumn { s_ST_OnOff }?
1456 w_CT_Headers = element header { w_CT_String }*
1457 w_CT_TrPrBase =
1458     (element cnfStyle { w_CT_Cnf }?
1459     | element divId { w_CT_DecimalNumber }?
1460     | element gridBefore { w_CT_DecimalNumber }?
1461     | element gridAfter { w_CT_DecimalNumber }?
1462     | element wBefore { w_CT_TblWidth }?
1463     | element wAfter { w_CT_TblWidth }?
1464     | element cantSplit { w_CT_OnOff }?)

```

```

1465 | element trHeight { w_CT_Height }?
1466 | element tblHeader { w_CT_OnOff }?
1467 | element tblCellSpacing { w_CT_TblWidth }?
1468 | element jc { w_CT_JcTable }?
1469 | element hidden { w_CT_OnOff }?)+
1470 w_CT_TrPr =
1471   w_CT_TrPrBase,
1472   element ins { w_CT_TrackChange }?,
1473   element del { w_CT_TrackChange }?,
1474   element trPrChange { w_CT_TrPrChange }?
1475 w_CT_Row =
1476   attribute w:rsidRPr { w_ST_LongHexNumber }?,
1477   attribute w:rsidR { w_ST_LongHexNumber }?,
1478   attribute w:rsidDel { w_ST_LongHexNumber }?,
1479   attribute w:rsidTr { w_ST_LongHexNumber }?,
1480   element tblPrEx { w_CT_TblPrEx }?,
1481   element trPr { w_CT_TrPr }?,
1482   w_EG_ContentCellContent*
1483 w_ST_TblLayoutType = string "fixed" | string "autofit"
1484 w_CT_TblLayoutType = attribute w:type { w_ST_TblLayoutType }?
1485 w_ST_TblOverlap = string "never" | string "overlap"
1486 w_CT_TblOverlap = attribute w:val { w_ST_TblOverlap }
1487 w_CT_TblPPr =
1488   attribute w:leftFromText { s_ST_TwipsMeasure }?,
1489   attribute w:rightFromText { s_ST_TwipsMeasure }?,
1490   attribute w:topFromText { s_ST_TwipsMeasure }?,
1491   attribute w:bottomFromText { s_ST_TwipsMeasure }?,
1492   attribute w:vertAnchor { w_ST_VAnchor }?,
1493   attribute w:horzAnchor { w_ST_HAnchor }?,
1494   attribute w:tblpXSpec { s_ST_XAlign }?,
1495   attribute w:tblpX { w_ST_SignedTwipsMeasure }?,
1496   attribute w:tblpYSpec { s_ST_YAlign }?,
1497   attribute w:tblpY { w_ST_SignedTwipsMeasure }?
1498 w_CT_TblCellMar =
1499   element top { w_CT_TblWidth }?,
1500   element start { w_CT_TblWidth }?,
1501   element left { w_CT_TblWidth }?,
1502   element bottom { w_CT_TblWidth }?,
1503   element end { w_CT_TblWidth }?,
1504   element right { w_CT_TblWidth }?
1505 w_CT_TblBorders =
1506   element top { w_CT_Border }?,
1507   element start { w_CT_Border }?,
1508   element left { w_CT_Border }?,
1509   element bottom { w_CT_Border }?,
1510   element end { w_CT_Border }?,
1511   element right { w_CT_Border }?,
1512   element insideH { w_CT_Border }?,
1513   element insideV { w_CT_Border }?
1514 w_CT_TblPrBase =
1515   element tblStyle { w_CT_String }?,
1516   element tblpPr { w_CT_TblPPr }?,
1517   element tblOverlap { w_CT_TblOverlap }?,

```

```

1518     element bidiVisual { w_CT_OnOff }?,
1519     element tblStyleRowBandSize { w_CT_DecimalNumber }?,
1520     element tblStyleColBandSize { w_CT_DecimalNumber }?,
1521     element tblW { w_CT_TblWidth }?,
1522     element jc { w_CT_JcTable }?,
1523     element tblCellSpacing { w_CT_TblWidth }?,
1524     element tblInd { w_CT_TblWidth }?,
1525     element tblBorders { w_CT_TblBorders }?,
1526     element shd { w_CT_Shadow }?,
1527     element tblLayout { w_CT_TblLayoutType }?,
1528     element tblCellMar { w_CT_TblCellMar }?,
1529     element tblLook { w_CT_TblLook }?,
1530     element tblCaption { w_CT_String }?,
1531     element tblDescription { w_CT_String }?
1532 w_CT_TblPr =
1533     w_CT_TblPrBase,
1534     element tblPrChange { w_CT_TblPrChange }?
1535 w_CT_TblPrExBase =
1536     element tblW { w_CT_TblWidth }?,
1537     element jc { w_CT_JcTable }?,
1538     element tblCellSpacing { w_CT_TblWidth }?,
1539     element tblInd { w_CT_TblWidth }?,
1540     element tblBorders { w_CT_TblBorders }?,
1541     element shd { w_CT_Shadow }?,
1542     element tblLayout { w_CT_TblLayoutType }?,
1543     element tblCellMar { w_CT_TblCellMar }?,
1544     element tblLook { w_CT_TblLook }?
1545 w_CT_TblPrEx =
1546     w_CT_TblPrExBase,
1547     element tblPrExChange { w_CT_TblPrExChange }?
1548 w_CT_Tbl =
1549     w_EG_RangeMarkupElements*,
1550     element tblPr { w_CT_TblPr },
1551     element tblGrid { w_CT_TblGrid },
1552     w_EG_ContentRowContent*
1553 w_CT_TblLook =
1554     attribute w:firstRow { s_ST_OnOff }?,
1555     attribute w:lastRow { s_ST_OnOff }?,
1556     attribute w:firstColumn { s_ST_OnOff }?,
1557     attribute w:lastColumn { s_ST_OnOff }?,
1558     attribute w:noHBand { s_ST_OnOff }?,
1559     attribute w:noVBand { s_ST_OnOff }?,
1560     attribute w:val { w_ST_ShortHexNumber }?
1561 w_ST_FtnPos =
1562     string "pageBottom"
1563     | string "beneathText"
1564     | string "sectEnd"
1565     | string "docEnd"
1566 w_CT_FtnPos = attribute w:val { w_ST_FtnPos }
1567 w_ST_EdnPos = string "sectEnd" | string "docEnd"
1568 w_CT_EdnPos = attribute w:val { w_ST_EdnPos }
1569 w_CT_NumFmt =
1570     attribute w:val { w_ST_NumberFormat },

```

```

1571     attribute w:format { s_ST_String }?
1572 w_ST_RestartNumber =
1573     string "continuous" | string "eachSect" | string "eachPage"
1574 w_CT_NumRestart = attribute w:val { w_ST_RestartNumber }
1575 w_CT_FtnEdnRef =
1576     attribute w:customMarkFollows { s_ST_OnOff }?,
1577     attribute w:id { w_ST_DecimalNumber }
1578 w_CT_FtnEdnSepRef = attribute w:id { w_ST_DecimalNumber }
1579 w_CT_FtnEdn =
1580     attribute w:type { w_ST_FtnEdn }?,
1581     attribute w:id { w_ST_DecimalNumber },
1582     w_EG_BlockLevelElts+
1583 w_EG_FtnEdnNumProps =
1584     element numStart { w_CT_DecimalNumber }?,
1585     element numRestart { w_CT_NumRestart }?
1586 w_CT_FtnProps =
1587     element pos { w_CT_FtnPos }?,
1588     element numFmt { w_CT_NumFmt }?,
1589     w_EG_FtnEdnNumProps?
1590 w_CT_EdnProps =
1591     element pos { w_CT_EdnPos }?,
1592     element numFmt { w_CT_NumFmt }?,
1593     w_EG_FtnEdnNumProps?
1594 w_CT_FtnDocProps =
1595     w_CT_FtnProps,
1596     element footnote { w_CT_FtnEdnSepRef }*
1597 w_CT_EdnDocProps =
1598     w_CT_EdnProps,
1599     element endnote { w_CT_FtnEdnSepRef }*
1600 w_CT_RecipientData =
1601     element active { w_CT_OnOff }?,
1602     element column { w_CT_DecimalNumber },
1603     element uniqueTag { xsd:base64Binary }
1604 w_CT_Recipients = element recipientData { w_CT_RecipientData }+
1605 w_recipients = element recipients { w_CT_Recipients }
1606 w_CT_OdsoFieldMapData =
1607     element type { w_CT_MailMergeOdsoFMDFieldType }?,
1608     element name { w_CT_String }?,
1609     element mappedName { w_CT_String }?,
1610     element column { w_CT_DecimalNumber }?,
1611     element lid { w_CT_Lang }?,
1612     element dynamicAddress { w_CT_OnOff }?
1613 w_ST_MailMergeSourceType =
1614     string "database"
1615     | string "addressBook"
1616     | string "document1"
1617     | string "document2"
1618     | string "text"
1619     | string "email"
1620     | string "native"
1621     | string "legacy"
1622     | string "master"
1623 w_CT_MailMergeSourceType = attribute w:val { w_ST_MailMergeSourceType }

```

```

1624 w_CT_Odso =
1625     element udl { w_CT_String }?,
1626     element table { w_CT_String }?,
1627     element src { w_CT_Rel }?,
1628     element colDelim { w_CT_DecimalNumber }?,
1629     element type { w_CT_MailMergeSourceType }?,
1630     element fHdr { w_CT_OnOff }?,
1631     element fieldMapData { w_CT_OdsoFieldMapData }*,
1632     element recipientData { w_CT_Rel }*
1633 w_CT_MailMerge =
1634     element mainDocumentType { w_CT_MailMergeDocType },
1635     element linkToQuery { w_CT_OnOff }?,
1636     element dataType { w_CT_MailMergeDataType },
1637     element connectString { w_CT_String }?,
1638     element query { w_CT_String }?,
1639     element dataSource { w_CT_Rel }?,
1640     element headerSource { w_CT_Rel }?,
1641     element doNotSuppressBlankLines { w_CT_OnOff }?,
1642     element destination { w_CT_MailMergeDest }?,
1643     element addressFieldName { w_CT_String }?,
1644     element mailSubject { w_CT_String }?,
1645     element mailAsAttachment { w_CT_OnOff }?,
1646     element viewMergedData { w_CT_OnOff }?,
1647     element activeRecord { w_CT_DecimalNumber }?,
1648     element checkErrors { w_CT_DecimalNumber }?,
1649     element odso { w_CT_Odso }?
1650 w_ST_TargetScreenSz =
1651     string "544x376"
1652     | string "640x480"
1653     | string "720x512"
1654     | string "800x600"
1655     | string "1024x768"
1656     | string "1152x882"
1657     | string "1152x900"
1658     | string "1280x1024"
1659     | string "1600x1200"
1660     | string "1800x1440"
1661     | string "1920x1200"
1662 w_CT_TargetScreenSz = attribute w:val { w_ST_TargetScreenSz }
1663 w_CT_Compat =
1664     element useSingleBorderforContiguousCells { w_CT_OnOff }?,
1665     element wpJustification { w_CT_OnOff }?,
1666     element noTabHangInd { w_CT_OnOff }?,
1667     element noLeading { w_CT_OnOff }?,
1668     element spaceForUL { w_CT_OnOff }?,
1669     element noColumnBalance { w_CT_OnOff }?,
1670     element balanceSingleByteDoubleByteWidth { w_CT_OnOff }?,
1671     element noExtraLineSpacing { w_CT_OnOff }?,
1672     element doNotLeaveBackslashAlone { w_CT_OnOff }?,
1673     element ulTrailSpace { w_CT_OnOff }?,
1674     element doNotExpandShiftReturn { w_CT_OnOff }?,
1675     element spacingInWholePoints { w_CT_OnOff }?,
1676     element lineWrapLikeWord6 { w_CT_OnOff }?,

```



```

1677 element printBodyTextBeforeHeader { w_CT_OnOff }?,
1678 element printColBlack { w_CT_OnOff }?,
1679 element wpSpaceWidth { w_CT_OnOff }?,
1680 element showBreaksInFrames { w_CT_OnOff }?,
1681 element subFontBySize { w_CT_OnOff }?,
1682 element suppressBottomSpacing { w_CT_OnOff }?,
1683 element suppressTopSpacing { w_CT_OnOff }?,
1684 element suppressSpacingAtTopOfPage { w_CT_OnOff }?,
1685 element suppressTopSpacingWP { w_CT_OnOff }?,
1686 element suppressSpBfAfterPgBrk { w_CT_OnOff }?,
1687 element swapBordersFacingPages { w_CT_OnOff }?,
1688 element convMailMergeEsc { w_CT_OnOff }?,
1689 element truncateFontHeightsLikeWP6 { w_CT_OnOff }?,
1690 element mwSmallCaps { w_CT_OnOff }?,
1691 element usePrinterMetrics { w_CT_OnOff }?,
1692 element doNotSuppressParagraphBorders { w_CT_OnOff }?,
1693 element wrapTrailSpaces { w_CT_OnOff }?,
1694 element footnoteLayoutLikeWW8 { w_CT_OnOff }?,
1695 element shapeLayoutLikeWW8 { w_CT_OnOff }?,
1696 element alignTablesRowByRow { w_CT_OnOff }?,
1697 element forgetLastTabAlignment { w_CT_OnOff }?,
1698 element adjustLineHeightInTable { w_CT_OnOff }?,
1699 element autoSpaceLikeWord95 { w_CT_OnOff }?,
1700 element noSpaceRaiseLower { w_CT_OnOff }?,
1701 element doNotUseHTMLParagraphAutoSpacing { w_CT_OnOff }?,
1702 element layoutRawTableWidth { w_CT_OnOff }?,
1703 element layoutTableRowsApart { w_CT_OnOff }?,
1704 element useWord97LineBreakRules { w_CT_OnOff }?,
1705 element doNotBreakWrappedTables { w_CT_OnOff }?,
1706 element doNotSnapToGridInCell { w_CT_OnOff }?,
1707 element selectFldWithFirstOrLastChar { w_CT_OnOff }?,
1708 element applyBreakingRules { w_CT_OnOff }?,
1709 element doNotWrapTextWithPunct { w_CT_OnOff }?,
1710 element doNotUseEastAsianBreakRules { w_CT_OnOff }?,
1711 element useWord2002TableStyleRules { w_CT_OnOff }?,
1712 element growAutofit { w_CT_OnOff }?,
1713 element useFELayout { w_CT_OnOff }?,
1714 element useNormalStyleForList { w_CT_OnOff }?,
1715 element doNotUseIndentAsNumberingTabStop { w_CT_OnOff }?,
1716 element useAltKinsokuLineBreakRules { w_CT_OnOff }?,
1717 element allowSpaceOfSameStyleInTable { w_CT_OnOff }?,
1718 element doNotSuppressIndentation { w_CT_OnOff }?,
1719 element doNotAutofitConstrainedTables { w_CT_OnOff }?,
1720 element autofitToFirstFixedWidthCell { w_CT_OnOff }?,
1721 element underlineTabInNumList { w_CT_OnOff }?,
1722 element displayHangulFixedWidth { w_CT_OnOff }?,
1723 element splitPgBreakAndParaMark { w_CT_OnOff }?,
1724 element doNotVertAlignCellWithSp { w_CT_OnOff }?,
1725 element doNotBreakConstrainedForcedTable { w_CT_OnOff }?,
1726 element doNotVertAlignInTxbx { w_CT_OnOff }?,
1727 element useAnsiKerningPairs { w_CT_OnOff }?,
1728 element cachedColBalance { w_CT_OnOff }?,
1729 element compatSetting { w_CT_CompatSetting }*

```

```

1730 w_CT_CompatSetting =
1731     attribute w:name { s_ST_String }?,
1732     attribute w:uri { s_ST_String }?,
1733     attribute w:val { s_ST_String }?
1734 w_CT_DocVar =
1735     attribute w:name { s_ST_String },
1736     attribute w:val { s_ST_String }
1737 w_CT_DocVars = element docVar { w_CT_DocVar }*
1738 w_CT_DocRsids =
1739     element rsidRoot { w_CT_LongHexNumber }?,
1740     element rsid { w_CT_LongHexNumber }*
1741 w_ST_CharacterSpacing =
1742     string "doNotCompress"
1743     | string "compressPunctuation"
1744     | string "compressPunctuationAndJapaneseKana"
1745 w_CT_CharacterSpacing = attribute w:val { w_ST_CharacterSpacing }
1746 w_CT_SaveThroughXslt =
1747     r_id?,
1748     attribute w:solutionID { s_ST_String }?
1749 w_CT_RPrDefault = element rPr { w_CT_RPr }?
1750 w_CT_PPrDefault = element pPr { w_CT_PPrGeneral }?
1751 w_CT_DocDefaults =
1752     element rPrDefault { w_CT_RPrDefault }?,
1753     element pPrDefault { w_CT_PPrDefault }?
1754 w_ST_WmlColorSchemeIndex =
1755     string "dark1"
1756     | string "light1"
1757     | string "dark2"
1758     | string "light2"
1759     | string "accent1"
1760     | string "accent2"
1761     | string "accent3"
1762     | string "accent4"
1763     | string "accent5"
1764     | string "accent6"
1765     | string "hyperlink"
1766     | string "followedHyperlink"
1767 w_CT_ColorSchemeMapping =
1768     attribute w:bg1 { w_ST_WmlColorSchemeIndex }?,
1769     attribute w:t1 { w_ST_WmlColorSchemeIndex }?,
1770     attribute w:bg2 { w_ST_WmlColorSchemeIndex }?,
1771     attribute w:t2 { w_ST_WmlColorSchemeIndex }?,
1772     attribute w:accent1 { w_ST_WmlColorSchemeIndex }?,
1773     attribute w:accent2 { w_ST_WmlColorSchemeIndex }?,
1774     attribute w:accent3 { w_ST_WmlColorSchemeIndex }?,
1775     attribute w:accent4 { w_ST_WmlColorSchemeIndex }?,
1776     attribute w:accent5 { w_ST_WmlColorSchemeIndex }?,
1777     attribute w:accent6 { w_ST_WmlColorSchemeIndex }?,
1778     attribute w:hyperlink { w_ST_WmlColorSchemeIndex }?,
1779     attribute w:followedHyperlink { w_ST_WmlColorSchemeIndex }?
1780 w_CT_ReadingModeInkLockDown =
1781     attribute w:actualPg { s_ST_OnOff },
1782     attribute w:w { w_ST_PixelsMeasure },

```

```

1783     attribute w:h { w_ST_PixelsMeasure },
1784     attribute w:fontSz { w_ST_DecimalNumberOrPercent }
1785 w_CT_WriteProtection =
1786     attribute w:recommended { s_ST_OnOff }?,
1787     w_AG_Password,
1788     w_AG_TransitionalPassword
1789 w_CT_Settings =
1790     element writeProtection { w_CT_WriteProtection }?,
1791     element view { w_CT_View }?,
1792     element zoom { w_CT_Zoom }?,
1793     element removePersonalInformation { w_CT_OnOff }?,
1794     element removeDateAndTime { w_CT_OnOff }?,
1795     element doNotDisplayPageBoundaries { w_CT_OnOff }?,
1796     element displayBackgroundShape { w_CT_OnOff }?,
1797     element printPostScriptOverText { w_CT_OnOff }?,
1798     element printFractionalCharacterWidth { w_CT_OnOff }?,
1799     element printFormsData { w_CT_OnOff }?,
1800     element embedTrueTypeFonts { w_CT_OnOff }?,
1801     element embedSystemFonts { w_CT_OnOff }?,
1802     element saveSubsetFonts { w_CT_OnOff }?,
1803     element saveFormsData { w_CT_OnOff }?,
1804     element mirrorMargins { w_CT_OnOff }?,
1805     element alignBordersAndEdges { w_CT_OnOff }?,
1806     element bordersDoNotSurroundHeader { w_CT_OnOff }?,
1807     element bordersDoNotSurroundFooter { w_CT_OnOff }?,
1808     element gutterAtTop { w_CT_OnOff }?,
1809     element hideSpellingErrors { w_CT_OnOff }?,
1810     element hideGrammaticalErrors { w_CT_OnOff }?,
1811     element activeWritingStyle { w_CT_WritingStyle }*,
1812     element proofState { w_CT_Proof }?,
1813     element formsDesign { w_CT_OnOff }?,
1814     element attachedTemplate { w_CT_Rel }?,
1815     element linkStyles { w_CT_OnOff }?,
1816     element stylePaneFormatFilter { w_CT_StylePaneFilter }?,
1817     element stylePaneSortMethod { w_CT_StyleSort }?,
1818     element documentType { w_CT_DocType }?,
1819     element mailMerge { w_CT_MailMerge }?,
1820     element revisionView { w_CT_TrackChangesView }?,
1821     element trackRevisions { w_CT_OnOff }?,
1822     element doNotTrackMoves { w_CT_OnOff }?,
1823     element doNotTrackFormatting { w_CT_OnOff }?,
1824     element documentProtection { w_CT_DocProtect }?,
1825     element autoFormatOverride { w_CT_OnOff }?,
1826     element styleLockTheme { w_CT_OnOff }?,
1827     element styleLockQFSet { w_CT_OnOff }?,
1828     element defaultTabStop { w_CT_TwipsMeasure }?,
1829     element autoHyphenation { w_CT_OnOff }?,
1830     element consecutiveHyphenLimit { w_CT_DecimalNumber }?,
1831     element hyphenationZone { w_CT_TwipsMeasure }?,
1832     element doNotHyphenateCaps { w_CT_OnOff }?,
1833     element showEnvelope { w_CT_OnOff }?,
1834     element summaryLength { w_CT_DecimalNumberOrPrecent }?,
1835     element clickAndTypeStyle { w_CT_String }?,

```

```

1836 element defaultTableStyle { w_CT_String }?,
1837 element evenAndOddHeaders { w_CT_OnOff }?,
1838 element bookFoldRevPrinting { w_CT_OnOff }?,
1839 element bookFoldPrinting { w_CT_OnOff }?,
1840 element bookFoldPrintingSheets { w_CT_DecimalNumber }?,
1841 element drawingGridHorizontalSpacing { w_CT_TwipsMeasure }?,
1842 element drawingGridVerticalSpacing { w_CT_TwipsMeasure }?,
1843 element displayHorizontalDrawingGridEvery { w_CT_DecimalNumber }?,
1844 element displayVerticalDrawingGridEvery { w_CT_DecimalNumber }?,
1845 element doNotUseMarginsForDrawingGridOrigin { w_CT_OnOff }?,
1846 element drawingGridHorizontalOrigin { w_CT_TwipsMeasure }?,
1847 element drawingGridVerticalOrigin { w_CT_TwipsMeasure }?,
1848 element doNotShadeFormData { w_CT_OnOff }?,
1849 element noPunctuationKerning { w_CT_OnOff }?,
1850 element characterSpacingControl { w_CT_CharacterSpacing }?,
1851 element printTwoOnOne { w_CT_OnOff }?,
1852 element strictFirstAndLastChars { w_CT_OnOff }?,
1853 element noLineBreaksAfter { w_CT_Kinsoku }?,
1854 element noLineBreaksBefore { w_CT_Kinsoku }?,
1855 element savePreviewPicture { w_CT_OnOff }?,
1856 element doNotValidateAgainstSchema { w_CT_OnOff }?,
1857 element saveInvalidXml { w_CT_OnOff }?,
1858 element ignoreMixedContent { w_CT_OnOff }?,
1859 element alwaysShowPlaceholderText { w_CT_OnOff }?,
1860 element doNotDemarcateInvalidXml { w_CT_OnOff }?,
1861 element saveXmlDataOnly { w_CT_OnOff }?,
1862 element useXSLTWhenSaving { w_CT_OnOff }?,
1863 element saveThroughXslt { w_CT_SaveThroughXslt }?,
1864 element showXMLTags { w_CT_OnOff }?,
1865 element alwaysMergeEmptyNamespace { w_CT_OnOff }?,
1866 element updateFields { w_CT_OnOff }?,
1867 element hdrShapeDefaults { w_CT_ShapeDefaults }?,
1868 element footnotePr { w_CT_FtnDocProps }?,
1869 element endnotePr { w_CT_EdnDocProps }?,
1870 element compat { w_CT_Compat }?,
1871 element docVars { w_CT_DocVars }?,
1872 element rsids { w_CT_DocRsids }?,
1873 m_mathPr?,
1874 element attachedSchema { w_CT_String }*,
1875 element themeFontLang { w_CT_Language }?,
1876 element clrSchemeMapping { w_CT_ColorSchemeMapping }?,
1877 element doNotIncludeSubdocsInStats { w_CT_OnOff }?,
1878 element doNotAutoCompressPictures { w_CT_OnOff }?,
1879 element forceUpgrade { w_CT_Empty }?,
1880 element captions { w_CT_Captions }?,
1881 element readModeInkLockDown { w_CT_ReadingModeInkLockDown }?,
1882 element smartTagType { w_CT_SmartTagType }*,
1883 sl_schemaLibrary?,
1884 element shapeDefaults { w_CT_ShapeDefaults }?,
1885 element doNotEmbedSmartTags { w_CT_OnOff }?,
1886 element decimalSymbol { w_CT_String }?,
1887 element listSeparator { w_CT_String }?
1888 w_CT_StyleSort = attribute w:val { w_ST_StyleSort }

```

```

1889 w_CT_StylePaneFilter =
1890     attribute w:allStyles { s_ST_OnOff }?,
1891     attribute w:customStyles { s_ST_OnOff }?,
1892     attribute w:latentStyles { s_ST_OnOff }?,
1893     attribute w:stylesInUse { s_ST_OnOff }?,
1894     attribute w:headingStyles { s_ST_OnOff }?,
1895     attribute w:numberingStyles { s_ST_OnOff }?,
1896     attribute w:tableStyles { s_ST_OnOff }?,
1897     attribute w:directFormattingOnRuns { s_ST_OnOff }?,
1898     attribute w:directFormattingOnParagraphs { s_ST_OnOff }?,
1899     attribute w:directFormattingOnNumbering { s_ST_OnOff }?,
1900     attribute w:directFormattingOnTables { s_ST_OnOff }?,
1901     attribute w:clearFormatting { s_ST_OnOff }?,
1902     attribute w:top3HeadingStyles { s_ST_OnOff }?,
1903     attribute w:visibleStyles { s_ST_OnOff }?,
1904     attribute w:alternateStyleNames { s_ST_OnOff }?,
1905     attribute w:val { w_ST_ShortHexNumber }?
1906 w_ST_StyleSort =
1907     string "name"
1908     | string "priority"
1909     | string "default"
1910     | string "font"
1911     | string "basedOn"
1912     | string "type"
1913     | string "0000"
1914     | string "0001"
1915     | string "0002"
1916     | string "0003"
1917     | string "0004"
1918     | string "0005"
1919 w_CT_WebSettings =
1920     element frameset { w_CT_Frameset }?,
1921     element divs { w_CT_Divs }?,
1922     element encoding { w_CT_String }?,
1923     element optimizeForBrowser { w_CT_OptimizeForBrowser }?,
1924     element relyOnVML { w_CT_OnOff }?,
1925     element allowPNG { w_CT_OnOff }?,
1926     element doNotRelyOnCSS { w_CT_OnOff }?,
1927     element doNotSaveAsSingleFile { w_CT_OnOff }?,
1928     element doNotOrganizeInFolder { w_CT_OnOff }?,
1929     element doNotUseLongFileNames { w_CT_OnOff }?,
1930     element pixelsPerInch { w_CT_DecimalNumber }?,
1931     element targetScreenSz { w_CT_TargetScreenSz }?,
1932     element saveSmartTagsAsXml { w_CT_OnOff }?
1933 w_ST_FrameScrollbar = string "on" | string "off" | string "auto"
1934 w_CT_FrameScrollbar = attribute w:val { w_ST_FrameScrollbar }
1935 w_CT_OptimizeForBrowser =
1936     w_CT_OnOff,
1937     attribute w:target { s_ST_String }?
1938 w_CT_Frame =
1939     element sz { w_CT_String }?,
1940     element name { w_CT_String }?,
1941     element title { w_CT_String }?,

```

```

1942     element longDesc { w_CT_Rel }?,
1943     element sourceFileName { w_CT_Rel }?,
1944     element marW { w_CT_PixelsMeasure }?,
1945     element marH { w_CT_PixelsMeasure }?,
1946     element scrollbar { w_CT_FrameScrollbar }?,
1947     element noResizeAllowed { w_CT_OnOff }?,
1948     element linkedToFile { w_CT_OnOff }?
1949 w_ST_FrameLayout = string "rows" | string "cols" | string "none"
1950 w_CT_FrameLayout = attribute w:val { w_ST_FrameLayout }
1951 w_CT_FramesetSplitbar =
1952     element w { w_CT_TwipsMeasure }?,
1953     element color { w_CT_Color }?,
1954     element noBorder { w_CT_OnOff }?,
1955     element flatBorders { w_CT_OnOff }?
1956 w_CT_Frameset =
1957     element sz { w_CT_String }?,
1958     element framesetSplitbar { w_CT_FramesetSplitbar }?,
1959     element frameLayout { w_CT_FrameLayout }?,
1960     element title { w_CT_String }?,
1961     (element frameset { w_CT_Frameset }*
1962      | element frame { w_CT_Frame }*)*
1963 w_CT_NumPicBullet =
1964     attribute w:numPicBulletId { w_ST_DecimalNumber },
1965     (element pict { w_CT_Picture }
1966      | element drawing { w_CT_Drawing })
1967 w_ST_LevelSuffix = string "tab" | string "space" | string "nothing"
1968 w_CT_LevelSuffix = attribute w:val { w_ST_LevelSuffix }
1969 w_CT_LevelText =
1970     attribute w:val { s_ST_String }?,
1971     attribute w:null { s_ST_OnOff }?
1972 w_CT_LvlLegacy =
1973     attribute w:legacy { s_ST_OnOff }?,
1974     attribute w:legacySpace { s_ST_TwipsMeasure }?,
1975     attribute w:legacyIndent { w_ST_SignedTwipsMeasure }?
1976 w_CT_Lvl =
1977     attribute w:ilvl { w_ST_DecimalNumber },
1978     attribute w:tplc { w_ST_LongHexNumber }?,
1979     attribute w:tentative { s_ST_OnOff }?,
1980     element start { w_CT_DecimalNumber }?,
1981     element numFmt { w_CT_NumFmt }?,
1982     element lvlRestart { w_CT_DecimalNumber }?,
1983     element pStyle { w_CT_String }?,
1984     element isLgl { w_CT_OnOff }?,
1985     element suff { w_CT_LevelSuffix }?,
1986     element lvlText { w_CT_LevelText }?,
1987     element lvlPicBulletId { w_CT_DecimalNumber }?,
1988     element legacy { w_CT_LvlLegacy }?,
1989     element lvlJc { w_CT_Jc }?,
1990     element pPr { w_CT_PPrGeneral }?,
1991     element rPr { w_CT_RPr }?
1992 w_ST_MultiLevelType =
1993     string "singleLevel" | string "multilevel" | string "hybridMultilevel"
1994 w_CT_MultiLevelType = attribute w:val { w_ST_MultiLevelType }

```

```

1995 w_CT_AbstractNum =
1996     attribute w:abstractNumId { w_ST_DecimalNumber },
1997     element nsid { w_CT_LongHexNumber }?,
1998     element multilevelType { w_CT_MultilevelType }?,
1999     element tpl { w_CT_LongHexNumber }?,
2000     element name { w_CT_String }?,
2001     element styleLink { w_CT_String }?,
2002     element numStyleLink { w_CT_String }?,
2003     element lvl { w_CT_Lvl }*
2004 w_CT_NumLvl =
2005     attribute w:ilvl { w_ST_DecimalNumber },
2006     element startOverride { w_CT_DecimalNumber }?,
2007     element lvl { w_CT_Lvl }?
2008 w_CT_Num =
2009     attribute w:numId { w_ST_DecimalNumber },
2010     element abstractNumId { w_CT_DecimalNumber },
2011     element lvlOverride { w_CT_NumLvl }*
2012 w_CT_Numbering =
2013     element numPicBullet { w_CT_NumPicBullet }*,
2014     element abstractNum { w_CT_AbstractNum }*,
2015     element num { w_CT_Num }*,
2016     element numIdMacAtCleanup { w_CT_DecimalNumber }?
2017 w_ST_TblStyleOverrideType =
2018     string "wholeTable"
2019     | string "firstRow"
2020     | string "lastRow"
2021     | string "firstCol"
2022     | string "lastCol"
2023     | string "band1Vert"
2024     | string "band2Vert"
2025     | string "band1Horz"
2026     | string "band2Horz"
2027     | string "neCell"
2028     | string "nwCell"
2029     | string "seCell"
2030     | string "swCell"
2031 w_CT_TblStylePr =
2032     attribute w:type { w_ST_TblStyleOverrideType },
2033     element pPr { w_CT_PPrGeneral }?,
2034     element rPr { w_CT_RPr }?,
2035     element tblPr { w_CT_TblPrBase }?,
2036     element trPr { w_CT_TrPr }?,
2037     element tcPr { w_CT_TcPr }?
2038 w_ST_StyleType =
2039     string "paragraph"
2040     | string "character"
2041     | string "table"
2042     | string "numbering"
2043 w_CT_Style =
2044     attribute w:type { w_ST_StyleType }?,
2045     attribute w:styleId { s_ST_String }?,
2046     attribute w:default { s_ST_OnOff }?,
2047     attribute w:customStyle { s_ST_OnOff }?,

```

```

2048     element name { w_CT_String }?,
2049     element aliases { w_CT_String }?,
2050     element basedOn { w_CT_String }?,
2051     element next { w_CT_String }?,
2052     element link { w_CT_String }?,
2053     element autoRedefine { w_CT_OnOff }?,
2054     element hidden { w_CT_OnOff }?,
2055     element uiPriority { w_CT_DecimalNumber }?,
2056     element semiHidden { w_CT_OnOff }?,
2057     element unhideWhenUsed { w_CT_OnOff }?,
2058     element qFormat { w_CT_OnOff }?,
2059     element locked { w_CT_OnOff }?,
2060     element personal { w_CT_OnOff }?,
2061     element personalCompose { w_CT_OnOff }?,
2062     element personalReply { w_CT_OnOff }?,
2063     element rsid { w_CT_LongHexNumber }?,
2064     element pPr { w_CT_PPrGeneral }?,
2065     element rPr { w_CT_RPr }?,
2066     element tblPr { w_CT_TblPrBase }?,
2067     element trPr { w_CT_TrPr }?,
2068     element tcPr { w_CT_TcPr }?,
2069     element tblStylePr { w_CT_TblStylePr }*
2070 w_CT_LsdException =
2071     attribute w:name { s_ST_String },
2072     attribute w:locked { s_ST_OnOff }?,
2073     attribute w:uiPriority { w_ST_DecimalNumber }?,
2074     attribute w:semiHidden { s_ST_OnOff }?,
2075     attribute w:unhideWhenUsed { s_ST_OnOff }?,
2076     attribute w:qFormat { s_ST_OnOff }?
2077 w_CT_LatentStyles =
2078     attribute w:defLockedState { s_ST_OnOff }?,
2079     attribute w:defUIPriority { w_ST_DecimalNumber }?,
2080     attribute w:defSemiHidden { s_ST_OnOff }?,
2081     attribute w:defUnhideWhenUsed { s_ST_OnOff }?,
2082     attribute w:defQFormat { s_ST_OnOff }?,
2083     attribute w:count { w_ST_DecimalNumber }?,
2084     element lsdException { w_CT_LsdException }*
2085 w_CT_Styles =
2086     element docDefaults { w_CT_DocDefaults }?,
2087     element latentStyles { w_CT_LatentStyles }?,
2088     element style { w_CT_Style }*
2089 w_CT_Panose = attribute w:val { s_ST_Panose }
2090 w_ST_FontFamily =
2091     string "decorative"
2092     | string "modern"
2093     | string "roman"
2094     | string "script"
2095     | string "swiss"
2096     | string "auto"
2097 w_CT_FontFamily = attribute w:val { w_ST_FontFamily }
2098 w_ST_Pitch = string "fixed" | string "variable" | string "default"
2099 w_CT_Pitch = attribute w:val { w_ST_Pitch }
2100 w_CT_FontSig =

```



```

2101 attribute w:usb0 { w_ST_LongHexNumber },
2102 attribute w:usb1 { w_ST_LongHexNumber },
2103 attribute w:usb2 { w_ST_LongHexNumber },
2104 attribute w:usb3 { w_ST_LongHexNumber },
2105 attribute w:csb0 { w_ST_LongHexNumber },
2106 attribute w:csb1 { w_ST_LongHexNumber }
2107 w_CT_FontRel =
2108   w_CT_Rel,
2109   attribute w:fontKey { s_ST_Guid }?,
2110   attribute w:subsetting { s_ST_OnOff }?
2111 w_CT_Font =
2112   attribute w:name { s_ST_String },
2113   element altName { w_CT_String }?,
2114   element panose1 { w_CT_Panose }?,
2115   element charset { w_CT_Charset }?,
2116   element family { w_CT_FontFamily }?,
2117   element notTrueType { w_CT_OnOff }?,
2118   element pitch { w_CT_Pitch }?,
2119   element sig { w_CT_FontSig }?,
2120   element embedRegular { w_CT_FontRel }?,
2121   element embedBold { w_CT_FontRel }?,
2122   element embedItalic { w_CT_FontRel }?,
2123   element embedBoldItalic { w_CT_FontRel }?
2124 w_CT_FontsList = element font { w_CT_Font }*
2125 w_CT_DivBdr =
2126   element top { w_CT_Border }?,
2127   element left { w_CT_Border }?,
2128   element bottom { w_CT_Border }?,
2129   element right { w_CT_Border }?
2130 w_CT_Div =
2131   attribute w:id { w_ST_DecimalNumber },
2132   element blockQuote { w_CT_OnOff }?,
2133   element bodyDiv { w_CT_OnOff }?,
2134   element marLeft { w_CT_SignedTwipsMeasure },
2135   element marRight { w_CT_SignedTwipsMeasure },
2136   element marTop { w_CT_SignedTwipsMeasure },
2137   element marBottom { w_CT_SignedTwipsMeasure },
2138   element divBdr { w_CT_DivBdr }?,
2139   element divsChild { w_CT_Divs }*
2140 w_CT_Divs = element div { w_CT_Div }+
2141 w_CT_TxbxContent = w_EG_BlockLevelElts+
2142 w_txbxContent = element txbxContent { w_CT_TxbxContent }
2143 w_EG_MathContent = m_oMathPara | m_oMath
2144 w_EG_BlockLevelChunkElts = w_EG_ContentBlockContent*
2145 w_EG_BlockLevelElts =
2146   w_EG_BlockLevelChunkElts*
2147   | element altChunk { w_CT_AltChunk }*
2148 w_EG_RunLevelElts =
2149   element proofErr { w_CT_ProofErr }?
2150   | element permStart { w_CT_PermStart }?
2151   | element permEnd { w_CT_Perm }?
2152   | w_EG_RangeMarkupElements*
2153   | element ins { w_CT_RunTrackChange }?

```

```

2154 | element del { w_CT_RunTrackChange }?
2155 | element moveFrom { w_CT_RunTrackChange }
2156 | element moveTo { w_CT_RunTrackChange }
2157 | w_EG_MathContent*
2158 w_CT_Body =
2159     w_EG_BlockLevelElts*,
2160     element sectPr { w_CT_SectPr }?
2161 w_CT_ShapeDefaults = (w_any_vml_office*)+
2162 w_CT_Comments = element comment { w_CT_Comment }*
2163 w_comments = element comments { w_CT_Comments }
2164 w_CT_Footnotes = element footnote { w_CT_FtnEdn }*
2165 w_footnotes = element footnotes { w_CT_Footnotes }
2166 w_CT_Endnotes = element endnote { w_CT_FtnEdn }*
2167 w_endnotes = element endnotes { w_CT_Endnotes }
2168 w_hdr = element hdr { w_CT_HdrFtr }
2169 w_ftr = element ftr { w_CT_HdrFtr }
2170 w_CT_SmartTagType =
2171     attribute w:namespaceuri { s_ST_String }?,
2172     attribute w:name { s_ST_String }?,
2173     attribute w:url { s_ST_String }?
2174 w_ST_ThemeColor =
2175     string "dark1"
2176     | string "light1"
2177     | string "dark2"
2178     | string "light2"
2179     | string "accent1"
2180     | string "accent2"
2181     | string "accent3"
2182     | string "accent4"
2183     | string "accent5"
2184     | string "accent6"
2185     | string "hyperlink"
2186     | string "followedHyperlink"
2187     | string "none"
2188     | string "background1"
2189     | string "text1"
2190     | string "background2"
2191     | string "text2"
2192 w_ST_DocPartBehavior = string "content" | string "p" | string "pg"
2193 w_CT_DocPartBehavior = attribute w:val { w_ST_DocPartBehavior }
2194 w_CT_DocPartBehaviors = element behavior { w_CT_DocPartBehavior }+
2195 w_ST_DocPartType =
2196     string "none"
2197     | string "normal"
2198     | string "autoExp"
2199     | string "toolbar"
2200     | string "speller"
2201     | string "formFld"
2202     | string "bbPlcHdr"
2203 w_CT_DocPartType = attribute w:val { w_ST_DocPartType }
2204 w_CT_DocPartTypes =
2205     attribute w:all { s_ST_OnOff }?,
2206     (element type { w_CT_DocPartType }+)

```

```

2207 w_ST_DocPartGallery =
2208     string "placeholder"
2209     | string "any"
2210     | string "default"
2211     | string "docParts"
2212     | string "coverPg"
2213     | string "eq"
2214     | string "ftrs"
2215     | string "hdrs"
2216     | string "pgNum"
2217     | string "tbls"
2218     | string "watermarks"
2219     | string "autoTxt"
2220     | string "txtBox"
2221     | string "pgNumT"
2222     | string "pgNumB"
2223     | string "pgNumMargins"
2224     | string "tblOfContents"
2225     | string "bib"
2226     | string "custQuickParts"
2227     | string "custCoverPg"
2228     | string "custEq"
2229     | string "custFtrs"
2230     | string "custHdrs"
2231     | string "custPgNum"
2232     | string "custTbls"
2233     | string "custWatermarks"
2234     | string "custAutoTxt"
2235     | string "custTxtBox"
2236     | string "custPgNumT"
2237     | string "custPgNumB"
2238     | string "custPgNumMargins"
2239     | string "custTblOfContents"
2240     | string "custBib"
2241     | string "custom1"
2242     | string "custom2"
2243     | string "custom3"
2244     | string "custom4"
2245     | string "custom5"
2246 w_CT_DocPartGallery = attribute w:val { w_ST_DocPartGallery }
2247 w_CT_DocPartCategory =
2248     element name { w_CT_String },
2249     element gallery { w_CT_DocPartGallery }
2250 w_CT_DocPartName =
2251     attribute w:val { s_ST_String },
2252     attribute w:decorated { s_ST_OnOff }?
2253 w_CT_DocPartPr =
2254     (element name { w_CT_DocPartName }
2255     | element style { w_CT_String }
2256     | element category { w_CT_DocPartCategory }
2257     | element types { w_CT_DocPartTypes }
2258     | element behaviors { w_CT_DocPartBehaviors }
2259     | element description { w_CT_String }

```

```

2260 | element guid { w_CT_Guid })+
2261 w_CT_DocPart =
2262   element docPartPr { w_CT_DocPartPr }?,
2263   element docPartBody { w_CT_Body }?
2264 w_CT_DocParts = element docPart { w_CT_DocPart }+
2265 w_settings = element settings { w_CT_Settings }
2266 w_webSettings = element webSettings { w_CT_WebSettings }
2267 w_fonts = element fonts { w_CT_Fontslist }
2268 w_numbering = element numbering { w_CT_Numbering }
2269 w_styles = element styles { w_CT_Styles }
2270 w_ST_CaptionPos =
2271   string "above" | string "below" | string "left" | string "right"
2272 w_CT_Caption =
2273   attribute w:name { s_ST_String },
2274   attribute w:pos { w_ST_CaptionPos }?,
2275   attribute w:chapNum { s_ST_OnOff }?,
2276   attribute w:heading { w_ST_DecimalNumber }?,
2277   attribute w:noLabel { s_ST_OnOff }?,
2278   attribute w:numFmt { w_ST_NumberFormat }?,
2279   attribute w:sep { w_ST_ChapterSep }?
2280 w_CT_AutoCaption =
2281   attribute w:name { s_ST_String },
2282   attribute w:caption { s_ST_String }
2283 w_CT_AutoCaptions = element autoCaption { w_CT_AutoCaption }+
2284 w_CT_Captions =
2285   element caption { w_CT_Caption }+,
2286   element autoCaptions { w_CT_AutoCaptions }?
2287 w_CT_DocumentBase = element background { w_CT_Background }?
2288 w_CT_Document =
2289   w_CT_DocumentBase,
2290   element body { w_CT_Body }?,
2291   attribute w:conformance { s_ST_ConformanceClass }?
2292 w_CT_GlossaryDocument =
2293   w_CT_DocumentBase,
2294   element docParts { w_CT_DocParts }?
2295 w_document = element document { w_CT_Document }
2296 w_glossaryDocument = element glossaryDocument { w_CT_GlossaryDocument }
2297 w_any_vml_office =
2298   o_shapedefaults
2299   | o_shapelayout
2300   | o_signatureline
2301   | o_ink
2302   | o_diagram
2303   | o_skew
2304   | o_extrusion
2305   | o_callout
2306   | o_lock
2307   | o_OLEObject
2308   | o_complex
2309   | o_left
2310   | o_top
2311   | o_right
2312   | o_bottom

```

```

2313 | o_column
2314 | o_clippath
2315 | o_fill
2316 w_any_vml_vml =
2317   v_shape
2318   | v_shapetype
2319   | v_group
2320   | v_background
2321   | v_fill
2322   | v_formulas
2323   | v_handles
2324   | v_imagedata
2325   | v_path
2326   | v_textbox
2327   | v_shadow
2328   | v_stroke
2329   | v_textpath
2330   | v_arc
2331   | v_curve
2332   | v_image
2333   | v_line
2334   | v_oval
2335   | v_polyline
2336   | v_rect
2337   | v_roundrect

```

B.1.1 Part Schemas

B.1.1.1 Comments Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_comments

```

B.1.1.2 Document Settings Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_settings

```

B.1.1.3 Endnotes Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_endnotes

```

B.1.1.4 Font Table Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_fonts

```

B.1.1.5 Footer Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_ftr

```

B.1.1.6 Footnotes Part

```

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 "dml-compatibility.rnc"

```

```

13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_footnotes

```

B.1.1.7 Glossary Document Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_glossaryDocument

```

B.1.1.8 Header Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"

```



```

19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_hdr

```

B.1.1.9 Mail Merge Recipient Data Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_recipients

```

B.1.1.10 Main Document Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_document

```

B.1.1.11 Numbering Definitions Part

```

include "wml.rnc"
include "shared-relationshipReference.rnc"
include "dml-wordprocessingDrawing.rnc"
include "dml-main.rnc"
include "dml-diagram.rnc"
include "shared-commonSimpleTypes.rnc"
include "dml-lockedCanvas.rnc"
include "any.rnc"
include "dml-chart.rnc"
include "dml-chartDrawing.rnc"
include "dml-picture.rnc"
include "dml-compatibility.rnc"
include "vml-presentationDrawing.rnc"
include "xml.rnc"
include "shared-customXmlSchemaProperties.rnc"
include "vml-officeDrawing.rnc"
include "vml-main.rnc"
include "vml-spreadsheetDrawing.rnc"
include "vml-wordprocessingDrawing.rnc"
include "shared-math.rnc"
start = w_numbering

```

B.1.1.12 Style Definitions Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_styles

```

B.1.1.13 Web Settings Part

```

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 "dml-compatibility.rnc"
13 include "vml-presentationDrawing.rnc"
14 include "xml.rnc"
15 include "shared-customXmlSchemaProperties.rnc"
16 include "vml-officeDrawing.rnc"
17 include "vml-main.rnc"
18 include "vml-spreadsheetDrawing.rnc"
19 include "vml-wordprocessingDrawing.rnc"
20 include "shared-math.rnc"
21 start = w_webSettings

```

B.2 SpreadsheetML

```

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 dxId { 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     attribute z-order { xsd:unsignedInt }?,
210     xdr_from,
211     xdr_to
212 sml_EG_ExtensionList = element ext { sml_CT_Extension }*
213 sml_CT_ExtensionList = sml_EG_ExtensionList?
214 sml_calcChain = element calcChain { sml_CT_CalcChain }
215 sml_CT_CalcChain =
216     element c { sml_CT_CalcCell }+,
217     element extList { sml_CT_ExtensionList }?
218 sml_CT_CalcCell =
219     attribute r { sml_ST_CellRef },
220
221     ## default value: 0
222     attribute i { xsd:int }?,
223
224     ## default value: false
225     attribute s { xsd:boolean }?,
226
227     ## default value: false
228     attribute l { xsd:boolean }?,
229
230     ## default value: false
231     attribute t { xsd:boolean }?,
232
233     ## default value: false
234     attribute a { xsd:boolean }?
235 sml_comments = element comments { sml_CT_Comments }
236 sml_CT_Comments =
237     element authors { sml_CT_Authors },
238     element commentList { sml_CT_CommentList },
239     element extList { sml_CT_ExtensionList }?
240 sml_CT_Authors = element author { s_ST_Xstring }*
241 sml_CT_CommentList = element comment { sml_CT_Comment }*
242 sml_CT_Comment =
243     attribute ref { sml_ST_Ref },
244     attribute authorId { xsd:unsignedInt },
245     attribute guid { s_ST_Guid }?,
246     attribute shapeId { xsd:unsignedInt }?,
247     element text { sml_CT_Rst },

```

```

248     element commentPr { sml_CT_CommentPr }?
249 sml_CT_CommentPr =
250
251     ## default value: true
252     attribute locked { xsd:boolean }?,
253
254     ## default value: true
255     attribute defaultSize { xsd:boolean }?,
256
257     ## default value: true
258     attribute print { xsd:boolean }?,
259
260     ## default value: false
261     attribute disabled { xsd:boolean }?,
262
263     ## default value: false
264     attribute uiObject { xsd:boolean }?,
265
266     ## default value: true
267     attribute autoFill { xsd:boolean }?,
268
269     ## default value: true
270     attribute autoLine { xsd:boolean }?,
271     attribute altText { s_ST_Xstring }?,
272
273     ## default value: left
274     attribute textHAlign { sml_ST_TextHAlign }?,
275
276     ## default value: top
277     attribute textVAlign { sml_ST_TextVAlign }?,
278
279     ## default value: true
280     attribute lockText { xsd:boolean }?,
281
282     ## default value: false
283     attribute justLastX { xsd:boolean }?,
284
285     ## default value: false
286     attribute autoScale { xsd:boolean }?,
287
288     ## default value: false
289     attribute rowHidden { xsd:boolean }?,
290
291     ## default value: false
292     attribute colHidden { xsd:boolean }?,
293     element anchor { sml_CT_ObjectAnchor }
294 sml_ST_TextHAlign =
295     string "left"
296     | string "center"
297     | string "right"
298     | string "justify"
299     | string "distributed"
300 sml_ST_TextVAlign =

```



```

301     string "top"
302     | string "center"
303     | string "bottom"
304     | string "justify"
305     | string "distributed"
306 sml_MapInfo = element MapInfo { sml_CT_MapInfo }
307 sml_CT_MapInfo =
308     attribute SelectionNamespaces { xsd:string },
309     element Schema { sml_CT_Schema }+,
310     element Map { sml_CT_Map }+
311 sml_CT_Schema =
312     mixed {
313         attribute ID { xsd:string },
314         attribute SchemaRef { xsd:string }?,
315         attribute Namespace { xsd:string }?,
316         attribute SchemaLanguage { xsd:token }?,
317         sml_CT_Schema_any
318     }
319 sml_CT_Schema_any =
320     element * - (o:* | v:* | w10:* | x:*) {
321         anyAttribute*,
322         mixed { anyElement* }
323     }
324 sml_CT_Map =
325     attribute ID { xsd:unsignedInt },
326     attribute Name { xsd:string },
327     attribute RootElement { xsd:string },
328     attribute SchemaID { xsd:string },
329     attribute ShowImportExportValidationErrors { xsd:boolean },
330     attribute AutoFit { xsd:boolean },
331     attribute Append { xsd:boolean },
332     attribute PreserveSortAFLayout { xsd:boolean },
333     attribute PreserveFormat { xsd:boolean },
334     element DataBinding { sml_CT_DataBinding }?
335 sml_CT_DataBinding =
336     attribute DataBindingName { xsd:string }?,
337     attribute FileBinding { xsd:boolean }?,
338     attribute ConnectionID { xsd:unsignedInt }?,
339     attribute FileBindingName { xsd:string }?,
340     attribute DataBindingLoadMode { xsd:unsignedInt },
341     sml_CT_DataBinding_any
342 sml_CT_DataBinding_any =
343     element * - (o:* | v:* | w10:* | x:*) {
344         anyAttribute*,
345         mixed { anyElement* }
346     }
347 sml_connections = element connections { sml_CT_Connections }
348 sml_CT_Connections = element connection { sml_CT_Connection }+
349 sml_CT_Connection =
350     attribute id { xsd:unsignedInt },
351     attribute sourceFile { s_ST_Xstring }?,
352     attribute odcFile { s_ST_Xstring }?,
353

```

```

354  ## default value: false
355  attribute keepAlive { xsd:boolean }?,
356
357  ## default value: 0
358  attribute interval { xsd:unsignedInt }?,
359  attribute name { s_ST_Xstring }?,
360  attribute description { s_ST_Xstring }?,
361  attribute type { xsd:unsignedInt }?,
362
363  ## default value: 1
364  attribute reconnectionMethod { xsd:unsignedInt }?,
365  attribute refreshedVersion { xsd:unsignedByte },
366
367  ## default value: 0
368  attribute minRefreshableVersion { xsd:unsignedByte }?,
369
370  ## default value: false
371  attribute savePassword { xsd:boolean }?,
372
373  ## default value: false
374  attribute new { xsd:boolean }?,
375
376  ## default value: false
377  attribute deleted { xsd:boolean }?,
378
379  ## default value: false
380  attribute onlyUseConnectionFile { xsd:boolean }?,
381
382  ## default value: false
383  attribute background { xsd:boolean }?,
384
385  ## default value: false
386  attribute refreshOnLoad { xsd:boolean }?,
387
388  ## default value: false
389  attribute saveData { xsd:boolean }?,
390
391  ## default value: integrated
392  attribute credentials { sml_ST_CredMethod }?,
393  attribute singleSignOnId { s_ST_Xstring }?,
394  element dbPr { sml_CT_DbPr }?,
395  element olapPr { sml_CT_OlapPr }?,
396  element webPr { sml_CT_WebPr }?,
397  element textPr { sml_CT_TextPr }?,
398  element parameters { sml_CT_Parameters }?,
399  element extLst { sml_CT_ExtensionList }?
400  sml_ST_CredMethod =
401    string "integrated"
402    | string "none"
403    | string "stored"
404    | string "prompt"
405  sml_CT_DbPr =
406    attribute connection { s_ST_Xstring },

```

```

407 attribute command { s_ST_Xstring }?,
408 attribute serverCommand { s_ST_Xstring }?,
409
410 ## default value: 2
411 attribute commandType { xsd:unsignedInt }?
412 sml_CT_OlapPr =
413
414 ## default value: false
415 attribute local { xsd:boolean }?,
416 attribute localConnection { s_ST_Xstring }?,
417
418 ## default value: true
419 attribute localRefresh { xsd:boolean }?,
420
421 ## default value: false
422 attribute sendLocale { xsd:boolean }?,
423 attribute rowDrillCount { xsd:unsignedInt }?,
424
425 ## default value: true
426 attribute serverFill { xsd:boolean }?,
427
428 ## default value: true
429 attribute serverNumberFormat { xsd:boolean }?,
430
431 ## default value: true
432 attribute serverFont { xsd:boolean }?,
433
434 ## default value: true
435 attribute serverFontColor { xsd:boolean }?
436 sml_CT_WebPr =
437
438 ## default value: false
439 attribute xml { xsd:boolean }?,
440
441 ## default value: false
442 attribute sourceData { xsd:boolean }?,
443
444 ## default value: false
445 attribute parsePre { xsd:boolean }?,
446
447 ## default value: false
448 attribute consecutive { xsd:boolean }?,
449
450 ## default value: false
451 attribute firstRow { xsd:boolean }?,
452
453 ## default value: false
454 attribute xl97 { xsd:boolean }?,
455
456 ## default value: false
457 attribute textDates { xsd:boolean }?,
458
459 ## default value: false

```

```

460 attribute xl2000 { xsd:boolean }?,
461 attribute url { s_ST_Xstring }?,
462 attribute post { s_ST_Xstring }?,
463
464 ## default value: false
465 attribute htmlTables { xsd:boolean }?,
466
467 ## default value: none
468 attribute htmlFormat { sml_ST_HtmlFmt }?,
469 attribute editPage { s_ST_Xstring }?,
470 element tables { sml_CT_Tables }?
471 sml_ST_HtmlFmt = string "none" | string "rtf" | string "all"
472 sml_CT_Parameters =
473   attribute count { xsd:unsignedInt }?,
474   element parameter { sml_CT_Parameter }+
475 sml_CT_Parameter =
476   attribute name { s_ST_Xstring }?,
477
478 ## default value: 0
479 attribute sqlType { xsd:int }?,
480
481 ## default value: prompt
482 attribute parameterType { sml_ST_ParameterType }?,
483
484 ## default value: false
485 attribute refreshOnChange { xsd:boolean }?,
486 attribute prompt { s_ST_Xstring }?,
487 attribute boolean { xsd:boolean }?,
488 attribute double { xsd:double }?,
489 attribute integer { xsd:int }?,
490 attribute string { s_ST_Xstring }?,
491 attribute cell { s_ST_Xstring }?
492 sml_ST_ParameterType = string "prompt" | string "value" | string "cell"
493 sml_CT_Tables =
494   attribute count { xsd:unsignedInt }?,
495   (element m { sml_CT_TableMissing }
496    | element s { sml_CT_XStringElement }
497    | element x { sml_CT_Index })+
498 sml_CT_TableMissing = empty
499 sml_CT_TextPr =
500
501 ## default value: true
502 attribute prompt { xsd:boolean }?,
503
504 ## default value: win
505 attribute fileType { sml_ST_FileType }?,
506
507 ## default value: 1252
508 attribute codePage { xsd:unsignedInt }?,
509 attribute characterSet { xsd:string }?,
510
511 ## default value: 1
512 attribute firstRow { xsd:unsignedInt }?,

```

```

513 attribute sourceFile { s_ST_Xstring }?,
514
515 ## default value: true
516 attribute delimited { xsd:boolean }?,
517
518 ## default value: .
519 attribute decimal { s_ST_Xstring }?,
520
521 ## default value: ,
522 attribute thousands { s_ST_Xstring }?,
523
524 ## default value: true
525 attribute tab { xsd:boolean }?,
526
527 ## default value: false
528 attribute space { xsd:boolean }?,
529
530 ## default value: false
531 attribute comma { xsd:boolean }?,
532
533 ## default value: false
534 attribute semicolon { xsd:boolean }?,
535
536 ## default value: false
537 attribute consecutive { xsd:boolean }?,
538
539 ## default value: doubleQuote
540 attribute qualifier { sml_ST_Qualifier }?,
541 attribute delimiter { s_ST_Xstring }?,
542 element textFields { sml_CT_TextFields }?
543 sml_ST_FileType =
544     string "mac"
545     | string "win"
546     | string "dos"
547     | string "lin"
548     | string "other"
549 sml_ST_Qualifier =
550     string "doubleQuote" | string "singleQuote" | string "none"
551 sml_CT_TextFields =
552
553 ## default value: 1
554 attribute count { xsd:unsignedInt }?,
555 element textField { sml_CT_TextField }+
556 sml_CT_TextField =
557
558 ## default value: general
559 attribute type { sml_ST_ExternalConnectionType }?,
560
561 ## default value: 0
562 attribute position { xsd:unsignedInt }?
563 sml_ST_ExternalConnectionType =
564     string "general"
565     | string "text"

```

```

566 | string "MDY"
567 | string "DMY"
568 | string "YMD"
569 | string "MYD"
570 | string "DYM"
571 | string "YDM"
572 | string "skip"
573 | string "EMD"
574 sml_pivotCacheDefinition =
575     element pivotCacheDefinition { sml_CT_PivotCacheDefinition }
576 sml_pivotCacheRecords =
577     element pivotCacheRecords { sml_CT_PivotCacheRecords }
578 sml_pivotTableDefinition =
579     element pivotTableDefinition { sml_CT_pivotTableDefinition }
580 sml_CT_PivotCacheDefinition =
581     r_id?,
582
583     ## default value: false
584     attribute invalid { xsd:boolean }?,
585
586     ## default value: true
587     attribute saveData { xsd:boolean }?,
588
589     ## default value: false
590     attribute refreshOnLoad { xsd:boolean }?,
591
592     ## default value: false
593     attribute optimizeMemory { xsd:boolean }?,
594
595     ## default value: true
596     attribute enableRefresh { xsd:boolean }?,
597     attribute refreshedBy { s_ST_Xstring }?,
598     attribute refreshedDate { xsd:double }?,
599     attribute refreshedDateIso { xsd:dateTime }?,
600
601     ## default value: false
602     attribute backgroundQuery { xsd:boolean }?,
603     attribute missingItemsLimit { xsd:unsignedInt }?,
604
605     ## default value: 0
606     attribute createdVersion { xsd:unsignedByte }?,
607
608     ## default value: 0
609     attribute refreshedVersion { xsd:unsignedByte }?,
610
611     ## default value: 0
612     attribute minRefreshableVersion { xsd:unsignedByte }?,
613     attribute recordCount { xsd:unsignedInt }?,
614
615     ## default value: false
616     attribute upgradeOnRefresh { xsd:boolean }?,
617
618     ## default value: false

```

```

619 attribute tupleCache { xsd:boolean }?,
620
621 ## default value: false
622 attribute supportSubquery { xsd:boolean }?,
623
624 ## default value: false
625 attribute supportAdvancedDrill { xsd:boolean }?,
626 element cacheSource { sml_CT_CacheSource },
627 element cacheFields { sml_CT_CacheFields },
628 element cacheHierarchies { sml_CT_CacheHierarchies }?,
629 element kpis { sml_CT_PCDKPIs }?,
630 element tupleCache { sml_CT_TupleCache }?,
631 element calculatedItems { sml_CT_CalculatedItems }?,
632 element calculatedMembers { sml_CT_CalculatedMembers }?,
633 element dimensions { sml_CT_Dimensions }?,
634 element measureGroups { sml_CT_MeasureGroups }?,
635 element maps { sml_CT_MeasureDimensionMaps }?,
636 element extLst { sml_CT_ExtensionList }?
637 sml_CT_CacheFields =
638   attribute count { xsd:unsignedInt }?,
639   element cacheField { sml_CT_CacheField }*
640 sml_CT_CacheField =
641   attribute name { s_ST_Xstring },
642   attribute caption { s_ST_Xstring }?,
643   attribute propertyName { s_ST_Xstring }?,
644
645   ## default value: false
646   attribute serverField { xsd:boolean }?,
647
648   ## default value: true
649   attribute uniqueList { xsd:boolean }?,
650   attribute numFmtId { sml_ST_NumFmtId }?,
651   attribute formula { s_ST_Xstring }?,
652
653   ## default value: 0
654   attribute sqlType { xsd:int }?,
655
656   ## default value: 0
657   attribute hierarchy { xsd:int }?,
658
659   ## default value: 0
660   attribute level { xsd:unsignedInt }?,
661
662   ## default value: true
663   attribute databaseField { xsd:boolean }?,
664   attribute mappingCount { xsd:unsignedInt }?,
665
666   ## default value: false
667   attribute memberPropertyField { xsd:boolean }?,
668   element sharedItems { sml_CT_SharedItems }?,
669   element fieldGroup { sml_CT_FieldGroup }?,
670   element mpMap { sml_CT_X }*,
671   element extLst { sml_CT_ExtensionList }?

```

```

672 sml_CT_CacheSource =
673     attribute type { sml_ST_SourceType },
674
675     ## default value: 0
676     attribute connectionId { xsd:unsignedInt }?,
677     (element worksheetSource { sml_CT_WorksheetSource }
678      | element consolidation { sml_CT_Consolidation }
679      | element extLst { sml_CT_ExtensionList }?)?
680 sml_ST_SourceType =
681     string "worksheet"
682     | string "external"
683     | string "consolidation"
684     | string "scenario"
685 sml_CT_WorksheetSource =
686     attribute ref { sml_ST_Ref }?,
687     attribute name { s_ST_Xstring }?,
688     attribute sheet { s_ST_Xstring }?,
689     r_id?
690 sml_CT_Consolidation =
691
692     ## default value: true
693     attribute autoPage { xsd:boolean }?,
694     element pages { sml_CT_Pages }?,
695     element rangeSets { sml_CT_RangeSets }
696 sml_CT_Pages =
697     attribute count { xsd:unsignedInt }?,
698     element page { sml_CT_PCDSCPage }+
699 sml_CT_PCDSCPage =
700     attribute count { xsd:unsignedInt }?,
701     element pageItem { sml_CT_PageItem }*
702 sml_CT_PageItem = attribute name { s_ST_Xstring }
703 sml_CT_RangeSets =
704     attribute count { xsd:unsignedInt }?,
705     element rangeSet { sml_CT_RangeSet }+
706 sml_CT_RangeSet =
707     attribute i1 { xsd:unsignedInt }?,
708     attribute i2 { xsd:unsignedInt }?,
709     attribute i3 { xsd:unsignedInt }?,
710     attribute i4 { xsd:unsignedInt }?,
711     attribute ref { sml_ST_Ref }?,
712     attribute name { s_ST_Xstring }?,
713     attribute sheet { s_ST_Xstring }?,
714     r_id?
715 sml_CT_SharedItems =
716
717     ## default value: true
718     attribute containsSemiMixedTypes { xsd:boolean }?,
719
720     ## default value: true
721     attribute containsNonDate { xsd:boolean }?,
722
723     ## default value: false
724     attribute containsDate { xsd:boolean }?,

```



```

725
726 ## default value: true
727 attribute containsString { xsd:boolean }?,
728
729 ## default value: false
730 attribute containsBlank { xsd:boolean }?,
731
732 ## default value: false
733 attribute containsMixedTypes { xsd:boolean }?,
734
735 ## default value: false
736 attribute containsNumber { xsd:boolean }?,
737
738 ## default value: false
739 attribute containsInteger { xsd:boolean }?,
740 attribute minValue { xsd:double }?,
741 attribute maxValue { xsd:double }?,
742 attribute minDate { xsd:dateTime }?,
743 attribute maxDate { xsd:dateTime }?,
744 attribute count { xsd:unsignedInt }?,
745
746 ## default value: false
747 attribute longText { xsd:boolean }?,
748 (element m { sml_CT_Missing }
749   | element n { sml_CT_Number }
750   | element b { sml_CT_Boolean }
751   | element e { sml_CT_Error }
752   | element s { sml_CT_String }
753   | element d { sml_CT_DateTime })*
754 sml_CT_Missing =
755   attribute u { xsd:boolean }?,
756   attribute f { xsd:boolean }?,
757   attribute c { s_ST_Xstring }?,
758   attribute cp { xsd:unsignedInt }?,
759   attribute in { xsd:unsignedInt }?,
760   attribute bc { sml_ST_UnsignedIntHex }?,
761   attribute fc { sml_ST_UnsignedIntHex }?,
762
763 ## default value: false
764 attribute i { xsd:boolean }?,
765
766 ## default value: false
767 attribute un { xsd:boolean }?,
768
769 ## default value: false
770 attribute st { xsd:boolean }?,
771
772 ## default value: false
773 attribute b { xsd:boolean }?,
774 element tpls { sml_CT_Tuples }*,
775 element x { sml_CT_X }*
776 sml_CT_Number =
777   attribute v { xsd:double },

```

```

778     attribute u { xsd:boolean }?,
779     attribute f { xsd:boolean }?,
780     attribute c { s_ST_Xstring }?,
781     attribute cp { xsd:unsignedInt }?,
782     attribute in { xsd:unsignedInt }?,
783     attribute bc { sml_ST_UnsignedIntHex }?,
784     attribute fc { sml_ST_UnsignedIntHex }?,
785
786     ## default value: false
787     attribute i { xsd:boolean }?,
788
789     ## default value: false
790     attribute un { xsd:boolean }?,
791
792     ## default value: false
793     attribute st { xsd:boolean }?,
794
795     ## default value: false
796     attribute b { xsd:boolean }?,
797     element tpls { sml_CT_Tuples }*,
798     element x { sml_CT_X }*
799 sml_CT_Boolean =
800     attribute v { xsd:boolean },
801     attribute u { xsd:boolean }?,
802     attribute f { xsd:boolean }?,
803     attribute c { s_ST_Xstring }?,
804     attribute cp { xsd:unsignedInt }?,
805     element x { sml_CT_X }*
806 sml_CT_Error =
807     attribute v { s_ST_Xstring },
808     attribute u { xsd:boolean }?,
809     attribute f { xsd:boolean }?,
810     attribute c { s_ST_Xstring }?,
811     attribute cp { xsd:unsignedInt }?,
812     attribute in { xsd:unsignedInt }?,
813     attribute bc { sml_ST_UnsignedIntHex }?,
814     attribute fc { sml_ST_UnsignedIntHex }?,
815
816     ## default value: false
817     attribute i { xsd:boolean }?,
818
819     ## default value: false
820     attribute un { xsd:boolean }?,
821
822     ## default value: false
823     attribute st { xsd:boolean }?,
824
825     ## default value: false
826     attribute b { xsd:boolean }?,
827     element tpls { sml_CT_Tuples }?,
828     element x { sml_CT_X }*
829 sml_CT_String =
830     attribute v { s_ST_Xstring },

```

```

831 attribute u { xsd:boolean }?,
832 attribute f { xsd:boolean }?,
833 attribute c { s_ST_Xstring }?,
834 attribute cp { xsd:unsignedInt }?,
835 attribute in { xsd:unsignedInt }?,
836 attribute bc { sml_ST_UnsignedIntHex }?,
837 attribute fc { sml_ST_UnsignedIntHex }?,
838
839 ## default value: false
840 attribute i { xsd:boolean }?,
841
842 ## default value: false
843 attribute un { xsd:boolean }?,
844
845 ## default value: false
846 attribute st { xsd:boolean }?,
847
848 ## default value: false
849 attribute b { xsd:boolean }?,
850 element tpls { sml_CT_Tuples }*,
851 element x { sml_CT_X }*
852 sml_CT_DateTime =
853 attribute v { xsd:dateTime },
854 attribute u { xsd:boolean }?,
855 attribute f { xsd:boolean }?,
856 attribute c { s_ST_Xstring }?,
857 attribute cp { xsd:unsignedInt }?,
858 element x { sml_CT_X }*
859 sml_CT_FieldGroup =
860 attribute par { xsd:unsignedInt }?,
861 attribute base { xsd:unsignedInt }?,
862 element rangePr { sml_CT_RangePr }?,
863 element discretePr { sml_CT_DiscretePr }?,
864 element groupItems { sml_CT_GroupItems }?
865 sml_CT_RangePr =
866
867 ## default value: true
868 attribute autoStart { xsd:boolean }?,
869
870 ## default value: true
871 attribute autoEnd { xsd:boolean }?,
872
873 ## default value: range
874 attribute groupBy { sml_ST_GroupBy }?,
875 attribute startNum { xsd:double }?,
876 attribute endNum { xsd:double }?,
877 attribute startDate { xsd:dateTime }?,
878 attribute endDate { xsd:dateTime }?,
879
880 ## default value: 1
881 attribute groupInterval { xsd:double }?
882 sml_ST_GroupBy =
883 string "range"

```

```

884 | string "seconds"
885 | string "minutes"
886 | string "hours"
887 | string "days"
888 | string "months"
889 | string "quarters"
890 | string "years"
891 sml_CT_DiscretePr =
892   attribute count { xsd:unsignedInt }?,
893   element x { sml_CT_Index }+
894 sml_CT_GroupItems =
895   attribute count { xsd:unsignedInt }?,
896   (element m { sml_CT_Missing }
897     | element n { sml_CT_Number }
898     | element b { sml_CT_Boolean }
899     | element e { sml_CT_Error }
900     | element s { sml_CT_String }
901     | element d { sml_CT_DateTime })+
902 sml_CT_PivotCacheRecords =
903   attribute count { xsd:unsignedInt }?,
904   element r { sml_CT_Record }*,
905   element extLst { sml_CT_ExtensionList }?
906 sml_CT_Record =
907   (element m { sml_CT_Missing }
908     | element n { sml_CT_Number }
909     | element b { sml_CT_Boolean }
910     | element e { sml_CT_Error }
911     | element s { sml_CT_String }
912     | element d { sml_CT_DateTime }
913     | element x { sml_CT_Index })+
914 sml_CT_PCDKPIs =
915   attribute count { xsd:unsignedInt }?,
916   element kpi { sml_CT_PCDKPI }*
917 sml_CT_PCDKPI =
918   attribute uniqueName { s_ST_Xstring },
919   attribute caption { s_ST_Xstring }?,
920   attribute displayFolder { s_ST_Xstring }?,
921   attribute measureGroup { s_ST_Xstring }?,
922   attribute parent { s_ST_Xstring }?,
923   attribute value { s_ST_Xstring },
924   attribute goal { s_ST_Xstring }?,
925   attribute status { s_ST_Xstring }?,
926   attribute trend { s_ST_Xstring }?,
927   attribute weight { s_ST_Xstring }?,
928   attribute time { s_ST_Xstring }?
929 sml_CT_CacheHierarchies =
930   attribute count { xsd:unsignedInt }?,
931   element cacheHierarchy { sml_CT_CacheHierarchy }*
932 sml_CT_CacheHierarchy =
933   attribute uniqueName { s_ST_Xstring },
934   attribute caption { s_ST_Xstring }?,
935
936   ## default value: false

```

```

937 attribute measure { xsd:boolean }?,
938
939 ## default value: false
940 attribute set { xsd:boolean }?,
941 attribute parentSet { xsd:unsignedInt }?,
942
943 ## default value: 0
944 attribute iconSet { xsd:int }?,
945
946 ## default value: false
947 attribute attribute { xsd:boolean }?,
948
949 ## default value: false
950 attribute time { xsd:boolean }?,
951
952 ## default value: false
953 attribute keyAttribute { xsd:boolean }?,
954 attribute defaultMemberUniqueName { s_ST_Xstring }?,
955 attribute allUniqueName { s_ST_Xstring }?,
956 attribute allCaption { s_ST_Xstring }?,
957 attribute dimensionUniqueName { s_ST_Xstring }?,
958 attribute displayFolder { s_ST_Xstring }?,
959 attribute measureGroup { s_ST_Xstring }?,
960
961 ## default value: false
962 attribute measures { xsd:boolean }?,
963 attribute count { xsd:unsignedInt },
964
965 ## default value: false
966 attribute oneField { xsd:boolean }?,
967 attribute memberValueDatatype { xsd:unsignedShort }?,
968 attribute unbalanced { xsd:boolean }?,
969 attribute unbalancedGroup { xsd:boolean }?,
970
971 ## default value: false
972 attribute hidden { xsd:boolean }?,
973 element fieldsUsage { sml_CT_FieldsUsage }?,
974 element groupLevels { sml_CT_GroupLevels }?,
975 element extLst { sml_CT_ExtensionList }?
976 sml_CT_FieldsUsage =
977     attribute count { xsd:unsignedInt }?,
978     element fieldUsage { sml_CT_FieldUsage }*
979 sml_CT_FieldUsage = attribute x { xsd:int }
980 sml_CT_GroupLevels =
981     attribute count { xsd:unsignedInt }?,
982     element groupLevel { sml_CT_GroupLevel }+
983 sml_CT_GroupLevel =
984     attribute uniqueName { s_ST_Xstring },
985     attribute caption { s_ST_Xstring },
986
987 ## default value: false
988 attribute user { xsd:boolean }?,
989

```

```

990  ## default value: false
991  attribute customRollUp { xsd:boolean }?,
992  element groups { sml_CT_Groups }?,
993  element extLst { sml_CT_ExtensionList }?
994 sml_CT_Groups =
995   attribute count { xsd:unsignedInt }?,
996   element group { sml_CT_LevelGroup }+
997 sml_CT_LevelGroup =
998   attribute name { s_ST_Xstring },
999   attribute uniqueName { s_ST_Xstring },
1000  attribute caption { s_ST_Xstring },
1001  attribute uniqueParent { s_ST_Xstring }?,
1002  attribute id { xsd:int }?,
1003  element groupMembers { sml_CT_GroupMembers }
1004 sml_CT_GroupMembers =
1005   attribute count { xsd:unsignedInt }?,
1006   element groupMember { sml_CT_GroupMember }+
1007 sml_CT_GroupMember =
1008   attribute uniqueName { s_ST_Xstring },
1009
1010  ## default value: false
1011  attribute group { xsd:boolean }?
1012 sml_CT_TupleCache =
1013   element entries { sml_CT_PCDSDTCEntries }?,
1014   element sets { sml_CT_Sets }?,
1015   element queryCache { sml_CT_QueryCache }?,
1016   element serverFormats { sml_CT_ServerFormats }?,
1017   element extLst { sml_CT_ExtensionList }?
1018 sml_CT_ServerFormat =
1019   attribute culture { s_ST_Xstring }?,
1020   attribute format { s_ST_Xstring }?
1021 sml_CT_ServerFormats =
1022   attribute count { xsd:unsignedInt }?,
1023   element serverFormat { sml_CT_ServerFormat }*
1024 sml_CT_PCDSDTCEntries =
1025   attribute count { xsd:unsignedInt }?,
1026   (element m { sml_CT_Missing }
1027    | element n { sml_CT_Number }
1028    | element e { sml_CT_Error }
1029    | element s { sml_CT_String })+
1030 sml_CT_Tuples =
1031   attribute c { xsd:unsignedInt }?,
1032   element tpl { sml_CT_Tuple }+
1033 sml_CT_Tuple =
1034   attribute fld { xsd:unsignedInt }?,
1035   attribute hier { xsd:unsignedInt }?,
1036   attribute item { xsd:unsignedInt }
1037 sml_CT_Sets =
1038   attribute count { xsd:unsignedInt }?,
1039   element set { sml_CT_Set }+
1040 sml_CT_Set =
1041   attribute count { xsd:unsignedInt }?,
1042   attribute maxRank { xsd:int },

```

```

1043 attribute setDefinition { s_ST_Xstring },
1044
1045 ## default value: none
1046 attribute sortType { sml_ST_SortType }?,
1047
1048 ## default value: false
1049 attribute queryFailed { xsd:boolean }?,
1050 element tpls { sml_CT_Tuples }*,
1051 element sortByTuple { sml_CT_Tuples }?
1052 sml_ST_SortType =
1053     string "none"
1054     | string "ascending"
1055     | string "descending"
1056     | string "ascendingAlpha"
1057     | string "descendingAlpha"
1058     | string "ascendingNatural"
1059     | string "descendingNatural"
1060 sml_CT_QueryCache =
1061     attribute count { xsd:unsignedInt }?,
1062     element query { sml_CT_Query }+
1063 sml_CT_Query =
1064     attribute mdx { s_ST_Xstring },
1065     element tpls { sml_CT_Tuples }?
1066 sml_CT_CalculatedItems =
1067     attribute count { xsd:unsignedInt }?,
1068     element calculatedItem { sml_CT_CalculatedItem }+
1069 sml_CT_CalculatedItem =
1070     attribute field { xsd:unsignedInt }?,
1071     attribute formula { s_ST_Xstring }?,
1072     element pivotArea { sml_CT_PivotArea },
1073     element extLst { sml_CT_ExtensionList }?
1074 sml_CT_CalculatedMembers =
1075     attribute count { xsd:unsignedInt }?,
1076     element calculatedMember { sml_CT_CalculatedMember }+
1077 sml_CT_CalculatedMember =
1078     attribute name { s_ST_Xstring },
1079     attribute mdx { s_ST_Xstring },
1080     attribute memberName { s_ST_Xstring }?,
1081     attribute hierarchy { s_ST_Xstring }?,
1082     attribute parent { s_ST_Xstring }?,
1083
1084 ## default value: 0
1085 attribute solveOrder { xsd:int }?,
1086
1087 ## default value: false
1088 attribute set { xsd:boolean }?,
1089 element extLst { sml_CT_ExtensionList }?
1090 sml_CT_pivotTableDefinition =
1091     attribute name { s_ST_Xstring },
1092     attribute cacheId { xsd:unsignedInt },
1093
1094 ## default value: false
1095 attribute dataOnRows { xsd:boolean }?,

```

```

1096 attribute dataPosition { xsd:unsignedInt }?,
1097 sml_AG_AutoFormat,
1098 attribute dataCaption { s_ST_Xstring },
1099 attribute grandTotalCaption { s_ST_Xstring }?,
1100 attribute errorCaption { s_ST_Xstring }?,
1101
1102 ## default value: false
1103 attribute showError { xsd:boolean }?,
1104 attribute missingCaption { s_ST_Xstring }?,
1105
1106 ## default value: true
1107 attribute showMissing { xsd:boolean }?,
1108 attribute pageStyle { s_ST_Xstring }?,
1109 attribute pivotTableStyle { s_ST_Xstring }?,
1110 attribute vacatedStyle { s_ST_Xstring }?,
1111 attribute tag { s_ST_Xstring }?,
1112
1113 ## default value: 0
1114 attribute updatedVersion { xsd:unsignedByte }?,
1115
1116 ## default value: 0
1117 attribute minRefreshableVersion { xsd:unsignedByte }?,
1118
1119 ## default value: false
1120 attribute asteriskTotals { xsd:boolean }?,
1121
1122 ## default value: true
1123 attribute showItems { xsd:boolean }?,
1124
1125 ## default value: false
1126 attribute editData { xsd:boolean }?,
1127
1128 ## default value: false
1129 attribute disableFieldList { xsd:boolean }?,
1130
1131 ## default value: true
1132 attribute showCalcMbrs { xsd:boolean }?,
1133
1134 ## default value: true
1135 attribute visualTotals { xsd:boolean }?,
1136
1137 ## default value: true
1138 attribute showMultipleLabel { xsd:boolean }?,
1139
1140 ## default value: true
1141 attribute showDataDropDown { xsd:boolean }?,
1142
1143 ## default value: true
1144 attribute showDrill { xsd:boolean }?,
1145
1146 ## default value: false
1147 attribute printDrill { xsd:boolean }?,
1148

```



```

1149 ## default value: true
1150 attribute showMemberPropertyTips { xsd:boolean }?,
1151
1152 ## default value: true
1153 attribute showDataTips { xsd:boolean }?,
1154
1155 ## default value: true
1156 attribute enableWizard { xsd:boolean }?,
1157
1158 ## default value: true
1159 attribute enableDrill { xsd:boolean }?,
1160
1161 ## default value: true
1162 attribute enableFieldProperties { xsd:boolean }?,
1163
1164 ## default value: true
1165 attribute preserveFormatting { xsd:boolean }?,
1166
1167 ## default value: false
1168 attribute useAutoFormatting { xsd:boolean }?,
1169
1170 ## default value: 0
1171 attribute pageWrap { xsd:unsignedInt }?,
1172
1173 ## default value: false
1174 attribute pageOverThenDown { xsd:boolean }?,
1175
1176 ## default value: false
1177 attribute subtotalHiddenItems { xsd:boolean }?,
1178
1179 ## default value: true
1180 attribute rowGrandTotals { xsd:boolean }?,
1181
1182 ## default value: true
1183 attribute colGrandTotals { xsd:boolean }?,
1184
1185 ## default value: false
1186 attribute fieldPrintTitles { xsd:boolean }?,
1187
1188 ## default value: false
1189 attribute itemPrintTitles { xsd:boolean }?,
1190
1191 ## default value: false
1192 attribute mergeItem { xsd:boolean }?,
1193
1194 ## default value: true
1195 attribute showDropZones { xsd:boolean }?,
1196
1197 ## default value: 0
1198 attribute createdVersion { xsd:unsignedByte }?,
1199
1200 ## default value: 1
1201 attribute indent { xsd:unsignedInt }?,

```

```

1202
1203   ## default value: false
1204   attribute showEmptyRow { xsd:boolean }?,
1205
1206   ## default value: false
1207   attribute showEmptyCol { xsd:boolean }?,
1208
1209   ## default value: true
1210   attribute showHeaders { xsd:boolean }?,
1211
1212   ## default value: true
1213   attribute compact { xsd:boolean }?,
1214
1215   ## default value: false
1216   attribute outline { xsd:boolean }?,
1217
1218   ## default value: false
1219   attribute outlineData { xsd:boolean }?,
1220
1221   ## default value: true
1222   attribute compactData { xsd:boolean }?,
1223
1224   ## default value: false
1225   attribute published { xsd:boolean }?,
1226
1227   ## default value: false
1228   attribute gridDropZones { xsd:boolean }?,
1229
1230   ## default value: true
1231   attribute immersive { xsd:boolean }?,
1232
1233   ## default value: true
1234   attribute multipleFieldFilters { xsd:boolean }?,
1235
1236   ## default value: 0
1237   attribute chartFormat { xsd:unsignedInt }?,
1238   attribute rowHeaderCaption { s_ST_Xstring }?,
1239   attribute colHeaderCaption { s_ST_Xstring }?,
1240
1241   ## default value: false
1242   attribute fieldListSortAscending { xsd:boolean }?,
1243
1244   ## default value: false
1245   attribute mdxSubqueries { xsd:boolean }?,
1246
1247   ## default value: true
1248   attribute customListSort { xsd:boolean }?,
1249   element location { sml_CT_Location },
1250   element pivotFields { sml_CT_PivotFields }?,
1251   element rowFields { sml_CT_RowFields }?,
1252   element rowItems { sml_CT_rowItems }?,
1253   element colFields { sml_CT_ColFields }?,
1254   element colItems { sml_CT_colItems }?,

```

```

1255 element pageFields { sml_CT_PageFields }?,
1256 element dataFields { sml_CT_DataFields }?,
1257 element formats { sml_CT_Formats }?,
1258 element conditionalFormats { sml_CT_ConditionalFormats }?,
1259 element chartFormats { sml_CT_ChartFormats }?,
1260 element pivotHierarchies { sml_CT_PivotHierarchies }?,
1261 element pivotTableStyleInfo { sml_CT_PivotTableStyle }?,
1262 element filters { sml_CT_PivotFilters }?,
1263 element rowHierarchiesUsage { sml_CT_RowHierarchiesUsage }?,
1264 element colHierarchiesUsage { sml_CT_ColHierarchiesUsage }?,
1265 element extLst { sml_CT_ExtensionList }?
1266 sml_CT_Location =
1267     attribute ref { sml_ST_Ref },
1268     attribute firstHeaderRow { xsd:unsignedInt },
1269     attribute firstDataRow { xsd:unsignedInt },
1270     attribute firstDataCol { xsd:unsignedInt },
1271
1272     ## default value: 0
1273     attribute rowPageCount { xsd:unsignedInt }?,
1274
1275     ## default value: 0
1276     attribute colPageCount { xsd:unsignedInt }?
1277 sml_CT_PivotFields =
1278     attribute count { xsd:unsignedInt }?,
1279     element pivotField { sml_CT_PivotField }+
1280 sml_CT_PivotField =
1281     attribute name { s_ST_Xstring }?,
1282     attribute axis { sml_ST_Axis }?,
1283
1284     ## default value: false
1285     attribute dataField { xsd:boolean }?,
1286     attribute subtotalCaption { s_ST_Xstring }?,
1287
1288     ## default value: true
1289     attribute showDropDowns { xsd:boolean }?,
1290
1291     ## default value: false
1292     attribute hiddenLevel { xsd:boolean }?,
1293     attribute uniqueMemberProperty { s_ST_Xstring }?,
1294
1295     ## default value: true
1296     attribute compact { xsd:boolean }?,
1297
1298     ## default value: false
1299     attribute allDrilled { xsd:boolean }?,
1300     attribute numFmtId { sml_ST_NumFmtId }?,
1301
1302     ## default value: true
1303     attribute outline { xsd:boolean }?,
1304
1305     ## default value: true
1306     attribute subtotalTop { xsd:boolean }?,
1307

```

```

1308 ## default value: true
1309 attribute dragToRow { xsd:boolean }?,
1310
1311 ## default value: true
1312 attribute dragToCol { xsd:boolean }?,
1313
1314 ## default value: false
1315 attribute multipleItemSelectionAllowed { xsd:boolean }?,
1316
1317 ## default value: true
1318 attribute dragToPage { xsd:boolean }?,
1319
1320 ## default value: true
1321 attribute dragToData { xsd:boolean }?,
1322
1323 ## default value: true
1324 attribute dragOff { xsd:boolean }?,
1325
1326 ## default value: true
1327 attribute showAll { xsd:boolean }?,
1328
1329 ## default value: false
1330 attribute insertBlankRow { xsd:boolean }?,
1331
1332 ## default value: false
1333 attribute serverField { xsd:boolean }?,
1334
1335 ## default value: false
1336 attribute insertPageBreak { xsd:boolean }?,
1337
1338 ## default value: false
1339 attribute autoShow { xsd:boolean }?,
1340
1341 ## default value: true
1342 attribute topAutoShow { xsd:boolean }?,
1343
1344 ## default value: false
1345 attribute hideNewItems { xsd:boolean }?,
1346
1347 ## default value: false
1348 attribute measureFilter { xsd:boolean }?,
1349
1350 ## default value: false
1351 attribute includeNewItemsInFilter { xsd:boolean }?,
1352
1353 ## default value: 10
1354 attribute itemPageCount { xsd:unsignedInt }?,
1355
1356 ## default value: manual
1357 attribute sortType { sml_ST_FieldSortType }?,
1358 attribute dataSourceSort { xsd:boolean }?,
1359
1360 ## default value: false

```

```

1361 attribute nonAutoSortDefault { xsd:boolean }?,
1362 attribute rankBy { xsd:unsignedInt }?,
1363
1364 ## default value: true
1365 attribute defaultSubtotal { xsd:boolean }?,
1366
1367 ## default value: false
1368 attribute sumSubtotal { xsd:boolean }?,
1369
1370 ## default value: false
1371 attribute countASubtotal { xsd:boolean }?,
1372
1373 ## default value: false
1374 attribute avgSubtotal { xsd:boolean }?,
1375
1376 ## default value: false
1377 attribute maxSubtotal { xsd:boolean }?,
1378
1379 ## default value: false
1380 attribute minSubtotal { xsd:boolean }?,
1381
1382 ## default value: false
1383 attribute productSubtotal { xsd:boolean }?,
1384
1385 ## default value: false
1386 attribute countSubtotal { xsd:boolean }?,
1387
1388 ## default value: false
1389 attribute stdDevSubtotal { xsd:boolean }?,
1390
1391 ## default value: false
1392 attribute stdDevPSubtotal { xsd:boolean }?,
1393
1394 ## default value: false
1395 attribute varSubtotal { xsd:boolean }?,
1396
1397 ## default value: false
1398 attribute varPSubtotal { xsd:boolean }?,
1399
1400 ## default value: false
1401 attribute showPropCell { xsd:boolean }?,
1402
1403 ## default value: false
1404 attribute showPropTip { xsd:boolean }?,
1405
1406 ## default value: false
1407 attribute showPropAsCaption { xsd:boolean }?,
1408
1409 ## default value: false
1410 attribute defaultAttributeDrillState { xsd:boolean }?,
1411 element items { sml_CT_Items }?,
1412 element autoSortScope { sml_CT_AutoSortScope }?,
1413 element extLst { sml_CT_ExtensionList }?

```

```

1414 sml_CT_AutoSortScope = element pivotArea { sml_CT_PivotArea }
1415 sml_CT_Items =
1416     attribute count { xsd:unsignedInt }?,
1417     element item { sml_CT_Item }+
1418 sml_CT_Item =
1419     attribute n { s_ST_Xstring }?,
1420
1421     ## default value: data
1422     attribute t { sml_ST_ItemType }?,
1423
1424     ## default value: false
1425     attribute h { xsd:boolean }?,
1426
1427     ## default value: false
1428     attribute s { xsd:boolean }?,
1429
1430     ## default value: true
1431     attribute sd { xsd:boolean }?,
1432
1433     ## default value: false
1434     attribute f { xsd:boolean }?,
1435
1436     ## default value: false
1437     attribute m { xsd:boolean }?,
1438
1439     ## default value: false
1440     attribute c { xsd:boolean }?,
1441     attribute x { xsd:unsignedInt }?,
1442
1443     ## default value: false
1444     attribute d { xsd:boolean }?,
1445
1446     ## default value: true
1447     attribute e { xsd:boolean }?
1448 sml_CT_PageFields =
1449     attribute count { xsd:unsignedInt }?,
1450     element pageField { sml_CT_PageField }+
1451 sml_CT_PageField =
1452     attribute fld { xsd:int },
1453     attribute item { xsd:unsignedInt }?,
1454     attribute hier { xsd:int }?,
1455     attribute name { s_ST_Xstring }?,
1456     attribute cap { s_ST_Xstring }?,
1457     element extLst { sml_CT_ExtensionList }?
1458 sml_CT_DataFields =
1459     attribute count { xsd:unsignedInt }?,
1460     element dataField { sml_CT_DataField }+
1461 sml_CT_DataField =
1462     attribute name { s_ST_Xstring }?,
1463     attribute fld { xsd:unsignedInt },
1464
1465     ## default value: sum
1466     attribute subtotal { sml_ST_DataConsolidateFunction }?,

```

```

1467
1468   ## default value: normal
1469   attribute showDataAs { sml_ST_ShowDataAs }?,
1470
1471   ## default value: -1
1472   attribute baseField { xsd:int }?,
1473
1474   ## default value: 1048832
1475   attribute baseItem { xsd:unsignedInt }?,
1476   attribute numFmtId { sml_ST_NumFmtId }?,
1477   element extLst { sml_CT_ExtensionList }?
1478 sml_CT_rowItems =
1479   attribute count { xsd:unsignedInt }?,
1480   element i { sml_CT_I }+
1481 sml_CT_colItems =
1482   attribute count { xsd:unsignedInt }?,
1483   element i { sml_CT_I }+
1484 sml_CT_I =
1485
1486   ## default value: data
1487   attribute t { sml_ST_ItemType }?,
1488
1489   ## default value: 0
1490   attribute r { xsd:unsignedInt }?,
1491
1492   ## default value: 0
1493   attribute i { xsd:unsignedInt }?,
1494   element x { sml_CT_X }*
1495 sml_CT_X =
1496
1497   ## default value: 0
1498   attribute v { xsd:int }?
1499 sml_CT_RowFields =
1500
1501   ## default value: 0
1502   attribute count { xsd:unsignedInt }?,
1503   element field { sml_CT_Field }+
1504 sml_CT_ColFields =
1505
1506   ## default value: 0
1507   attribute count { xsd:unsignedInt }?,
1508   element field { sml_CT_Field }+
1509 sml_CT_Field = attribute x { xsd:int }
1510 sml_CT_Formats =
1511
1512   ## default value: 0
1513   attribute count { xsd:unsignedInt }?,
1514   element format { sml_CT_Format }+
1515 sml_CT_Format =
1516
1517   ## default value: formatting
1518   attribute action { sml_ST_FormatAction }?,
1519   attribute dxId { sml_ST_DxfId }?,

```

```

1520     element pivotArea { sml_CT_PivotArea },
1521     element extLst { sml_CT_ExtensionList }?
1522 sml_CT_ConditionalFormats =
1523
1524     ## default value: 0
1525     attribute count { xsd:unsignedInt }?,
1526     element conditionalFormat { sml_CT_ConditionalFormat }+
1527 sml_CT_ConditionalFormat =
1528
1529     ## default value: selection
1530     attribute scope { sml_ST_Scope }?,
1531
1532     ## default value: none
1533     attribute type { sml_ST_Type }?,
1534     attribute priority { xsd:unsignedInt },
1535     element pivotAreas { sml_CT_PivotAreas },
1536     element extLst { sml_CT_ExtensionList }?
1537 sml_CT_PivotAreas =
1538     attribute count { xsd:unsignedInt }?,
1539     element pivotArea { sml_CT_PivotArea }*
1540 sml_ST_Scope = string "selection" | string "data" | string "field"
1541 sml_ST_Type =
1542     string "none" | string "all" | string "row" | string "column"
1543 sml_CT_ChartFormats =
1544
1545     ## default value: 0
1546     attribute count { xsd:unsignedInt }?,
1547     element chartFormat { sml_CT_ChartFormat }+
1548 sml_CT_ChartFormat =
1549     attribute chart { xsd:unsignedInt },
1550     attribute format { xsd:unsignedInt },
1551
1552     ## default value: false
1553     attribute series { xsd:boolean }?,
1554     element pivotArea { sml_CT_PivotArea }
1555 sml_CT_PivotHierarchies =
1556     attribute count { xsd:unsignedInt }?,
1557     element pivotHierarchy { sml_CT_PivotHierarchy }+
1558 sml_CT_PivotHierarchy =
1559
1560     ## default value: false
1561     attribute outline { xsd:boolean }?,
1562
1563     ## default value: false
1564     attribute multipleItemSelectionAllowed { xsd:boolean }?,
1565
1566     ## default value: false
1567     attribute subtotalTop { xsd:boolean }?,
1568
1569     ## default value: true
1570     attribute showInFieldList { xsd:boolean }?,
1571
1572     ## default value: true

```



```

1573 attribute dragToRow { xsd:boolean }?,
1574
1575 ## default value: true
1576 attribute dragToCol { xsd:boolean }?,
1577
1578 ## default value: true
1579 attribute dragToPage { xsd:boolean }?,
1580
1581 ## default value: false
1582 attribute dragToData { xsd:boolean }?,
1583
1584 ## default value: true
1585 attribute dragOff { xsd:boolean }?,
1586
1587 ## default value: false
1588 attribute includeNewItemInFilter { xsd:boolean }?,
1589 attribute caption { s_ST_Xstring }?,
1590 element mps { sml_CT_MemberProperties }?,
1591 element members { sml_CT_Members }*,
1592 element extLst { sml_CT_ExtensionList }?
1593 sml_CT_RowHierarchiesUsage =
1594 attribute count { xsd:unsignedInt }?,
1595 element rowHierarchyUsage { sml_CT_HierarchyUsage }+
1596 sml_CT_ColHierarchiesUsage =
1597 attribute count { xsd:unsignedInt }?,
1598 element colHierarchyUsage { sml_CT_HierarchyUsage }+
1599 sml_CT_HierarchyUsage = attribute hierarchyUsage { xsd:int }
1600 sml_CT_MemberProperties =
1601 attribute count { xsd:unsignedInt }?,
1602 element mp { sml_CT_MemberProperty }+
1603 sml_CT_MemberProperty =
1604 attribute name { s_ST_Xstring }?,
1605
1606 ## default value: false
1607 attribute showCell { xsd:boolean }?,
1608
1609 ## default value: false
1610 attribute showTip { xsd:boolean }?,
1611
1612 ## default value: false
1613 attribute showAsCaption { xsd:boolean }?,
1614 attribute nameLen { xsd:unsignedInt }?,
1615 attribute pPos { xsd:unsignedInt }?,
1616 attribute pLen { xsd:unsignedInt }?,
1617 attribute level { xsd:unsignedInt }?,
1618 attribute field { xsd:unsignedInt }
1619 sml_CT_Members =
1620 attribute count { xsd:unsignedInt }?,
1621 attribute level { xsd:unsignedInt }?,
1622 element member { sml_CT_Member }+
1623 sml_CT_Member = attribute name { s_ST_Xstring }
1624 sml_CT_Dimensions =
1625 attribute count { xsd:unsignedInt }?,

```

```

1626     element dimension { sml_CT_PivotDimension }*
1627 sml_CT_PivotDimension =
1628
1629     ## default value: false
1630     attribute measure { xsd:boolean }?,
1631     attribute name { s_ST_Xstring },
1632     attribute uniqueName { s_ST_Xstring },
1633     attribute caption { s_ST_Xstring }
1634 sml_CT_MeasureGroups =
1635     attribute count { xsd:unsignedInt }?,
1636     element measureGroup { sml_CT_MeasureGroup }*
1637 sml_CT_MeasureDimensionMaps =
1638     attribute count { xsd:unsignedInt }?,
1639     element map { sml_CT_MeasureDimensionMap }*
1640 sml_CT_MeasureGroup =
1641     attribute name { s_ST_Xstring },
1642     attribute caption { s_ST_Xstring }
1643 sml_CT_MeasureDimensionMap =
1644     attribute measureGroup { xsd:unsignedInt }?,
1645     attribute dimension { xsd:unsignedInt }?
1646 sml_CT_PivotTableStyle =
1647     attribute name { xsd:string }?,
1648     attribute showRowHeaders { xsd:boolean }?,
1649     attribute showColHeaders { xsd:boolean }?,
1650     attribute showRowStripes { xsd:boolean }?,
1651     attribute showColStripes { xsd:boolean }?,
1652     attribute showLastColumn { xsd:boolean }?
1653 sml_CT_PivotFilters =
1654
1655     ## default value: 0
1656     attribute count { xsd:unsignedInt }?,
1657     element filter { sml_CT_PivotFilter }*
1658 sml_CT_PivotFilter =
1659     attribute fld { xsd:unsignedInt },
1660     attribute mpFld { xsd:unsignedInt }?,
1661     attribute type { sml_ST_PivotFilterType },
1662
1663     ## default value: 0
1664     attribute evalOrder { xsd:int }?,
1665     attribute id { xsd:unsignedInt },
1666     attribute iMeasureHier { xsd:unsignedInt }?,
1667     attribute iMeasureFld { xsd:unsignedInt }?,
1668     attribute name { s_ST_Xstring },
1669     attribute description { s_ST_Xstring },
1670     attribute stringValue1 { s_ST_Xstring },
1671     attribute stringValue2 { s_ST_Xstring },
1672     element autoFilter { sml_CT_AutoFilter },
1673     element extLst { sml_CT_ExtensionList }?
1674 sml_ST_ShowDataAs =
1675     string "normal"
1676     | string "difference"
1677     | string "percent"
1678     | string "percentDiff"

```

```

1679 | string "runTotal"
1680 | string "percentOfRow"
1681 | string "percentOfCol"
1682 | string "percentOfTotal"
1683 | string "index"
1684 sml_ST_ItemType =
1685     string "data"
1686     | string "default"
1687     | string "sum"
1688     | string "countA"
1689     | string "avg"
1690     | string "max"
1691     | string "min"
1692     | string "product"
1693     | string "count"
1694     | string "stdDev"
1695     | string "stdDevP"
1696     | string "var"
1697     | string "varP"
1698     | string "grand"
1699     | string "blank"
1700 sml_ST_FormatAction =
1701     string "blank"
1702     | string "formatting"
1703     | string "drill"
1704     | string "formula"
1705 sml_ST_FieldSortType =
1706     string "manual" | string "ascending" | string "descending"
1707 sml_ST_PivotFilterType =
1708     string "unknown"
1709     | string "count"
1710     | string "percent"
1711     | string "sum"
1712     | string "captionEqual"
1713     | string "captionNotEqual"
1714     | string "captionBeginsWith"
1715     | string "captionNotBeginsWith"
1716     | string "captionEndsWith"
1717     | string "captionNotEndsWith"
1718     | string "captionContains"
1719     | string "captionNotContains"
1720     | string "captionGreaterThan"
1721     | string "captionGreaterThanOrEqual"
1722     | string "captionLessThan"
1723     | string "captionLessThanOrEqual"
1724     | string "captionBetween"
1725     | string "captionNotBetween"
1726     | string "valueEqual"
1727     | string "valueNotEqual"
1728     | string "valueGreaterThan"
1729     | string "valueGreaterThanOrEqual"
1730     | string "valueLessThan"
1731     | string "valueLessThanOrEqual"

```

```

1732 | string "valueBetween"
1733 | string "valueNotBetween"
1734 | string "dateEqual"
1735 | string "dateNotEqual"
1736 | string "dateOlderThan"
1737 | string "dateOlderThanOrEqual"
1738 | string "dateNewerThan"
1739 | string "dateNewerThanOrEqual"
1740 | string "dateBetween"
1741 | string "dateNotBetween"
1742 | string "tomorrow"
1743 | string "today"
1744 | string "yesterday"
1745 | string "nextWeek"
1746 | string "thisWeek"
1747 | string "lastWeek"
1748 | string "nextMonth"
1749 | string "thisMonth"
1750 | string "lastMonth"
1751 | string "nextQuarter"
1752 | string "thisQuarter"
1753 | string "lastQuarter"
1754 | string "nextYear"
1755 | string "thisYear"
1756 | string "lastYear"
1757 | string "yearToDate"
1758 | string "Q1"
1759 | string "Q2"
1760 | string "Q3"
1761 | string "Q4"
1762 | string "M1"
1763 | string "M2"
1764 | string "M3"
1765 | string "M4"
1766 | string "M5"
1767 | string "M6"
1768 | string "M7"
1769 | string "M8"
1770 | string "M9"
1771 | string "M10"
1772 | string "M11"
1773 | string "M12"
1774 sml_CT_PivotArea =
1775   attribute field { xsd:int }?,
1776
1777   ## default value: normal
1778   attribute type { sml_ST_PivotAreaType }?,
1779
1780   ## default value: true
1781   attribute dataOnly { xsd:boolean }?,
1782
1783   ## default value: false
1784   attribute labelOnly { xsd:boolean }?,

```

```

1785
1786   ## default value: false
1787   attribute grandRow { xsd:boolean }?,
1788
1789   ## default value: false
1790   attribute grandCol { xsd:boolean }?,
1791
1792   ## default value: false
1793   attribute cacheIndex { xsd:boolean }?,
1794
1795   ## default value: true
1796   attribute outline { xsd:boolean }?,
1797   attribute offset { sml_ST_Ref }?,
1798
1799   ## default value: false
1800   attribute collapsedLevelsAreSubtotals { xsd:boolean }?,
1801   attribute axis { sml_ST_Axis }?,
1802   attribute fieldPosition { xsd:unsignedInt }?,
1803   element references { sml_CT_PivotAreaReferences }?,
1804   element extLst { sml_CT_ExtensionList }?
1805 sml_ST_PivotAreaType =
1806     string "none"
1807     | string "normal"
1808     | string "data"
1809     | string "all"
1810     | string "origin"
1811     | string "button"
1812     | string "topEnd"
1813     | string "topRight"
1814 sml_CT_PivotAreaReferences =
1815     attribute count { xsd:unsignedInt }?,
1816     element reference { sml_CT_PivotAreaReference }+
1817 sml_CT_PivotAreaReference =
1818     attribute field { xsd:unsignedInt }?,
1819     attribute count { xsd:unsignedInt }?,
1820
1821     ## default value: true
1822     attribute selected { xsd:boolean }?,
1823
1824     ## default value: false
1825     attribute byPosition { xsd:boolean }?,
1826
1827     ## default value: false
1828     attribute relative { xsd:boolean }?,
1829
1830     ## default value: false
1831     attribute defaultSubtotal { xsd:boolean }?,
1832
1833     ## default value: false
1834     attribute sumSubtotal { xsd:boolean }?,
1835
1836     ## default value: false
1837     attribute countASubtotal { xsd:boolean }?,

```

```

1838
1839   ## default value: false
1840   attribute avgSubtotal { xsd:boolean }?,
1841
1842   ## default value: false
1843   attribute maxSubtotal { xsd:boolean }?,
1844
1845   ## default value: false
1846   attribute minSubtotal { xsd:boolean }?,
1847
1848   ## default value: false
1849   attribute productSubtotal { xsd:boolean }?,
1850
1851   ## default value: false
1852   attribute countSubtotal { xsd:boolean }?,
1853
1854   ## default value: false
1855   attribute stdDevSubtotal { xsd:boolean }?,
1856
1857   ## default value: false
1858   attribute stdDevPSubtotal { xsd:boolean }?,
1859
1860   ## default value: false
1861   attribute varSubtotal { xsd:boolean }?,
1862
1863   ## default value: false
1864   attribute varPSubtotal { xsd:boolean }?,
1865   element x { sml_CT_Index }*,
1866   element extLst { sml_CT_ExtensionList }?
1867 sml_CT_Index = attribute v { xsd:unsignedInt }
1868 sml_ST_Axis =
1869   string "axisRow"
1870   | string "axisCol"
1871   | string "axisPage"
1872   | string "axisValues"
1873 sml_queryTable = element queryTable { sml_CT_QueryTable }
1874 sml_CT_QueryTable =
1875   attribute name { s_ST_Xstring },
1876
1877   ## default value: true
1878   attribute headers { xsd:boolean }?,
1879
1880   ## default value: false
1881   attribute rowNumbers { xsd:boolean }?,
1882
1883   ## default value: false
1884   attribute disableRefresh { xsd:boolean }?,
1885
1886   ## default value: true
1887   attribute backgroundRefresh { xsd:boolean }?,
1888
1889   ## default value: false
1890   attribute firstBackgroundRefresh { xsd:boolean }?,

```

```

1891
1892 ## default value: false
1893 attribute refreshOnLoad { xsd:boolean }?,
1894
1895 ## default value: insertDelete
1896 attribute growShrinkType { sml_ST_GrowShrinkType }?,
1897
1898 ## default value: false
1899 attribute fillFormulas { xsd:boolean }?,
1900
1901 ## default value: false
1902 attribute removeDataOnSave { xsd:boolean }?,
1903
1904 ## default value: false
1905 attribute disableEdit { xsd:boolean }?,
1906
1907 ## default value: true
1908 attribute preserveFormatting { xsd:boolean }?,
1909
1910 ## default value: true
1911 attribute adjustColumnWidth { xsd:boolean }?,
1912
1913 ## default value: false
1914 attribute intermediate { xsd:boolean }?,
1915 attribute connectionId { xsd:unsignedInt },
1916 sml_AG_AutoFormat,
1917 element queryTableRefresh { sml_CT_QueryTableRefresh }?,
1918 element extLst { sml_CT_ExtensionList }?
1919 sml_CT_QueryTableRefresh =
1920
1921 ## default value: true
1922 attribute preserveSortFilterLayout { xsd:boolean }?,
1923
1924 ## default value: false
1925 attribute fieldIdWrapped { xsd:boolean }?,
1926
1927 ## default value: true
1928 attribute headersInLastRefresh { xsd:boolean }?,
1929
1930 ## default value: 0
1931 attribute minimumVersion { xsd:unsignedByte }?,
1932
1933 ## default value: 1
1934 attribute nextId { xsd:unsignedInt }?,
1935
1936 ## default value: 0
1937 attribute unboundColumnsLeft { xsd:unsignedInt }?,
1938
1939 ## default value: 0
1940 attribute unboundColumnsRight { xsd:unsignedInt }?,
1941 element queryTableFields { sml_CT_QueryTableFields },
1942 element queryTableDeletedFields { sml_CT_QueryTableDeletedFields }?,
1943 element sortState { sml_CT_SortState }?,

```

```

1944     element extLst { sml_CT_ExtensionList }?
1945 sml_CT_QueryTableDeletedFields =
1946     attribute count { xsd:unsignedInt }?,
1947     element deletedField { sml_CT_DeletedField }+
1948 sml_CT_DeletedField = attribute name { s_ST_Xstring }
1949 sml_CT_QueryTableFields =
1950
1951     ## default value: 0
1952     attribute count { xsd:unsignedInt }?,
1953     element queryTableField { sml_CT_QueryTableField }*
1954 sml_CT_QueryTableField =
1955     attribute id { xsd:unsignedInt },
1956     attribute name { s_ST_Xstring }?,
1957
1958     ## default value: true
1959     attribute dataBound { xsd:boolean }?,
1960
1961     ## default value: false
1962     attribute rowNumbers { xsd:boolean }?,
1963
1964     ## default value: false
1965     attribute fillFormulas { xsd:boolean }?,
1966
1967     ## default value: false
1968     attribute clipped { xsd:boolean }?,
1969
1970     ## default value: 0
1971     attribute tableColumnId { xsd:unsignedInt }?,
1972     element extLst { sml_CT_ExtensionList }?
1973 sml_ST_GrowShrinkType =
1974     string "insertDelete" | string "insertClear" | string "overwriteClear"
1975 sml_sst = element sst { sml_CT_Sst }
1976 sml_CT_Sst =
1977     attribute count { xsd:unsignedInt }?,
1978     attribute uniqueCount { xsd:unsignedInt }?,
1979     element si { sml_CT_Rst }*,
1980     element extLst { sml_CT_ExtensionList }?
1981 sml_ST_PhoneticType =
1982     string "halfwidthKatakana"
1983     | string "fullwidthKatakana"
1984     | string "Hiragana"
1985     | string "noConversion"
1986 sml_ST_PhoneticAlignment =
1987     string "noControl"
1988     | string "left"
1989     | string "center"
1990     | string "distributed"
1991 sml_CT_PhoneticRun =
1992     attribute sb { xsd:unsignedInt },
1993     attribute eb { xsd:unsignedInt },
1994     element t { s_ST_Xstring }
1995 sml_CT_RElt =
1996     element rPr { sml_CT_RPrElt }?,

```



```

1997     element t { s_ST_Xstring }
1998 sml_CT_RPrElt =
1999     (element rFont { sml_CT_FontName }?
2000     | element charset { sml_CT_IntProperty }?
2001     | element family { sml_CT_IntProperty }?
2002     | element b { sml_CT_BooleanProperty }?
2003     | element i { sml_CT_BooleanProperty }?
2004     | element strike { sml_CT_BooleanProperty }?
2005     | element outline { sml_CT_BooleanProperty }?
2006     | element shadow { sml_CT_BooleanProperty }?
2007     | element condense { sml_CT_BooleanProperty }?
2008     | element extend { sml_CT_BooleanProperty }?
2009     | element color { sml_CT_Color }?
2010     | element sz { sml_CT_FontSize }?
2011     | element u { sml_CT_UnderlineProperty }?
2012     | element vertAlign { sml_CT_VerticalAlignFontProperty }?
2013     | element scheme { sml_CT_FontScheme }?)+
2014 sml_CT_Rst =
2015     element t { s_ST_Xstring }?,
2016     element r { sml_CT_RElt }*,
2017     element rPh { sml_CT_PhoneticRun }*,
2018     element phoneticPr { sml_CT_PhoneticPr }?
2019 sml_CT_PhoneticPr =
2020     attribute fontId { sml_ST_FontId },
2021
2022     ## default value: fullwidthKatakana
2023     attribute type { sml_ST_PhoneticType }?,
2024
2025     ## default value: left
2026     attribute alignment { sml_ST_PhoneticAlignment }?
2027 sml_headers = element headers { sml_CT_RevisionHeaders }
2028 sml_revisions = element revisions { sml_CT_Revisions }
2029 sml_CT_RevisionHeaders =
2030     attribute guid { s_ST_Guid },
2031     attribute lastGuid { s_ST_Guid }?,
2032
2033     ## default value: true
2034     attribute shared { xsd:boolean }?,
2035
2036     ## default value: false
2037     attribute diskRevisions { xsd:boolean }?,
2038
2039     ## default value: true
2040     attribute history { xsd:boolean }?,
2041
2042     ## default value: true
2043     attribute trackRevisions { xsd:boolean }?,
2044
2045     ## default value: false
2046     attribute exclusive { xsd:boolean }?,
2047
2048     ## default value: 0
2049     attribute revisionId { xsd:unsignedInt }?,

```

```

2050
2051   ## default value: 1
2052   attribute version { xsd:int }?,
2053
2054   ## default value: true
2055   attribute keepChangeHistory { xsd:boolean }?,
2056
2057   ## default value: false
2058   attribute protected { xsd:boolean }?,
2059
2060   ## default value: 30
2061   attribute preserveHistory { xsd:unsignedInt }?,
2062   element header { sml_CT_RevisionHeader }+
2063 sml_CT_Revisions =
2064   (element rrc { sml_CT_RevisionRowColumn }*
2065    | element rm { sml_CT_RevisionMove }*
2066    | element rcv { sml_CT_RevisionCustomView }*
2067    | element rsnm { sml_CT_RevisionSheetRename }*
2068    | element ris { sml_CT_RevisionInsertSheet }*
2069    | element rcc { sml_CT_RevisionCellChange }*
2070    | element rfmt { sml_CT_RevisionFormatting }*
2071    | element raf { sml_CT_RevisionAutoFormatting }*
2072    | element rdn { sml_CT_RevisionDefinedName }*
2073    | element rcmt { sml_CT_RevisionComment }*
2074    | element rqt { sml_CT_RevisionQueryTableField }*
2075    | element rcft { sml_CT_RevisionConflict }*)+
2076 sml_AG_RevData =
2077   attribute rId { xsd:unsignedInt },
2078
2079   ## default value: false
2080   attribute ua { xsd:boolean }?,
2081
2082   ## default value: false
2083   attribute ra { xsd:boolean }?
2084 sml_CT_RevisionHeader =
2085   attribute guid { s_ST_Guid },
2086   attribute dateTime { xsd:dateTime },
2087   attribute maxSheetId { xsd:unsignedInt },
2088   attribute userName { s_ST_Xstring },
2089   r_id,
2090   attribute minRId { xsd:unsignedInt }?,
2091   attribute maxRId { xsd:unsignedInt }?,
2092   element sheetIdMap { sml_CT_SheetIdMap },
2093   element reviewedList { sml_CT_ReviewedRevisions }?,
2094   element extLst { sml_CT_ExtensionList }?
2095 sml_CT_SheetIdMap =
2096   attribute count { xsd:unsignedInt }?,
2097   element sheetId { sml_CT_SheetId }+
2098 sml_CT_SheetId = attribute val { xsd:unsignedInt }
2099 sml_CT_ReviewedRevisions =
2100   attribute count { xsd:unsignedInt }?,
2101   element reviewed { sml_CT_Reviewed }+
2102 sml_CT_Reviewed = attribute rId { xsd:unsignedInt }

```

```

2103 sml_CT_UndoInfo =
2104     attribute index { xsd:unsignedInt },
2105     attribute exp { sml_ST_FormulaExpression },
2106
2107     ## default value: false
2108     attribute ref3D { xsd:boolean }?,
2109
2110     ## default value: false
2111     attribute array { xsd:boolean }?,
2112
2113     ## default value: false
2114     attribute v { xsd:boolean }?,
2115
2116     ## default value: false
2117     attribute nf { xsd:boolean }?,
2118
2119     ## default value: false
2120     attribute cs { xsd:boolean }?,
2121     attribute dr { sml_ST_RefA },
2122     attribute dn { s_ST_Xstring }?,
2123     attribute r { sml_ST_CellRef }?,
2124     attribute sId { xsd:unsignedInt }?
2125 sml_CT_RevisionRowColumn =
2126     sml_AG_RevData,
2127     attribute sId { xsd:unsignedInt },
2128
2129     ## default value: false
2130     attribute eol { xsd:boolean }?,
2131     attribute ref { sml_ST_Ref },
2132     attribute action { sml_ST_rwColActionType },
2133
2134     ## default value: false
2135     attribute edge { xsd:boolean }?,
2136     (element undo { sml_CT_UndoInfo }*
2137      | element rcc { sml_CT_RevisionCellChange }*
2138      | element rfmt { sml_CT_RevisionFormatting }*)*
2139 sml_CT_RevisionMove =
2140     sml_AG_RevData,
2141     attribute sheetId { xsd:unsignedInt },
2142     attribute source { sml_ST_Ref },
2143     attribute destination { sml_ST_Ref },
2144
2145     ## default value: 0
2146     attribute sourceSheetId { xsd:unsignedInt }?,
2147     (element undo { sml_CT_UndoInfo }*
2148      | element rcc { sml_CT_RevisionCellChange }*
2149      | element rfmt { sml_CT_RevisionFormatting }*)*
2150 sml_CT_RevisionCustomView =
2151     attribute guid { s_ST_Guid },
2152     attribute action { sml_ST_RevisionAction }
2153 sml_CT_RevisionSheetRename =
2154     sml_AG_RevData,
2155     attribute sheetId { xsd:unsignedInt },

```

```

2156     attribute oldName { s_ST_Xstring },
2157     attribute newName { s_ST_Xstring },
2158     element extLst { sml_CT_ExtensionList }?
2159 sml_CT_RevisionInsertSheet =
2160     sml_AG_RevData,
2161     attribute sheetId { xsd:unsignedInt },
2162     attribute name { s_ST_Xstring },
2163     attribute sheetPosition { xsd:unsignedInt }
2164 sml_CT_RevisionCellChange =
2165     sml_AG_RevData,
2166     attribute sId { xsd:unsignedInt },
2167
2168     ## default value: false
2169     attribute odf { xsd:boolean }?,
2170
2171     ## default value: false
2172     attribute xfdxf { xsd:boolean }?,
2173
2174     ## default value: false
2175     attribute s { xsd:boolean }?,
2176
2177     ## default value: false
2178     attribute dxf { xsd:boolean }?,
2179     attribute numFmtId { sml_ST_NumFmtId }?,
2180
2181     ## default value: false
2182     attribute quotePrefix { xsd:boolean }?,
2183
2184     ## default value: false
2185     attribute oldQuotePrefix { xsd:boolean }?,
2186
2187     ## default value: false
2188     attribute ph { xsd:boolean }?,
2189
2190     ## default value: false
2191     attribute oldPh { xsd:boolean }?,
2192
2193     ## default value: false
2194     attribute endOfListFormulaUpdate { xsd:boolean }?,
2195     element oc { sml_CT_Cell },
2196     element nc { sml_CT_Cell },
2197     element odf { sml_CT_Dxf },
2198     element ndxf { sml_CT_Dxf },
2199     element extLst { sml_CT_ExtensionList }?
2200 sml_CT_RevisionFormatting =
2201     attribute sheetId { xsd:unsignedInt },
2202
2203     ## default value: false
2204     attribute xfdxf { xsd:boolean }?,
2205
2206     ## default value: false
2207     attribute s { xsd:boolean }?,
2208     attribute sqref { sml_ST_Sqref },

```

```

2209     attribute start { xsd:unsignedInt }?,
2210     attribute length { xsd:unsignedInt }?,
2211     element dxf { sml_CT_Dxf }?,
2212     element extLst { sml_CT_ExtensionList }?
2213 sml_CT_RevisionAutoFormatting =
2214     attribute sheetId { xsd:unsignedInt },
2215     sml_AG_AutoFormat,
2216     attribute ref { sml_ST_Ref }
2217 sml_CT_RevisionComment =
2218     attribute sheetId { xsd:unsignedInt },
2219     attribute cell { sml_ST_CellRef },
2220     attribute guid { s_ST_Guid },
2221
2222     ## default value: add
2223     attribute action { sml_ST_RevisionAction }?,
2224
2225     ## default value: false
2226     attribute alwaysShow { xsd:boolean }?,
2227
2228     ## default value: false
2229     attribute old { xsd:boolean }?,
2230
2231     ## default value: false
2232     attribute hiddenRow { xsd:boolean }?,
2233
2234     ## default value: false
2235     attribute hiddenColumn { xsd:boolean }?,
2236     attribute author { s_ST_Xstring },
2237
2238     ## default value: 0
2239     attribute oldLength { xsd:unsignedInt }?,
2240
2241     ## default value: 0
2242     attribute newLength { xsd:unsignedInt }?
2243 sml_CT_RevisionDefinedName =
2244     sml_AG_RevData,
2245     attribute localSheetId { xsd:unsignedInt }?,
2246
2247     ## default value: false
2248     attribute customView { xsd:boolean }?,
2249     attribute name { s_ST_Xstring },
2250
2251     ## default value: false
2252     attribute function { xsd:boolean }?,
2253
2254     ## default value: false
2255     attribute oldFunction { xsd:boolean }?,
2256     attribute functionGroupId { xsd:unsignedByte }?,
2257     attribute oldFunctionGroupId { xsd:unsignedByte }?,
2258     attribute shortcutKey { xsd:unsignedByte }?,
2259     attribute oldShortcutKey { xsd:unsignedByte }?,
2260
2261     ## default value: false

```

```

2262 attribute hidden { xsd:boolean }?,
2263
2264 ## default value: false
2265 attribute oldHidden { xsd:boolean }?,
2266 attribute customMenu { s_ST_Xstring }?,
2267 attribute oldCustomMenu { s_ST_Xstring }?,
2268 attribute description { s_ST_Xstring }?,
2269 attribute oldDescription { s_ST_Xstring }?,
2270 attribute help { s_ST_Xstring }?,
2271 attribute oldHelp { s_ST_Xstring }?,
2272 attribute statusBar { s_ST_Xstring }?,
2273 attribute oldStatusBar { s_ST_Xstring }?,
2274 attribute comment { s_ST_Xstring }?,
2275 attribute oldComment { s_ST_Xstring }?,
2276 element formula { sml_ST_Formula }?,
2277 element oldFormula { sml_ST_Formula }?,
2278 element extLst { sml_CT_ExtensionList }?
2279 sml_CT_RevisionConflict =
2280     sml_AG_RevData,
2281     attribute sheetId { xsd:unsignedInt }?
2282 sml_CT_RevisionQueryTableField =
2283     attribute sheetId { xsd:unsignedInt },
2284     attribute ref { sml_ST_Ref },
2285     attribute fieldId { xsd:unsignedInt }
2286 sml_ST_rwColActionType =
2287     string "insertRow"
2288     | string "deleteRow"
2289     | string "insertCol"
2290     | string "deleteCol"
2291 sml_ST_RevisionAction = string "add" | string "delete"
2292 sml_ST_FormulaExpression =
2293     string "ref"
2294     | string "refError"
2295     | string "area"
2296     | string "areaError"
2297     | string "computedArea"
2298 sml_users = element users { sml_CT_Users }
2299 sml_CT_Users =
2300     attribute count { xsd:unsignedInt }?,
2301     element userInfo { sml_CT_SharedUser }*
2302 sml_CT_SharedUser =
2303     attribute guid { s_ST_Guid },
2304     attribute name { s_ST_Xstring },
2305     attribute id { xsd:int },
2306     attribute dateTime { xsd:dateTime },
2307     element extLst { sml_CT_ExtensionList }?
2308 sml_worksheet = element worksheet { sml_CT_Worksheet }
2309 sml_chartsheet = element chartsheet { sml_CT_Chartsheet }
2310 sml_dialogsheet = element dialogsheet { sml_CT_Dialogsheet }
2311 sml_CT_Macrosheet =
2312     element sheetPr { sml_CT_SheetPr }?,
2313     element dimension { sml_CT_SheetDimension }?,
2314     element sheetViews { sml_CT_SheetViews }?,

```

```

2315     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2316     element cols { sml_CT_Cols }*,
2317     element sheetData { sml_CT_SheetData },
2318     element sheetProtection { sml_CT_SheetProtection }?,
2319     element autoFilter { sml_CT_AutoFilter }?,
2320     element sortState { sml_CT_SortState }?,
2321     element dataConsolidate { sml_CT_DataConsolidate }?,
2322     element customSheetViews { sml_CT_CustomSheetViews }?,
2323     element phoneticPr { sml_CT_PhoneticPr }?,
2324     element conditionalFormatting { sml_CT_ConditionalFormatting }*,
2325     element printOptions { sml_CT_PrintOptions }?,
2326     element pageMargins { sml_CT_PageMargins }?,
2327     element pageSetup { sml_CT_PageSetup }?,
2328     element headerFooter { sml_CT_HeaderFooter }?,
2329     element rowBreaks { sml_CT_PageBreak }?,
2330     element colBreaks { sml_CT_PageBreak }?,
2331     element customProperties { sml_CT_CustomProperties }?,
2332     element drawing { sml_CT_Drawing }?,
2333     element legacyDrawing { sml_CT_LegacyDrawing }?,
2334     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2335     element drawingHF { sml_CT_DrawingHF }?,
2336     element picture { sml_CT_SheetBackgroundPicture }?,
2337     element oleObjects { sml_CT_OleObjects }?,
2338     element extLst { sml_CT_ExtensionList }?
2339 sml_CT_Dialogsheet =
2340     element sheetPr { sml_CT_SheetPr }?,
2341     element sheetViews { sml_CT_SheetViews }?,
2342     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2343     element sheetProtection { sml_CT_SheetProtection }?,
2344     element customSheetViews { sml_CT_CustomSheetViews }?,
2345     element printOptions { sml_CT_PrintOptions }?,
2346     element pageMargins { sml_CT_PageMargins }?,
2347     element pageSetup { sml_CT_PageSetup }?,
2348     element headerFooter { sml_CT_HeaderFooter }?,
2349     element drawing { sml_CT_Drawing }?,
2350     element legacyDrawing { sml_CT_LegacyDrawing }?,
2351     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2352     element drawingHF { sml_CT_DrawingHF }?,
2353     element oleObjects { sml_CT_OleObjects }?,
2354     element extLst { sml_CT_ExtensionList }?
2355 sml_CT_Worksheet =
2356     element sheetPr { sml_CT_SheetPr }?,
2357     element dimension { sml_CT_SheetDimension }?,
2358     element sheetViews { sml_CT_SheetViews }?,
2359     element sheetFormatPr { sml_CT_SheetFormatPr }?,
2360     element cols { sml_CT_Cols }*,
2361     element sheetData { sml_CT_SheetData },
2362     element sheetCalcPr { sml_CT_SheetCalcPr }?,
2363     element sheetProtection { sml_CT_SheetProtection }?,
2364     element protectedRanges { sml_CT_ProtectedRanges }?,
2365     element scenarios { sml_CT_Scenarios }?,
2366     element autoFilter { sml_CT_AutoFilter }?,
2367     element sortState { sml_CT_SortState }?,

```

```

2368     element dataConsolidate { sml_CT_DataConsolidate }?,
2369     element customSheetViews { sml_CT_CustomSheetViews }?,
2370     element mergeCells { sml_CT_MergeCells }?,
2371     element phoneticPr { sml_CT_PhoneticPr }?,
2372     element conditionalFormatting { sml_CT_ConditionalFormatting }*,
2373     element dataValidations { sml_CT_DataValidations }?,
2374     element hyperlinks { sml_CT_Hyperlinks }?,
2375     element printOptions { sml_CT_PrintOptions }?,
2376     element pageMargins { sml_CT_PageMargins }?,
2377     element pageSetup { sml_CT_PageSetup }?,
2378     element headerFooter { sml_CT_HeaderFooter }?,
2379     element rowBreaks { sml_CT_PageBreak }?,
2380     element colBreaks { sml_CT_PageBreak }?,
2381     element customProperties { sml_CT_CustomProperties }?,
2382     element cellWatches { sml_CT_CellWatches }?,
2383     element ignoredErrors { sml_CT_IgnoredErrors }?,
2384     element smartTags { sml_CT_SmartTags }?,
2385     element drawing { sml_CT_Drawing }?,
2386     element legacyDrawing { sml_CT_LegacyDrawing }?,
2387     element legacyDrawingHF { sml_CT_LegacyDrawing }?,
2388     element drawingHF { sml_CT_DrawingHF }?,
2389     element picture { sml_CT_SheetBackgroundPicture }?,
2390     element oleObjects { sml_CT_OleObjects }?,
2391     element controls { sml_CT_Controls }?,
2392     element webPublishItems { sml_CT_WebPublishItems }?,
2393     element tableParts { sml_CT_TableParts }?,
2394     element extLst { sml_CT_ExtensionList }?
2395 sml_CT_SheetData = element row { sml_CT_Row }*
2396 sml_CT_SheetCalcPr =
2397
2398     ## default value: false
2399     attribute fullCalcOnLoad { xsd:boolean }?
2400 sml_CT_SheetFormatPr =
2401
2402     ## default value: 8
2403     attribute baseColWidth { xsd:unsignedInt }?,
2404     attribute defaultColWidth { xsd:double }?,
2405     attribute defaultRowHeight { xsd:double },
2406
2407     ## default value: false
2408     attribute customHeight { xsd:boolean }?,
2409
2410     ## default value: false
2411     attribute zeroHeight { xsd:boolean }?,
2412
2413     ## default value: false
2414     attribute thickTop { xsd:boolean }?,
2415
2416     ## default value: false
2417     attribute thickBottom { xsd:boolean }?,
2418
2419     ## default value: 0
2420     attribute outlineLevelRow { xsd:unsignedByte }?,

```



```

2421
2422     ## default value: 0
2423     attribute outlineLevelCol { xsd:unsignedByte }?
2424 sml_CT_Cols = element col { sml_CT_Col }+
2425 sml_CT_Col =
2426     attribute min { xsd:unsignedInt },
2427     attribute max { xsd:unsignedInt },
2428     attribute width { xsd:double }?,
2429
2430     ## default value: 0
2431     attribute style { xsd:unsignedInt }?,
2432
2433     ## default value: false
2434     attribute hidden { xsd:boolean }?,
2435
2436     ## default value: false
2437     attribute bestFit { xsd:boolean }?,
2438
2439     ## default value: false
2440     attribute customWidth { xsd:boolean }?,
2441
2442     ## default value: false
2443     attribute phonetic { xsd:boolean }?,
2444
2445     ## default value: 0
2446     attribute outlineLevel { xsd:unsignedByte }?,
2447
2448     ## default value: false
2449     attribute collapsed { xsd:boolean }?
2450 sml_ST_CellSpan = xsd:string
2451 sml_ST_CellSpans = list { sml_ST_CellSpan* }
2452 sml_CT_Row =
2453     attribute r { xsd:unsignedInt }?,
2454     attribute spans { sml_ST_CellSpans }?,
2455
2456     ## default value: 0
2457     attribute s { xsd:unsignedInt }?,
2458
2459     ## default value: false
2460     attribute customFormat { xsd:boolean }?,
2461     attribute ht { xsd:double }?,
2462
2463     ## default value: false
2464     attribute hidden { xsd:boolean }?,
2465
2466     ## default value: false
2467     attribute customHeight { xsd:boolean }?,
2468
2469     ## default value: 0
2470     attribute outlineLevel { xsd:unsignedByte }?,
2471
2472     ## default value: false
2473     attribute collapsed { xsd:boolean }?,

```

```

2474
2475   ## default value: false
2476   attribute thickTop { xsd:boolean }?,
2477
2478   ## default value: false
2479   attribute thickBot { xsd:boolean }?,
2480
2481   ## default value: false
2482   attribute ph { xsd:boolean }?,
2483   element c { sml_CT_Cell }*,
2484   element extLst { sml_CT_ExtensionList }?
2485 sml_CT_Cell =
2486   attribute r { sml_ST_CellRef }?,
2487
2488   ## default value: 0
2489   attribute s { xsd:unsignedInt }?,
2490
2491   ## default value: n
2492   attribute t { sml_ST_CellType }?,
2493
2494   ## default value: 0
2495   attribute cm { xsd:unsignedInt }?,
2496
2497   ## default value: 0
2498   attribute vm { xsd:unsignedInt }?,
2499
2500   ## default value: false
2501   attribute ph { xsd:boolean }?,
2502   element f { sml_CT_CellFormula }?,
2503   element v { s_ST_Xstring }?,
2504   element is { sml_CT_Rst }?,
2505   element extLst { sml_CT_ExtensionList }?
2506 sml_ST_CellType =
2507   string "b"
2508   | string "d"
2509   | string "n"
2510   | string "e"
2511   | string "s"
2512   | string "str"
2513   | string "inlineStr"
2514 sml_ST_CellFormulaType =
2515   string "normal"
2516   | string "array"
2517   | string "dataTable"
2518   | string "shared"
2519 sml_CT_SheetPr =
2520
2521   ## default value: false
2522   attribute syncHorizontal { xsd:boolean }?,
2523
2524   ## default value: false
2525   attribute syncVertical { xsd:boolean }?,
2526   attribute syncRef { sml_ST_Ref }?,

```

```

2527
2528   ## default value: false
2529   attribute transitionEvaluation { xsd:boolean }?,
2530
2531   ## default value: false
2532   attribute transitionEntry { xsd:boolean }?,
2533
2534   ## default value: true
2535   attribute published { xsd:boolean }?,
2536   attribute codeName { xsd:string }?,
2537
2538   ## default value: false
2539   attribute filterMode { xsd:boolean }?,
2540
2541   ## default value: true
2542   attribute enableFormatConditionsCalculation { xsd:boolean }?,
2543   element tabColor { sml_CT_Color }?,
2544   element outlinePr { sml_CT_OutlinePr }?,
2545   element pageSetUpPr { sml_CT_PageSetUpPr }?
2546 sml_CT_SheetDimension = attribute ref { sml_ST_Ref }
2547 sml_CT_SheetViews =
2548   element sheetView { sml_CT_SheetView }+,
2549   element extLst { sml_CT_ExtensionList }?
2550 sml_CT_SheetView =
2551
2552   ## default value: false
2553   attribute windowProtection { xsd:boolean }?,
2554
2555   ## default value: false
2556   attribute showFormulas { xsd:boolean }?,
2557
2558   ## default value: true
2559   attribute showGridLines { xsd:boolean }?,
2560
2561   ## default value: true
2562   attribute showRowColHeaders { xsd:boolean }?,
2563
2564   ## default value: true
2565   attribute showZeros { xsd:boolean }?,
2566
2567   ## default value: false
2568   attribute rightToLeft { xsd:boolean }?,
2569
2570   ## default value: false
2571   attribute tabSelected { xsd:boolean }?,
2572
2573   ## default value: true
2574   attribute showRuler { xsd:boolean }?,
2575
2576   ## default value: true
2577   attribute showOutlineSymbols { xsd:boolean }?,
2578
2579   ## default value: true

```

```

2580     attribute defaultGridColor { xsd:boolean }?,
2581
2582     ## default value: true
2583     attribute showWhiteSpace { xsd:boolean }?,
2584
2585     ## default value: normal
2586     attribute view { sml_ST_SheetViewType }?,
2587     attribute topLeftCell { sml_ST_CellRef }?,
2588
2589     ## default value: 64
2590     attribute colorId { xsd:unsignedInt }?,
2591
2592     ## default value: 100
2593     attribute zoomScale { xsd:unsignedInt }?,
2594
2595     ## default value: 0
2596     attribute zoomScaleNormal { xsd:unsignedInt }?,
2597
2598     ## default value: 0
2599     attribute zoomScaleSheetLayoutView { xsd:unsignedInt }?,
2600
2601     ## default value: 0
2602     attribute zoomScalePageLayoutView { xsd:unsignedInt }?,
2603     attribute workbookViewId { xsd:unsignedInt },
2604     element pane { sml_CT_Pane }?,
2605     element selection { sml_CT_Selection }*,
2606     element pivotSelection { sml_CT_PivotSelection }*,
2607     element extLst { sml_CT_ExtensionList }?
2608 sml_CT_Pane =
2609
2610     ## default value: 0
2611     attribute xSplit { xsd:double }?,
2612
2613     ## default value: 0
2614     attribute ySplit { xsd:double }?,
2615     attribute topLeftCell { sml_ST_CellRef }?,
2616
2617     ## default value: topLeft
2618     attribute activePane { sml_ST_Pane }?,
2619
2620     ## default value: split
2621     attribute state { sml_ST_PaneState }?
2622 sml_CT_PivotSelection =
2623
2624     ## default value: topLeft
2625     attribute pane { sml_ST_Pane }?,
2626
2627     ## default value: false
2628     attribute showHeader { xsd:boolean }?,
2629
2630     ## default value: false
2631     attribute label { xsd:boolean }?,
2632

```

```

2633 ## default value: false
2634 attribute data { xsd:boolean }?,
2635
2636 ## default value: false
2637 attribute extendable { xsd:boolean }?,
2638
2639 ## default value: 0
2640 attribute count { xsd:unsignedInt }?,
2641 attribute axis { sml_ST_Axis }?,
2642
2643 ## default value: 0
2644 attribute dimension { xsd:unsignedInt }?,
2645
2646 ## default value: 0
2647 attribute start { xsd:unsignedInt }?,
2648
2649 ## default value: 0
2650 attribute min { xsd:unsignedInt }?,
2651
2652 ## default value: 0
2653 attribute max { xsd:unsignedInt }?,
2654
2655 ## default value: 0
2656 attribute activeRow { xsd:unsignedInt }?,
2657
2658 ## default value: 0
2659 attribute activeCol { xsd:unsignedInt }?,
2660
2661 ## default value: 0
2662 attribute previousRow { xsd:unsignedInt }?,
2663
2664 ## default value: 0
2665 attribute previousCol { xsd:unsignedInt }?,
2666
2667 ## default value: 0
2668 attribute click { xsd:unsignedInt }?,
2669 r_id?,
2670 element pivotArea { sml_CT_PivotArea }
2671 sml_CT_Selection =
2672
2673 ## default value: topLeft
2674 attribute pane { sml_ST_Pane }?,
2675 attribute activeCell { sml_ST_CellRef }?,
2676
2677 ## default value: 0
2678 attribute activeCellId { xsd:unsignedInt }?,
2679
2680 ## default value: A1
2681 attribute sqref { sml_ST_Sqref }?
2682 sml_ST_Pane =
2683 string "bottomRight"
2684 | string "topRight"
2685 | string "bottomLeft"

```

```

2686 | string "topLeft"
2687 sml_CT_PageBreak =
2688
2689 ## default value: 0
2690 attribute count { xsd:unsignedInt }?,
2691
2692 ## default value: 0
2693 attribute manualBreakCount { xsd:unsignedInt }?,
2694 element brk { sml_CT_Break }*
2695 sml_CT_Break =
2696
2697 ## default value: 0
2698 attribute id { xsd:unsignedInt }?,
2699
2700 ## default value: 0
2701 attribute min { xsd:unsignedInt }?,
2702
2703 ## default value: 0
2704 attribute max { xsd:unsignedInt }?,
2705
2706 ## default value: false
2707 attribute man { xsd:boolean }?,
2708
2709 ## default value: false
2710 attribute pt { xsd:boolean }?
2711 sml_ST_SheetViewType =
2712 string "normal" | string "pageBreakPreview" | string "pageLayout"
2713 sml_CT_OutlinePr =
2714
2715 ## default value: false
2716 attribute applyStyles { xsd:boolean }?,
2717
2718 ## default value: true
2719 attribute summaryBelow { xsd:boolean }?,
2720
2721 ## default value: true
2722 attribute summaryRight { xsd:boolean }?,
2723
2724 ## default value: true
2725 attribute showOutlineSymbols { xsd:boolean }?
2726 sml_CT_PageSetUpPr =
2727
2728 ## default value: true
2729 attribute autoPageBreaks { xsd:boolean }?,
2730
2731 ## default value: false
2732 attribute fitToPage { xsd:boolean }?
2733 sml_CT_DataConsolidate =
2734
2735 ## default value: sum
2736 attribute function { sml_ST_DataConsolidateFunction }?,
2737
2738 ## default value: false

```

```

2739     attribute startLabels { xsd:boolean }?,
2740
2741     ## default value: false
2742     attribute leftLabels { xsd:boolean }?,
2743
2744     ## default value: false
2745     attribute topLabels { xsd:boolean }?,
2746
2747     ## default value: false
2748     attribute link { xsd:boolean }?,
2749     element dataRefs { sml_CT_DataRefs }?
2750 sml_ST_DataConsolidateFunction =
2751     string "average"
2752     | string "count"
2753     | string "countNums"
2754     | string "max"
2755     | string "min"
2756     | string "product"
2757     | string "stdDev"
2758     | string "stdDevp"
2759     | string "sum"
2760     | string "var"
2761     | string "varp"
2762 sml_CT_DataRefs =
2763     attribute count { xsd:unsignedInt }?,
2764     element dataRef { sml_CT_DataRef }*
2765 sml_CT_DataRef =
2766     attribute ref { sml_ST_Ref }?,
2767     attribute name { s_ST_Xstring }?,
2768     attribute sheet { s_ST_Xstring }?,
2769     r_id?
2770 sml_CT_MergeCells =
2771     attribute count { xsd:unsignedInt }?,
2772     element mergeCell { sml_CT_MergeCell }+
2773 sml_CT_MergeCell = attribute ref { sml_ST_Ref }
2774 sml_CT_SmartTags = element cellSmartTags { sml_CT_CellSmartTags }+
2775 sml_CT_CellSmartTags =
2776     attribute r { sml_ST_CellRef },
2777     element cellSmartTag { sml_CT_CellSmartTag }+
2778 sml_CT_CellSmartTag =
2779     attribute type { xsd:unsignedInt },
2780
2781     ## default value: false
2782     attribute deleted { xsd:boolean }?,
2783
2784     ## default value: false
2785     attribute xmlBased { xsd:boolean }?,
2786     element cellSmartTagPr { sml_CT_CellSmartTagPr }*
2787 sml_CT_CellSmartTagPr =
2788     attribute key { s_ST_Xstring },
2789     attribute val { s_ST_Xstring }
2790 sml_CT_Drawing = r_id
2791 sml_CT_LegacyDrawing = r_id

```

```

2792 sml_CT_DrawingHF =
2793     r_id,
2794     attribute lho { xsd:unsignedInt }?,
2795     attribute lhe { xsd:unsignedInt }?,
2796     attribute lhf { xsd:unsignedInt }?,
2797     attribute cho { xsd:unsignedInt }?,
2798     attribute che { xsd:unsignedInt }?,
2799     attribute chf { xsd:unsignedInt }?,
2800     attribute rho { xsd:unsignedInt }?,
2801     attribute rhe { xsd:unsignedInt }?,
2802     attribute rhf { xsd:unsignedInt }?,
2803     attribute lfo { xsd:unsignedInt }?,
2804     attribute lfe { xsd:unsignedInt }?,
2805     attribute lff { xsd:unsignedInt }?,
2806     attribute cfo { xsd:unsignedInt }?,
2807     attribute cfe { xsd:unsignedInt }?,
2808     attribute cff { xsd:unsignedInt }?,
2809     attribute rfo { xsd:unsignedInt }?,
2810     attribute rfe { xsd:unsignedInt }?,
2811     attribute rff { xsd:unsignedInt }?
2812 sml_CT_CustomSheetViews =
2813     element customSheetView { sml_CT_CustomSheetView }+
2814 sml_CT_CustomSheetView =
2815     attribute guid { s_ST_Guid },
2816
2817     ## default value: 100
2818     attribute scale { xsd:unsignedInt }?,
2819
2820     ## default value: 64
2821     attribute colorId { xsd:unsignedInt }?,
2822
2823     ## default value: false
2824     attribute showPageBreaks { xsd:boolean }?,
2825
2826     ## default value: false
2827     attribute showFormulas { xsd:boolean }?,
2828
2829     ## default value: true
2830     attribute showGridLines { xsd:boolean }?,
2831
2832     ## default value: true
2833     attribute showRowCol { xsd:boolean }?,
2834
2835     ## default value: true
2836     attribute outlineSymbols { xsd:boolean }?,
2837
2838     ## default value: true
2839     attribute zeroValues { xsd:boolean }?,
2840
2841     ## default value: false
2842     attribute fitToPage { xsd:boolean }?,
2843
2844     ## default value: false

```



```

2845 attribute printArea { xsd:boolean }?,
2846
2847 ## default value: false
2848 attribute filter { xsd:boolean }?,
2849
2850 ## default value: false
2851 attribute showAutoFilter { xsd:boolean }?,
2852
2853 ## default value: false
2854 attribute hiddenRows { xsd:boolean }?,
2855
2856 ## default value: false
2857 attribute hiddenColumns { xsd:boolean }?,
2858
2859 ## default value: visible
2860 attribute state { sml_ST_SheetState }?,
2861
2862 ## default value: false
2863 attribute filterUnique { xsd:boolean }?,
2864
2865 ## default value: normal
2866 attribute view { sml_ST_SheetViewType }?,
2867
2868 ## default value: true
2869 attribute showRuler { xsd:boolean }?,
2870 attribute topLeftCell { sml_ST_CellRef }?,
2871 element pane { sml_CT_Pane }?,
2872 element selection { sml_CT_Selection }?,
2873 element rowBreaks { sml_CT_PageBreak }?,
2874 element colBreaks { sml_CT_PageBreak }?,
2875 element pageMargins { sml_CT_PageMargins }?,
2876 element printOptions { sml_CT_PrintOptions }?,
2877 element pageSetup { sml_CT_PageSetup }?,
2878 element headerFooter { sml_CT_HeaderFooter }?,
2879 element autoFilter { sml_CT_AutoFilter }?,
2880 element extLst { sml_CT_ExtensionList }?
2881 sml_CT_DataValidations =
2882
2883 ## default value: false
2884 attribute disablePrompts { xsd:boolean }?,
2885 attribute xWindow { xsd:unsignedInt }?,
2886 attribute yWindow { xsd:unsignedInt }?,
2887 attribute count { xsd:unsignedInt }?,
2888 element dataValidation { sml_CT_DataValidation }+
2889 sml_CT_DataValidation =
2890
2891 ## default value: none
2892 attribute type { sml_ST_DataValidationType }?,
2893
2894 ## default value: stop
2895 attribute errorStyle { sml_ST_DataValidationErrorStyle }?,
2896
2897 ## default value: noControl

```

```

2898     attribute imeMode { sml_ST_DataValidationImeMode }?,
2899
2900     ## default value: between
2901     attribute operator { sml_ST_DataValidationOperator }?,
2902
2903     ## default value: false
2904     attribute allowBlank { xsd:boolean }?,
2905
2906     ## default value: false
2907     attribute showDropDown { xsd:boolean }?,
2908
2909     ## default value: false
2910     attribute showInputMessage { xsd:boolean }?,
2911
2912     ## default value: false
2913     attribute showErrorMessage { xsd:boolean }?,
2914     attribute errorTitle { s_ST_Xstring }?,
2915     attribute error { s_ST_Xstring }?,
2916     attribute promptTitle { s_ST_Xstring }?,
2917     attribute prompt { s_ST_Xstring }?,
2918     attribute sqref { sml_ST_Sqref },
2919     element formula1 { sml_ST_Formula }?,
2920     element formula2 { sml_ST_Formula }?
2921 sml_ST_DataValidationType =
2922     string "none"
2923     | string "whole"
2924     | string "decimal"
2925     | string "list"
2926     | string "date"
2927     | string "time"
2928     | string "textLength"
2929     | string "custom"
2930 sml_ST_DataValidationOperator =
2931     string "between"
2932     | string "notBetween"
2933     | string "equal"
2934     | string "notEqual"
2935     | string "lessThan"
2936     | string "lessThanOrEqual"
2937     | string "greaterThan"
2938     | string "greaterThanOrEqual"
2939 sml_ST_DataValidationErrorStyle =
2940     string "stop" | string "warning" | string "information"
2941 sml_ST_DataValidationImeMode =
2942     string "noControl"
2943     | string "off"
2944     | string "on"
2945     | string "disabled"
2946     | string "hiragana"
2947     | string "fullKatakana"
2948     | string "halfKatakana"
2949     | string "fullAlpha"
2950     | string "halfAlpha"

```

```

2951 | string "fullHangul"
2952 | string "halfHangul"
2953 sml_ST_CfType =
2954     string "expression"
2955     | string "cellIs"
2956     | string "colorScale"
2957     | string "dataBar"
2958     | string "iconSet"
2959     | string "top10"
2960     | string "uniqueValues"
2961     | string "duplicateValues"
2962     | string "containsText"
2963     | string "notContainsText"
2964     | string "beginsWith"
2965     | string "endsWith"
2966     | string "containsBlanks"
2967     | string "notContainsBlanks"
2968     | string "containsErrors"
2969     | string "notContainsErrors"
2970     | string "timePeriod"
2971     | string "aboveAverage"
2972 sml_ST_TimePeriod =
2973     string "today"
2974     | string "yesterday"
2975     | string "tomorrow"
2976     | string "last7Days"
2977     | string "thisMonth"
2978     | string "lastMonth"
2979     | string "nextMonth"
2980     | string "thisWeek"
2981     | string "lastWeek"
2982     | string "nextWeek"
2983 sml_ST_ConditionalFormattingOperator =
2984     string "lessThan"
2985     | string "lessThanOrEqual"
2986     | string "equal"
2987     | string "notEqual"
2988     | string "greaterThanOrEqual"
2989     | string "greaterThan"
2990     | string "between"
2991     | string "notBetween"
2992     | string "containsText"
2993     | string "notContains"
2994     | string "beginsWith"
2995     | string "endsWith"
2996 sml_ST_CfvoType =
2997     string "num"
2998     | string "percent"
2999     | string "max"
3000     | string "min"
3001     | string "formula"
3002     | string "percentile"
3003 sml_CT_ConditionalFormatting =

```

```

3004
3005     ## default value: false
3006     attribute pivot { xsd:boolean }?,
3007     attribute sqref { sml_ST_Sqref }?,
3008     element cfRule { sml_CT_CfRule }+,
3009     element extLst { sml_CT_ExtensionList }?
3010 sml_CT_CfRule =
3011     attribute type { sml_ST_CfType }?,
3012     attribute dxId { sml_ST_DxId }?,
3013     attribute priority { xsd:int },
3014
3015     ## default value: false
3016     attribute stopIfTrue { xsd:boolean }?,
3017
3018     ## default value: true
3019     attribute aboveAverage { xsd:boolean }?,
3020
3021     ## default value: false
3022     attribute percent { xsd:boolean }?,
3023
3024     ## default value: false
3025     attribute bottom { xsd:boolean }?,
3026     attribute operator { sml_ST_ConditionalFormattingOperator }?,
3027     attribute text { xsd:string }?,
3028     attribute timePeriod { sml_ST_TimePeriod }?,
3029     attribute rank { xsd:unsignedInt }?,
3030     attribute stdDev { xsd:int }?,
3031
3032     ## default value: false
3033     attribute equalAverage { xsd:boolean }?,
3034     element formula { sml_ST_Formula }*,
3035     element colorScale { sml_CT_ColorScale }?,
3036     element dataBar { sml_CT_DataBar }?,
3037     element iconSet { sml_CT_IconSet }?,
3038     element extLst { sml_CT_ExtensionList }?
3039 sml_CT_Hyperlinks = element hyperlink { sml_CT_Hyperlink }+
3040 sml_CT_Hyperlink =
3041     attribute ref { sml_ST_Ref },
3042     r_id?,
3043     attribute location { s_ST_Xstring }?,
3044     attribute tooltip { s_ST_Xstring }?,
3045     attribute display { s_ST_Xstring }?
3046 sml_CT_CellFormula =
3047     sml_ST_Formula,
3048
3049     ## default value: normal
3050     attribute t { sml_ST_CellFormulaType }?,
3051
3052     ## default value: false
3053     attribute aca { xsd:boolean }?,
3054     attribute ref { sml_ST_Ref }?,
3055
3056     ## default value: false

```

```

3057 attribute dt2D { xsd:boolean }?,
3058
3059 ## default value: false
3060 attribute dtr { xsd:boolean }?,
3061
3062 ## default value: false
3063 attribute del1 { xsd:boolean }?,
3064
3065 ## default value: false
3066 attribute del2 { xsd:boolean }?,
3067 attribute r1 { sml_ST_CellRef }?,
3068 attribute r2 { sml_ST_CellRef }?,
3069
3070 ## default value: false
3071 attribute ca { xsd:boolean }?,
3072 attribute si { xsd:unsignedInt }?,
3073
3074 ## default value: false
3075 attribute bx { xsd:boolean }?
3076 sml_CT_ColorScale =
3077   element cfvo { sml_CT_Cfvo }+,
3078   element color { sml_CT_Color }+
3079 sml_CT_DataBar =
3080
3081 ## default value: 10
3082 attribute minLength { xsd:unsignedInt }?,
3083
3084 ## default value: 90
3085 attribute maxLength { xsd:unsignedInt }?,
3086
3087 ## default value: true
3088 attribute showValue { xsd:boolean }?,
3089 element cfvo { sml_CT_Cfvo }+,
3090 element color { sml_CT_Color }
3091 sml_CT_IconSet =
3092
3093 ## default value: 3TrafficLights1
3094 attribute iconSet { sml_ST_IconSetType }?,
3095
3096 ## default value: true
3097 attribute showValue { xsd:boolean }?,
3098
3099 ## default value: true
3100 attribute percent { xsd:boolean }?,
3101
3102 ## default value: false
3103 attribute reverse { xsd:boolean }?,
3104 element cfvo { sml_CT_Cfvo }+
3105 sml_CT_Cfvo =
3106   attribute type { sml_ST_CfvoType },
3107   attribute val { s_ST_Xstring }?,
3108
3109 ## default value: true

```

```

3110     attribute gte { xsd:boolean }?,
3111     element extLst { sml_CT_ExtensionList }?
3112 sml_CT_PageMargins =
3113     attribute left { xsd:double },
3114     attribute right { xsd:double },
3115     attribute top { xsd:double },
3116     attribute bottom { xsd:double },
3117     attribute header { xsd:double },
3118     attribute footer { xsd:double }
3119 sml_CT_PrintOptions =
3120
3121     ## default value: false
3122     attribute horizontalCentered { xsd:boolean }?,
3123
3124     ## default value: false
3125     attribute verticalCentered { xsd:boolean }?,
3126
3127     ## default value: false
3128     attribute headings { xsd:boolean }?,
3129
3130     ## default value: false
3131     attribute gridLines { xsd:boolean }?,
3132
3133     ## default value: true
3134     attribute gridLinesSet { xsd:boolean }?
3135 sml_CT_PageSetup =
3136
3137     ## default value: 1
3138     attribute paperSize { xsd:unsignedInt }?,
3139     attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
3140     attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
3141
3142     ## default value: 100
3143     attribute scale { xsd:unsignedInt }?,
3144
3145     ## default value: 1
3146     attribute firstPageNumber { xsd:unsignedInt }?,
3147
3148     ## default value: 1
3149     attribute fitToWidth { xsd:unsignedInt }?,
3150
3151     ## default value: 1
3152     attribute fitToHeight { xsd:unsignedInt }?,
3153
3154     ## default value: downThenOver
3155     attribute pageOrder { sml_ST_PageOrder }?,
3156
3157     ## default value: default
3158     attribute orientation { sml_ST_Orientation }?,
3159
3160     ## default value: true
3161     attribute usePrinterDefaults { xsd:boolean }?,
3162

```

```

3163 ## default value: false
3164 attribute blackAndWhite { xsd:boolean }?,
3165
3166 ## default value: false
3167 attribute draft { xsd:boolean }?,
3168
3169 ## default value: none
3170 attribute cellComments { sml_ST_CellComments }?,
3171
3172 ## default value: false
3173 attribute useFirstPageNumber { xsd:boolean }?,
3174
3175 ## default value: displayed
3176 attribute errors { sml_ST_PrintError }?,
3177
3178 ## default value: 600
3179 attribute horizontalDpi { xsd:unsignedInt }?,
3180
3181 ## default value: 600
3182 attribute verticalDpi { xsd:unsignedInt }?,
3183
3184 ## default value: 1
3185 attribute copies { xsd:unsignedInt }?,
3186 r_id?
3187 sml_ST_PageOrder = string "downThenOver" | string "overThenDown"
3188 sml_ST_Orientation =
3189     string "default" | string "portrait" | string "landscape"
3190 sml_ST_CellComments =
3191     string "none" | string "asDisplayed" | string "atEnd"
3192 sml_CT_HeaderFooter =
3193
3194 ## default value: false
3195 attribute differentOddEven { xsd:boolean }?,
3196
3197 ## default value: false
3198 attribute differentFirst { xsd:boolean }?,
3199
3200 ## default value: true
3201 attribute scaleWithDoc { xsd:boolean }?,
3202
3203 ## default value: true
3204 attribute alignWithMargins { xsd:boolean }?,
3205 element oddHeader { s_ST_Xstring }?,
3206 element oddFooter { s_ST_Xstring }?,
3207 element evenHeader { s_ST_Xstring }?,
3208 element evenFooter { s_ST_Xstring }?,
3209 element firstHeader { s_ST_Xstring }?,
3210 element firstFooter { s_ST_Xstring }?
3211 sml_ST_PrintError =
3212     string "displayed" | string "blank" | string "dash" | string "NA"
3213 sml_CT_Scenarios =
3214     attribute current { xsd:unsignedInt }?,
3215     attribute show { xsd:unsignedInt }?,

```

```

3216     attribute sqref { sml_ST_Sqref }?,
3217     element scenario { sml_CT_Scenario }+
3218 sml_CT_SheetProtection =
3219     attribute password { sml_ST_UnsignedShortHex }?,
3220     attribute algorithmName { s_ST_Xstring }?,
3221     attribute hashValue { xsd:base64Binary }?,
3222     attribute saltValue { xsd:base64Binary }?,
3223     attribute spinCount { xsd:unsignedInt }?,
3224
3225     ## default value: false
3226     attribute sheet { xsd:boolean }?,
3227
3228     ## default value: false
3229     attribute objects { xsd:boolean }?,
3230
3231     ## default value: false
3232     attribute scenarios { xsd:boolean }?,
3233
3234     ## default value: true
3235     attribute formatCells { xsd:boolean }?,
3236
3237     ## default value: true
3238     attribute formatColumns { xsd:boolean }?,
3239
3240     ## default value: true
3241     attribute formatRows { xsd:boolean }?,
3242
3243     ## default value: true
3244     attribute insertColumns { xsd:boolean }?,
3245
3246     ## default value: true
3247     attribute insertRows { xsd:boolean }?,
3248
3249     ## default value: true
3250     attribute insertHyperlinks { xsd:boolean }?,
3251
3252     ## default value: true
3253     attribute deleteColumns { xsd:boolean }?,
3254
3255     ## default value: true
3256     attribute deleteRows { xsd:boolean }?,
3257
3258     ## default value: false
3259     attribute selectLockedCells { xsd:boolean }?,
3260
3261     ## default value: true
3262     attribute sort { xsd:boolean }?,
3263
3264     ## default value: true
3265     attribute autoFilter { xsd:boolean }?,
3266
3267     ## default value: true
3268     attribute pivotTables { xsd:boolean }?,

```



```

3269
3270   ## default value: false
3271   attribute selectUnlockedCells { xsd:boolean }?
3272 sml_CT_ProtectedRanges =
3273   element protectedRange { sml_CT_ProtectedRange }+
3274 sml_CT_ProtectedRange =
3275   attribute password { sml_ST_UnsignedShortHex }?,
3276   attribute sqref { sml_ST_Sqref },
3277   attribute name { s_ST_Xstring },
3278   attribute securityDescriptor { xsd:string }?,
3279   attribute algorithmName { s_ST_Xstring }?,
3280   attribute hashValue { xsd:base64Binary }?,
3281   attribute saltValue { xsd:base64Binary }?,
3282   attribute spinCount { xsd:unsignedInt }?,
3283   element securityDescriptor { xsd:string }*
3284 sml_CT_Scenario =
3285   attribute name { s_ST_Xstring },
3286
3287   ## default value: false
3288   attribute locked { xsd:boolean }?,
3289
3290   ## default value: false
3291   attribute hidden { xsd:boolean }?,
3292   attribute count { xsd:unsignedInt }?,
3293   attribute user { s_ST_Xstring }?,
3294   attribute comment { s_ST_Xstring }?,
3295   element inputCells { sml_CT_InputCells }+
3296 sml_CT_InputCells =
3297   attribute r { sml_ST_CellRef },
3298
3299   ## default value: false
3300   attribute deleted { xsd:boolean }?,
3301
3302   ## default value: false
3303   attribute undone { xsd:boolean }?,
3304   attribute val { s_ST_Xstring },
3305   attribute numFmtId { sml_ST_NumFmtId }?
3306 sml_CT_CellWatches = element cellWatch { sml_CT_CellWatch }+
3307 sml_CT_CellWatch = attribute r { sml_ST_CellRef }
3308 sml_CT_Chartsheet =
3309   element sheetPr { sml_CT_ChartsheetPr }?,
3310   element sheetViews { sml_CT_ChartsheetViews },
3311   element sheetProtection { sml_CT_ChartsheetProtection }?,
3312   element customSheetViews { sml_CT_CustomChartsheetViews }?,
3313   element pageMargins { sml_CT_PageMargins }?,
3314   element pageSetup { sml_CT-CsPageSetup }?,
3315   element headerFooter { sml_CT_HeaderFooter }?,
3316   element drawing { sml_CT_Drawing },
3317   element legacyDrawing { sml_CT_LegacyDrawing }?,
3318   element legacyDrawingHF { sml_CT_LegacyDrawing }?,
3319   element drawingHF { sml_CT_DrawingHF }?,
3320   element picture { sml_CT_SheetBackgroundPicture }?,
3321   element webPublishItems { sml_CT_WebPublishItems }?,

```

```

3322     element extLst { sml_CT_ExtensionList }?
3323 sml_CT_ChartsheetPr =
3324
3325     ## default value: true
3326     attribute published { xsd:boolean }?,
3327     attribute codeName { xsd:string }?,
3328     element tabColor { sml_CT_Color }?
3329 sml_CT_ChartsheetViews =
3330     element sheetView { sml_CT_ChartsheetView }+,
3331     element extLst { sml_CT_ExtensionList }?
3332 sml_CT_ChartsheetView =
3333
3334     ## default value: false
3335     attribute tabSelected { xsd:boolean }?,
3336
3337     ## default value: 100
3338     attribute zoomScale { xsd:unsignedInt }?,
3339     attribute workbookViewId { xsd:unsignedInt },
3340
3341     ## default value: false
3342     attribute zoomToFit { xsd:boolean }?,
3343     element extLst { sml_CT_ExtensionList }?
3344 sml_CT_ChartsheetProtection =
3345     attribute password { sml_ST_UnsignedShortHex }?,
3346     attribute algorithmName { s_ST_Xstring }?,
3347     attribute hashValue { xsd:base64Binary }?,
3348     attribute saltValue { xsd:base64Binary }?,
3349     attribute spinCount { xsd:unsignedInt }?,
3350
3351     ## default value: false
3352     attribute content { xsd:boolean }?,
3353
3354     ## default value: false
3355     attribute objects { xsd:boolean }?
3356 sml_CT-CsPageSetup =
3357
3358     ## default value: 1
3359     attribute paperSize { xsd:unsignedInt }?,
3360     attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
3361     attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
3362
3363     ## default value: 1
3364     attribute firstPageNumber { xsd:unsignedInt }?,
3365
3366     ## default value: default
3367     attribute orientation { sml_ST_Orientation }?,
3368
3369     ## default value: true
3370     attribute usePrinterDefaults { xsd:boolean }?,
3371
3372     ## default value: false
3373     attribute blackAndWhite { xsd:boolean }?,
3374

```

```

3375 ## default value: false
3376 attribute draft { xsd:boolean }?,
3377
3378 ## default value: false
3379 attribute useFirstPageNumber { xsd:boolean }?,
3380
3381 ## default value: 600
3382 attribute horizontalDpi { xsd:unsignedInt }?,
3383
3384 ## default value: 600
3385 attribute verticalDpi { xsd:unsignedInt }?,
3386
3387 ## default value: 1
3388 attribute copies { xsd:unsignedInt }?,
3389 r_id?
3390 sml_CT_CustomChartsheetViews =
3391   element customSheetView { sml_CT_CustomChartsheetView }*
3392 sml_CT_CustomChartsheetView =
3393   attribute guid { s_ST_Guid },
3394
3395 ## default value: 100
3396 attribute scale { xsd:unsignedInt }?,
3397
3398 ## default value: visible
3399 attribute state { sml_ST_SheetState }?,
3400
3401 ## default value: false
3402 attribute zoomToFit { xsd:boolean }?,
3403 element pageMargins { sml_CT_PageMargins }?,
3404 element pageSetup { sml_CT-CsPageSetup }?,
3405 element headerFooter { sml_CT_HeaderFooter }?
3406 sml_CT_CustomProperties = element customPr { sml_CT_CustomProperty }+
3407 sml_CT_CustomProperty =
3408   attribute name { s_ST_Xstring },
3409   r_id
3410 sml_CT_OleObjects = element oleObject { sml_CT_OleObject }+
3411 sml_CT_OleObject =
3412   attribute progId { xsd:string }?,
3413
3414 ## default value: DVASPECT_CONTENT
3415 attribute dvAspect { sml_ST_DvAspect }?,
3416 attribute link { s_ST_Xstring }?,
3417 attribute oleUpdate { sml_ST_OleUpdate }?,
3418
3419 ## default value: false
3420 attribute autoLoad { xsd:boolean }?,
3421 attribute shapeId { xsd:unsignedInt },
3422   r_id?,
3423   element objectPr { sml_CT_ObjectPr }?
3424 sml_CT_ObjectPr =
3425
3426 ## default value: true
3427 attribute locked { xsd:boolean }?,

```

```

3428
3429   ## default value: true
3430   attribute defaultSize { xsd:boolean }?,
3431
3432   ## default value: true
3433   attribute print { xsd:boolean }?,
3434
3435   ## default value: false
3436   attribute disabled { xsd:boolean }?,
3437
3438   ## default value: false
3439   attribute uiObject { xsd:boolean }?,
3440
3441   ## default value: true
3442   attribute autoFill { xsd:boolean }?,
3443
3444   ## default value: true
3445   attribute autoLine { xsd:boolean }?,
3446
3447   ## default value: true
3448   attribute autoPict { xsd:boolean }?,
3449   attribute macro { sml_ST_Formula }?,
3450   attribute altText { s_ST_Xstring }?,
3451
3452   ## default value: false
3453   attribute dde { xsd:boolean }?,
3454   r_id?,
3455   element anchor { sml_CT_ObjectAnchor }
3456 sml_ST_DvAspect = string "DVASPECT_CONTENT" | string "DVASPECT_ICON"
3457 sml_ST_OleUpdate = string "OLEUPDATE_ALWAYS" | string "OLEUPDATE_ONCALL"
3458 sml_CT_WebPublishItems =
3459     attribute count { xsd:unsignedInt }?,
3460     element webPublishItem { sml_CT_WebPublishItem }+
3461 sml_CT_WebPublishItem =
3462     attribute id { xsd:unsignedInt },
3463     attribute divId { s_ST_Xstring },
3464     attribute sourceType { sml_ST_WebSourceType },
3465     attribute sourceRef { sml_ST_Ref }?,
3466     attribute sourceObject { s_ST_Xstring }?,
3467     attribute destinationFile { s_ST_Xstring },
3468     attribute title { s_ST_Xstring }?,
3469
3470   ## default value: false
3471   attribute autoRepublish { xsd:boolean }?
3472 sml_CT_Controls = element control { sml_CT_Control }+
3473 sml_CT_Control =
3474     attribute shapeId { xsd:unsignedInt },
3475     r_id,
3476     attribute name { xsd:string }?,
3477     element controlPr { sml_CT_ControlPr }?
3478 sml_CT_ControlPr =
3479
3480   ## default value: true

```

```

3481 attribute locked { xsd:boolean }?,
3482
3483 ## default value: true
3484 attribute defaultSize { xsd:boolean }?,
3485
3486 ## default value: true
3487 attribute print { xsd:boolean }?,
3488
3489 ## default value: false
3490 attribute disabled { xsd:boolean }?,
3491
3492 ## default value: false
3493 attribute recalcAlways { xsd:boolean }?,
3494
3495 ## default value: false
3496 attribute uiObject { xsd:boolean }?,
3497
3498 ## default value: true
3499 attribute autoFill { xsd:boolean }?,
3500
3501 ## default value: true
3502 attribute autoLine { xsd:boolean }?,
3503
3504 ## default value: true
3505 attribute autoPict { xsd:boolean }?,
3506 attribute macro { sml_ST_Formula }?,
3507 attribute altText { s_ST_Xstring }?,
3508 attribute linkedCell { sml_ST_Formula }?,
3509 attribute listFillRange { sml_ST_Formula }?,
3510
3511 ## default value: pict
3512 attribute cf { s_ST_Xstring }?,
3513 r_id?,
3514 element anchor { sml_CT_ObjectAnchor }
3515 sml_ST_WebSourceType =
3516     string "sheet"
3517     | string "printArea"
3518     | string "autoFilter"
3519     | string "range"
3520     | string "chart"
3521     | string "pivotTable"
3522     | string "query"
3523     | string "label"
3524 sml_CT_IgnoredErrors =
3525     element ignoredError { sml_CT_IgnoredError }+,
3526     element extLst { sml_CT_ExtensionList }?
3527 sml_CT_IgnoredError =
3528     attribute sqref { sml_ST_Sqref },
3529
3530 ## default value: false
3531 attribute evalError { xsd:boolean }?,
3532
3533 ## default value: false

```

```

3534     attribute twoDigitTextYear { xsd:boolean }?,
3535
3536     ## default value: false
3537     attribute numberStoredAsText { xsd:boolean }?,
3538
3539     ## default value: false
3540     attribute formula { xsd:boolean }?,
3541
3542     ## default value: false
3543     attribute formulaRange { xsd:boolean }?,
3544
3545     ## default value: false
3546     attribute unlockedFormula { xsd:boolean }?,
3547
3548     ## default value: false
3549     attribute emptyCellReference { xsd:boolean }?,
3550
3551     ## default value: false
3552     attribute listDataValidation { xsd:boolean }?,
3553
3554     ## default value: false
3555     attribute calculatedColumn { xsd:boolean }?
3556 sml_ST_PaneState =
3557     string "split" | string "frozen" | string "frozenSplit"
3558 sml_CT_TableParts =
3559     attribute count { xsd:unsignedInt }?,
3560     element tablePart { sml_CT_TablePart }*
3561 sml_CT_TablePart = r_id
3562 sml_metadata = element metadata { sml_CT_Metadata }
3563 sml_CT_Metadata =
3564     element metadataTypes { sml_CT_MetadataTypes }?,
3565     element metadataStrings { sml_CT_MetadataStrings }?,
3566     element mdxMetadata { sml_CT_MdxMetadata }?,
3567     element futureMetadata { sml_CT_FutureMetadata }*,
3568     element cellMetadata { sml_CT_MetadataBlocks }?,
3569     element valueMetadata { sml_CT_MetadataBlocks }?,
3570     element extLst { sml_CT_ExtensionList }?
3571 sml_CT_MetadataTypes =
3572
3573     ## default value: 0
3574     attribute count { xsd:unsignedInt }?,
3575     element metadataType { sml_CT_MetadataType }+
3576 sml_CT_MetadataType =
3577     attribute name { s_ST_Xstring },
3578     attribute minSupportedVersion { xsd:unsignedInt },
3579
3580     ## default value: false
3581     attribute ghostRow { xsd:boolean }?,
3582
3583     ## default value: false
3584     attribute ghostCol { xsd:boolean }?,
3585
3586     ## default value: false

```

```

3587 attribute edit { xsd:boolean }?,
3588
3589 ## default value: false
3590 attribute delete { xsd:boolean }?,
3591
3592 ## default value: false
3593 attribute copy { xsd:boolean }?,
3594
3595 ## default value: false
3596 attribute pasteAll { xsd:boolean }?,
3597
3598 ## default value: false
3599 attribute pasteFormulas { xsd:boolean }?,
3600
3601 ## default value: false
3602 attribute pasteValues { xsd:boolean }?,
3603
3604 ## default value: false
3605 attribute pasteFormats { xsd:boolean }?,
3606
3607 ## default value: false
3608 attribute pasteComments { xsd:boolean }?,
3609
3610 ## default value: false
3611 attribute pasteDataValidation { xsd:boolean }?,
3612
3613 ## default value: false
3614 attribute pasteBorders { xsd:boolean }?,
3615
3616 ## default value: false
3617 attribute pasteColWidths { xsd:boolean }?,
3618
3619 ## default value: false
3620 attribute pasteNumberFormats { xsd:boolean }?,
3621
3622 ## default value: false
3623 attribute merge { xsd:boolean }?,
3624
3625 ## default value: false
3626 attribute splitFirst { xsd:boolean }?,
3627
3628 ## default value: false
3629 attribute splitAll { xsd:boolean }?,
3630
3631 ## default value: false
3632 attribute rowColShift { xsd:boolean }?,
3633
3634 ## default value: false
3635 attribute clearAll { xsd:boolean }?,
3636
3637 ## default value: false
3638 attribute clearFormats { xsd:boolean }?,
3639

```

```

3640  ## default value: false
3641  attribute clearContents { xsd:boolean }?,
3642
3643  ## default value: false
3644  attribute clearComments { xsd:boolean }?,
3645
3646  ## default value: false
3647  attribute assign { xsd:boolean }?,
3648
3649  ## default value: false
3650  attribute coerce { xsd:boolean }?,
3651
3652  ## default value: false
3653  attribute adjust { xsd:boolean }?,
3654
3655  ## default value: false
3656  attribute cellMeta { xsd:boolean }?
3657 sml_CT_MetadataBlocks =
3658
3659  ## default value: 0
3660  attribute count { xsd:unsignedInt }?,
3661  element bk { sml_CT_MetadataBlock }+
3662 sml_CT_MetadataBlock = element rc { sml_CT_MetadataRecord }+
3663 sml_CT_MetadataRecord =
3664   attribute t { xsd:unsignedInt },
3665   attribute v { xsd:unsignedInt }
3666 sml_CT_FutureMetadata =
3667   attribute name { s_ST_Xstring },
3668
3669  ## default value: 0
3670  attribute count { xsd:unsignedInt }?,
3671  element bk { sml_CT_FutureMetadataBlock }*,
3672  element extLst { sml_CT_ExtensionList }?
3673 sml_CT_FutureMetadataBlock = element extLst { sml_CT_ExtensionList }?
3674 sml_CT_MdxMetadata =
3675
3676  ## default value: 0
3677  attribute count { xsd:unsignedInt }?,
3678  element mdx { sml_CT_Mdx }+
3679 sml_CT_Mdx =
3680   attribute n { xsd:unsignedInt },
3681   attribute f { sml_ST_MdxFunctionType },
3682   (element t { sml_CT_MdxTuple }
3683    | element ms { sml_CT_MdxSet }
3684    | element p { sml_CT_MdxMemeberProp }
3685    | element k { sml_CT_MdxKPI })
3686 sml_ST_MdxFunctionType =
3687   string "m"
3688   | string "v"
3689   | string "s"
3690   | string "c"
3691   | string "r"
3692   | string "p"

```



```

3693 | string "k"
3694 sml_CT_MdxTuple =
3695
3696 ## default value: 0
3697 attribute c { xsd:unsignedInt }?,
3698 attribute ct { s_ST_Xstring }?,
3699 attribute si { xsd:unsignedInt }?,
3700 attribute fi { xsd:unsignedInt }?,
3701 attribute bc { sml_ST_UnsignedIntHex }?,
3702 attribute fc { sml_ST_UnsignedIntHex }?,
3703
3704 ## default value: false
3705 attribute i { xsd:boolean }?,
3706
3707 ## default value: false
3708 attribute u { xsd:boolean }?,
3709
3710 ## default value: false
3711 attribute st { xsd:boolean }?,
3712
3713 ## default value: false
3714 attribute b { xsd:boolean }?,
3715 element n { sml_CT_MetadataStringIndex }*
3716 sml_CT_MdxSet =
3717 attribute ns { xsd:unsignedInt },
3718
3719 ## default value: 0
3720 attribute c { xsd:unsignedInt }?,
3721
3722 ## default value: u
3723 attribute o { sml_ST_MdxSetOrder }?,
3724 element n { sml_CT_MetadataStringIndex }*
3725 sml_ST_MdxSetOrder =
3726 string "u"
3727 | string "a"
3728 | string "d"
3729 | string "aa"
3730 | string "ad"
3731 | string "na"
3732 | string "nd"
3733 sml_CT_MdxMemeberProp =
3734 attribute n { xsd:unsignedInt },
3735 attribute np { xsd:unsignedInt }
3736 sml_CT_MdxKPI =
3737 attribute n { xsd:unsignedInt },
3738 attribute np { xsd:unsignedInt },
3739 attribute p { sml_ST_MdxKPIProperty }
3740 sml_ST_MdxKPIProperty =
3741 string "v"
3742 | string "g"
3743 | string "s"
3744 | string "t"
3745 | string "w"

```

```

3746 | string "m"
3747 sml_CT_MetadadataStringIndex =
3748   attribute x { xsd:unsignedInt },
3749
3750   ## default value: false
3751   attribute s { xsd:boolean }?
3752 sml_CT_MetadadataStrings =
3753
3754   ## default value: 0
3755   attribute count { xsd:unsignedInt }?,
3756   element s { sml_CT_XStringElement }+
3757 sml_singleXmlCells = element singleXmlCells { sml_CT_SingleXmlCells }
3758 sml_CT_SingleXmlCells = element singleXmlCell { sml_CT_SingleXmlCell }+
3759 sml_CT_SingleXmlCell =
3760   attribute id { xsd:unsignedInt },
3761   attribute r { sml_ST_CellRef },
3762   attribute connectionId { xsd:unsignedInt },
3763   element xmlCellPr { sml_CT_XmlCellPr },
3764   element extLst { sml_CT_ExtensionList }?
3765 sml_CT_XmlCellPr =
3766   attribute id { xsd:unsignedInt },
3767   attribute uniqueName { s_ST_Xstring }?,
3768   element xmlPr { sml_CT_XmlPr },
3769   element extLst { sml_CT_ExtensionList }?
3770 sml_CT_XmlPr =
3771   attribute mapId { xsd:unsignedInt },
3772   attribute xpath { s_ST_Xstring },
3773   attribute xmlDataType { sml_ST_XmlDataType },
3774   element extLst { sml_CT_ExtensionList }?
3775 sml_styleSheet = element styleSheet { sml_CT_Stylesheet }
3776 sml_CT_Stylesheet =
3777   element numFmts { sml_CT_NumFmts }?,
3778   element fonts { sml_CT_Fonts }?,
3779   element fills { sml_CT_Fills }?,
3780   element borders { sml_CT_Borders }?,
3781   element cellStyleXfs { sml_CT_CellStyleXfs }?,
3782   element cellXfs { sml_CT_CellXfs }?,
3783   element cellStyles { sml_CT_CellStyles }?,
3784   element dxfs { sml_CT_Dxfs }?,
3785   element tableStyles { sml_CT_TableStyles }?,
3786   element colors { sml_CT_Colors }?,
3787   element extLst { sml_CT_ExtensionList }?
3788 sml_CT_CellAlignment =
3789   attribute horizontal { sml_ST_HorizontalAlignment }?,
3790   attribute vertical { sml_ST_VerticalAlignment }?,
3791   attribute textRotation { xsd:unsignedInt }?,
3792   attribute wrapText { xsd:boolean }?,
3793   attribute indent { xsd:unsignedInt }?,
3794   attribute relativeIndent { xsd:int }?,
3795   attribute justifyLastLine { xsd:boolean }?,
3796   attribute shrinkToFit { xsd:boolean }?,
3797   attribute readingOrder { xsd:unsignedInt }?
3798 sml_ST_BorderStyle =

```

```

3799     string "none"
3800     | string "thin"
3801     | string "medium"
3802     | string "dashed"
3803     | string "dotted"
3804     | string "thick"
3805     | string "double"
3806     | string "hair"
3807     | string "mediumDashed"
3808     | string "dashDot"
3809     | string "mediumDashDot"
3810     | string "dashDotDot"
3811     | string "mediumDashDotDot"
3812     | string "slantDashDot"
3813 sml_CT_Borders =
3814     attribute count { xsd:unsignedInt }?,
3815     element border { sml_CT_Border }*
3816 sml_CT_Border =
3817     attribute diagonalUp { xsd:boolean }?,
3818     attribute diagonalDown { xsd:boolean }?,
3819
3820     ## default value: true
3821     attribute outline { xsd:boolean }?,
3822     element start { sml_CT_BorderPr }?,
3823     element end { sml_CT_BorderPr }?,
3824     element left { sml_CT_BorderPr }?,
3825     element right { sml_CT_BorderPr }?,
3826     element top { sml_CT_BorderPr }?,
3827     element bottom { sml_CT_BorderPr }?,
3828     element diagonal { sml_CT_BorderPr }?,
3829     element vertical { sml_CT_BorderPr }?,
3830     element horizontal { sml_CT_BorderPr }?
3831 sml_CT_BorderPr =
3832
3833     ## default value: none
3834     attribute style { sml_ST_BorderStyle }?,
3835     element color { sml_CT_Color }?
3836 sml_CT_CellProtection =
3837     attribute locked { xsd:boolean }?,
3838     attribute hidden { xsd:boolean }?
3839 sml_CT_Fonts =
3840     attribute count { xsd:unsignedInt }?,
3841     element font { sml_CT_Font }*
3842 sml_CT_Fills =
3843     attribute count { xsd:unsignedInt }?,
3844     element fill { sml_CT_Fill }*
3845 sml_CT_Fill =
3846     element patternFill { sml_CT_PatternFill }?
3847     | element gradientFill { sml_CT_GradientFill }?
3848 sml_CT_PatternFill =
3849     attribute patternType { sml_ST_PatternType }?,
3850     element fgColor { sml_CT_Color }?,
3851     element bgColor { sml_CT_Color }?

```

```

3852 sml_CT_Color =
3853     attribute auto { xsd:boolean }?,
3854     attribute indexed { xsd:unsignedInt }?,
3855     attribute rgb { sml_ST_UnsignedIntHex }?,
3856     attribute theme { xsd:unsignedInt }?,
3857
3858     ## default value: 0.0
3859     attribute tint { xsd:double }?
3860 sml_ST_PatternType =
3861     string "none"
3862     | string "solid"
3863     | string "mediumGray"
3864     | string "darkGray"
3865     | string "lightGray"
3866     | string "darkHorizontal"
3867     | string "darkVertical"
3868     | string "darkDown"
3869     | string "darkUp"
3870     | string "darkGrid"
3871     | string "darkTrellis"
3872     | string "lightHorizontal"
3873     | string "lightVertical"
3874     | string "lightDown"
3875     | string "lightUp"
3876     | string "lightGrid"
3877     | string "lightTrellis"
3878     | string "gray125"
3879     | string "gray0625"
3880 sml_CT_GradientFill =
3881
3882     ## default value: linear
3883     attribute type { sml_ST_GradientType }?,
3884
3885     ## default value: 0
3886     attribute degree { xsd:double }?,
3887
3888     ## default value: 0
3889     attribute left { xsd:double }?,
3890
3891     ## default value: 0
3892     attribute right { xsd:double }?,
3893
3894     ## default value: 0
3895     attribute top { xsd:double }?,
3896
3897     ## default value: 0
3898     attribute bottom { xsd:double }?,
3899     element stop { sml_CT_GradientStop }*
3900 sml_CT_GradientStop =
3901     attribute position { xsd:double },
3902     element color { sml_CT_Color }
3903 sml_ST_GradientType = string "linear" | string "path"
3904 sml_ST_HorizontalAlignment =

```

```

3905     string "general"
3906     | string "left"
3907     | string "center"
3908     | string "right"
3909     | string "fill"
3910     | string "justify"
3911     | string "centerContinuous"
3912     | string "distributed"
3913 sml_ST_VerticalAlignment =
3914     string "top"
3915     | string "center"
3916     | string "bottom"
3917     | string "justify"
3918     | string "distributed"
3919 sml_CT_NumFmts =
3920     attribute count { xsd:unsignedInt }?,
3921     element numFmt { sml_CT_NumFmt }*
3922 sml_CT_NumFmt =
3923     attribute numFmtId { sml_ST_NumFmtId },
3924     attribute formatCode { s_ST_Xstring }
3925 sml_CT_CellStyleXfs =
3926     attribute count { xsd:unsignedInt }?,
3927     element xf { sml_CT_Xf }+
3928 sml_CT_CellXfs =
3929     attribute count { xsd:unsignedInt }?,
3930     element xf { sml_CT_Xf }+
3931 sml_CT_Xf =
3932     attribute numFmtId { sml_ST_NumFmtId }?,
3933     attribute fontId { sml_ST_FontId }?,
3934     attribute fillId { sml_ST_FillId }?,
3935     attribute borderId { sml_ST_BorderId }?,
3936     attribute xfId { sml_ST_CellStyleXfId }?,
3937
3938     ## default value: false
3939     attribute quotePrefix { xsd:boolean }?,
3940
3941     ## default value: false
3942     attribute pivotButton { xsd:boolean }?,
3943     attribute applyNumberFormat { xsd:boolean }?,
3944     attribute applyFont { xsd:boolean }?,
3945     attribute applyFill { xsd:boolean }?,
3946     attribute applyBorder { xsd:boolean }?,
3947     attribute applyAlignment { xsd:boolean }?,
3948     attribute applyProtection { xsd:boolean }?,
3949     element alignment { sml_CT_CellAlignment }?,
3950     element protection { sml_CT_CellProtection }?,
3951     element extLst { sml_CT_ExtensionList }?
3952 sml_CT_CellStyles =
3953     attribute count { xsd:unsignedInt }?,
3954     element cellStyle { sml_CT_CellStyle }+
3955 sml_CT_CellStyle =
3956     attribute name { s_ST_Xstring }?,
3957     attribute xfId { sml_ST_CellStyleXfId },

```

```

3958     attribute builtinId { xsd:unsignedInt }?,
3959     attribute iLevel { xsd:unsignedInt }?,
3960     attribute hidden { xsd:boolean }?,
3961     attribute customBuiltin { xsd:boolean }?,
3962     element extLst { sml_CT_ExtensionList }?
3963 sml_CT_Dxfs =
3964     attribute count { xsd:unsignedInt }?,
3965     element dxf { sml_CT_Dxf }*
3966 sml_CT_Dxf =
3967     element font { sml_CT_Font }?,
3968     element numFmt { sml_CT_NumFmt }?,
3969     element fill { sml_CT_Fill }?,
3970     element alignment { sml_CT_CellAlignment }?,
3971     element border { sml_CT_Border }?,
3972     element protection { sml_CT_CellProtection }?,
3973     element extLst { sml_CT_ExtensionList }?
3974 sml_ST_NumFmtId = xsd:unsignedInt
3975 sml_ST_FontId = xsd:unsignedInt
3976 sml_ST_FillId = xsd:unsignedInt
3977 sml_ST_BorderId = xsd:unsignedInt
3978 sml_ST_CellStyleXfId = xsd:unsignedInt
3979 sml_ST_DxfId = xsd:unsignedInt
3980 sml_CT_Colors =
3981     element indexedColors { sml_CT_IndexedColors }?,
3982     element mruColors { sml_CT_MRUColors }?
3983 sml_CT_IndexedColors = element rgbColor { sml_CT_RgbColor }+
3984 sml_CT_MRUColors = element color { sml_CT_Color }+
3985 sml_CT_RgbColor = attribute rgb { sml_ST_UnsignedIntHex }?
3986 sml_CT_TableStyles =
3987     attribute count { xsd:unsignedInt }?,
3988     attribute defaultTableStyle { xsd:string }?,
3989     attribute defaultPivotStyle { xsd:string }?,
3990     element tableStyle { sml_CT_TableStyle }*
3991 sml_CT_TableStyle =
3992     attribute name { xsd:string },
3993
3994     ## default value: true
3995     attribute pivot { xsd:boolean }?,
3996
3997     ## default value: true
3998     attribute table { xsd:boolean }?,
3999     attribute count { xsd:unsignedInt }?,
4000     element tableStyleElement { sml_CT_TableStyleElement }*
4001 sml_CT_TableStyleElement =
4002     attribute type { sml_ST_TableStyleType },
4003
4004     ## default value: 1
4005     attribute size { xsd:unsignedInt }?,
4006     attribute dxfId { sml_ST_DxfId }?
4007 sml_ST_TableStyleType =
4008     string "wholeTable"
4009     | string "headerRow"
4010     | string "totalRow"

```

```

4011 | string "firstColumn"
4012 | string "lastColumn"
4013 | string "firstRowStripe"
4014 | string "secondRowStripe"
4015 | string "firstColumnStripe"
4016 | string "secondColumnStripe"
4017 | string "firstHeaderCell"
4018 | string "lastHeaderCell"
4019 | string "firstTotalCell"
4020 | string "lastTotalCell"
4021 | string "firstSubtotalColumn"
4022 | string "secondSubtotalColumn"
4023 | string "thirdSubtotalColumn"
4024 | string "firstSubtotalRow"
4025 | string "secondSubtotalRow"
4026 | string "thirdSubtotalRow"
4027 | string "blankRow"
4028 | string "firstColumnSubheading"
4029 | string "secondColumnSubheading"
4030 | string "thirdColumnSubheading"
4031 | string "firstRowSubheading"
4032 | string "secondRowSubheading"
4033 | string "thirdRowSubheading"
4034 | string "pageFieldLabels"
4035 | string "pageFieldValues"
4036 sml_CT_BooleanProperty =
4037
4038     ## default value: true
4039     attribute val { xsd:boolean }?
4040 sml_CT_FontSize = attribute val { xsd:double }
4041 sml_CT_IntProperty = attribute val { xsd:int }
4042 sml_CT_FontName = attribute val { s_ST_Xstring }
4043 sml_CT_VerticalAlignFontProperty =
4044     attribute val { s_ST_VerticalAlignRun }
4045 sml_CT_FontScheme = attribute val { sml_ST_FontScheme }
4046 sml_ST_FontScheme = string "none" | string "major" | string "minor"
4047 sml_CT_UnderlineProperty =
4048
4049     ## default value: single
4050     attribute val { sml_ST_UnderlineValues }?
4051 sml_ST_UnderlineValues =
4052     string "single"
4053     | string "double"
4054     | string "singleAccounting"
4055     | string "doubleAccounting"
4056     | string "none"
4057 sml_CT_Font =
4058     (element name { sml_CT_FontName })?
4059     | element charset { sml_CT_IntProperty }?
4060     | element family { sml_CT_IntProperty }?
4061     | element b { sml_CT_BooleanProperty }?
4062     | element i { sml_CT_BooleanProperty }?
4063     | element strike { sml_CT_BooleanProperty }?

```

```

4064 | element outline { sml_CT_BooleanProperty }?
4065 | element shadow { sml_CT_BooleanProperty }?
4066 | element condense { sml_CT_BooleanProperty }?
4067 | element extend { sml_CT_BooleanProperty }?
4068 | element color { sml_CT_Color }?
4069 | element sz { sml_CT_FontSize }?
4070 | element u { sml_CT_UnderlineProperty }?
4071 | element vertAlign { sml_CT_VerticalAlignFontProperty }?
4072 | element scheme { sml_CT_FontScheme }?)+
4073 sml_AG_AutoFormat =
4074   attribute autoFormatId { xsd:unsignedInt }?,
4075   attribute applyNumberFormats { xsd:boolean }?,
4076   attribute applyBorderFormats { xsd:boolean }?,
4077   attribute applyFontFormats { xsd:boolean }?,
4078   attribute applyPatternFormats { xsd:boolean }?,
4079   attribute applyAlignmentFormats { xsd:boolean }?,
4080   attribute applyWidthHeightFormats { xsd:boolean }?
4081 sml_externallink = element externalLink { sml_CT_ExternalLink }
4082 sml_CT_ExternalLink =
4083   element externalBook { sml_CT_ExternalBook }?
4084   | element ddeLink { sml_CT_DdeLink }?
4085   | element oleLink { sml_CT_OleLink }?
4086   | element extLst { sml_CT_ExtensionList }?
4087 sml_CT_ExternalBook =
4088   r_id,
4089   element sheetNames { sml_CT_ExternalSheetNames }?,
4090   element definedNames { sml_CT_ExternalDefinedNames }?,
4091   element sheetDataSet { sml_CT_ExternalSheetDataSet }?
4092 sml_CT_ExternalSheetNames =
4093   element sheetName { sml_CT_ExternalSheetName }+
4094 sml_CT_ExternalSheetName = attribute val { s_ST_Xstring }?
4095 sml_CT_ExternalDefinedNames =
4096   element definedName { sml_CT_ExternalDefinedName }*
4097 sml_CT_ExternalDefinedName =
4098   attribute name { s_ST_Xstring },
4099   attribute refersTo { s_ST_Xstring }?,
4100   attribute sheetId { xsd:unsignedInt }?
4101 sml_CT_ExternalSheetDataSet =
4102   element sheetData { sml_CT_ExternalSheetData }+
4103 sml_CT_ExternalSheetData =
4104   attribute sheetId { xsd:unsignedInt },
4105
4106   ## default value: false
4107   attribute refreshError { xsd:boolean }?,
4108   element row { sml_CT_ExternalRow }*
4109 sml_CT_ExternalRow =
4110   attribute r { xsd:unsignedInt },
4111   element cell { sml_CT_ExternalCell }*
4112 sml_CT_ExternalCell =
4113   attribute r { sml_ST_CellRef }?,
4114
4115   ## default value: n
4116   attribute t { sml_ST_CellType }?,

```



```

4117
4118     ## default value: 0
4119     attribute vm { xsd:unsignedInt }?,
4120     element v { s_ST_Xstring }?
4121 sml_CT_DdeLink =
4122     attribute ddeService { s_ST_Xstring },
4123     attribute ddeTopic { s_ST_Xstring },
4124     element ddeItems { sml_CT_DdeItems }?
4125 sml_CT_DdeItems = element ddeItem { sml_CT_DdeItem }*
4126 sml_CT_DdeItem =
4127
4128     ## default value: 0
4129     attribute name { s_ST_Xstring }?,
4130
4131     ## default value: false
4132     attribute ole { xsd:boolean }?,
4133
4134     ## default value: false
4135     attribute advise { xsd:boolean }?,
4136
4137     ## default value: false
4138     attribute preferPic { xsd:boolean }?,
4139     element values { sml_CT_DdeValues }?
4140 sml_CT_DdeValues =
4141
4142     ## default value: 1
4143     attribute rows { xsd:unsignedInt }?,
4144
4145     ## default value: 1
4146     attribute cols { xsd:unsignedInt }?,
4147     element value { sml_CT_DdeValue }+
4148 sml_CT_DdeValue =
4149
4150     ## default value: n
4151     attribute t { sml_ST_DdeValueType }?,
4152     element val { s_ST_Xstring }
4153 sml_ST_DdeValueType =
4154     string "nil" | string "b" | string "n" | string "e" | string "str"
4155 sml_CT_OleLink =
4156     r_id,
4157     attribute progId { s_ST_Xstring },
4158     element oleItems { sml_CT_OleItems }?
4159 sml_CT_OleItems = element oleItem { sml_CT_OleItem }*
4160 sml_CT_OleItem =
4161     attribute name { s_ST_Xstring },
4162
4163     ## default value: false
4164     attribute icon { xsd:boolean }?,
4165
4166     ## default value: false
4167     attribute advise { xsd:boolean }?,
4168
4169     ## default value: false

```

```

4170     attribute preferPic { xsd:boolean }?
4171 sml_table = element table { sml_CT_Table }
4172 sml_CT_Table =
4173     attribute id { xsd:unsignedInt },
4174     attribute name { s_ST_Xstring }?,
4175     attribute displayName { s_ST_Xstring },
4176     attribute comment { s_ST_Xstring }?,
4177     attribute ref { sml_ST_Ref },
4178
4179     ## default value: worksheet
4180     attribute tableType { sml_ST_TableType }?,
4181
4182     ## default value: 1
4183     attribute headerRowCount { xsd:unsignedInt }?,
4184
4185     ## default value: false
4186     attribute insertRow { xsd:boolean }?,
4187
4188     ## default value: false
4189     attribute insertRowShift { xsd:boolean }?,
4190
4191     ## default value: 0
4192     attribute totalsRowCount { xsd:unsignedInt }?,
4193
4194     ## default value: true
4195     attribute totalsRowShown { xsd:boolean }?,
4196
4197     ## default value: false
4198     attribute published { xsd:boolean }?,
4199     attribute headerRowDxfId { sml_ST_DxfId }?,
4200     attribute dataDxfId { sml_ST_DxfId }?,
4201     attribute totalsRowDxfId { sml_ST_DxfId }?,
4202     attribute headerRowBorderDxfId { sml_ST_DxfId }?,
4203     attribute tableBorderDxfId { sml_ST_DxfId }?,
4204     attribute totalsRowBorderDxfId { sml_ST_DxfId }?,
4205     attribute headerRowCellStyle { s_ST_Xstring }?,
4206     attribute dataCellStyle { s_ST_Xstring }?,
4207     attribute totalsRowCellStyle { s_ST_Xstring }?,
4208     attribute connectionId { xsd:unsignedInt }?,
4209     element autoFilter { sml_CT_AutoFilter }?,
4210     element sortState { sml_CT_SortState }?,
4211     element tableColumns { sml_CT_TableColumns },
4212     element tableStyleInfo { sml_CT_TableStyleInfo }?,
4213     element extLst { sml_CT_ExtensionList }?
4214 sml_ST_TableType =
4215     string "worksheet" | string "xml" | string "queryTable"
4216 sml_CT_TableStyleInfo =
4217     attribute name { s_ST_Xstring }?,
4218     attribute showFirstColumn { xsd:boolean }?,
4219     attribute showLastColumn { xsd:boolean }?,
4220     attribute showRowStripes { xsd:boolean }?,
4221     attribute showColumnStripes { xsd:boolean }?
4222 sml_CT_TableColumns =

```

```

4223     attribute count { xsd:unsignedInt }?,
4224     element tableColumn { sml_CT_TableColumn }+
4225 sml_CT_TableColumn =
4226     attribute id { xsd:unsignedInt },
4227     attribute uniqueName { s_ST_Xstring }?,
4228     attribute name { s_ST_Xstring },
4229
4230     ## default value: none
4231     attribute totalsRowFunction { sml_ST_TotalsRowFunction }?,
4232     attribute totalsRowLabel { s_ST_Xstring }?,
4233     attribute queryTableFieldId { xsd:unsignedInt }?,
4234     attribute headerRowDxfId { sml_ST_DxfId }?,
4235     attribute dataDxfId { sml_ST_DxfId }?,
4236     attribute totalsRowDxfId { sml_ST_DxfId }?,
4237     attribute headerRowCellStyle { s_ST_Xstring }?,
4238     attribute dataCellStyle { s_ST_Xstring }?,
4239     attribute totalsRowCellStyle { s_ST_Xstring }?,
4240     element calculatedColumnFormula { sml_CT_TableFormula }?,
4241     element totalsRowFormula { sml_CT_TableFormula }?,
4242     element xmlColumnPr { sml_CT_XmlColumnPr }?,
4243     element extLst { sml_CT_ExtensionList }?
4244 sml_CT_TableFormula =
4245     sml_ST_Formula,
4246
4247     ## default value: false
4248     attribute array { xsd:boolean }?
4249 sml_ST_TotalsRowFunction =
4250     string "none"
4251     | string "sum"
4252     | string "min"
4253     | string "max"
4254     | string "average"
4255     | string "count"
4256     | string "countNums"
4257     | string "stdDev"
4258     | string "var"
4259     | string "custom"
4260 sml_CT_XmlColumnPr =
4261     attribute mapId { xsd:unsignedInt },
4262     attribute xpath { s_ST_Xstring },
4263
4264     ## default value: false
4265     attribute denormalized { xsd:boolean }?,
4266     attribute xmlDataType { sml_ST_XmlDataType },
4267     element extLst { sml_CT_ExtensionList }?
4268 sml_ST_XmlDataType = xsd:string
4269 sml_volTypes = element volTypes { sml_CT_VolTypes }
4270 sml_CT_VolTypes =
4271     element volType { sml_CT_VolType }+,
4272     element extLst { sml_CT_ExtensionList }?
4273 sml_CT_VolType =
4274     attribute type { sml_ST_VolDepType },
4275     element main { sml_CT_VolMain }+

```

```

4276 sml_CT_VolMain =
4277     attribute first { s_ST_Xstring },
4278     element tp { sml_CT_VolTopic }+
4279 sml_CT_VolTopic =
4280
4281     ## default value: n
4282     attribute t { sml_ST_VolValueType }?,
4283     element v { s_ST_Xstring },
4284     element stp { s_ST_Xstring }*,
4285     element tr { sml_CT_VolTopicRef }+
4286 sml_CT_VolTopicRef =
4287     attribute r { sml_ST_CellRef },
4288     attribute s { xsd:unsignedInt }
4289 sml_ST_VolDepType = string "realTimeData" | string "olapFunctions"
4290 sml_ST_VolValueType = string "b" | string "n" | string "e" | string "s"
4291 sml_workbook = element workbook { sml_CT_Workbook }
4292 sml_CT_Workbook =
4293     attribute conformance { s_ST_ConformanceClass }?,
4294     element fileVersion { sml_CT_FileVersion }?,
4295     element fileSharing { sml_CT_FileSharing }?,
4296     element workbookPr { sml_CT_WorkbookPr }?,
4297     element workbookProtection { sml_CT_WorkbookProtection }?,
4298     element bookViews { sml_CT_BookViews }?,
4299     element sheets { sml_CT_Sheets },
4300     element functionGroups { sml_CT_FunctionGroups }?,
4301     element externalReferences { sml_CT_ExternalReferences }?,
4302     element definedNames { sml_CT_DefinedNames }?,
4303     element calcPr { sml_CT_CalcPr }?,
4304     element oleSize { sml_CT_OleSize }?,
4305     element customWorkbookViews { sml_CT_CustomWorkbookViews }?,
4306     element pivotCaches { sml_CT_PivotCaches }?,
4307     element smartTagPr { sml_CT_SmartTagPr }?,
4308     element smartTagTypes { sml_CT_SmartTagTypes }?,
4309     element webPublishing { sml_CT_WebPublishing }?,
4310     element fileRecoveryPr { sml_CT_FileRecoveryPr }*,
4311     element webPublishObjects { sml_CT_WebPublishObjects }?,
4312     element extLst { sml_CT_ExtensionList }?
4313 sml_CT_FileVersion =
4314     attribute appName { xsd:string }?,
4315     attribute lastEdited { xsd:string }?,
4316     attribute lowestEdited { xsd:string }?,
4317     attribute rupBuild { xsd:string }?,
4318     attribute codeName { s_ST_Guid }?
4319 sml_CT_BookViews = element workbookView { sml_CT_BookView }+
4320 sml_CT_BookView =
4321
4322     ## default value: visible
4323     attribute visibility { sml_ST_Visibility }?,
4324
4325     ## default value: false
4326     attribute minimized { xsd:boolean }?,
4327
4328     ## default value: true

```

```

4329 attribute showHorizontalScroll { xsd:boolean }?,
4330
4331 ## default value: true
4332 attribute showVerticalScroll { xsd:boolean }?,
4333
4334 ## default value: true
4335 attribute showSheetTabs { xsd:boolean }?,
4336 attribute xWindow { xsd:int }?,
4337 attribute yWindow { xsd:int }?,
4338 attribute windowWidth { xsd:unsignedInt }?,
4339 attribute windowHeight { xsd:unsignedInt }?,
4340
4341 ## default value: 600
4342 attribute tabRatio { xsd:unsignedInt }?,
4343
4344 ## default value: 0
4345 attribute firstSheet { xsd:unsignedInt }?,
4346
4347 ## default value: 0
4348 attribute activeTab { xsd:unsignedInt }?,
4349
4350 ## default value: true
4351 attribute autoFilterDateGrouping { xsd:boolean }?,
4352 element extLst { sml_CT_ExtensionList }?
4353 sml_ST_Visibility =
4354     string "visible" | string "hidden" | string "veryHidden"
4355 sml_CT_CustomWorkbookViews =
4356     element customWorkbookView { sml_CT_CustomWorkbookView }+
4357 sml_CT_CustomWorkbookView =
4358     attribute name { s_ST_Xstring },
4359     attribute guid { s_ST_Guid },
4360
4361 ## default value: false
4362 attribute autoUpdate { xsd:boolean }?,
4363 attribute mergeInterval { xsd:unsignedInt }?,
4364
4365 ## default value: false
4366 attribute changesSavedWin { xsd:boolean }?,
4367
4368 ## default value: false
4369 attribute onlySync { xsd:boolean }?,
4370
4371 ## default value: false
4372 attribute personalView { xsd:boolean }?,
4373
4374 ## default value: true
4375 attribute includePrintSettings { xsd:boolean }?,
4376
4377 ## default value: true
4378 attribute includeHiddenRowCol { xsd:boolean }?,
4379
4380 ## default value: false
4381 attribute maximized { xsd:boolean }?,

```

```

4382
4383   ## default value: false
4384   attribute minimized { xsd:boolean }?,
4385
4386   ## default value: true
4387   attribute showHorizontalScroll { xsd:boolean }?,
4388
4389   ## default value: true
4390   attribute showVerticalScroll { xsd:boolean }?,
4391
4392   ## default value: true
4393   attribute showSheetTabs { xsd:boolean }?,
4394
4395   ## default value: 0
4396   attribute xWindow { xsd:int }?,
4397
4398   ## default value: 0
4399   attribute yWindow { xsd:int }?,
4400   attribute windowWidth { xsd:unsignedInt },
4401   attribute windowHeight { xsd:unsignedInt },
4402
4403   ## default value: 600
4404   attribute tabRatio { xsd:unsignedInt }?,
4405   attribute activeSheetId { xsd:unsignedInt },
4406
4407   ## default value: true
4408   attribute showFormulaBar { xsd:boolean }?,
4409
4410   ## default value: true
4411   attribute showStatusbar { xsd:boolean }?,
4412
4413   ## default value: commIndicator
4414   attribute showComments { sml_ST_Comments }?,
4415
4416   ## default value: all
4417   attribute showObjects { sml_ST_Objects }?,
4418   element extLst { sml_CT_ExtensionList }?
4419 sml_ST_Comments =
4420   string "commNone"
4421   | string "commIndicator"
4422   | string "commIndAndComment"
4423 sml_ST_Objects = string "all" | string "placeholders" | string "none"
4424 sml_CT_Sheets = element sheet { sml_CT_Sheet }+
4425 sml_CT_Sheet =
4426   attribute name { s_ST_Xstring },
4427   attribute sheetId { xsd:unsignedInt },
4428
4429   ## default value: visible
4430   attribute state { sml_ST_SheetState }?,
4431   r_id
4432 sml_ST_SheetState =
4433   string "visible" | string "hidden" | string "veryHidden"
4434 sml_CT_WorkbookPr =

```

```

4435
4436 ## default value: false
4437 attribute date1904 { xsd:boolean }?,
4438
4439 ## default value: true
4440 attribute dateCompatibility { xsd:boolean }?,
4441
4442 ## default value: all
4443 attribute showObjects { sml_ST_Objects }?,
4444
4445 ## default value: true
4446 attribute showBorderUnselectedTables { xsd:boolean }?,
4447
4448 ## default value: false
4449 attribute filterPrivacy { xsd:boolean }?,
4450
4451 ## default value: false
4452 attribute promptedSolutions { xsd:boolean }?,
4453
4454 ## default value: true
4455 attribute showInkAnnotation { xsd:boolean }?,
4456
4457 ## default value: false
4458 attribute backupFile { xsd:boolean }?,
4459
4460 ## default value: true
4461 attribute saveExternalLinkValues { xsd:boolean }?,
4462
4463 ## default value: userSet
4464 attribute updateLinks { sml_ST_UpdateLinks }?,
4465 attribute codeName { xsd:string }?,
4466
4467 ## default value: false
4468 attribute hidePivotFieldList { xsd:boolean }?,
4469
4470 ## default value: false
4471 attribute showPivotChartFilter { xsd:boolean }?,
4472
4473 ## default value: false
4474 attribute allowRefreshQuery { xsd:boolean }?,
4475
4476 ## default value: false
4477 attribute publishItems { xsd:boolean }?,
4478
4479 ## default value: false
4480 attribute checkCompatibility { xsd:boolean }?,
4481
4482 ## default value: true
4483 attribute autoCompressPictures { xsd:boolean }?,
4484
4485 ## default value: false
4486 attribute refreshAllConnections { xsd:boolean }?,
4487 attribute defaultThemeVersion { xsd:unsignedInt }?

```

```

4488 sml_ST_UpdateLinks = string "userSet" | string "never" | string "always"
4489 sml_CT_SmartTagPr =
4490
4491     ## default value: false
4492     attribute embed { xsd:boolean }?,
4493
4494     ## default value: all
4495     attribute show { sml_ST_SmartTagShow }?
4496 sml_ST_SmartTagShow =
4497     string "all" | string "none" | string "noIndicator"
4498 sml_CT_SmartTagTypes = element smartTagType { sml_CT_SmartTagType }*
4499 sml_CT_SmartTagType =
4500     attribute namespaceUri { s_ST_Xstring }?,
4501     attribute name { s_ST_Xstring }?,
4502     attribute url { s_ST_Xstring }?
4503 sml_CT_FileRecoveryPr =
4504
4505     ## default value: true
4506     attribute autoRecover { xsd:boolean }?,
4507
4508     ## default value: false
4509     attribute crashSave { xsd:boolean }?,
4510
4511     ## default value: false
4512     attribute dataExtractLoad { xsd:boolean }?,
4513
4514     ## default value: false
4515     attribute repairLoad { xsd:boolean }?
4516 sml_CT_CalcPr =
4517     attribute calcId { xsd:unsignedInt }?,
4518
4519     ## default value: auto
4520     attribute calcMode { sml_ST_CalcMode }?,
4521
4522     ## default value: false
4523     attribute fullCalcOnLoad { xsd:boolean }?,
4524
4525     ## default value: A1
4526     attribute refMode { sml_ST_RefMode }?,
4527
4528     ## default value: false
4529     attribute iterate { xsd:boolean }?,
4530
4531     ## default value: 100
4532     attribute iterateCount { xsd:unsignedInt }?,
4533
4534     ## default value: 0.001
4535     attribute iterateDelta { xsd:double }?,
4536
4537     ## default value: true
4538     attribute fullPrecision { xsd:boolean }?,
4539
4540     ## default value: true

```



```

4541     attribute calcCompleted { xsd:boolean }?,
4542
4543     ## default value: true
4544     attribute calcOnSave { xsd:boolean }?,
4545
4546     ## default value: true
4547     attribute concurrentCalc { xsd:boolean }?,
4548     attribute concurrentManualCount { xsd:unsignedInt }?,
4549     attribute forceFullCalc { xsd:boolean }?
4550 sml_ST_CalcMode = string "manual" | string "auto" | string "autoNoTable"
4551 sml_ST_RefMode = string "A1" | string "R1C1"
4552 sml_CT_DefinedNames = element definedName { sml_CT_DefinedName }*
4553 sml_CT_DefinedName =
4554     sml_ST_Formula,
4555     attribute name { s_ST_Xstring },
4556     attribute comment { s_ST_Xstring }?,
4557     attribute customMenu { s_ST_Xstring }?,
4558     attribute description { s_ST_Xstring }?,
4559     attribute help { s_ST_Xstring }?,
4560     attribute statusBar { s_ST_Xstring }?,
4561     attribute localSheetId { xsd:unsignedInt }?,
4562
4563     ## default value: false
4564     attribute hidden { xsd:boolean }?,
4565
4566     ## default value: false
4567     attribute function { xsd:boolean }?,
4568
4569     ## default value: false
4570     attribute vbProcedure { xsd:boolean }?,
4571
4572     ## default value: false
4573     attribute xlm { xsd:boolean }?,
4574     attribute functionGroupId { xsd:unsignedInt }?,
4575     attribute shortcutKey { s_ST_Xstring }?,
4576
4577     ## default value: false
4578     attribute publishToServer { xsd:boolean }?,
4579
4580     ## default value: false
4581     attribute workbookParameter { xsd:boolean }?
4582 sml_CT_ExternalReferences =
4583     element externalReference { sml_CT_ExternalReference }+
4584 sml_CT_ExternalReference = r_id
4585 sml_CT_SheetBackgroundPicture = r_id
4586 sml_CT_PivotCaches = element pivotCache { sml_CT_PivotCache }+
4587 sml_CT_PivotCache =
4588     attribute cacheId { xsd:unsignedInt },
4589     r_id
4590 sml_CT_FileSharing =
4591
4592     ## default value: false
4593     attribute readOnlyRecommended { xsd:boolean }?,

```

```

4594     attribute userName { s_ST_Xstring }?,
4595     attribute reservationPassword { sml_ST_UnsignedShortHex }?,
4596     attribute algorithmName { s_ST_Xstring }?,
4597     attribute hashValue { xsd:base64Binary }?,
4598     attribute saltValue { xsd:base64Binary }?,
4599     attribute spinCount { xsd:unsignedInt }?
4600 sml_CT_OleSize = attribute ref { sml_ST_Ref }
4601 sml_CT_WorkbookProtection =
4602     attribute workbookPassword { sml_ST_UnsignedShortHex }?,
4603     attribute workbookPasswordCharacterSet { xsd:string }?,
4604     attribute revisionsPassword { sml_ST_UnsignedShortHex }?,
4605     attribute revisionsPasswordCharacterSet { xsd:string }?,
4606
4607     ## default value: false
4608     attribute lockStructure { xsd:boolean }?,
4609
4610     ## default value: false
4611     attribute lockWindows { xsd:boolean }?,
4612
4613     ## default value: false
4614     attribute lockRevision { xsd:boolean }?,
4615     attribute revisionsAlgorithmName { s_ST_Xstring }?,
4616     attribute revisionsHashValue { xsd:base64Binary }?,
4617     attribute revisionsSaltValue { xsd:base64Binary }?,
4618     attribute revisionsSpinCount { xsd:unsignedInt }?,
4619     attribute workbookAlgorithmName { s_ST_Xstring }?,
4620     attribute workbookHashValue { xsd:base64Binary }?,
4621     attribute workbookSaltValue { xsd:base64Binary }?,
4622     attribute workbookSpinCount { xsd:unsignedInt }?
4623 sml_CT_WebPublishing =
4624
4625     ## default value: true
4626     attribute css { xsd:boolean }?,
4627
4628     ## default value: true
4629     attribute thicket { xsd:boolean }?,
4630
4631     ## default value: true
4632     attribute longFileNames { xsd:boolean }?,
4633
4634     ## default value: false
4635     attribute vml { xsd:boolean }?,
4636
4637     ## default value: false
4638     attribute allowPng { xsd:boolean }?,
4639
4640     ## default value: 800x600
4641     attribute targetScreenSize { sml_ST_TargetScreenSize }?,
4642
4643     ## default value: 96
4644     attribute dpi { xsd:unsignedInt }?,
4645     attribute codePage { xsd:unsignedInt }?,
4646     attribute characterSet { xsd:string }?

```

```

4647 sml_ST_TargetScreenSize =
4648     string "544x376"
4649     | string "640x480"
4650     | string "720x512"
4651     | string "800x600"
4652     | string "1024x768"
4653     | string "1152x882"
4654     | string "1152x900"
4655     | string "1280x1024"
4656     | string "1600x1200"
4657     | string "1800x1440"
4658     | string "1920x1200"
4659 sml_CT_FunctionGroups =
4660
4661     ## default value: 16
4662     attribute builtInGroupCount { xsd:unsignedInt }?,
4663     element functionGroup { sml_CT_FunctionGroup }*
4664 sml_CT_FunctionGroup = attribute name { s_ST_Xstring }?
4665 sml_CT_WebPublishObjects =
4666     attribute count { xsd:unsignedInt }?,
4667     element webPublishObject { sml_CT_WebPublishObject }+
4668 sml_CT_WebPublishObject =
4669     attribute id { xsd:unsignedInt },
4670     attribute divId { s_ST_Xstring },
4671     attribute sourceObject { s_ST_Xstring }?,
4672     attribute destinationFile { s_ST_Xstring },
4673     attribute title { s_ST_Xstring }?,
4674
4675     ## default value: false
4676     attribute autoRepublish { xsd:boolean }?

```

B.2.1 Part Schemas

B.2.1.1 Calculation Chain Part

```

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 include "dml-compatibility.rnc"
13 start = sml_calcChain

```

B.2.1.2 Chartsheet Part

```

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 include "dml-compatibility.rnc"
13 start = sml_chartsheet

```

B.2.1.3 Comments Part

```

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 include "dml-compatibility.rnc"
13 start = sml_comments

```

B.2.1.4 Connections Part

```

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 include "dml-compatibility.rnc"
13 start = sml_connections

```

B.2.1.5 Custom XML Mappings Part

```

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 include "dml-compatibility.rnc"
13 start = sml_MapInfo

```

B.2.1.6 Dialogsheet Part

```

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 include "dml-compatibility.rnc"
13 start = sml_dialogsheet

```

B.2.1.7 Drawing Part

```

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 include "dml-compatibility.rnc"
12 start = xdr_wsDr

```

B.2.1.8 External Workbook References Part

```

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 include "dml-compatibility.rnc"
13 start = sml_externalLink

```

B.2.1.9 Metadata Part

```

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 include "dml-compatibility.rnc"
13 start = sml_metadata

```

B.2.1.10 Pivot Table Part

```

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 include "dml-compatibility.rnc"
13 start = sml_pivotTableDefinition

```

B.2.1.11 Pivot Table Cache Definition Part

```

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 include "dml-compatibility.rnc"
13 start = sml_pivotCacheDefinition

```

B.2.1.12 Pivot Table Cache Records Part

```

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 include "dml-compatibility.rnc"
13 start = sml_pivotCacheRecords

```

B.2.1.13 Query Table Part

```

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 include "dml-compatibility.rnc"
13 start = sml_queryTable

```

B.2.1.14 Shared String Table Part

```

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

```

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 include "dml-compatibility.rnc"
13 start = sml_headers

```

B.2.1.16 Shared Workbook Revision Log Part

```

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 include "dml-compatibility.rnc"
13 start = sml_revisions

```

B.2.1.17 Shared Workbook User Data Part

```

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 include "dml-compatibility.rnc"
13 start = sml_users

```

B.2.1.18 Single Cell Table Definitions Part

```

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 include "dml-compatibility.rnc"
13 start = sml_singleXmlCells

```

B.2.1.19 Styles Part

```

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 include "dml-compatibility.rnc"
13 start = sml_styleSheet

```

B.2.1.20 Table Definitions Part

```

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 include "dml-compatibility.rnc"
13 start = sml_table

```

B.2.1.21 Volatile Dependencies Part

```

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 include "dml-compatibility.rnc"
13 start = sml_volTypes

```

B.2.1.22 Workbook Part

```

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 include "dml-compatibility.rnc"
13 start = sml_workbook

```

B.2.1.23 Worksheet Part

```

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 include "dml-compatibility.rnc"
13 start = sml_worksheet

```

B.3 PresentationML

```

14 default namespace =
15     "http://schemas.openxmlformats.org/presentationml/2006/main"
16 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
17 namespace o = "urn:schemas-microsoft-com:office:office"
18 namespace p =
19     "http://schemas.openxmlformats.org/presentationml/2006/main"
20 namespace r =
21     "http://schemas.openxmlformats.org/officeDocument/2006/relationships"
22 namespace s =
23     "http://schemas.openxmlformats.org/officeDocument/2006/sharedTypes"
24 namespace v = "urn:schemas-microsoft-com:vml"
25 namespace w10 = "urn:schemas-microsoft-com:office:word"
26 namespace x = "urn:schemas-microsoft-com:office:excel"
27
28
29 p_ST_TransitionSideDirectionType = "l" | "u" | "r" | "d"
30 p_ST_TransitionCornerDirectionType = "lu" | "ru" | "ld" | "rd"
31 p_ST_TransitionInOutDirectionType = "out" | "in"
32 p_CT_SideDirectionTransition =
33
34     ## default value: l
35     attribute dir { p_ST_TransitionSideDirectionType }?
36 p_CT_CornerDirectionTransition =
37
38     ## default value: lu
39     attribute dir { p_ST_TransitionCornerDirectionType }?
40 p_ST_TransitionEightDirectionType =
41     p_ST_TransitionSideDirectionType | p_ST_TransitionCornerDirectionType
42 p_CT_EightDirectionTransition =
43
44     ## default value: l
45     attribute dir { p_ST_TransitionEightDirectionType }?
46 p_CT_OrientationTransition =

```

```

47
48   ## default value: horz
49   attribute dir { p_ST_Direction }?
50 p_CT_InOutTransition =
51
52   ## default value: out
53   attribute dir { p_ST_TransitionInOutDirectionType }?
54 p_CT_OptionalBlackTransition =
55
56   ## default value: false
57   attribute thruBlk { xsd:boolean }?
58 p_CT_SplitTransition =
59
60   ## default value: horz
61   attribute orient { p_ST_Direction }?,
62
63   ## default value: out
64   attribute dir { p_ST_TransitionInOutDirectionType }?
65 p_CT_WheelTransition =
66
67   ## default value: 4
68   attribute spokes { xsd:unsignedInt }?
69 p_CT_TransitionStartSoundAction =
70
71   ## default value: false
72   attribute loop { xsd:boolean }?,
73   element snd { a_CT_EmbeddedWAVAudioFile }
74 p_CT_TransitionSoundAction =
75   element stSnd { p_CT_TransitionStartSoundAction }
76   | element endSnd { p_CT_Empty }
77 p_ST_TransitionSpeed = "slow" | "med" | "fast"
78 p_CT_SlideTransition =
79
80   ## default value: fast
81   attribute spd { p_ST_TransitionSpeed }?,
82
83   ## default value: true
84   attribute advClick { xsd:boolean }?,
85   attribute advTm { xsd:unsignedInt }?,
86   (element blinds { p_CT_OrientationTransition }
87   | element checker { p_CT_OrientationTransition }
88   | element circle { p_CT_Empty }
89   | element dissolve { p_CT_Empty }
90   | element comb { p_CT_OrientationTransition }
91   | element cover { p_CT_EightDirectionTransition }
92   | element cut { p_CT_OptionalBlackTransition }
93   | element diamond { p_CT_Empty }
94   | element fade { p_CT_OptionalBlackTransition }
95   | element newsflash { p_CT_Empty }
96   | element plus { p_CT_Empty }
97   | element pull { p_CT_EightDirectionTransition }
98   | element push { p_CT_SideDirectionTransition }
99   | element random { p_CT_Empty }

```

```

100 | element randomBar { p_CT_OrientationTransition }
101 | element split { p_CT_SplitTransition }
102 | element strips { p_CT_CornerDirectionTransition }
103 | element wedge { p_CT_Empty }
104 | element wheel { p_CT_WheelTransition }
105 | element wipe { p_CT_SideDirectionTransition }
106 | element zoom { p_CT_InOutTransition }}?,
107 element sndAc { p_CT_TransitionSoundAction }?,
108 element extLst { p_CT_ExtensionListModify }?
109 p_ST_TLTimeIndefinite = "indefinite"
110 p_ST_TLTime = xsd:unsignedInt | p_ST_TLTimeIndefinite
111 p_ST_TLTimeNodeID = xsd:unsignedInt
112 p_CT_TLIterateIntervalTime = attribute val { p_ST_TLTime }
113 p_CT_TLIterateIntervalPercentage =
114     attribute val { a_ST_PositivePercentage }
115 p_ST_IterateType = "el" | "wd" | "lt"
116 p_CT_TLIterateData =
117
118     ## default value: el
119     attribute type { p_ST_IterateType }?,
120
121     ## default value: false
122     attribute backwards { xsd:boolean }?,
123     (element tmAbs { p_CT_TLIterateIntervalTime }
124     | element tmPct { p_CT_TLIterateIntervalPercentage })
125 p_CT_TLSubShapeId = attribute spid { a_ST_ShapeID }
126 p_CT_TLTextTargetElement =
127     (element charRg { p_CT_IndexRange }
128     | element pRg { p_CT_IndexRange })?
129 p_ST_TLChartSubelementType =
130     "gridLegend" | "series" | "category" | "ptInSeries" | "ptInCategory"
131 p_CT_TLOleChartTargetElement =
132     attribute type { p_ST_TLChartSubelementType },
133
134     ## default value: 0
135     attribute lvl { xsd:unsignedInt }?
136 p_CT_TLShapeTargetElement =
137     attribute spid { a_ST_ShapeID },
138     (element bg { p_CT_Empty }
139     | element subSp { p_CT_TLSubShapeId }
140     | element oleChartEl { p_CT_TLOleChartTargetElement }
141     | element txEl { p_CT_TLTextTargetElement }
142     | element graphicEl { a_CT_AnimationElementChoice })?
143 p_CT_TLTimeTargetElement =
144     element sldTgt { p_CT_Empty }
145     | element sndTgt { a_CT_EmbeddedWAVAudioFile }
146     | element spTgt { p_CT_TLShapeTargetElement }
147     | element inkTgt { p_CT_TLSubShapeId }
148 p_CT_TLTriggerTimeNodeID = attribute val { p_ST_TLTimeNodeID }
149 p_ST_TLTriggerRuntimeNode = "first" | "last" | "all"
150 p_CT_TLTriggerRuntimeNode = attribute val { p_ST_TLTriggerRuntimeNode }
151 p_ST_TLTriggerEvent =
152     "onBegin"

```

```

153 | "onEnd"
154 | "begin"
155 | "end"
156 | "onClick"
157 | "onDbClick"
158 | "onMouseOver"
159 | "onMouseOut"
160 | "onNext"
161 | "onPrev"
162 | "onStopAudio"
163 p_CT_TLTimeCondition =
164   attribute evt { p_ST_TLTriggerEvent }?,
165   attribute delay { p_ST_TLTime }?,
166   (element tgtEl { p_CT_TLTimeTargetElement }
167     | element tn { p_CT_TLTriggerTimeNodeID }
168     | element rtn { p_CT_TLTriggerRuntimeNode })?
169 p_CT_TLTimeConditionList = element cond { p_CT_TLTimeCondition }+
170 p_CT_TimeNodeList =
171   (element par { p_CT_TLTimeNodeParallel }
172     | element seq { p_CT_TLTimeNodeSequence }
173     | element excl { p_CT_TLTimeNodeExclusive }
174     | element anim { p_CT_TLAnimateBehavior }
175     | element animClr { p_CT_TLAnimateColorBehavior }
176     | element animEffect { p_CT_TLAnimateEffectBehavior }
177     | element animMotion { p_CT_TLAnimateMotionBehavior }
178     | element animRot { p_CT_TLAnimateRotationBehavior }
179     | element animScale { p_CT_TLAnimateScaleBehavior }
180     | element cmd { p_CT_TLCommandBehavior }
181     | element set { p_CT_TLSetBehavior }
182     | element audio { p_CT_TLMediaNodeAudio }
183     | element video { p_CT_TLMediaNodeVideo })+
184 p_ST_TLTimeNodePresetClassType =
185   "entr" | "exit" | "emph" | "path" | "verb" | "mediacall"
186 p_ST_TLTimeNodeRestartType = "always" | "whenNotActive" | "never"
187 p_ST_TLTimeNodeFillType = "remove" | "freeze" | "hold" | "transition"
188 p_ST_TLTimeNodeSyncType = "canSlip" | "locked"
189 p_ST_TLTimeNodeMasterRelation = "sameClick" | "lastClick" | "nextClick"
190 p_ST_TLTimeNodeType =
191   "clickEffect"
192   | "withEffect"
193   | "afterEffect"
194   | "mainSeq"
195   | "interactiveSeq"
196   | "clickPar"
197   | "withGroup"
198   | "afterGroup"
199   | "tmRoot"
200 p_CT_TLCommonTimeNodeData =
201   attribute id { p_ST_TLTimeNodeID }?,
202   attribute presetID { xsd:int }?,
203   attribute presetClass { p_ST_TLTimeNodePresetClassType }?,
204   attribute presetSubtype { xsd:int }?,
205   attribute dur { p_ST_TLTime }?,

```

```

206
207   ## default value: 1000
208   attribute repeatCount { p_ST_TLTime }?,
209   attribute repeatDur { p_ST_TLTime }?,
210
211   ## default value: 100%
212   attribute spd { a_ST_Percentage }?,
213
214   ## default value: 0%
215   attribute accel { a_ST_PositiveFixedPercentage }?,
216
217   ## default value: 0%
218   attribute decel { a_ST_PositiveFixedPercentage }?,
219
220   ## default value: false
221   attribute autoRev { xsd:boolean }?,
222   attribute restart { p_ST_TLTimeNodeRestartType }?,
223   attribute fill { p_ST_TLTimeNodeFillType }?,
224   attribute syncBehavior { p_ST_TLTimeNodeSyncType }?,
225   attribute tmFilter { xsd:string }?,
226   attribute evtFilter { xsd:string }?,
227   attribute display { xsd:boolean }?,
228   attribute masterRel { p_ST_TLTimeNodeMasterRelation }?,
229   attribute bldLvl { xsd:int }?,
230   attribute grpId { xsd:unsignedInt }?,
231   attribute afterEffect { xsd:boolean }?,
232   attribute nodeType { p_ST_TLTimeNodeType }?,
233   attribute nodePh { xsd:boolean }?,
234   element stCondLst { p_CT_TLTimeConditionList }?,
235   element endCondLst { p_CT_TLTimeConditionList }?,
236   element endSync { p_CT_TLTimeCondition }?,
237   element iterate { p_CT_TLIterateData }?,
238   element childTnLst { p_CT_TimeNodeList }?,
239   element subTnLst { p_CT_TimeNodeList }?
240 p_CT_TLTimeNodeParallel = element cTn { p_CT_TLCommonTimeNodeData }
241 p_ST_TLNextActionType = "none" | "seek"
242 p_ST_TLPreviousActionType = "none" | "skipTimed"
243 p_CT_TLTimeNodeSequence =
244   attribute concurrent { xsd:boolean }?,
245   attribute prevAc { p_ST_TLPreviousActionType }?,
246   attribute nextAc { p_ST_TLNextActionType }?,
247   element cTn { p_CT_TLCommonTimeNodeData },
248   element prevCondLst { p_CT_TLTimeConditionList }?,
249   element nextCondLst { p_CT_TLTimeConditionList }?
250 p_CT_TLTimeNodeExclusive = element cTn { p_CT_TLCommonTimeNodeData }
251 p_CT_TLBehaviorAttributeNameList = element attrName { xsd:string }+
252 p_ST_TLBehaviorAdditiveType = "base" | "sum" | "repl" | "mult" | "none"
253 p_ST_TLBehaviorAccumulateType = "none" | "always"
254 p_ST_TLBehaviorTransformType = "pt" | "img"
255 p_ST_TLBehaviorOverrideType = "normal" | "childStyle"
256 p_CT_TLCommonBehaviorData =
257   attribute additive { p_ST_TLBehaviorAdditiveType }?,
258   attribute accumulate { p_ST_TLBehaviorAccumulateType }?,

```

```

259     attribute xfrmType { p_ST_TLBehaviorTransformType }?,
260     attribute from { xsd:string }?,
261     attribute to { xsd:string }?,
262     attribute by { xsd:string }?,
263     attribute rctx { xsd:string }?,
264     attribute override { p_ST_TLBehaviorOverrideType }?,
265     element cTn { p_CT_TLCommonTimeNodeData },
266     element tgtEl { p_CT_TLTimeTargetElement },
267     element attrNameLst { p_CT_TLBehaviorAttributeNameList }?
268 p_CT_TLAnimVariantBooleanVal = attribute val { xsd:boolean }
269 p_CT_TLAnimVariantIntegerVal = attribute val { xsd:int }
270 p_CT_TLAnimVariantFloatVal = attribute val { xsd:float }
271 p_CT_TLAnimVariantStringVal = attribute val { xsd:string }
272 p_CT_TLAnimVariant =
273     element boolVal { p_CT_TLAnimVariantBooleanVal }
274     | element intVal { p_CT_TLAnimVariantIntegerVal }
275     | element fltVal { p_CT_TLAnimVariantFloatVal }
276     | element strVal { p_CT_TLAnimVariantStringVal }
277     | element clrVal { a_CT_Color }
278 p_ST_TLTimeAnimateValueType =
279     a_ST_PositiveFixedPercentage | p_ST_TLTimeIndefinite
280 p_CT_TLTimeAnimateValue =
281
282     ## default value: indefinite
283     attribute tm { p_ST_TLTimeAnimateValueType }?,
284     attribute fmla { xsd:string }?,
285     element val { p_CT_TLAnimVariant }?
286 p_CT_TLTimeAnimateValueList = element tav { p_CT_TLTimeAnimateValue }*
287 p_ST_TLAnimateBehaviorCalcMode = "discrete" | "lin" | "fmla"
288 p_ST_TLAnimateBehaviorValueType = "str" | "num" | "clr"
289 p_CT_TLAnimateBehavior =
290     attribute by { xsd:string }?,
291     attribute from { xsd:string }?,
292     attribute to { xsd:string }?,
293     attribute calcmode { p_ST_TLAnimateBehaviorCalcMode }?,
294     attribute valueType { p_ST_TLAnimateBehaviorValueType }?,
295     element cBhvr { p_CT_TLCommonBehaviorData },
296     element tavLst { p_CT_TLTimeAnimateValueList }?
297 p_CT_TLByRgbColorTransform =
298     attribute r { a_ST_FixedPercentage },
299     attribute g { a_ST_FixedPercentage },
300     attribute b { a_ST_FixedPercentage }
301 p_CT_TLByHslColorTransform =
302     attribute h { a_ST_Angle },
303     attribute s { a_ST_FixedPercentage },
304     attribute l { a_ST_FixedPercentage }
305 p_CT_TLByAnimateColorTransform =
306     element rgb { p_CT_TLByRgbColorTransform }
307     | element hsl { p_CT_TLByHslColorTransform }
308 p_ST_TLAnimateColorSpace = "rgb" | "hsl"
309 p_ST_TLAnimateColorDirection = "cw" | "ccw"
310 p_CT_TLAnimateColorBehavior =
311     attribute clrSpc { p_ST_TLAnimateColorSpace }?,

```

```

312     attribute dir { p_ST_TLAnimateColorDirection }?,
313     element cBhvr { p_CT_TLCommonBehaviorData },
314     element by { p_CT_TLByAnimateColorTransform }?,
315     element from { a_CT_Color }?,
316     element to { a_CT_Color }?
317 p_ST_TLAnimateEffectTransition = "in" | "out" | "none"
318 p_CT_TLAnimateEffectBehavior =
319     attribute transition { p_ST_TLAnimateEffectTransition }?,
320     attribute filter { xsd:string }?,
321     attribute prLst { xsd:string }?,
322     element cBhvr { p_CT_TLCommonBehaviorData },
323     element progress { p_CT_TLAnimVariant }?
324 p_ST_TLAnimateMotionBehaviorOrigin = "parent" | "layout"
325 p_ST_TLAnimateMotionPathEditMode = "relative" | "fixed"
326 p_CT_TLPoint =
327     attribute x { a_ST_Percentage },
328     attribute y { a_ST_Percentage }
329 p_CT_TLAnimateMotionBehavior =
330     attribute origin { p_ST_TLAnimateMotionBehaviorOrigin }?,
331     attribute path { xsd:string }?,
332     attribute pathEditMode { p_ST_TLAnimateMotionPathEditMode }?,
333     attribute rAng { a_ST_Angle }?,
334     attribute ptsTypes { xsd:string }?,
335     element cBhvr { p_CT_TLCommonBehaviorData },
336     element by { p_CT_TLPoint }?,
337     element from { p_CT_TLPoint }?,
338     element to { p_CT_TLPoint }?,
339     element rCtr { p_CT_TLPoint }?
340 p_CT_TLAnimateRotationBehavior =
341     attribute by { a_ST_Angle }?,
342     attribute from { a_ST_Angle }?,
343     attribute to { a_ST_Angle }?,
344     element cBhvr { p_CT_TLCommonBehaviorData }
345 p_CT_TLAnimateScaleBehavior =
346     attribute zoomContents { xsd:boolean }?,
347     element cBhvr { p_CT_TLCommonBehaviorData },
348     element by { p_CT_TLPoint }?,
349     element from { p_CT_TLPoint }?,
350     element to { p_CT_TLPoint }?
351 p_ST_TLCommandType = "evt" | "call" | "verb"
352 p_CT_TLCommandBehavior =
353     attribute type { p_ST_TLCommandType }?,
354     attribute cmd { xsd:string }?,
355     element cBhvr { p_CT_TLCommonBehaviorData }
356 p_CT_TLSetBehavior =
357     element cBhvr { p_CT_TLCommonBehaviorData },
358     element to { p_CT_TLAnimVariant }?
359 p_CT_TLCommonMediaNodeData =
360
361     ## default value: 50%
362     attribute vol { a_ST_PositiveFixedPercentage }?,
363
364     ## default value: false

```



```

365 attribute mute { xsd:boolean }?,
366
367 ## default value: 1
368 attribute numSld { xsd:unsignedInt }?,
369
370 ## default value: true
371 attribute showWhenStopped { xsd:boolean }?,
372 element cTn { p_CT_TLCommonTimeNodeData },
373 element tgtEl { p_CT_TLTimeTargetElement }
374 p_CT_TLMediaNodeAudio =
375
376 ## default value: false
377 attribute isNarration { xsd:boolean }?,
378 element cMediaNode { p_CT_TLCommonMediaNodeData }
379 p_CT_TLMediaNodeVideo =
380
381 ## default value: false
382 attribute fullScrn { xsd:boolean }?,
383 element cMediaNode { p_CT_TLCommonMediaNodeData }
384 p_AG_TLBuild =
385 attribute spid { a_ST_ShapeID },
386 attribute grpId { xsd:unsignedInt },
387
388 ## default value: false
389 attribute uiExpand { xsd:boolean }?
390 p_CT_TLTemplate =
391
392 ## default value: 0
393 attribute lvl { xsd:unsignedInt }?,
394 element tnLst { p_CT_TimeNodeList }
395 p_CT_TLTemplateList = element tpl { p_CT_TLTemplate }*
396 p_ST_TLParaBuildType = "allAtOnce" | "p" | "cust" | "whole"
397 p_CT_TLBuildParagraph =
398   p_AG_TLBuild,
399
400 ## default value: whole
401 attribute build { p_ST_TLParaBuildType }?,
402
403 ## default value: 1
404 attribute bldLvl { xsd:unsignedInt }?,
405
406 ## default value: false
407 attribute animBg { xsd:boolean }?,
408
409 ## default value: true
410 attribute autoUpdateAnimBg { xsd:boolean }?,
411
412 ## default value: false
413 attribute rev { xsd:boolean }?,
414
415 ## default value: indefinite
416 attribute advAuto { p_ST_TLTime }?,
417 element tplLst { p_CT_TLTemplateList }?

```

```

418 p_ST_TLDiagramBuildType =
419     "whole"
420     | "depthByNode"
421     | "depthByBranch"
422     | "breadthByNode"
423     | "breadthByLvl"
424     | "cw"
425     | "cwIn"
426     | "cwOut"
427     | "ccw"
428     | "ccwIn"
429     | "ccwOut"
430     | "inByRing"
431     | "outByRing"
432     | "up"
433     | "down"
434     | "allAtOnce"
435     | "cust"
436 p_CT_TLBuildDiagram =
437     p_AG_TLBuild,
438
439     ## default value: whole
440     attribute bld { p_ST_TLDiagramBuildType }?
441 p_ST_TLOleChartBuildType =
442     "allAtOnce" | "series" | "category" | "seriesEl" | "categoryEl"
443 p_CT_TLOleBuildChart =
444     p_AG_TLBuild,
445
446     ## default value: allAtOnce
447     attribute bld { p_ST_TLOleChartBuildType }?,
448
449     ## default value: true
450     attribute animBg { xsd:boolean }?
451 p_CT_TLGraphicalObjectBuild =
452     p_AG_TLBuild,
453     (element bldAsOne { p_CT_Empty }
454     | element bldSub { a_CT_AnimationGraphicalObjectBuildProperties })
455 p_CT_BuildList =
456     (element bldP { p_CT_TLBuildParagraph }
457     | element bldDgm { p_CT_TLBuildDiagram }
458     | element bldOleChart { p_CT_TLOleBuildChart }
459     | element bldGraphic { p_CT_TLGraphicalObjectBuild })+
460 p_CT_SlideTiming =
461     element tnLst { p_CT_TimeNodeList }?,
462     element bldLst { p_CT_BuildList }?,
463     element extLst { p_CT_ExtensionListModify }?
464 p_CT_Empty = empty
465 p_ST_Name = xsd:string
466 p_ST_Direction = "horz" | "vert"
467 p_ST_Index = xsd:unsignedInt
468 p_CT_IndexRange =
469     attribute st { p_ST_Index },
470     attribute end { p_ST_Index }

```

```

471 p_CT_SlideRelationshipListEntry = r_id
472 p_CT_SlideRelationshipList =
473     element sld { p_CT_SlideRelationshipListEntry }*
474 p_CT_CustomShowId = attribute id { xsd:unsignedInt }
475 p_EG_SlideListChoice =
476     element sldAll { p_CT_Empty }
477     | element sldRg { p_CT_IndexRange }
478     | element custShow { p_CT_CustomShowId }
479 p_CT_CustomerData = r_id
480 p_CT_TagsData = r_id
481 p_CT_CustomerDataList =
482     (element custData { p_CT_CustomerData }*,
483     element tags { p_CT_TagsData }?)?
484 p_CT_Extension =
485     attribute uri { xsd:token }?,
486     p_CT_Extension_any
487 p_CT_Extension_any =
488     element * - (o:* | v:* | w10:* | x:*) {
489         anyAttribute*,
490         mixed { anyElement* }
491     }
492 p_EG_ExtensionList = element ext { p_CT_Extension }*
493 p_CT_ExtensionList = p_EG_ExtensionList?
494 p_CT_ExtensionListModify =
495
496     ## default value: false
497     attribute mod { xsd:boolean }?,
498     p_EG_ExtensionList?
499 p_CT_CommentAuthor =
500     attribute id { xsd:unsignedInt },
501     attribute name { p_ST_Name },
502     attribute initials { p_ST_Name },
503     attribute lastIdx { xsd:unsignedInt },
504     attribute clrIdx { xsd:unsignedInt },
505     element extLst { p_CT_ExtensionList }?
506 p_CT_CommentAuthorList = element cmAuthor { p_CT_CommentAuthor }*
507 p_cmAuthorLst = element cmAuthorLst { p_CT_CommentAuthorList }
508 p_CT_Comment =
509     attribute authorId { xsd:unsignedInt },
510     attribute dt { xsd:dateTime }?,
511     attribute idx { p_ST_Index },
512     element pos { a_CT_Point2D },
513     element text { xsd:string },
514     element extLst { p_CT_ExtensionListModify }?
515 p_CT_CommentList = element cm { p_CT_Comment }*
516 p_cmLst = element cmLst { p_CT_CommentList }
517 p_AG_Ole =
518     attribute spid { a_ST_ShapeID },
519     attribute name { xsd:string }?,
520
521     ## default value: false
522     attribute showAsIcon { xsd:boolean }?,
523     r_id?,

```

```

524     attribute imgW { a_ST_PositiveCoordinate32 }?,
525     attribute imgH { a_ST_PositiveCoordinate32 }?
526 p_ST_OleObjectFollowColorScheme = "none" | "full" | "textAndBackground"
527 p_CT_OleObjectEmbed =
528
529     ## default value: none
530     attribute followColorScheme { p_ST_OleObjectFollowColorScheme }?,
531     element extLst { p_CT_ExtensionList }?
532 p_CT_OleObjectLink =
533
534     ## default value: false
535     attribute updateAutomatic { xsd:boolean }?,
536     element extLst { p_CT_ExtensionList }?
537 p_CT_OleObject =
538     p_AG_Ole,
539     attribute progId { xsd:string }?,
540     (element embed { p_CT_OleObjectEmbed }
541      | element link { p_CT_OleObjectLink }
542      | element pic { p_CT_Picture })
543 p_oleObj = element oleObj { p_CT_OleObject }
544 p_CT_Control =
545     p_AG_Ole,
546     element extLst { p_CT_ExtensionList }?,
547     element pic { p_CT_Picture }?
548 p_CT_ControlList = element control { p_CT_Control }*
549 p_ST_SlideId =
550     xsd:unsignedInt { minInclusive = "256" maxExclusive = "2147483648" }
551 p_CT_SlideIdListEntry =
552     attribute id { p_ST_SlideId },
553     r_id,
554     element extLst { p_CT_ExtensionList }?
555 p_CT_SlideIdList = element sldId { p_CT_SlideIdListEntry }*
556 p_ST_SlideMasterId = xsd:unsignedInt { minInclusive = "2147483648" }
557 p_CT_SlideMasterIdListEntry =
558     attribute id { p_ST_SlideMasterId }?,
559     r_id,
560     element extLst { p_CT_ExtensionList }?
561 p_CT_SlideMasterIdList =
562     element sldMasterId { p_CT_SlideMasterIdListEntry }*
563 p_CT_NotesMasterIdListEntry =
564     r_id,
565     element extLst { p_CT_ExtensionList }?
566 p_CT_NotesMasterIdList =
567     element notesMasterId { p_CT_NotesMasterIdListEntry }?
568 p_CT_HandoutMasterIdListEntry =
569     r_id,
570     element extLst { p_CT_ExtensionList }?
571 p_CT_HandoutMasterIdList =
572     element handoutMasterId { p_CT_HandoutMasterIdListEntry }?
573 p_CT_EmbeddedFontDataId = r_id
574 p_CT_EmbeddedFontListEntry =
575     element font { a_CT_TextFont },
576     element regular { p_CT_EmbeddedFontDataId }?,

```

```

577     element bold { p_CT_EmbeddedFontDataId }?,
578     element italic { p_CT_EmbeddedFontDataId }?,
579     element boldItalic { p_CT_EmbeddedFontDataId }?
580 p_CT_EmbeddedFontList =
581     element embeddedFont { p_CT_EmbeddedFontListEntry }*
582 p_CT_SmartTags = r_id
583 p_CT_CustomShow =
584     attribute name { p_ST_Name },
585     attribute id { xsd:unsignedInt },
586     element sldLst { p_CT_SlideRelationshipList },
587     element extLst { p_CT_ExtensionList }?
588 p_CT_CustomShowList = element custShow { p_CT_CustomShow }*
589 p_ST_PhotoAlbumLayout =
590     "fitToSlide"
591     | "1pic"
592     | "2pic"
593     | "4pic"
594     | "1picTitle"
595     | "2picTitle"
596     | "4picTitle"
597 p_ST_PhotoAlbumFrameShape =
598     "frameStyle1"
599     | "frameStyle2"
600     | "frameStyle3"
601     | "frameStyle4"
602     | "frameStyle5"
603     | "frameStyle6"
604     | "frameStyle7"
605 p_CT_PhotoAlbum =
606
607     ## default value: false
608     attribute bw { xsd:boolean }?,
609
610     ## default value: false
611     attribute showCaptions { xsd:boolean }?,
612
613     ## default value: fitToSlide
614     attribute layout { p_ST_PhotoAlbumLayout }?,
615
616     ## default value: frameStyle1
617     attribute frame { p_ST_PhotoAlbumFrameShape }?,
618     element extLst { p_CT_ExtensionList }?
619 p_ST_SlideSizeCoordinate =
620     xsd:int {
621         minInclusive = "0"
622         minInclusive = "914400"
623         maxInclusive = "51206400"
624     }
625 p_ST_SlideSizeType =
626     "screen4x3"
627     | "letter"
628     | "A4"
629     | "35mm"

```

```

630 | "overhead"
631 | "banner"
632 | "custom"
633 | "ledger"
634 | "A3"
635 | "B4ISO"
636 | "B5ISO"
637 | "B4JIS"
638 | "B5JIS"
639 | "hagakiCard"
640 | "screen16x9"
641 | "screen16x10"
642 p_CT_SlideSize =
643     attribute cx { p_ST_SlideSizeCoordinate },
644     attribute cy { p_ST_SlideSizeCoordinate },
645
646     ## default value: custom
647     attribute type { p_ST_SlideSizeType }?
648 p_CT_Kinsoku =
649     attribute lang { xsd:string }?,
650     attribute invalStChars { xsd:string },
651     attribute invalEndChars { xsd:string }
652 p_ST_BookmarkIdSeed =
653     xsd:unsignedInt { minInclusive = "1" maxExclusive = "2147483648" }
654 p_CT_ModifyVerifier =
655     attribute algorithmName { xsd:string }?,
656     attribute hashValue { xsd:base64Binary }?,
657     attribute saltValue { xsd:base64Binary }?,
658     attribute spinValue { xsd:unsignedInt }?,
659     attribute cryptProviderType { s_ST_CryptProv }?,
660     attribute cryptAlgorithmClass { s_ST_AlgorithmClass }?,
661     attribute cryptAlgorithmType { s_ST_AlgorithmType }?,
662     attribute cryptAlgorithmSid { xsd:unsignedInt }?,
663     attribute spinCount { xsd:unsignedInt }?,
664     attribute saltData { xsd:base64Binary }?,
665     attribute hashData { xsd:base64Binary }?,
666     attribute cryptProvider { xsd:string }?,
667     attribute algIdExt { xsd:unsignedInt }?,
668     attribute algIdExtSource { xsd:string }?,
669     attribute cryptProviderTypeExt { xsd:unsignedInt }?,
670     attribute cryptProviderTypeExtSource { xsd:string }?
671 p_CT_Presentation =
672
673     ## default value: 50%
674     attribute serverZoom { a_ST_Percentage }?,
675
676     ## default value: 1
677     attribute firstSlideNum { xsd:int }?,
678
679     ## default value: true
680     attribute showSpecialPlsOnTitleSld { xsd:boolean }?,
681
682     ## default value: false

```

```

683 attribute rtl { xsd:boolean }?,
684
685 ## default value: false
686 attribute removePersonalInfoOnSave { xsd:boolean }?,
687
688 ## default value: false
689 attribute compatMode { xsd:boolean }?,
690
691 ## default value: true
692 attribute strictFirstAndLastChars { xsd:boolean }?,
693
694 ## default value: false
695 attribute embedTrueTypeFonts { xsd:boolean }?,
696
697 ## default value: false
698 attribute saveSubsetFonts { xsd:boolean }?,
699
700 ## default value: true
701 attribute autoCompressPictures { xsd:boolean }?,
702
703 ## default value: 1
704 attribute bookmarkIdSeed { p_ST_BookmarkIdSeed }?,
705 attribute conformance { s_ST_ConformanceClass }?,
706 element sldMasterIdLst { p_CT_SlideMasterIdList }?,
707 element notesMasterIdLst { p_CT_NotesMasterIdList }?,
708 element handoutMasterIdLst { p_CT_HandoutMasterIdList }?,
709 element sldIdLst { p_CT_SlideIdList }?,
710 element sldSz { p_CT_SlideSize }?,
711 element notesSz { a_CT_PositiveSize2D },
712 element smartTags { p_CT_SmartTags }?,
713 element embeddedFontLst { p_CT_EmbeddedFontList }?,
714 element custShowLst { p_CT_CustomShowList }?,
715 element photoAlbum { p_CT_PhotoAlbum }?,
716 element custDataLst { p_CT_CustomerDataList }?,
717 element kinsoku { p_CT_Kinsoku }?,
718 element defaultTextStyle { a_CT_TextListStyle }?,
719 element modifyVerifier { p_CT_ModifyVerifier }?,
720 element extLst { p_CT_ExtensionList }?
721 p_presentation = element presentation { p_CT_Presentation }
722 p_CT_HtmlPublishProperties =
723
724 ## default value: true
725 attribute showSpeakerNotes { xsd:boolean }?,
726 attribute target { xsd:string }?,
727 attribute title { xsd:string }?,
728 r_id,
729 p_EG_SlideListChoice,
730 element extLst { p_CT_ExtensionList }?
731 p_ST_WebColorType =
732 "none"
733 | "browser"
734 | "presentationText"
735 | "presentationAccent"

```

```

736 | "whiteTextOnBlack"
737 | "blackTextOnWhite"
738 p_ST_WebScreenSize =
739 "544x376"
740 | "640x480"
741 | "720x512"
742 | "800x600"
743 | "1024x768"
744 | "1152x882"
745 | "1152x900"
746 | "1280x1024"
747 | "1600x1200"
748 | "1800x1400"
749 | "1920x1200"
750 p_ST_WebEncoding = xsd:string
751 p_CT_WebProperties =
752
753 ## default value: false
754 attribute showAnimation { xsd:boolean }?,
755
756 ## default value: true
757 attribute resizeGraphics { xsd:boolean }?,
758
759 ## default value: false
760 attribute allowPng { xsd:boolean }?,
761
762 ## default value: false
763 attribute relyOnVm1 { xsd:boolean }?,
764
765 ## default value: true
766 attribute organizeInFolders { xsd:boolean }?,
767
768 ## default value: true
769 attribute useLongFileNames { xsd:boolean }?,
770
771 ## default value: 800x600
772 attribute imgSz { p_ST_WebScreenSize }?,
773 attribute encoding { p_ST_WebEncoding }?,
774
775 ## default value: whiteTextOnBlack
776 attribute clr { p_ST_WebColorType }?,
777 element extLst { p_CT_ExtensionList }?
778 p_ST_PrintWhat =
779 "slides"
780 | "handouts1"
781 | "handouts2"
782 | "handouts3"
783 | "handouts4"
784 | "handouts6"
785 | "handouts9"
786 | "notes"
787 | "outline"
788 p_ST_PrintColorMode = "bw" | "gray" | "clr"

```



```

789 p_CT_PrintProperties =
790
791   ## default value: slides
792   attribute prnWhat { p_ST_PrintWhat }?,
793
794   ## default value: clr
795   attribute clrMode { p_ST_PrintColorMode }?,
796
797   ## default value: false
798   attribute hiddenSlides { xsd:boolean }?,
799
800   ## default value: false
801   attribute scaleToFitPaper { xsd:boolean }?,
802
803   ## default value: false
804   attribute framesSlides { xsd:boolean }?,
805   element extLst { p_CT_ExtensionList }?
806 p_CT_ShowInfoBrowse =
807
808   ## default value: true
809   attribute showScrollbar { xsd:boolean }?
810 p_CT_ShowInfoKiosk =
811
812   ## default value: 300000
813   attribute restart { xsd:unsignedInt }?
814 p_EG_ShowType =
815   element present { p_CT_Empty }
816   | element browse { p_CT_ShowInfoBrowse }
817   | element kiosk { p_CT_ShowInfoKiosk }
818 p_CT_ShowProperties =
819
820   ## default value: false
821   attribute loop { xsd:boolean }?,
822
823   ## default value: false
824   attribute showNarration { xsd:boolean }?,
825
826   ## default value: true
827   attribute showAnimation { xsd:boolean }?,
828
829   ## default value: true
830   attribute useTimings { xsd:boolean }?,
831   (p_EG_ShowType?,
832   p_EG_SlideListChoice?,
833   element penClr { a_CT_Color }?,
834   element extLst { p_CT_ExtensionList }?)?
835 p_CT_PresentationProperties =
836   element htmlPubPr { p_CT_HtmlPublishProperties }?,
837   element webPr { p_CT_WebProperties }?,
838   element prnPr { p_CT_PrintProperties }?,
839   element showPr { p_CT_ShowProperties }?,
840   element clrMru { a_CT_ColorMRU }?,
841   element extLst { p_CT_ExtensionList }?

```

```

842 p_presentationPr =
843     element presentationPr { p_CT_PresentationProperties }
844 p_CT_HeaderFooter =
845
846     ## default value: true
847     attribute sldNum { xsd:boolean }?,
848
849     ## default value: true
850     attribute hdr { xsd:boolean }?,
851
852     ## default value: true
853     attribute ftr { xsd:boolean }?,
854
855     ## default value: true
856     attribute dt { xsd:boolean }?,
857     element extLst { p_CT_ExtensionListModify }?
858 p_ST_PlaceholderType =
859     "title"
860     | "body"
861     | "ctrTitle"
862     | "subTitle"
863     | "dt"
864     | "sldNum"
865     | "ftr"
866     | "hdr"
867     | "obj"
868     | "chart"
869     | "tbl"
870     | "clipArt"
871     | "dgm"
872     | "media"
873     | "sldImg"
874     | "pic"
875 p_ST_PlaceholderSize = "full" | "half" | "quarter"
876 p_CT_Placeholder =
877
878     ## default value: obj
879     attribute type { p_ST_PlaceholderType }?,
880
881     ## default value: horz
882     attribute orient { p_ST_Direction }?,
883
884     ## default value: full
885     attribute sz { p_ST_PlaceholderSize }?,
886
887     ## default value: 0
888     attribute idx { xsd:unsignedInt }?,
889
890     ## default value: false
891     attribute hasCustomPrompt { xsd:boolean }?,
892     element extLst { p_CT_ExtensionListModify }?
893 p_CT_ApplicationNonVisualDrawingProps =
894

```

```

895  ## default value: false
896  attribute isPhoto { xsd:boolean }?,
897
898  ## default value: false
899  attribute userDrawn { xsd:boolean }?,
900  element ph { p_CT_Placeholder }?,
901  a_EG_Media?,
902  element custDataLst { p_CT_CustomerDataList }?,
903  element extLst { p_CT_ExtensionList }?
904  p_CT_ShapeNonVisual =
905    element cNvPr { a_CT_NonVisualDrawingProps },
906    element cNvSpPr { a_CT_NonVisualDrawingShapeProps },
907    element nvPr { p_CT_ApplicationNonVisualDrawingProps }
908  p_CT_Shape =
909
910  ## default value: false
911  attribute useBgFill { xsd:boolean }?,
912  element nvSpPr { p_CT_ShapeNonVisual },
913  element spPr { a_CT_ShapeProperties },
914  element style { a_CT_ShapeStyle }?,
915  element txBdy { a_CT_TextBody }?,
916  element extLst { p_CT_ExtensionListModify }?
917  p_CT_ConnectorNonVisual =
918    element cNvPr { a_CT_NonVisualDrawingProps },
919    element cNvCxnSpPr { a_CT_NonVisualConnectorProperties },
920    element nvPr { p_CT_ApplicationNonVisualDrawingProps }
921  p_CT_Connector =
922    element nvCxnSpPr { p_CT_ConnectorNonVisual },
923    element spPr { a_CT_ShapeProperties },
924    element style { a_CT_ShapeStyle }?,
925    element extLst { p_CT_ExtensionListModify }?
926  p_CT_PictureNonVisual =
927    element cNvPr { a_CT_NonVisualDrawingProps },
928    element cNvPicPr { a_CT_NonVisualPictureProperties },
929    element nvPr { p_CT_ApplicationNonVisualDrawingProps }
930  p_CT_Picture =
931    element nvPicPr { p_CT_PictureNonVisual },
932    element blipFill { a_CT_BlipFillProperties },
933    element spPr { a_CT_ShapeProperties },
934    element style { a_CT_ShapeStyle }?,
935    element extLst { p_CT_ExtensionListModify }?
936  p_CT_GraphicalObjectFrameNonVisual =
937    element cNvPr { a_CT_NonVisualDrawingProps },
938    element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties },
939    element nvPr { p_CT_ApplicationNonVisualDrawingProps }
940  p_CT_GraphicalObjectFrame =
941    element nvGraphicFramePr { p_CT_GraphicalObjectFrameNonVisual },
942    element xfrm { a_CT_Transform2D },
943    a_graphic,
944    element extLst { p_CT_ExtensionListModify }?
945  p_CT_GroupShapeNonVisual =
946    element cNvPr { a_CT_NonVisualDrawingProps },
947    element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps },

```

```

948     element nvPr { p_CT_ApplicationNonVisualDrawingProps }
949 p_CT_GroupShape =
950     element nvGrpSpPr { p_CT_GroupShapeNonVisual },
951     element grpSpPr { a_CT_GroupShapeProperties },
952     (element sp { p_CT_Shape }
953     | element grpSp { p_CT_GroupShape }
954     | element graphicFrame { p_CT_GraphicalObjectFrame }
955     | element cxnSp { p_CT_Connector }
956     | element pic { p_CT_Picture }
957     | element contentPart { p_CT_Rel })*,
958     element extLst { p_CT_ExtensionListModify }?
959 p_CT_Rel = r_id
960 p_EG_TopLevelSlide = element clrMap { a_CT_ColorMapping }
961 p_EG_ChildSlide = element clrMapOvr { a_CT_ColorMappingOverride }?
962 p_AG_ChildSlide =
963
964     ## default value: true
965     attribute showMasterSp { xsd:boolean }?,
966
967     ## default value: true
968     attribute showMasterPhAnim { xsd:boolean }?
969 p_CT_BackgroundProperties =
970
971     ## default value: false
972     attribute shadeToTitle { xsd:boolean }?,
973     a_EG_FillProperties,
974     a_EG_EffectProperties?,
975     element extLst { p_CT_ExtensionList }?
976 p_EG_Background =
977     element bgPr { p_CT_BackgroundProperties }
978     | element bgRef { a_CT_StyleMatrixReference }
979 p_CT_Background =
980
981     ## default value: white
982     attribute bwMode { a_ST_BlackWhiteMode }?,
983     p_EG_Background
984 p_CT_CommonSlideData =
985     attribute name { xsd:string }?,
986     element bg { p_CT_Background }?,
987     element spTree { p_CT_GroupShape },
988     element custDataLst { p_CT_CustomerDataList }?,
989     element controls { p_CT_ControlList }?,
990     element extLst { p_CT_ExtensionList }?
991 p_CT_Slide =
992     p_AG_ChildSlide,
993
994     ## default value: true
995     attribute show { xsd:boolean }?,
996     element cSld { p_CT_CommonSlideData },
997     p_EG_ChildSlide?,
998     element transition { p_CT_SlideTransition }?,
999     element timing { p_CT_SlideTiming }?,
1000     element extLst { p_CT_ExtensionListModify }?

```

```

1001 p_sld = element sld { p_CT_Slide }
1002 p_ST_SlideLayoutType =
1003   "title"
1004   | "tx"
1005   | "twoColTx"
1006   | "tbl"
1007   | "txAndChart"
1008   | "chartAndTx"
1009   | "dgm"
1010   | "chart"
1011   | "txAndClipArt"
1012   | "clipArtAndTx"
1013   | "titleOnly"
1014   | "blank"
1015   | "txAndObj"
1016   | "objAndTx"
1017   | "objOnly"
1018   | "obj"
1019   | "txAndMedia"
1020   | "mediaAndTx"
1021   | "objOverTx"
1022   | "txOverObj"
1023   | "txAndTwoObj"
1024   | "twoObjAndTx"
1025   | "twoObjOverTx"
1026   | "fourObj"
1027   | "vertTx"
1028   | "clipArtAndVertTx"
1029   | "vertTitleAndTx"
1030   | "vertTitleAndTxOverChart"
1031   | "twoObj"
1032   | "objAndTwoObj"
1033   | "twoObjAndObj"
1034   | "cust"
1035   | "secHead"
1036   | "twoTxTwoObj"
1037   | "objTx"
1038   | "picTx"
1039 p_CT_SlideLayout =
1040   p_AG_ChildSlide,
1041   attribute matchingName { xsd:string }?,
1042
1043   ## default value: cust
1044   attribute type { p_ST_SlideLayoutType }?,
1045
1046   ## default value: false
1047   attribute preserve { xsd:boolean }?,
1048
1049   ## default value: false
1050   attribute userDrawn { xsd:boolean }?,
1051   element cSld { p_CT_CommonSlideData },
1052   p_EG_ChildSlide?,
1053   element transition { p_CT_SlideTransition }?,

```

```

1054     element timing { p_CT_SlideTiming }?,
1055     element hf { p_CT_HeaderFooter }?,
1056     element extLst { p_CT_ExtensionListModify }?
1057 p_sldLayout = element sldLayout { p_CT_SlideLayout }
1058 p_CT_SlideMasterTextStyles =
1059     element titleStyle { a_CT_TextListStyle }?,
1060     element bodyStyle { a_CT_TextListStyle }?,
1061     element otherStyle { a_CT_TextListStyle }?,
1062     element extLst { p_CT_ExtensionList }?
1063 p_ST_SlideLayoutId = xsd:unsignedInt { minInclusive = "2147483648" }
1064 p_CT_SlideLayoutIdListEntry =
1065     attribute id { p_ST_SlideLayoutId }?,
1066     r_id,
1067     element extLst { p_CT_ExtensionList }?
1068 p_CT_SlideLayoutIdList =
1069     element sldLayoutId { p_CT_SlideLayoutIdListEntry }*
1070 p_CT_SlideMaster =
1071
1072     ## default value: false
1073     attribute preserve { xsd:boolean }?,
1074     element cSld { p_CT_CommonSlideData },
1075     p_EG_TopLevelSlide,
1076     element sldLayoutIdLst { p_CT_SlideLayoutIdList }?,
1077     element transition { p_CT_SlideTransition }?,
1078     element timing { p_CT_SlideTiming }?,
1079     element hf { p_CT_HeaderFooter }?,
1080     element txStyles { p_CT_SlideMasterTextStyles }?,
1081     element extLst { p_CT_ExtensionListModify }?
1082 p_sldMaster = element sldMaster { p_CT_SlideMaster }
1083 p_CT_HandoutMaster =
1084     element cSld { p_CT_CommonSlideData },
1085     p_EG_TopLevelSlide,
1086     element hf { p_CT_HeaderFooter }?,
1087     element extLst { p_CT_ExtensionListModify }?
1088 p_handoutMaster = element handoutMaster { p_CT_HandoutMaster }
1089 p_CT_NotesMaster =
1090     element cSld { p_CT_CommonSlideData },
1091     p_EG_TopLevelSlide,
1092     element hf { p_CT_HeaderFooter }?,
1093     element notesStyle { a_CT_TextListStyle }?,
1094     element extLst { p_CT_ExtensionListModify }?
1095 p_notesMaster = element notesMaster { p_CT_NotesMaster }
1096 p_CT_NotesSlide =
1097     p_AG_ChildSlide,
1098     element cSld { p_CT_CommonSlideData },
1099     p_EG_ChildSlide?,
1100     element extLst { p_CT_ExtensionListModify }?
1101 p_notes = element notes { p_CT_NotesSlide }
1102 p_CT_SlideSyncProperties =
1103     attribute serverSldId { xsd:string },
1104     attribute serverSldModifiedTime { xsd:dateTime },
1105     attribute clientInsertedTime { xsd:dateTime },
1106     element extLst { p_CT_ExtensionList }?

```

```

1107 p_sldSyncPr = element sldSyncPr { p_CT_SlideSyncProperties }
1108 p_CT_StringTag =
1109     attribute name { xsd:string }?,
1110     attribute val { xsd:string }?
1111 p_CT_TagList = element tag { p_CT_StringTag }*
1112 p_tagLst = element tagLst { p_CT_TagList }
1113 p_ST_SplitterBarState = "minimized" | "restored" | "maximized"
1114 p_ST_ViewType =
1115     "sldView"
1116     | "sldMasterView"
1117     | "notesView"
1118     | "handoutView"
1119     | "notesMasterView"
1120     | "outlineView"
1121     | "sldSorterView"
1122     | "sldThumbnailView"
1123 p_CT_NormalViewPortion =
1124     attribute sz { a_ST_PositiveFixedPercentage },
1125
1126     ## default value: true
1127     attribute autoAdjust { xsd:boolean }?
1128 p_CT_NormalViewProperties =
1129
1130     ## default value: true
1131     attribute showOutlineIcons { xsd:boolean }?,
1132
1133     ## default value: false
1134     attribute snapVertSplitter { xsd:boolean }?,
1135
1136     ## default value: restored
1137     attribute vertBarState { p_ST_SplitterBarState }?,
1138
1139     ## default value: restored
1140     attribute horzBarState { p_ST_SplitterBarState }?,
1141
1142     ## default value: false
1143     attribute preferSingleView { xsd:boolean }?,
1144     element restoredLeft { p_CT_NormalViewPortion },
1145     element restoredTop { p_CT_NormalViewPortion },
1146     element extLst { p_CT_ExtensionList }?
1147 p_CT_CommonViewProperties =
1148
1149     ## default value: false
1150     attribute varScale { xsd:boolean }?,
1151     element scale { a_CT_Scale2D },
1152     element origin { a_CT_Point2D }
1153 p_CT_NotesTextViewProperties =
1154     element cViewPr { p_CT_CommonViewProperties },
1155     element extLst { p_CT_ExtensionList }?
1156 p_CT_OutlineViewSlideEntry =
1157     r_id,
1158
1159     ## default value: false

```

```

1160     attribute collapse { xsd:boolean }?
1161 p_CT_OutlineViewSlideList = element sld { p_CT_OutlineViewSlideEntry }*
1162 p_CT_OutlineViewProperties =
1163     element cViewPr { p_CT_CommonViewProperties },
1164     element sldLst { p_CT_OutlineViewSlideList }?,
1165     element extLst { p_CT_ExtensionList }?
1166 p_CT_SlideSorterViewProperties =
1167
1168     ## default value: true
1169     attribute showFormatting { xsd:boolean }?,
1170     element cViewPr { p_CT_CommonViewProperties },
1171     element extLst { p_CT_ExtensionList }?
1172 p_CT_Guide =
1173
1174     ## default value: vert
1175     attribute orient { p_ST_Direction }?,
1176
1177     ## default value: 0
1178     attribute pos { a_ST_Coordinate32 }?
1179 p_CT_GuideList = element guide { p_CT_Guide }*
1180 p_CT_CommonSlideViewProperties =
1181
1182     ## default value: true
1183     attribute snapToGrid { xsd:boolean }?,
1184
1185     ## default value: false
1186     attribute snapToObjects { xsd:boolean }?,
1187
1188     ## default value: false
1189     attribute showGuides { xsd:boolean }?,
1190     element cViewPr { p_CT_CommonViewProperties },
1191     element guideLst { p_CT_GuideList }?
1192 p_CT_SlideViewProperties =
1193     element cSldViewPr { p_CT_CommonSlideViewProperties },
1194     element extLst { p_CT_ExtensionList }?
1195 p_CT_NotesViewProperties =
1196     element cSldViewPr { p_CT_CommonSlideViewProperties },
1197     element extLst { p_CT_ExtensionList }?
1198 p_CT_ViewProperties =
1199
1200     ## default value: sldView
1201     attribute lastView { p_ST_ViewType }?,
1202
1203     ## default value: true
1204     attribute showComments { xsd:boolean }?,
1205     (element normalViewPr { p_CT_NormalViewProperties }?,
1206     element slideViewPr { p_CT_SlideViewProperties }?,
1207     element outlineViewPr { p_CT_OutlineViewProperties }?,
1208     element notesTextViewPr { p_CT_NotesTextViewProperties }?,
1209     element sorterViewPr { p_CT_SlideSorterViewProperties }?,
1210     element notesViewPr { p_CT_NotesViewProperties }?,
1211     element gridSpacing { a_CT_PositiveSize2D }?,
1212     element extLst { p_CT_ExtensionList }?)?

```



```
1213 p_viewPr = element viewPr { p_CT_ViewProperties }
```

B.3.1 Part Schemas

B.3.1.1 Comment Authors Part

```
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 include "dml-compatibility.rnc"
12 start = p_cmAuthorLst
```

B.3.1.2 Comments Part

```
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 include "dml-compatibility.rnc"
12 start = p_cmlst
```

B.3.1.3 Handout Master Part

```
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 include "dml-compatibility.rnc"
12 start = p_handoutMaster
```

B.3.1.4 Notes Master Part

```
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 include "dml-compatibility.rnc"
12 start = p_notesMaster

```

B.3.1.5 Notes Slide Part

```

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 include "dml-compatibility.rnc"
12 start = p_notes

```

B.3.1.6 Presentation Part

```

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 include "dml-compatibility.rnc"
12 start = p_presentation

```

B.3.1.7 Presentation Properties Part

```

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 include "dml-compatibility.rnc"
12 start = p_presentationPr

```

B.3.1.8 Slide Part

```

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 include "dml-compatibility.rnc"
12 start = p_sld

```

B.3.1.9 Slide Layout Part

```

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 include "dml-compatibility.rnc"
12 start = p_sldLayout

```

B.3.1.10 Slide Master Part

```

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 include "dml-compatibility.rnc"
12 start = p_sldMaster

```

B.3.1.11 Slide Synchronization Data Part

```

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 include "dml-compatibility.rnc"
12 start = p_sldSyncPr

```

B.3.1.12 User Defined Tags Part

```

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 include "dml-compatibility.rnc"
12 start = p_tagLst

```

B.3.1.13 View Properties Part

```

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 include "dml-compatibility.rnc"
12 start = p_viewPr

```

B.4 DrawingML - Framework

B.4.1 DrawingML - Main

```

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 noUngrp { 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_NonVisualDrawingProps =
648     attribute id { a_ST_DrawingElementId },
649     attribute name { xsd:string },
650     attribute descr { xsd:string }?,

```



```

651
652   ## default value: false
653   attribute hidden { xsd:boolean }?,
654   attribute title { xsd:string }?,
655   element hlinkClick { a_CT_Hyperlink }?,
656   element hlinkHover { a_CT_Hyperlink }?,
657   element extLst { a_CT_OfficeArtExtensionList }?
658 a_CT_NonVisualDrawingShapeProps =
659
660   ## default value: false
661   attribute txBox { xsd:boolean }?,
662   element spLocks { a_CT_ShapeLocking }?,
663   element extLst { a_CT_OfficeArtExtensionList }?
664 a_CT_NonVisualConnectorProperties =
665   element cxnSpLocks { a_CT_ConnectorLocking }?,
666   element stCxn { a_CT_Connection }?,
667   element endCxn { a_CT_Connection }?,
668   element extLst { a_CT_OfficeArtExtensionList }?
669 a_CT_NonVisualPictureProperties =
670
671   ## default value: true
672   attribute preferRelativeResize { xsd:boolean }?,
673   element picLocks { a_CT_PictureLocking }?,
674   element extLst { a_CT_OfficeArtExtensionList }?
675 a_CT_NonVisualGroupDrawingShapeProps =
676   element grpSpLocks { a_CT_GroupLocking }?,
677   element extLst { a_CT_OfficeArtExtensionList }?
678 a_CT_NonVisualGraphicFrameProperties =
679   element graphicFrameLocks { a_CT_GraphicalObjectFrameLocking }?,
680   element extLst { a_CT_OfficeArtExtensionList }?
681 a_CT_GraphicalObjectData =
682   attribute uri { xsd:token }?,
683   a_CT_GraphicalObjectData_any*
684 a_CT_GraphicalObjectData_any =
685   element * - (o:* | v:* | w10:* | x:*) {
686     anyAttribute*,
687     mixed { anyElement* }
688   }
689 a_CT_GraphicalObject = element graphicData { a_CT_GraphicalObjectData }
690 a_graphic = element graphic { a_CT_GraphicalObject }
691 a_ST_ChartBuildStep =
692   "category"
693   | "ptInCategory"
694   | "series"
695   | "ptInSeries"
696   | "allPts"
697   | "gridLegend"
698 a_ST_DgmBuildStep = "sp" | "bg"
699 a_CT_AnimationDgmElement =
700
701   ## default value: {00000000-0000-0000-0000-000000000000}
702   attribute id { s_ST_Guid }?,
703

```

```

704   ## default value: sp
705   attribute bldStep { a_ST_DgmBuildStep }?
706 a_CT_AnimationChartElement =
707
708   ## default value: -1
709   attribute seriesIdx { xsd:int }?,
710
711   ## default value: -1
712   attribute categoryIdx { xsd:int }?,
713   attribute bldStep { a_ST_ChartBuildStep }
714 a_CT_AnimationElementChoice =
715   element dgm { a_CT_AnimationDgmElement }
716   | element chart { a_CT_AnimationChartElement }
717 a_ST_AnimationBuildType = "allAtOnce"
718 a_ST_AnimationDgmOnlyBuildType = "one" | "lvlOne" | "lvlAtOnce"
719 a_ST_AnimationDgmBuildType =
720   a_ST_AnimationBuildType | a_ST_AnimationDgmOnlyBuildType
721 a_CT_AnimationDgmBuildProperties =
722
723   ## default value: allAtOnce
724   attribute bld { a_ST_AnimationDgmBuildType }?,
725
726   ## default value: false
727   attribute rev { xsd:boolean }?
728 a_ST_AnimationChartOnlyBuildType =
729   "series" | "category" | "seriesEl" | "categoryEl"
730 a_ST_AnimationChartBuildType =
731   a_ST_AnimationBuildType | a_ST_AnimationChartOnlyBuildType
732 a_CT_AnimationChartBuildProperties =
733
734   ## default value: allAtOnce
735   attribute bld { a_ST_AnimationChartBuildType }?,
736
737   ## default value: true
738   attribute animBg { xsd:boolean }?
739 a_CT_AnimationGraphicalObjectBuildProperties =
740   element bldDgm { a_CT_AnimationDgmBuildProperties }
741   | element bldChart { a_CT_AnimationChartBuildProperties }
742 a_CT_BackgroundFormatting = a_EG_FillProperties?, a_EG_EffectProperties?
743 a_CT_WholeE2oFormatting =
744   element ln { a_CT_LineProperties }?,
745   a_EG_EffectProperties?
746 a_CT_GvmlUseShapeRectangle = empty
747 a_CT_GvmlTextShape =
748   element txBody { a_CT_TextBody },
749   (element useSpRect { a_CT_GvmlUseShapeRectangle }
750   | element xfrm { a_CT_Transform2D } ),
751   element extLst { a_CT_OfficeArtExtensionList }?
752 a_CT_GvmlShapeNonVisual =
753   element cNvPr { a_CT_NonVisualDrawingProps },
754   element cNvSpPr { a_CT_NonVisualDrawingShapeProps }
755 a_CT_GvmlShape =
756   element nvSpPr { a_CT_GvmlShapeNonVisual },

```

```

757     element spPr { a_CT_ShapeProperties },
758     element txSp { a_CT_GvmlTextShape }?,
759     element style { a_CT_ShapeStyle }?,
760     element extLst { a_CT_OfficeArtExtensionList }?
761 a_CT_GvmlConnectorNonVisual =
762     element cNvPr { a_CT_NonVisualDrawingProps },
763     element cNvCxnSpPr { a_CT_NonVisualConnectorProperties }
764 a_CT_GvmlConnector =
765     element nvCxnSpPr { a_CT_GvmlConnectorNonVisual },
766     element spPr { a_CT_ShapeProperties },
767     element style { a_CT_ShapeStyle }?,
768     element extLst { a_CT_OfficeArtExtensionList }?
769 a_CT_GvmlPictureNonVisual =
770     element cNvPr { a_CT_NonVisualDrawingProps },
771     element cNvPicPr { a_CT_NonVisualPictureProperties }
772 a_CT_GvmlPicture =
773     element nvPicPr { a_CT_GvmlPictureNonVisual },
774     element blipFill { a_CT_BlipFillProperties },
775     element spPr { a_CT_ShapeProperties },
776     element style { a_CT_ShapeStyle }?,
777     element extLst { a_CT_OfficeArtExtensionList }?
778 a_CT_GvmlGraphicFrameNonVisual =
779     element cNvPr { a_CT_NonVisualDrawingProps },
780     element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }
781 a_CT_GvmlGraphicalObjectFrame =
782     element nvGraphicFramePr { a_CT_GvmlGraphicFrameNonVisual },
783     a_graphic,
784     element xfrm { a_CT_Transform2D },
785     element extLst { a_CT_OfficeArtExtensionList }?
786 a_CT_GvmlGroupShapeNonVisual =
787     element cNvPr { a_CT_NonVisualDrawingProps },
788     element cNvGrpSpPr { a_CT_NonVisualGroupDrawingShapeProps }
789 a_CT_GvmlGroupShape =
790     element nvGrpSpPr { a_CT_GvmlGroupShapeNonVisual },
791     element grpSpPr { a_CT_GroupShapeProperties },
792     (element txSp { a_CT_GvmlTextShape }
793      | element sp { a_CT_GvmlShape }
794      | element cxnSp { a_CT_GvmlConnector }
795      | element pic { a_CT_GvmlPicture }
796      | element graphicFrame { a_CT_GvmlGraphicalObjectFrame }
797      | element grpSp { a_CT_GvmlGroupShape })*,
798     element extLst { a_CT_OfficeArtExtensionList }?
799 a_ST_PresetCameraType =
800     "legacyObliqueTopLeft"
801     | "legacyObliqueTop"
802     | "legacyObliqueTopRight"
803     | "legacyObliqueLeft"
804     | "legacyObliqueFront"
805     | "legacyObliqueRight"
806     | "legacyObliqueBottomLeft"
807     | "legacyObliqueBottom"
808     | "legacyObliqueBottomRight"
809     | "legacyPerspectiveTopLeft"

```

```

810 | "legacyPerspectiveTop"
811 | "legacyPerspectiveTopRight"
812 | "legacyPerspectiveLeft"
813 | "legacyPerspectiveFront"
814 | "legacyPerspectiveRight"
815 | "legacyPerspectiveBottomLeft"
816 | "legacyPerspectiveBottom"
817 | "legacyPerspectiveBottomRight"
818 | "orthographicFront"
819 | "isometricTopUp"
820 | "isometricTopDown"
821 | "isometricBottomUp"
822 | "isometricBottomDown"
823 | "isometricLeftUp"
824 | "isometricLeftDown"
825 | "isometricRightUp"
826 | "isometricRightDown"
827 | "isometricOffAxis1Left"
828 | "isometricOffAxis1Right"
829 | "isometricOffAxis1Top"
830 | "isometricOffAxis2Left"
831 | "isometricOffAxis2Right"
832 | "isometricOffAxis2Top"
833 | "isometricOffAxis3Left"
834 | "isometricOffAxis3Right"
835 | "isometricOffAxis3Bottom"
836 | "isometricOffAxis4Left"
837 | "isometricOffAxis4Right"
838 | "isometricOffAxis4Bottom"
839 | "obliqueTopLeft"
840 | "obliqueTop"
841 | "obliqueTopRight"
842 | "obliqueLeft"
843 | "obliqueRight"
844 | "obliqueBottomLeft"
845 | "obliqueBottom"
846 | "obliqueBottomRight"
847 | "perspectiveFront"
848 | "perspectiveLeft"
849 | "perspectiveRight"
850 | "perspectiveAbove"
851 | "perspectiveBelow"
852 | "perspectiveAboveLeftFacing"
853 | "perspectiveAboveRightFacing"
854 | "perspectiveContrastingLeftFacing"
855 | "perspectiveContrastingRightFacing"
856 | "perspectiveHeroicLeftFacing"
857 | "perspectiveHeroicRightFacing"
858 | "perspectiveHeroicExtremeLeftFacing"
859 | "perspectiveHeroicExtremeRightFacing"
860 | "perspectiveRelaxed"
861 | "perspectiveRelaxedModerately"
862 a_ST_FOVAngle = xsd:int { minInclusive = "0" maxInclusive = "10800000" }

```

```

863 a_CT_Camera =
864     attribute prst { a_ST_PresetCameraType },
865     attribute fov { a_ST_FOVAngle }?,
866
867     ## default value: 100%
868     attribute zoom { a_ST_PositivePercentage }?,
869     element rot { a_CT_SphereCoords }?
870 a_ST_LightRigDirection =
871     "tl" | "t" | "tr" | "l" | "r" | "bl" | "b" | "br"
872 a_ST_LightRigType =
873     "legacyFlat1"
874     | "legacyFlat2"
875     | "legacyFlat3"
876     | "legacyFlat4"
877     | "legacyNormal1"
878     | "legacyNormal2"
879     | "legacyNormal3"
880     | "legacyNormal4"
881     | "legacyHarsh1"
882     | "legacyHarsh2"
883     | "legacyHarsh3"
884     | "legacyHarsh4"
885     | "threePt"
886     | "balanced"
887     | "soft"
888     | "harsh"
889     | "flood"
890     | "contrasting"
891     | "morning"
892     | "sunrise"
893     | "sunset"
894     | "chilly"
895     | "freezing"
896     | "flat"
897     | "twoPt"
898     | "glow"
899     | "brightRoom"
900 a_CT_LightRig =
901     attribute rig { a_ST_LightRigType },
902     attribute dir { a_ST_LightRigDirection },
903     element rot { a_CT_SphereCoords }?
904 a_CT_Scene3D =
905     element camera { a_CT_Camera },
906     element lightRig { a_CT_LightRig },
907     element backdrop { a_CT_Backdrop }?,
908     element extLst { a_CT_OfficeArtExtensionList }?
909 a_CT_Backdrop =
910     element anchor { a_CT_Point3D },
911     element norm { a_CT_Vector3D },
912     element up { a_CT_Vector3D },
913     element extLst { a_CT_OfficeArtExtensionList }?
914 a_ST_BevelPresetType =
915     "relaxedInset"

```

```

916 | "circle"
917 | "slope"
918 | "cross"
919 | "angle"
920 | "softRound"
921 | "convex"
922 | "coolSlant"
923 | "divot"
924 | "ribblet"
925 | "hardEdge"
926 | "artDeco"
927 a_CT_Bevel =
928
929   ## default value: 76200
930   attribute w { a_ST_PositiveCoordinate }?,
931
932   ## default value: 76200
933   attribute h { a_ST_PositiveCoordinate }?,
934
935   ## default value: circle
936   attribute prst { a_ST_BevelPresetType }?
937 a_ST_PresetMaterialType =
938 "legacyMatte"
939 | "legacyPlastic"
940 | "legacyMetal"
941 | "legacyWireframe"
942 | "matte"
943 | "plastic"
944 | "metal"
945 | "warmMatte"
946 | "translucentPowder"
947 | "powder"
948 | "dkEdge"
949 | "softEdge"
950 | "clear"
951 | "flat"
952 | "softmetal"
953 a_CT_Shape3D =
954
955   ## default value: 0
956   attribute z { a_ST_Coordinate }?,
957
958   ## default value: 0
959   attribute extrusionH { a_ST_PositiveCoordinate }?,
960
961   ## default value: 0
962   attribute contourW { a_ST_PositiveCoordinate }?,
963
964   ## default value: warmMatte
965   attribute prstMaterial { a_ST_PresetMaterialType }?,
966   element bevelT { a_CT_Bevel }?,
967   element bevelB { a_CT_Bevel }?,
968   element extrusionClr { a_CT_Color }?,

```

```

969     element contourClr { a_CT_Color }?,
970     element extLst { a_CT_OfficeArtExtensionList }?
971 a_CT_FlatText =
972
973     ## default value: 0
974     attribute z { a_ST_Coordinate }?
975 a_EG_Text3D =
976     element sp3d { a_CT_Shape3D }
977     | element flatTx { a_CT_FlatText }
978 a_CT_AlphaBiLevelEffect =
979     attribute thresh { a_ST_PositiveFixedPercentage }
980 a_CT_AlphaCeilingEffect = empty
981 a_CT_AlphaFloorEffect = empty
982 a_CT_AlphaInverseEffect = a_EG_ColorChoice?
983 a_CT_AlphaModulateFixedEffect =
984
985     ## default value: 100%
986     attribute amt { a_ST_PositivePercentage }?
987 a_CT_AlphaOutsetEffect =
988
989     ## default value: 0
990     attribute rad { a_ST_Coordinate }?
991 a_CT_AlphaReplaceEffect = attribute a { a_ST_PositiveFixedPercentage }
992 a_CT_BiLevelEffect = attribute thresh { a_ST_PositiveFixedPercentage }
993 a_CT_BlurEffect =
994
995     ## default value: 0
996     attribute rad { a_ST_PositiveCoordinate }?,
997
998     ## default value: true
999     attribute grow { xsd:boolean }?
1000 a_CT_ColorChangeEffect =
1001
1002     ## default value: true
1003     attribute useA { xsd:boolean }?,
1004     element clrFrom { a_CT_Color },
1005     element clrTo { a_CT_Color }
1006 a_CT_ColorReplaceEffect = a_EG_ColorChoice
1007 a_CT_DuotoneEffect = a_EG_ColorChoice+
1008 a_CT_GlowEffect =
1009
1010     ## default value: 0
1011     attribute rad { a_ST_PositiveCoordinate }?,
1012     a_EG_ColorChoice
1013 a_CT_GrayscaleEffect = empty
1014 a_CT_HSLEffect =
1015
1016     ## default value: 0
1017     attribute hue { a_ST_PositiveFixedAngle }?,
1018
1019     ## default value: 0%
1020     attribute sat { a_ST_FixedPercentage }?,
1021

```

```

1022  ## default value: 0%
1023  attribute lum { a_ST_FixedPercentage }?
1024  a_CT_InnerShadowEffect =
1025
1026  ## default value: 0
1027  attribute blurRad { a_ST_PositiveCoordinate }?,
1028
1029  ## default value: 0
1030  attribute dist { a_ST_PositiveCoordinate }?,
1031
1032  ## default value: 0
1033  attribute dir { a_ST_PositiveFixedAngle }?,
1034  a_EG_ColorChoice
1035  a_CT_LuminanceEffect =
1036
1037  ## default value: 0%
1038  attribute bright { a_ST_FixedPercentage }?,
1039
1040  ## default value: 0%
1041  attribute contrast { a_ST_FixedPercentage }?
1042  a_CT_OuterShadowEffect =
1043
1044  ## default value: 0
1045  attribute blurRad { a_ST_PositiveCoordinate }?,
1046
1047  ## default value: 0
1048  attribute dist { a_ST_PositiveCoordinate }?,
1049
1050  ## default value: 0
1051  attribute dir { a_ST_PositiveFixedAngle }?,
1052
1053  ## default value: 100%
1054  attribute sx { a_ST_Percentage }?,
1055
1056  ## default value: 100%
1057  attribute sy { a_ST_Percentage }?,
1058
1059  ## default value: 0
1060  attribute kx { a_ST_FixedAngle }?,
1061
1062  ## default value: 0
1063  attribute ky { a_ST_FixedAngle }?,
1064
1065  ## default value: b
1066  attribute algn { a_ST_RectAlignment }?,
1067
1068  ## default value: true
1069  attribute rotWithShape { xsd:boolean }?,
1070  a_EG_ColorChoice
1071  a_ST_PresetShadowVal =
1072  "shdw1"
1073  | "shdw2"
1074  | "shdw3"

```



```

1075 | "shdw4"
1076 | "shdw5"
1077 | "shdw6"
1078 | "shdw7"
1079 | "shdw8"
1080 | "shdw9"
1081 | "shdw10"
1082 | "shdw11"
1083 | "shdw12"
1084 | "shdw13"
1085 | "shdw14"
1086 | "shdw15"
1087 | "shdw16"
1088 | "shdw17"
1089 | "shdw18"
1090 | "shdw19"
1091 | "shdw20"
1092 a_CT_PresetShadowEffect =
1093     attribute prst { a_ST_PresetShadowVal },
1094
1095     ## default value: 0
1096     attribute dist { a_ST_PositiveCoordinate }?,
1097
1098     ## default value: 0
1099     attribute dir { a_ST_PositiveFixedAngle }?,
1100     a_EG_ColorChoice
1101 a_CT_ReflectionEffect =
1102
1103     ## default value: 0
1104     attribute blurRad { a_ST_PositiveCoordinate }?,
1105
1106     ## default value: 100%
1107     attribute stA { a_ST_PositiveFixedPercentage }?,
1108
1109     ## default value: 0%
1110     attribute stPos { a_ST_PositiveFixedPercentage }?,
1111
1112     ## default value: 0%
1113     attribute endA { a_ST_PositiveFixedPercentage }?,
1114
1115     ## default value: 100%
1116     attribute endPos { a_ST_PositiveFixedPercentage }?,
1117
1118     ## default value: 0
1119     attribute dist { a_ST_PositiveCoordinate }?,
1120
1121     ## default value: 0
1122     attribute dir { a_ST_PositiveFixedAngle }?,
1123
1124     ## default value: 5400000
1125     attribute fadeDir { a_ST_PositiveFixedAngle }?,
1126
1127     ## default value: 100%

```

```

1128     attribute sx { a_ST_Percentage }?,
1129
1130     ## default value: 100%
1131     attribute sy { a_ST_Percentage }?,
1132
1133     ## default value: 0
1134     attribute kx { a_ST_FixedAngle }?,
1135
1136     ## default value: 0
1137     attribute ky { a_ST_FixedAngle }?,
1138
1139     ## default value: b
1140     attribute algn { a_ST_RectAlignment }?,
1141
1142     ## default value: true
1143     attribute rotWithShape { xsd:boolean }?
1144 a_CT_RelativeOffsetEffect =
1145
1146     ## default value: 0%
1147     attribute tx { a_ST_Percentage }?,
1148
1149     ## default value: 0%
1150     attribute ty { a_ST_Percentage }?
1151 a_CT_SoftEdgesEffect = attribute rad { a_ST_PositiveCoordinate }
1152 a_CT_TintEffect =
1153
1154     ## default value: 0
1155     attribute hue { a_ST_PositiveFixedAngle }?,
1156
1157     ## default value: 0%
1158     attribute amt { a_ST_FixedPercentage }?
1159 a_CT_TransformEffect =
1160
1161     ## default value: 100%
1162     attribute sx { a_ST_Percentage }?,
1163
1164     ## default value: 100%
1165     attribute sy { a_ST_Percentage }?,
1166
1167     ## default value: 0
1168     attribute kx { a_ST_FixedAngle }?,
1169
1170     ## default value: 0
1171     attribute ky { a_ST_FixedAngle }?,
1172
1173     ## default value: 0
1174     attribute tx { a_ST_Coordinate }?,
1175
1176     ## default value: 0
1177     attribute ty { a_ST_Coordinate }?
1178 a_CT_NoFillProperties = empty
1179 a_CT_SolidColorFillProperties = a_EG_ColorChoice?
1180 a_CT_LinearShadeProperties =

```

```

1181     attribute ang { a_ST_PositiveFixedAngle }?,
1182     attribute scaled { xsd:boolean }?
1183 a_ST_PathShadeType = "shape" | "circle" | "rect"
1184 a_CT_PathShadeProperties =
1185     attribute path { a_ST_PathShadeType }?,
1186     element fillToRect { a_CT_RelativeRect }?
1187 a_EG_ShadeProperties =
1188     element lin { a_CT_LinearShadeProperties }
1189     | element path { a_CT_PathShadeProperties }
1190 a_ST_TileFlipMode = "none" | "x" | "y" | "xy"
1191 a_CT_GradientStop =
1192     attribute pos { a_ST_PositiveFixedPercentage },
1193     a_EG_ColorChoice
1194 a_CT_GradientStopList = element gs { a_CT_GradientStop }+
1195 a_CT_GradientFillProperties =
1196     attribute flip { a_ST_TileFlipMode }?,
1197     attribute rotWithShape { xsd:boolean }?,
1198     element gsLst { a_CT_GradientStopList }?,
1199     a_EG_ShadeProperties?,
1200     element tileRect { a_CT_RelativeRect }?
1201 a_CT_TileInfoProperties =
1202     attribute tx { a_ST_Coordinate }?,
1203     attribute ty { a_ST_Coordinate }?,
1204     attribute sx { a_ST_Percentage }?,
1205     attribute sy { a_ST_Percentage }?,
1206     attribute flip { a_ST_TileFlipMode }?,
1207     attribute algn { a_ST_RectAlignment }?
1208 a_CT_StretchInfoProperties = element fillRect { a_CT_RelativeRect }?
1209 a_EG_FillModeProperties =
1210     element tile { a_CT_TileInfoProperties }
1211     | element stretch { a_CT_StretchInfoProperties }
1212 a_ST_BlipCompression = "email" | "screen" | "print" | "hqprint" | "none"
1213 a_CT_Blip =
1214     a_AG_Blob,
1215
1216     ## default value: none
1217     attribute cstate { a_ST_BlipCompression }?,
1218     (element alphaBiLevel { a_CT_AlphaBiLevelEffect }
1219     | element alphaCeiling { a_CT_AlphaCeilingEffect }
1220     | element alphaFloor { a_CT_AlphaFloorEffect }
1221     | element alphaInv { a_CT_AlphaInverseEffect }
1222     | element alphaMod { a_CT_AlphaModulateEffect }
1223     | element alphaModFix { a_CT_AlphaModulateFixedEffect }
1224     | element alphaRepl { a_CT_AlphaReplaceEffect }
1225     | element biLevel { a_CT_BiLevelEffect }
1226     | element blur { a_CT_BlurEffect }
1227     | element clrChange { a_CT_ColorChangeEffect }
1228     | element clrRepl { a_CT_ColorReplaceEffect }
1229     | element duotone { a_CT_DuotoneEffect }
1230     | element fillOverlay { a_CT_FillOverlayEffect }
1231     | element grayscl { a_CT_GrayscaleEffect }
1232     | element hsl { a_CT_HSLEffect }
1233     | element lum { a_CT_LuminanceEffect }

```

```

1234 | element tint { a_CT_TintEffect })*,
1235 element extLst { a_CT_OfficeArtExtensionList }?
1236 a_CT_BlipFillProperties =
1237 attribute dpi { xsd:unsignedInt }?,
1238 attribute rotWithShape { xsd:boolean }?,
1239 element blip { a_CT_Blip }?,
1240 element srcRect { a_CT_RelativeRect }?,
1241 a_EG_FillModeProperties?
1242 a_ST_PresetPatternVal =
1243 "pct5"
1244 | "pct10"
1245 | "pct20"
1246 | "pct25"
1247 | "pct30"
1248 | "pct40"
1249 | "pct50"
1250 | "pct60"
1251 | "pct70"
1252 | "pct75"
1253 | "pct80"
1254 | "pct90"
1255 | "horz"
1256 | "vert"
1257 | "ltHorz"
1258 | "ltVert"
1259 | "dkHorz"
1260 | "dkVert"
1261 | "narHorz"
1262 | "narVert"
1263 | "dashHorz"
1264 | "dashVert"
1265 | "cross"
1266 | "dnDiag"
1267 | "upDiag"
1268 | "ltDnDiag"
1269 | "ltUpDiag"
1270 | "dkDnDiag"
1271 | "dkUpDiag"
1272 | "wdDnDiag"
1273 | "wdUpDiag"
1274 | "dashDnDiag"
1275 | "dashUpDiag"
1276 | "diagCross"
1277 | "smCheck"
1278 | "lgCheck"
1279 | "smGrid"
1280 | "lgGrid"
1281 | "dotGrid"
1282 | "smConfetti"
1283 | "lgConfetti"
1284 | "horzBrick"
1285 | "diagBrick"
1286 | "solidDmnd"

```

```

1287 | "openDmnd"
1288 | "dotDmnd"
1289 | "plaid"
1290 | "sphere"
1291 | "weave"
1292 | "divot"
1293 | "shingle"
1294 | "wave"
1295 | "trellis"
1296 | "zigZag"
1297 a_CT_PatternFillProperties =
1298   attribute prst { a_ST_PresetPatternVal }?,
1299   element fgClr { a_CT_Color }?,
1300   element bgClr { a_CT_Color }?
1301 a_CT_GroupFillProperties = empty
1302 a_EG_FillProperties =
1303   element noFill { a_CT_NoFillProperties }
1304   | element solidFill { a_CT_SolidColorFillProperties }
1305   | element gradFill { a_CT_GradientFillProperties }
1306   | element blipFill { a_CT_BlipFillProperties }
1307   | element pattFill { a_CT_PatternFillProperties }
1308   | element grpFill { a_CT_GroupFillProperties }
1309 a_CT_FillProperties = a_EG_FillProperties
1310 a_CT_FillEffect = a_EG_FillProperties
1311 a_ST_BlendMode = "over" | "mult" | "screen" | "darken" | "lighten"
1312 a_CT_FillOverlayEffect =
1313   attribute blend { a_ST_BlendMode },
1314   a_EG_FillProperties
1315 a_CT_EffectReference = attribute ref { xsd:token }?
1316 a_EG_Effect =
1317   element cont { a_CT_EffectContainer }
1318   | element effect { a_CT_EffectReference }
1319   | element alphaBiLevel { a_CT_AlphaBiLevelEffect }
1320   | element alphaCeiling { a_CT_AlphaCeilingEffect }
1321   | element alphaFloor { a_CT_AlphaFloorEffect }
1322   | element alphaInv { a_CT_AlphaInverseEffect }
1323   | element alphaMod { a_CT_AlphaModulateEffect }
1324   | element alphaModFix { a_CT_AlphaModulateFixedEffect }
1325   | element alphaOutset { a_CT_AlphaOutsetEffect }
1326   | element alphaRepl { a_CT_AlphaReplaceEffect }
1327   | element biLevel { a_CT_BiLevelEffect }
1328   | element blend { a_CT_BlendEffect }
1329   | element blur { a_CT_BlurEffect }
1330   | element clrChange { a_CT_ColorChangeEffect }
1331   | element clrRepl { a_CT_ColorReplaceEffect }
1332   | element duotone { a_CT_DuotoneEffect }
1333   | element fill { a_CT_FillEffect }
1334   | element fillOverlay { a_CT_FillOverlayEffect }
1335   | element glow { a_CT_GlowEffect }
1336   | element grayscl { a_CT_GrayscaleEffect }
1337   | element hsl { a_CT_HSLEffect }
1338   | element innerShdw { a_CT_InnerShadowEffect }
1339   | element lum { a_CT_LuminanceEffect }

```

```

1340 | element outerShdw { a_CT_OuterShadowEffect }
1341 | element prstShdw { a_CT_PresetShadowEffect }
1342 | element reflection { a_CT_ReflectionEffect }
1343 | element relOff { a_CT_RelativeOffsetEffect }
1344 | element softEdge { a_CT_SoftEdgesEffect }
1345 | element tint { a_CT_TintEffect }
1346 | element xfrm { a_CT_TransformEffect }
1347 a_ST_EffectContainerType = "sib" | "tree"
1348 a_CT_EffectContainer =
1349
1350     ## default value: sib
1351     attribute type { a_ST_EffectContainerType }?,
1352     attribute name { xsd:token }?,
1353     a_EG_Effect*
1354 a_CT_AlphaModulateEffect = element cont { a_CT_EffectContainer }
1355 a_CT_BlendEffect =
1356     attribute blend { a_ST_BlendMode },
1357     element cont { a_CT_EffectContainer }
1358 a_CT_EffectList =
1359     element blur { a_CT_BlurEffect }?,
1360     element fillOverlay { a_CT_FillOverlayEffect }?,
1361     element glow { a_CT_GlowEffect }?,
1362     element innerShdw { a_CT_InnerShadowEffect }?,
1363     element outerShdw { a_CT_OuterShadowEffect }?,
1364     element prstShdw { a_CT_PresetShadowEffect }?,
1365     element reflection { a_CT_ReflectionEffect }?,
1366     element softEdge { a_CT_SoftEdgesEffect }?
1367 a_EG_EffectProperties =
1368     element effectLst { a_CT_EffectList }
1369     | element effectDag { a_CT_EffectContainer }
1370 a_CT_EffectProperties = a_EG_EffectProperties
1371 a_blip = element blip { a_CT_Blip }
1372 a_ST_ShapeType =
1373     "line"
1374     | "lineInv"
1375     | "triangle"
1376     | "rtTriangle"
1377     | "rect"
1378     | "diamond"
1379     | "parallelogram"
1380     | "trapezoid"
1381     | "nonIsoscelesTrapezoid"
1382     | "pentagon"
1383     | "hexagon"
1384     | "heptagon"
1385     | "octagon"
1386     | "decagon"
1387     | "dodecagon"
1388     | "star4"
1389     | "star5"
1390     | "star6"
1391     | "star7"
1392     | "star8"

```

1393	"star10"
1394	"star12"
1395	"star16"
1396	"star24"
1397	"star32"
1398	"roundRect"
1399	"round1Rect"
1400	"round2SameRect"
1401	"round2DiagRect"
1402	"snipRoundRect"
1403	"snip1Rect"
1404	"snip2SameRect"
1405	"snip2DiagRect"
1406	"plaque"
1407	"ellipse"
1408	"teardrop"
1409	"homePlate"
1410	"chevron"
1411	"pieWedge"
1412	"pie"
1413	"blockArc"
1414	"donut"
1415	"noSmoking"
1416	"rightArrow"
1417	"leftArrow"
1418	"upArrow"
1419	"downArrow"
1420	"stripedRightArrow"
1421	"notchedRightArrow"
1422	"bentUpArrow"
1423	"leftRightArrow"
1424	"upDownArrow"
1425	"leftUpArrow"
1426	"leftRightUpArrow"
1427	"quadArrow"
1428	"leftArrowCallout"
1429	"rightArrowCallout"
1430	"upArrowCallout"
1431	"downArrowCallout"
1432	"leftRightArrowCallout"
1433	"upDownArrowCallout"
1434	"quadArrowCallout"
1435	"bentArrow"
1436	"uturnArrow"
1437	"circularArrow"
1438	"leftCircularArrow"
1439	"leftRightCircularArrow"
1440	"curvedRightArrow"
1441	"curvedLeftArrow"
1442	"curvedUpArrow"
1443	"curvedDownArrow"
1444	"swooshArrow"
1445	"cube"

1446	"can"
1447	"lightningBolt"
1448	"heart"
1449	"sun"
1450	"moon"
1451	"smileyFace"
1452	"irregularSeal1"
1453	"irregularSeal2"
1454	"foldedCorner"
1455	"bevel"
1456	"frame"
1457	"halfFrame"
1458	"corner"
1459	"diagStripe"
1460	"chord"
1461	"arc"
1462	"leftBracket"
1463	"rightBracket"
1464	"leftBrace"
1465	"rightBrace"
1466	"bracketPair"
1467	"bracePair"
1468	"straightConnector1"
1469	"bentConnector2"
1470	"bentConnector3"
1471	"bentConnector4"
1472	"bentConnector5"
1473	"curvedConnector2"
1474	"curvedConnector3"
1475	"curvedConnector4"
1476	"curvedConnector5"
1477	"callout1"
1478	"callout2"
1479	"callout3"
1480	"accentCallout1"
1481	"accentCallout2"
1482	"accentCallout3"
1483	"borderCallout1"
1484	"borderCallout2"
1485	"borderCallout3"
1486	"accentBorderCallout1"
1487	"accentBorderCallout2"
1488	"accentBorderCallout3"
1489	"wedgeRectCallout"
1490	"wedgeRoundRectCallout"
1491	"wedgeEllipseCallout"
1492	"cloudCallout"
1493	"cloud"
1494	"ribbon"
1495	"ribbon2"
1496	"ellipseRibbon"
1497	"ellipseRibbon2"
1498	"leftRightRibbon"

1499	"verticalScroll"
1500	"horizontalScroll"
1501	"wave"
1502	"doubleWave"
1503	"plus"
1504	"flowChartProcess"
1505	"flowChartDecision"
1506	"flowChartInputOutput"
1507	"flowChartPredefinedProcess"
1508	"flowChartInternalStorage"
1509	"flowChartDocument"
1510	"flowChartMultidocument"
1511	"flowChartTerminator"
1512	"flowChartPreparation"
1513	"flowChartManualInput"
1514	"flowChartManualOperation"
1515	"flowChartConnector"
1516	"flowChartPunchedCard"
1517	"flowChartPunchedTape"
1518	"flowChartSummingJunction"
1519	"flowChartOr"
1520	"flowChartCollate"
1521	"flowChartSort"
1522	"flowChartExtract"
1523	"flowChartMerge"
1524	"flowChartOfflineStorage"
1525	"flowChartOnlineStorage"
1526	"flowChartMagneticTape"
1527	"flowChartMagneticDisk"
1528	"flowChartMagneticDrum"
1529	"flowChartDisplay"
1530	"flowChartDelay"
1531	"flowChartAlternateProcess"
1532	"flowChartOffpageConnector"
1533	"actionButtonBlank"
1534	"actionButtonHome"
1535	"actionButtonHelp"
1536	"actionButtonInformation"
1537	"actionButtonForwardNext"
1538	"actionButtonBackPrevious"
1539	"actionButtonEnd"
1540	"actionButtonBeginning"
1541	"actionButtonReturn"
1542	"actionButtonDocument"
1543	"actionButtonSound"
1544	"actionButtonMovie"
1545	"gear6"
1546	"gear9"
1547	"funnel"
1548	"mathPlus"
1549	"mathMinus"
1550	"mathMultiply"
1551	"mathDivide"

```

1552 | "mathEqual"
1553 | "mathNotEqual"
1554 | "cornerTabs"
1555 | "squareTabs"
1556 | "plaqueTabs"
1557 | "chartX"
1558 | "chartStar"
1559 | "chartPlus"
1560 a_ST_TextShapeType =
1561   "textNoShape"
1562   | "textPlain"
1563   | "textStop"
1564   | "textTriangle"
1565   | "textTriangleInverted"
1566   | "textChevron"
1567   | "textChevronInverted"
1568   | "textRingInside"
1569   | "textRingOutside"
1570   | "textArchUp"
1571   | "textArchDown"
1572   | "textCircle"
1573   | "textButton"
1574   | "textArchUpPour"
1575   | "textArchDownPour"
1576   | "textCirclePour"
1577   | "textButtonPour"
1578   | "textCurveUp"
1579   | "textCurveDown"
1580   | "textCanUp"
1581   | "textCanDown"
1582   | "textWave1"
1583   | "textWave2"
1584   | "textDoubleWave1"
1585   | "textWave4"
1586   | "textInflate"
1587   | "textDeflate"
1588   | "textInflateBottom"
1589   | "textDeflateBottom"
1590   | "textInflateTop"
1591   | "textDeflateTop"
1592   | "textDeflateInflate"
1593   | "textDeflateInflateDeflate"
1594   | "textFadeRight"
1595   | "textFadeLeft"
1596   | "textFadeUp"
1597   | "textFadeDown"
1598   | "textSlantUp"
1599   | "textSlantDown"
1600   | "textCascadeUp"
1601   | "textCascadeDown"
1602 a_ST_GeomGuideName = xsd:token
1603 a_ST_GeomGuideFormula = xsd:string
1604 a_CT_GeomGuide =

```

```

1605     attribute name { a_ST_GeomGuideName },
1606     attribute fmla { a_ST_GeomGuideFormula }
1607 a_CT_GeomGuideList = element gd { a_CT_GeomGuide }*
1608 a_ST_AdjCoordinate = a_ST_Coordinate | a_ST_GeomGuideName
1609 a_ST_AdjAngle = a_ST_Angle | a_ST_GeomGuideName
1610 a_CT_AdjPoint2D =
1611     attribute x { a_ST_AdjCoordinate },
1612     attribute y { a_ST_AdjCoordinate }
1613 a_CT_GeomRect =
1614     attribute l { a_ST_AdjCoordinate },
1615     attribute t { a_ST_AdjCoordinate },
1616     attribute r { a_ST_AdjCoordinate },
1617     attribute b { a_ST_AdjCoordinate }
1618 a_CT_XYAdjustHandle =
1619     attribute gdRefX { a_ST_GeomGuideName }?,
1620     attribute minX { a_ST_AdjCoordinate }?,
1621     attribute maxX { a_ST_AdjCoordinate }?,
1622     attribute gdRefY { a_ST_GeomGuideName }?,
1623     attribute minY { a_ST_AdjCoordinate }?,
1624     attribute maxY { a_ST_AdjCoordinate }?,
1625     element pos { a_CT_AdjPoint2D }
1626 a_CT_PolarAdjustHandle =
1627     attribute gdRefR { a_ST_GeomGuideName }?,
1628     attribute minR { a_ST_AdjCoordinate }?,
1629     attribute maxR { a_ST_AdjCoordinate }?,
1630     attribute gdRefAng { a_ST_GeomGuideName }?,
1631     attribute minAng { a_ST_AdjAngle }?,
1632     attribute maxAng { a_ST_AdjAngle }?,
1633     element pos { a_CT_AdjPoint2D }
1634 a_CT_ConnectionSite =
1635     attribute ang { a_ST_AdjAngle },
1636     element pos { a_CT_AdjPoint2D }
1637 a_CT_AdjustHandleList =
1638     (element ahXY { a_CT_XYAdjustHandle }
1639     | element ahPolar { a_CT_PolarAdjustHandle })*
1640 a_CT_ConnectionSiteList = element cxn { a_CT_ConnectionSite }*
1641 a_CT_Connection =
1642     attribute id { a_ST_DrawingElementId },
1643     attribute idx { xsd:unsignedInt }
1644 a_CT_Path2DMoveTo = element pt { a_CT_AdjPoint2D }
1645 a_CT_Path2DLineTo = element pt { a_CT_AdjPoint2D }
1646 a_CT_Path2DArcTo =
1647     attribute wR { a_ST_AdjCoordinate },
1648     attribute hR { a_ST_AdjCoordinate },
1649     attribute stAng { a_ST_AdjAngle },
1650     attribute swAng { a_ST_AdjAngle }
1651 a_CT_Path2DQuadBezierTo = element pt { a_CT_AdjPoint2D }+
1652 a_CT_Path2DCubicBezierTo = element pt { a_CT_AdjPoint2D }+
1653 a_CT_Path2DClose = empty
1654 a_ST_PathFillMode =
1655     "none" | "norm" | "lighten" | "lightenLess" | "darken" | "darkenLess"
1656 a_CT_Path2D =
1657

```

```

1658 ## default value: 0
1659 attribute w { a_ST_PositiveCoordinate }?,
1660
1661 ## default value: 0
1662 attribute h { a_ST_PositiveCoordinate }?,
1663
1664 ## default value: norm
1665 attribute fill { a_ST_PathFillMode }?,
1666
1667 ## default value: true
1668 attribute stroke { xsd:boolean }?,
1669
1670 ## default value: true
1671 attribute extrusionOk { xsd:boolean }?,
1672 (element close { a_CT_Path2DClose }
1673   | element moveTo { a_CT_Path2DMoveTo }
1674   | element lnTo { a_CT_Path2DLineTo }
1675   | element arcTo { a_CT_Path2DArcTo }
1676   | element quadBezTo { a_CT_Path2DQuadBezierTo }
1677   | element cubicBezTo { a_CT_Path2DCubicBezierTo })*
1678 a_CT_Path2DList = element path { a_CT_Path2D }*
1679 a_CT_PresetGeometry2D =
1680   attribute prst { a_ST_ShapeType },
1681   element avLst { a_CT_GeomGuideList }?
1682 a_CT_PresetTextShape =
1683   attribute prst { a_ST_TextShapeType },
1684   element avLst { a_CT_GeomGuideList }?
1685 a_CT_CustomGeometry2D =
1686   element avLst { a_CT_GeomGuideList }?,
1687   element gdLst { a_CT_GeomGuideList }?,
1688   element ahLst { a_CT_AdjustHandleList }?,
1689   element cxnLst { a_CT_ConnectionSiteList }?,
1690   element rect { a_CT_GeomRect }?,
1691   element pathLst { a_CT_Path2DList }
1692 a_EG_Geometry =
1693   element custGeom { a_CT_CustomGeometry2D }
1694   | element prstGeom { a_CT_PresetGeometry2D }
1695 a_EG_TextGeometry =
1696   element custGeom { a_CT_CustomGeometry2D }
1697   | element prstTxWarp { a_CT_PresetTextShape }
1698 a_ST_LineEndType =
1699   "none" | "triangle" | "stealth" | "diamond" | "oval" | "arrow"
1700 a_ST_LineEndWidth = "sm" | "med" | "lg"
1701 a_ST_LineEndLength = "sm" | "med" | "lg"
1702 a_CT_LineEndProperties =
1703   attribute type { a_ST_LineEndType }?,
1704   attribute w { a_ST_LineEndWidth }?,
1705   attribute len { a_ST_LineEndLength }?
1706 a_EG_LineFillProperties =
1707   element noFill { a_CT_NoFillProperties }
1708   | element solidFill { a_CT_SolidColorFillProperties }
1709   | element gradFill { a_CT_GradientFillProperties }
1710   | element pattFill { a_CT_PatternFillProperties }

```

```

1711 a_CT_LineJoinBevel = empty
1712 a_CT_LineJoinRound = empty
1713 a_CT_LineJoinMiterProperties =
1714     attribute lim { a_ST_PositivePercentage }?
1715 a_EG_LineJoinProperties =
1716     element round { a_CT_LineJoinRound }
1717     | element bevel { a_CT_LineJoinBevel }
1718     | element miter { a_CT_LineJoinMiterProperties }
1719 a_ST_PresetLineDashVal =
1720     "solid"
1721     | "dot"
1722     | "dash"
1723     | "lgDash"
1724     | "dashDot"
1725     | "lgDashDot"
1726     | "lgDashDotDot"
1727     | "sysDash"
1728     | "sysDot"
1729     | "sysDashDot"
1730     | "sysDashDotDot"
1731 a_CT_PresetLineDashProperties =
1732     attribute val { a_ST_PresetLineDashVal }?
1733 a_CT_DashStop =
1734     attribute d { a_ST_PositivePercentage },
1735     attribute sp { a_ST_PositivePercentage }
1736 a_CT_DashStopList = element ds { a_CT_DashStop }*
1737 a_EG_LineDashProperties =
1738     element prstDash { a_CT_PresetLineDashProperties }
1739     | element custDash { a_CT_DashStopList }
1740 a_ST_LineCap = "rnd" | "sq" | "flat"
1741 a_ST_LineWidth =
1742     xsd:int { minInclusive = "0" maxInclusive = "20116800" }
1743 a_ST_PenAlignment = "ctr" | "in"
1744 a_ST_CompoundLine = "sng" | "dbl" | "thickThin" | "thinThick" | "tri"
1745 a_CT_LineProperties =
1746     attribute w { a_ST_LineWidth }?,
1747     attribute cap { a_ST_LineCap }?,
1748     attribute cmpd { a_ST_CompoundLine }?,
1749     attribute algn { a_ST_PenAlignment }?,
1750     a_EG_LineFillProperties?,
1751     a_EG_LineDashProperties?,
1752     a_EG_LineJoinProperties?,
1753     element headEnd { a_CT_LineEndProperties }?,
1754     element tailEnd { a_CT_LineEndProperties }?,
1755     element extLst { a_CT_OfficeArtExtensionList }?
1756 a_ST_ShapeID = xsd:token
1757 a_CT_ShapeProperties =
1758     attribute bwMode { a_ST_BlackWhiteMode }?,
1759     element xfrm { a_CT_Transform2D }?,
1760     a_EG_Geometry?,
1761     a_EG_FillProperties?,
1762     element ln { a_CT_LineProperties }?,
1763     a_EG_EffectProperties?,

```

```

1764     element scene3d { a_CT_Scene3D }?,
1765     element sp3d { a_CT_Shape3D }?,
1766     element extLst { a_CT_OfficeArtExtensionList }?
1767 a_CT_GroupShapeProperties =
1768     attribute bwMode { a_ST_BlackWhiteMode }?,
1769     element xfrm { a_CT_GroupTransform2D }?,
1770     a_EG_FillProperties?,
1771     a_EG_EffectProperties?,
1772     element scene3d { a_CT_Scene3D }?,
1773     element extLst { a_CT_OfficeArtExtensionList }?
1774 a_CT_StyleMatrixReference =
1775     attribute idx { a_ST_StyleMatrixColumnIndex },
1776     a_EG_ColorChoice?
1777 a_CT_FontReference =
1778     attribute idx { a_ST_FontCollectionIndex },
1779     a_EG_ColorChoice?
1780 a_CT_ShapeStyle =
1781     element lnRef { a_CT_StyleMatrixReference },
1782     element fillRef { a_CT_StyleMatrixReference },
1783     element effectRef { a_CT_StyleMatrixReference },
1784     element fontRef { a_CT_FontReference }
1785 a_CT_DefaultShapeDefinition =
1786     element spPr { a_CT_ShapeProperties },
1787     element bodyPr { a_CT_TextBodyProperties },
1788     element lstStyle { a_CT_TextListStyle },
1789     element style { a_CT_ShapeStyle }?,
1790     element extLst { a_CT_OfficeArtExtensionList }?
1791 a_CT_ObjectStyleDefaults =
1792     element spDef { a_CT_DefaultShapeDefinition }?,
1793     element lnDef { a_CT_DefaultShapeDefinition }?,
1794     element txDef { a_CT_DefaultShapeDefinition }?,
1795     element extLst { a_CT_OfficeArtExtensionList }?
1796 a_CT_EmptyElement = empty
1797 a_CT_ColorMapping =
1798     attribute bg1 { a_ST_ColorSchemeIndex },
1799     attribute tx1 { a_ST_ColorSchemeIndex },
1800     attribute bg2 { a_ST_ColorSchemeIndex },
1801     attribute tx2 { a_ST_ColorSchemeIndex },
1802     attribute accent1 { a_ST_ColorSchemeIndex },
1803     attribute accent2 { a_ST_ColorSchemeIndex },
1804     attribute accent3 { a_ST_ColorSchemeIndex },
1805     attribute accent4 { a_ST_ColorSchemeIndex },
1806     attribute accent5 { a_ST_ColorSchemeIndex },
1807     attribute accent6 { a_ST_ColorSchemeIndex },
1808     attribute hlink { a_ST_ColorSchemeIndex },
1809     attribute folHlink { a_ST_ColorSchemeIndex },
1810     element extLst { a_CT_OfficeArtExtensionList }?
1811 a_CT_ColorMappingOverride =
1812     element masterClrMapping { a_CT_EmptyElement }
1813     | element overrideClrMapping { a_CT_ColorMapping }
1814 a_CT_ColorSchemeAndMapping =
1815     element clrScheme { a_CT_ColorScheme },
1816     element clrMap { a_CT_ColorMapping }?

```

```

1817 a_CT_ColorSchemeList =
1818     element extraClrScheme { a_CT_ColorSchemeAndMapping }*
1819 a_CT_OfficeStyleSheet =
1820     attribute name { xsd:string }?,
1821     element themeElements { a_CT_BaseStyles },
1822     element objectDefaults { a_CT_ObjectStyleDefaults }?,
1823     element extraClrSchemeLst { a_CT_ColorSchemeList }?,
1824     element custClrLst { a_CT_CustomColorList }?,
1825     element extLst { a_CT_OfficeArtExtensionList }?
1826 a_CT_BaseStylesOverride =
1827     element clrScheme { a_CT_ColorScheme }?,
1828     element fontScheme { a_CT_FontScheme }?,
1829     element fmtScheme { a_CT_StyleMatrix }?
1830 a_CT_ClipboardStyleSheet =
1831     element themeElements { a_CT_BaseStyles },
1832     element clrMap { a_CT_ColorMapping }
1833 a_theme = element theme { a_CT_OfficeStyleSheet }
1834 a_themeOverride = element themeOverride { a_CT_BaseStylesOverride }
1835 a_themeManager = element themeManager { a_CT_EmptyElement }
1836 a_CT_TableCellProperties =
1837
1838     ## default value: 91440
1839     attribute marL { a_ST_Coordinate32 }?,
1840
1841     ## default value: 91440
1842     attribute marR { a_ST_Coordinate32 }?,
1843
1844     ## default value: 45720
1845     attribute marT { a_ST_Coordinate32 }?,
1846
1847     ## default value: 45720
1848     attribute marB { a_ST_Coordinate32 }?,
1849
1850     ## default value: horz
1851     attribute vert { a_ST_TextVerticalType }?,
1852
1853     ## default value: t
1854     attribute anchor { a_ST_TextAnchoringType }?,
1855
1856     ## default value: false
1857     attribute anchorCtr { xsd:boolean }?,
1858
1859     ## default value: clip
1860     attribute horzOverflow { a_ST_TextHorzOverflowType }?,
1861     element lnL { a_CT_LineProperties }?,
1862     element lnR { a_CT_LineProperties }?,
1863     element lnT { a_CT_LineProperties }?,
1864     element lnB { a_CT_LineProperties }?,
1865     element lnTlToBr { a_CT_LineProperties }?,
1866     element lnBlToTr { a_CT_LineProperties }?,
1867     element cell3D { a_CT_Cell3D }?,
1868     a_EG_FillProperties?,
1869     element headers { a_CT_Headers }?,

```

```

1870     element extLst { a_CT_OfficeArtExtensionList }?
1871 a_CT_Headers = element header { xsd:string }*
1872 a_CT_TableCol =
1873     attribute w { a_ST_Coordinate },
1874     element extLst { a_CT_OfficeArtExtensionList }?
1875 a_CT_TableGrid = element gridCol { a_CT_TableCol }*
1876 a_CT_TableCell =
1877
1878     ## default value: 1
1879     attribute rowSpan { xsd:int }?,
1880
1881     ## default value: 1
1882     attribute gridSpan { xsd:int }?,
1883
1884     ## default value: false
1885     attribute hMerge { xsd:boolean }?,
1886
1887     ## default value: false
1888     attribute vMerge { xsd:boolean }?,
1889     attribute id { xsd:string }?,
1890     element txBdy { a_CT_TextBody }?,
1891     element tcPr { a_CT_TableCellProperties }?,
1892     element extLst { a_CT_OfficeArtExtensionList }?
1893 a_CT_TableRow =
1894     attribute h { a_ST_Coordinate },
1895     element tc { a_CT_TableCell }*,
1896     element extLst { a_CT_OfficeArtExtensionList }?
1897 a_CT_TableProperties =
1898
1899     ## default value: false
1900     attribute rtl { xsd:boolean }?,
1901
1902     ## default value: false
1903     attribute firstRow { xsd:boolean }?,
1904
1905     ## default value: false
1906     attribute firstCol { xsd:boolean }?,
1907
1908     ## default value: false
1909     attribute lastRow { xsd:boolean }?,
1910
1911     ## default value: false
1912     attribute lastCol { xsd:boolean }?,
1913
1914     ## default value: false
1915     attribute bandRow { xsd:boolean }?,
1916
1917     ## default value: false
1918     attribute bandCol { xsd:boolean }?,
1919     a_EG_FillProperties?,
1920     a_EG_EffectProperties?,
1921     (element tableStyle { a_CT_TableStyle }
1922       | element tableStyleId { s_ST_Guid })?,

```



```

1923     element extLst { a_CT_OfficeArtExtensionList }?
1924 a_CT_Table =
1925     element tblPr { a_CT_TableProperties }?,
1926     element tblGrid { a_CT_TableGrid },
1927     element tr { a_CT_TableRow }*
1928 a_tbl = element tbl { a_CT_Table }
1929 a_CT_Cell3D =
1930
1931     ## default value: plastic
1932     attribute prstMaterial { a_ST_PresetMaterialType }?,
1933     element bevel { a_CT_Bevel },
1934     element lightRig { a_CT_LightRig }?,
1935     element extLst { a_CT_OfficeArtExtensionList }?
1936 a_EG_ThemeableFillStyle =
1937     element fill { a_CT_FillProperties }
1938     | element fillRef { a_CT_StyleMatrixReference }
1939 a_CT_ThemeableLineStyle =
1940     element ln { a_CT_LineProperties }
1941     | element lnRef { a_CT_StyleMatrixReference }
1942 a_EG_ThemeableEffectStyle =
1943     element effect { a_CT_EffectProperties }
1944     | element effectRef { a_CT_StyleMatrixReference }
1945 a_EG_ThemeableFontStyles =
1946     element font { a_CT_FontCollection }
1947     | element fontRef { a_CT_FontReference }
1948 a_ST_OnOffStyleType = "on" | "off" | "def"
1949 a_CT_TableStyleTextStyle =
1950
1951     ## default value: def
1952     attribute b { a_ST_OnOffStyleType }?,
1953
1954     ## default value: def
1955     attribute i { a_ST_OnOffStyleType }?,
1956     a_EG_ThemeableFontStyles?,
1957     a_EG_ColorChoice?,
1958     element extLst { a_CT_OfficeArtExtensionList }?
1959 a_CT_TableCellBorderStyle =
1960     element left { a_CT_ThemeableLineStyle }?,
1961     element right { a_CT_ThemeableLineStyle }?,
1962     element top { a_CT_ThemeableLineStyle }?,
1963     element bottom { a_CT_ThemeableLineStyle }?,
1964     element insideH { a_CT_ThemeableLineStyle }?,
1965     element insideV { a_CT_ThemeableLineStyle }?,
1966     element tl2br { a_CT_ThemeableLineStyle }?,
1967     element tr2bl { a_CT_ThemeableLineStyle }?,
1968     element extLst { a_CT_OfficeArtExtensionList }?
1969 a_CT_TableBackgroundStyle =
1970     a_EG_ThemeableFillStyle?, a_EG_ThemeableEffectStyle?
1971 a_CT_TableStyleCellStyle =
1972     element tcBdr { a_CT_TableCellBorderStyle }?,
1973     a_EG_ThemeableFillStyle?,
1974     element cell3D { a_CT_Cell3D }?
1975 a_CT_TablePartStyle =

```

```

1976     element tcTxStyle { a_CT_TableStyleTextStyle }?,
1977     element tcStyle { a_CT_TableStyleCellStyle }?
1978 a_CT_TableStyle =
1979     attribute styleId { s_ST_Guid },
1980     attribute styleName { xsd:string },
1981     element tblBg { a_CT_TableBackgroundStyle }?,
1982     element wholeTbl { a_CT_TablePartStyle }?,
1983     element band1H { a_CT_TablePartStyle }?,
1984     element band2H { a_CT_TablePartStyle }?,
1985     element band1V { a_CT_TablePartStyle }?,
1986     element band2V { a_CT_TablePartStyle }?,
1987     element lastCol { a_CT_TablePartStyle }?,
1988     element firstCol { a_CT_TablePartStyle }?,
1989     element lastRow { a_CT_TablePartStyle }?,
1990     element seCell { a_CT_TablePartStyle }?,
1991     element swCell { a_CT_TablePartStyle }?,
1992     element firstRow { a_CT_TablePartStyle }?,
1993     element neCell { a_CT_TablePartStyle }?,
1994     element nwCell { a_CT_TablePartStyle }?,
1995     element extLst { a_CT_OfficeArtExtensionList }?
1996 a_CT_TableStyleList =
1997     attribute def { s_ST_Guid },
1998     element tblStyle { a_CT_TableStyle }*
1999 a_tblStyleLst = element tblStyleLst { a_CT_TableStyleList }
2000 a_CT_TextParagraph =
2001     element pPr { a_CT_TextParagraphProperties }?,
2002     a_EG_TextRun*,
2003     element endParaRPr { a_CT_TextCharacterProperties }?
2004 a_ST_TextAnchoringType = "t" | "ctr" | "b" | "just" | "dist"
2005 a_ST_TextVertOverflowType = "overflow" | "ellipsis" | "clip"
2006 a_ST_TextHorzOverflowType = "overflow" | "clip"
2007 a_ST_TextVerticalType =
2008     "horz"
2009     | "vert"
2010     | "vert270"
2011     | "wordArtVert"
2012     | "eaVert"
2013     | "mongolianVert"
2014     | "wordArtVertRtl"
2015 a_ST_TextWrappingType = "none" | "square"
2016 a_ST_TextColumnCount =
2017     xsd:int { minInclusive = "1" maxInclusive = "16" }
2018 a_CT_TextListStyle =
2019     element defPPr { a_CT_TextParagraphProperties }?,
2020     element lvl1pPr { a_CT_TextParagraphProperties }?,
2021     element lvl2pPr { a_CT_TextParagraphProperties }?,
2022     element lvl3pPr { a_CT_TextParagraphProperties }?,
2023     element lvl4pPr { a_CT_TextParagraphProperties }?,
2024     element lvl5pPr { a_CT_TextParagraphProperties }?,
2025     element lvl6pPr { a_CT_TextParagraphProperties }?,
2026     element lvl7pPr { a_CT_TextParagraphProperties }?,
2027     element lvl8pPr { a_CT_TextParagraphProperties }?,
2028     element lvl9pPr { a_CT_TextParagraphProperties }?,

```

```

2029     element extLst { a_CT_OfficeArtExtensionList }?
2030 a_ST_TextFontScalePercentOrPercentString =
2031     a_ST_TextFontScalePercent | s_ST_Percentage
2032 a_ST_TextFontScalePercent =
2033     xsd:int { minInclusive = "1000" maxInclusive = "100000" }
2034 a_CT_TextNormalAutofit =
2035
2036     ## default value: 100%
2037     attribute fontScale { a_ST_TextFontScalePercentOrPercentString }?,
2038
2039     ## default value: 0%
2040     attribute lnSpcReduction { a_ST_TextSpacingPercentOrPercentString }?
2041 a_CT_TextShapeAutofit = empty
2042 a_CT_TextNoAutofit = empty
2043 a_EG_TextAutofit =
2044     element noAutofit { a_CT_TextNoAutofit }
2045     | element normAutofit { a_CT_TextNormalAutofit }
2046     | element spAutoFit { a_CT_TextShapeAutofit }
2047 a_CT_TextBodyProperties =
2048     attribute rot { a_ST_Angle }?,
2049     attribute spcFirstLastPara { xsd:boolean }?,
2050     attribute vertOverflow { a_ST_TextVertOverflowType }?,
2051     attribute horzOverflow { a_ST_TextHorzOverflowType }?,
2052     attribute vert { a_ST_TextVerticalType }?,
2053     attribute wrap { a_ST_TextWrappingType }?,
2054     attribute lIns { a_ST_Coordinate32 }?,
2055     attribute tIns { a_ST_Coordinate32 }?,
2056     attribute rIns { a_ST_Coordinate32 }?,
2057     attribute bIns { a_ST_Coordinate32 }?,
2058     attribute numCol { a_ST_TextColumnCount }?,
2059     attribute spcCol { a_ST_PositiveCoordinate32 }?,
2060     attribute rtlCol { xsd:boolean }?,
2061     attribute fromWordArt { xsd:boolean }?,
2062     attribute anchor { a_ST_TextAnchoringType }?,
2063     attribute anchorCtr { xsd:boolean }?,
2064     attribute forceAA { xsd:boolean }?,
2065
2066     ## default value: false
2067     attribute upright { xsd:boolean }?,
2068     attribute compatLnSpc { xsd:boolean }?,
2069     element prstTxWarp { a_CT_PresetTextShape }?,
2070     a_EG_TextAutofit?,
2071     element scene3d { a_CT_Scene3D }?,
2072     a_EG_Text3D?,
2073     element extLst { a_CT_OfficeArtExtensionList }?
2074 a_CT_TextBody =
2075     element bodyPr { a_CT_TextBodyProperties },
2076     element lstStyle { a_CT_TextListStyle }?,
2077     element p { a_CT_TextParagraph }+
2078 a_ST_TextBulletStartAtNum =
2079     xsd:int { minInclusive = "1" maxInclusive = "32767" }
2080 a_ST_TextAutonumberScheme =
2081     "alphaLcParenBoth"

```

```

2082 | "alphaUcParenBoth"
2083 | "alphaLcParenR"
2084 | "alphaUcParenR"
2085 | "alphaLcPeriod"
2086 | "alphaUcPeriod"
2087 | "arabicParenBoth"
2088 | "arabicParenR"
2089 | "arabicPeriod"
2090 | "arabicPlain"
2091 | "romanLcParenBoth"
2092 | "romanUcParenBoth"
2093 | "romanLcParenR"
2094 | "romanUcParenR"
2095 | "romanLcPeriod"
2096 | "romanUcPeriod"
2097 | "circleNumDbPlain"
2098 | "circleNumWdBlackPlain"
2099 | "circleNumWdWhitePlain"
2100 | "arabicDbPeriod"
2101 | "arabicDbPlain"
2102 | "ea1ChsPeriod"
2103 | "ea1ChsPlain"
2104 | "ea1ChtPeriod"
2105 | "ea1ChtPlain"
2106 | "ea1JpnChsDbPeriod"
2107 | "ea1JpnKorPlain"
2108 | "ea1JpnKorPeriod"
2109 | "arabic1Minus"
2110 | "arabic2Minus"
2111 | "hebrew2Minus"
2112 | "thaiAlphaPeriod"
2113 | "thaiAlphaParenR"
2114 | "thaiAlphaParenBoth"
2115 | "thaiNumPeriod"
2116 | "thaiNumParenR"
2117 | "thaiNumParenBoth"
2118 | "hindiAlphaPeriod"
2119 | "hindiNumPeriod"
2120 | "hindiNumParenR"
2121 | "hindiAlpha1Period"
2122 a_CT_TextBulletColorFollowText = empty
2123 a_EG_TextBulletColor =
2124   element buClrTx { a_CT_TextBulletColorFollowText }
2125   | element buClr { a_CT_Color }
2126 a_ST_TextBulletSizePercent =
2127   xsd:int { minInclusive = "25000" maxInclusive = "400000" }
2128 a_CT_TextBulletSizeFollowText = empty
2129 a_CT_TextBulletSizePercent =
2130   attribute val { a_ST_TextBulletSizePercent }?
2131 a_CT_TextBulletSizePoint = attribute val { a_ST_TextFontSize }?
2132 a_EG_TextBulletSize =
2133   element buSzTx { a_CT_TextBulletSizeFollowText }
2134   | element buSzPct { a_CT_TextBulletSizePercent }

```

```

2135 | element buSzPts { a_CT_TextBulletSizePoint }
2136 a_CT_TextBulletTypefaceFollowText = empty
2137 a_EG_TextBulletTypeface =
2138   element buFontTx { a_CT_TextBulletTypefaceFollowText }
2139   | element buFont { a_CT_TextFont }
2140 a_CT_TextAutonumberBullet =
2141   attribute type { a_ST_TextAutonumberScheme },
2142
2143   ## default value: 1
2144   attribute startAt { a_ST_TextBulletStartAtNum }?
2145 a_CT_TextCharBullet = attribute char { xsd:string }
2146 a_CT_TextBlipBullet = element blip { a_CT_Blip }
2147 a_CT_TextNoBullet = empty
2148 a_EG_TextBullet =
2149   element buNone { a_CT_TextNoBullet }
2150   | element buAutoNum { a_CT_TextAutonumberBullet }
2151   | element buChar { a_CT_TextCharBullet }
2152   | element buBlip { a_CT_TextBlipBullet }
2153 a_ST_TextPoint = a_ST_TextPointUnqualified | s_ST_UniversalMeasure
2154 a_ST_TextPointUnqualified =
2155   xsd:int { minInclusive = "-400000" maxInclusive = "400000" }
2156 a_ST_TextNonNegativePoint =
2157   xsd:int { minInclusive = "0" maxInclusive = "400000" }
2158 a_ST_TextFontSize =
2159   xsd:int { minInclusive = "100" maxInclusive = "400000" }
2160 a_ST_TextTypeface = xsd:string
2161 a_CT_TextFont =
2162   attribute typeface { a_ST_TextTypeface }?,
2163   attribute panose { s_ST_Panose }?,
2164
2165   ## default value: 0
2166   attribute pitchFamily { xsd:byte }?,
2167
2168   ## default value: 1
2169   attribute charset { xsd:byte }?
2170 a_ST_TextUnderlineType =
2171   "none"
2172   | "words"
2173   | "sng"
2174   | "dbl"
2175   | "heavy"
2176   | "dotted"
2177   | "dottedHeavy"
2178   | "dash"
2179   | "dashHeavy"
2180   | "dashLong"
2181   | "dashLongHeavy"
2182   | "dotDash"
2183   | "dotDashHeavy"
2184   | "dotDotDash"
2185   | "dotDotDashHeavy"
2186   | "wavy"
2187   | "wavyHeavy"

```

```

2188 | "wavyDbl"
2189 a_CT_TextUnderlineLineFollowText = empty
2190 a_CT_TextUnderlineFillFollowText = empty
2191 a_CT_TextUnderlineFillGroupWrapper = a_EG_FillProperties
2192 a_EG_TextUnderlineLine =
2193   element uLnTx { a_CT_TextUnderlineLineFollowText }
2194   | element uLn { a_CT_LineProperties }?
2195 a_EG_TextUnderlineFill =
2196   element uFillTx { a_CT_TextUnderlineFillFollowText }
2197   | element uFill { a_CT_TextUnderlineFillGroupWrapper }
2198 a_ST_TextStrikeType = "noStrike" | "sngStrike" | "dblStrike"
2199 a_ST_TextCapsType = "none" | "small" | "all"
2200 a_CT_TextCharacterProperties =
2201   attribute kumimoji { xsd:boolean }?,
2202   attribute lang { s_ST_Lang }?,
2203   attribute altLang { s_ST_Lang }?,
2204   attribute sz { a_ST_TextFontSize }?,
2205   attribute b { xsd:boolean }?,
2206   attribute i { xsd:boolean }?,
2207   attribute u { a_ST_TextUnderlineType }?,
2208   attribute strike { a_ST_TextStrikeType }?,
2209   attribute kern { a_ST_TextNonNegativePoint }?,
2210   attribute cap { a_ST_TextCapsType }?,
2211   attribute spc { a_ST_TextPoint }?,
2212   attribute normalizeH { xsd:boolean }?,
2213   attribute baseline { a_ST_Percentage }?,
2214   attribute noProof { xsd:boolean }?,
2215
2216   ## default value: true
2217   attribute dirty { xsd:boolean }?,
2218
2219   ## default value: false
2220   attribute err { xsd:boolean }?,
2221
2222   ## default value: true
2223   attribute smtClean { xsd:boolean }?,
2224
2225   ## default value: 0
2226   attribute smtId { xsd:unsignedInt }?,
2227   attribute bmk { xsd:string }?,
2228   element ln { a_CT_LineProperties }?,
2229   a_EG_FillProperties?,
2230   a_EG_EffectProperties?,
2231   element highlight { a_CT_Color }?,
2232   a_EG_TextUnderlineLine?,
2233   a_EG_TextUnderlineFill?,
2234   element latin { a_CT_TextFont }?,
2235   element ea { a_CT_TextFont }?,
2236   element cs { a_CT_TextFont }?,
2237   element sym { a_CT_TextFont }?,
2238   element hlinkClick { a_CT_Hyperlink }?,
2239   element hlinkMouseOver { a_CT_Hyperlink }?,
2240   element rtl { a_CT_Boolean }?,

```

```

2241     element extLst { a_CT_OfficeArtExtensionList }?
2242 a_CT_Boolean =
2243
2244     ## default value: 0
2245     attribute val { s_ST_OnOff }?
2246 a_ST_TextSpacingPoint =
2247     xsd:int { minInclusive = "0" maxInclusive = "158400" }
2248 a_ST_TextSpacingPercentOrPercentString =
2249     a_ST_TextSpacingPercent | s_ST_Percentage
2250 a_ST_TextSpacingPercent =
2251     xsd:int { minInclusive = "0" maxInclusive = "13200000" }
2252 a_CT_TextSpacingPercent =
2253     attribute val { a_ST_TextSpacingPercentOrPercentString }
2254 a_CT_TextSpacingPoint = attribute val { a_ST_TextSpacingPoint }
2255 a_ST_TextMargin =
2256     xsd:int { minInclusive = "0" maxInclusive = "51206400" }
2257 a_ST_TextIndent =
2258     xsd:int { minInclusive = "-51206400" maxInclusive = "51206400" }
2259 a_ST_TextTabAlignType = "l" | "ctr" | "r" | "dec"
2260 a_CT_TextTabStop =
2261     attribute pos { a_ST_Coordinate32 }?,
2262     attribute algn { a_ST_TextTabAlignType }?
2263 a_CT_TextTabStopList = element tab { a_CT_TextTabStop }*
2264 a_CT_TextLineBreak = element rPr { a_CT_TextCharacterProperties }?
2265 a_CT_TextSpacing =
2266     element spcPct { a_CT_TextSpacingPercent }
2267     | element spcPts { a_CT_TextSpacingPoint }
2268 a_ST_TextAlignType =
2269     "l" | "ctr" | "r" | "just" | "justLow" | "dist" | "thaiDist"
2270 a_ST_TextFontAlignType = "auto" | "t" | "ctr" | "base" | "b"
2271 a_ST_TextIndentLevelType =
2272     xsd:int { minInclusive = "0" maxInclusive = "8" }
2273 a_CT_TextParagraphProperties =
2274     attribute marL { a_ST_TextMargin }?,
2275     attribute marR { a_ST_TextMargin }?,
2276     attribute lvl { a_ST_TextIndentLevelType }?,
2277     attribute indent { a_ST_TextIndent }?,
2278     attribute algn { a_ST_TextAlignType }?,
2279     attribute defTabSz { a_ST_Coordinate32 }?,
2280     attribute rtl { xsd:boolean }?,
2281     attribute eaLnBrk { xsd:boolean }?,
2282     attribute fontAlgn { a_ST_TextFontAlignType }?,
2283     attribute latinLnBrk { xsd:boolean }?,
2284     attribute hangingPunct { xsd:boolean }?,
2285     element lnSpc { a_CT_TextSpacing }?,
2286     element spcBef { a_CT_TextSpacing }?,
2287     element spcAft { a_CT_TextSpacing }?,
2288     a_EG_TextBulletColor?,
2289     a_EG_TextBulletSize?,
2290     a_EG_TextBulletTypeface?,
2291     a_EG_TextBullet?,
2292     element tabLst { a_CT_TextTabStopList }?,
2293     element defRPr { a_CT_TextCharacterProperties }?,

```

```

2294     element extLst { a_CT_OfficeArtExtensionList }?
2295 a_CT_TextField =
2296     attribute id { s_ST_Guid },
2297     attribute type { xsd:string }?,
2298     element rPr { a_CT_TextCharacterProperties }?,
2299     element pPr { a_CT_TextParagraphProperties }?,
2300     element t { xsd:string }?
2301 a_EG_TextRun =
2302     element r { a_CT_RegularTextRun }
2303     | element br { a_CT_TextLineBreak }
2304     | element fld { a_CT_TextField }
2305 a_CT_RegularTextRun =
2306     element rPr { a_CT_TextCharacterProperties }?,
2307     element t { xsd:string }

```

B.4.1.1 Part Schemas

B.4.1.1.1 Table Styles Part

```

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 include "dml-compatibility.rnc"
11 start = a_tblStyleLst

```

B.4.1.1.2 Theme Part

```

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 include "dml-compatibility.rnc"
11 start = a_theme

```

B.4.1.1.3 Theme Override Part

```

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 include "dml-compatibility.rnc"
11 start = a_theme

```

B.4.2 DrawingML – Compatibility

```

1 default namespace =
2   "http://schemas.openxmlformats.org/drawingml/2006/compatibility"
3 namespace a = "http://schemas.openxmlformats.org/drawingml/2006/main"
4 namespace dcmpt =
5   "http://schemas.openxmlformats.org/drawingml/2006/compatibility"
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:vml"
10 namespace w10 = "urn:schemas-microsoft-com:office:word"
11 namespace x = "urn:schemas-microsoft-com:office:excel"
12
13 dcmpt_CT_Compat = attribute spid { a_ST_ShapeID }
14 dcmpt_legacyDrawing = element legacyDrawing { dcmpt_CT_Compat }

```

B.4.3 DrawingML - Picture

```

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:vml"
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.4.4 DrawingML - Locked Canvas

```

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:vml"

```

```

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_GvmlGroupShape }

```

B.4.5 DrawingML - Wordprocessing Drawing

```

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 v = "urn:schemas-microsoft-com:vm1"
6 namespace w =
7   "http://schemas.openxmlformats.org/wordprocessingml/2006/main"
8 namespace w10 = "urn:schemas-microsoft-com:office:word"
9 namespace wp =
10   "http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing"
11 namespace x = "urn:schemas-microsoft-com:office:excel"
12
13 wp_CT_EffectExtent =
14   attribute l { a_ST_Coordinate },
15   attribute t { a_ST_Coordinate },
16   attribute r { a_ST_Coordinate },
17   attribute b { a_ST_Coordinate }
18 wp_ST_WrapDistance = xsd:unsignedInt
19 wp_CT_Inline =
20   attribute distT { wp_ST_WrapDistance }?,
21   attribute distB { wp_ST_WrapDistance }?,
22   attribute distL { wp_ST_WrapDistance }?,
23   attribute distR { wp_ST_WrapDistance }?,
24   element extent { a_CT_PositiveSize2D },
25   element effectExtent { wp_CT_EffectExtent }?,
26   element docPr { a_CT_NonVisualDrawingProps },
27   element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }?,
28   a_graphic
29 wp_ST_WrapText = "bothSides" | "left" | "right" | "largest"
30 wp_CT_WrapPath =
31   attribute edited { xsd:boolean }?,
32   element start { a_CT_Point2D },
33   element lineTo { a_CT_Point2D }+
34 wp_CT_WrapNone = empty
35 wp_CT_WrapSquare =
36   attribute wrapText { wp_ST_WrapText },
37   attribute distT { wp_ST_WrapDistance }?,
38   attribute distB { wp_ST_WrapDistance }?,
39   attribute distL { wp_ST_WrapDistance }?,
40   attribute distR { wp_ST_WrapDistance }?,
41   element effectExtent { wp_CT_EffectExtent }?
42 wp_CT_WrapTight =
43   attribute wrapText { wp_ST_WrapText },
44   attribute distL { wp_ST_WrapDistance }?,
45   attribute distR { wp_ST_WrapDistance }?,
46   element wrapPolygon { wp_CT_WrapPath }

```

```

47 wp_CT_WrapThrough =
48   attribute wrapText { wp_ST_WrapText },
49   attribute distL { wp_ST_WrapDistance }?,
50   attribute distR { wp_ST_WrapDistance }?,
51   element wrapPolygon { wp_CT_WrapPath }
52 wp_CT_WrapTopBottom =
53   attribute distT { wp_ST_WrapDistance }?,
54   attribute distB { wp_ST_WrapDistance }?,
55   element effectExtent { wp_CT_EffectExtent }?
56 wp_EG_WrapType =
57   element wrapNone { wp_CT_WrapNone }
58   | element wrapSquare { wp_CT_WrapSquare }
59   | element wrapTight { wp_CT_WrapTight }
60   | element wrapThrough { wp_CT_WrapThrough }
61   | element wrapTopAndBottom { wp_CT_WrapTopBottom }
62 wp_ST_PositionOffset = xsd:int
63 wp_ST_AlignH = "left" | "right" | "center" | "inside" | "outside"
64 wp_ST_RelFromH =
65   "margin"
66   | "page"
67   | "column"
68   | "character"
69   | "leftMargin"
70   | "rightMargin"
71   | "insideMargin"
72   | "outsideMargin"
73 wp_CT_PosH =
74   attribute relativeFrom { wp_ST_RelFromH },
75   (element align { wp_ST_AlignH }
76    | element posOffset { wp_ST_PositionOffset })
77 wp_ST_AlignV = "top" | "bottom" | "center" | "inside" | "outside"
78 wp_ST_RelFromV =
79   "margin"
80   | "page"
81   | "paragraph"
82   | "line"
83   | "topMargin"
84   | "bottomMargin"
85   | "insideMargin"
86   | "outsideMargin"
87 wp_CT_PosV =
88   attribute relativeFrom { wp_ST_RelFromV },
89   (element align { wp_ST_AlignV }
90    | element posOffset { wp_ST_PositionOffset })
91 wp_CT_Anchor =
92   attribute distT { wp_ST_WrapDistance }?,
93   attribute distB { wp_ST_WrapDistance }?,
94   attribute distL { wp_ST_WrapDistance }?,
95   attribute distR { wp_ST_WrapDistance }?,
96   attribute simplePos { xsd:boolean }?,
97   attribute relativeHeight { xsd:unsignedInt },
98   attribute behindDoc { xsd:boolean },
99   attribute locked { xsd:boolean },

```

```

100 attribute layoutInCell { xsd:boolean },
101 attribute hidden { xsd:boolean }?,
102 attribute allowOverlap { xsd:boolean },
103 element simplePos { a_CT_Point2D },
104 element positionH { wp_CT_PosH },
105 element positionV { wp_CT_PosV },
106 element extent { a_CT_PositiveSize2D },
107 element effectExtent { wp_CT_EffectExtent }?,
108 wp_EG_WrapType,
109 element docPr { a_CT_NonVisualDrawingProps },
110 element cNvGraphicFramePr { a_CT_NonVisualGraphicFrameProperties }?,
111 a_graphic
112 wp_inline = element inline { wp_CT_Inline }
113 wp_anchor = element anchor { wp_CT_Anchor }

```

B.4.6 DrawingML - Spreadsheet Drawing

```

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 txBdy { 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

```

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   xsd:unsignedShort { minInclusive = "5" maxInclusive = "500" }
120 dchrt_CT_HPercent =
121

```



```

122  ## default value: 100
123  attribute val { dchrt_ST_HPercent }?
124  dchrt_ST_RotY =
125    xsd:unsignedShort { minInclusive = "0" maxInclusive = "360" }
126  dchrt_CT_RotY =
127
128  ## default value: 0
129  attribute val { dchrt_ST_RotY }?
130  dchrt_ST_DepthPercent =
131    xsd:unsignedShort { minInclusive = "20" maxInclusive = "2000" }
132  dchrt_CT_DepthPercent =
133
134  ## default value: 100
135  attribute val { dchrt_ST_DepthPercent }?
136  dchrt_ST_Perspective =
137    xsd:unsignedByte { minInclusive = "0" maxInclusive = "240" }
138  dchrt_CT_Perspective =
139
140  ## default value: 30
141  attribute val { dchrt_ST_Perspective }?
142  dchrt_CT_View3D =
143    element rotX { dchrt_CT_RotX }?,
144    element hPercent { dchrt_CT_HPercent }?,
145    element rotY { dchrt_CT_RotY }?,
146    element depthPercent { dchrt_CT_DepthPercent }?,
147    element rAngAx { dchrt_CT_Boolean }?,
148    element perspective { dchrt_CT_Perspective }?,
149    element extLst { dchrt_CT_ExtensionList }?
150  dchrt_CT_Surface =
151    element thickness { dchrt_CT_UnsignedInt }?,
152    element spPr { a_CT_ShapeProperties }?,
153    element pictureOptions { dchrt_CT_PictureOptions }?,
154    element extLst { dchrt_CT_ExtensionList }?
155  dchrt_CT_DTable =
156    element showHorzBorder { dchrt_CT_Boolean }?,
157    element showVertBorder { dchrt_CT_Boolean }?,
158    element showOutline { dchrt_CT_Boolean }?,
159    element showKeys { dchrt_CT_Boolean }?,
160    element spPr { a_CT_ShapeProperties }?,
161    element txPr { a_CT_TextBody }?,
162    element extLst { dchrt_CT_ExtensionList }?
163  dchrt_ST_GapAmount =
164    xsd:unsignedShort { minInclusive = "0" maxInclusive = "500" }
165  dchrt_CT_GapAmount =
166
167  ## default value: 150
168  attribute val { dchrt_ST_GapAmount }?
169  dchrt_ST_Overlap =
170    xsd:byte { minInclusive = "-100" maxInclusive = "100" }
171  dchrt_CT_Overlap =
172
173  ## default value: 0
174  attribute val { dchrt_ST_Overlap }?

```

```

175 dchrt_ST_BubbleScale =
176     xsd:unsignedInt { minInclusive = "0" maxInclusive = "300" }
177 dchrt_CT_BubbleScale =
178
179     ## default value: 100
180     attribute val { dchrt_ST_BubbleScale }?
181 dchrt_ST_SizeRepresents = string "area" | string "w"
182 dchrt_CT_SizeRepresents =
183
184     ## default value: area
185     attribute val { dchrt_ST_SizeRepresents }?
186 dchrt_ST_FirstSliceAng =
187     xsd:unsignedShort { minInclusive = "0" maxInclusive = "360" }
188 dchrt_CT_FirstSliceAng =
189
190     ## default value: 0
191     attribute val { dchrt_ST_FirstSliceAng }?
192 dchrt_ST_HoleSize =
193     xsd:unsignedByte { minInclusive = "10" maxInclusive = "90" }
194 dchrt_CT_HoleSize =
195
196     ## default value: 10
197     attribute val { dchrt_ST_HoleSize }?
198 dchrt_ST_SplitType =
199     string "auto"
200     | string "cust"
201     | string "percent"
202     | string "pos"
203     | string "val"
204 dchrt_CT_SplitType =
205
206     ## default value: auto
207     attribute val { dchrt_ST_SplitType }?
208 dchrt_CT_CustSplit = element secondPiePt { dchrt_CT_UnsignedInt }*
209 dchrt_ST_SecondPieSize =
210     xsd:unsignedShort { minInclusive = "5" maxInclusive = "200" }
211 dchrt_CT_SecondPieSize =
212
213     ## default value: 75
214     attribute val { dchrt_ST_SecondPieSize }?
215 dchrt_CT_NumFmt =
216     attribute formatCode { s_ST_Xstring },
217     attribute sourceLinked { xsd:boolean }?
218 dchrt_ST_LblAlgn = string "ctr" | string "l" | string "r"
219 dchrt_CT_LblAlgn = attribute val { dchrt_ST_LblAlgn }
220 dchrt_ST_DLblPos =
221     string "bestFit"
222     | string "b"
223     | string "ctr"
224     | string "inBase"
225     | string "inEnd"
226     | string "l"
227     | string "outEnd"

```

```

228 | string "r"
229 | string "t"
230 dchrt_CT_DLblPos = attribute val { dchrt_ST_DLblPos }
231 dchrt_EG_DLblShared =
232     element numFmt { dchrt_CT_NumFmt }?,
233     element spPr { a_CT_ShapeProperties }?,
234     element txPr { a_CT_TextBody }?,
235     element dLblPos { dchrt_CT_DLblPos }?,
236     element showLegendKey { dchrt_CT_Boolean }?,
237     element showVal { dchrt_CT_Boolean }?,
238     element showCatName { dchrt_CT_Boolean }?,
239     element showSerName { dchrt_CT_Boolean }?,
240     element showPercent { dchrt_CT_Boolean }?,
241     element showBubbleSize { dchrt_CT_Boolean }?,
242     element separator { xsd:string }?
243 dchrt_Group_DLbl =
244     element layout { dchrt_CT_Layout }?,
245     element tx { dchrt_CT_Tx }?,
246     dchrt_EG_DLblShared
247 dchrt_CT_DLbl =
248     element idx { dchrt_CT_UnsignedInt },
249     (element delete { dchrt_CT_Boolean }
250      | dchrt_Group_DLbl),
251     element extLst { dchrt_CT_ExtensionList }?
252 dchrt_Group_DLbls =
253     dchrt_EG_DLblShared,
254     element showLeaderLines { dchrt_CT_Boolean }?,
255     element leaderLines { dchrt_CT_ChartLines }?
256 dchrt_CT_DLbls =
257     element dLbl { dchrt_CT_DLbl }*,
258     (element delete { dchrt_CT_Boolean }
259      | dchrt_Group_DLbls),
260     element extLst { dchrt_CT_ExtensionList }?
261 dchrt_ST_MarkerStyle =
262     string "circle"
263     | string "dash"
264     | string "diamond"
265     | string "dot"
266     | string "none"
267     | string "picture"
268     | string "plus"
269     | string "square"
270     | string "star"
271     | string "triangle"
272     | string "x"
273 dchrt_CT_MarkerStyle = attribute val { dchrt_ST_MarkerStyle }
274 dchrt_ST_MarkerSize =
275     xsd:unsignedByte { minInclusive = "2" maxInclusive = "72" }
276 dchrt_CT_MarkerSize =
277
278     ## default value: 5
279     attribute val { dchrt_ST_MarkerSize }?
280 dchrt_CT_Marker =

```

```

281     element symbol { dchrt_CT_MarkerStyle }?,
282     element size { dchrt_CT_MarkerSize }?,
283     element spPr { a_CT_ShapeProperties }?,
284     element extLst { dchrt_CT_ExtensionList }?
285 dchrt_CT_DPt =
286     element idx { dchrt_CT_UnsignedInt },
287     element invertIfNegative { dchrt_CT_Boolean }?,
288     element marker { dchrt_CT_Marker }?,
289     element bubble3D { dchrt_CT_Boolean }?,
290     element explosion { dchrt_CT_UnsignedInt }?,
291     element spPr { a_CT_ShapeProperties }?,
292     element pictureOptions { dchrt_CT_PictureOptions }?,
293     element extLst { dchrt_CT_ExtensionList }?
294 dchrt_ST_TrendlineType =
295     string "exp"
296     | string "linear"
297     | string "log"
298     | string "movingAvg"
299     | string "poly"
300     | string "power"
301 dchrt_CT_TrendlineType =
302
303     ## default value: linear
304     attribute val { dchrt_ST_TrendlineType }?
305 dchrt_ST_Order =
306     xsd:unsignedByte { minInclusive = "2" maxInclusive = "6" }
307 dchrt_CT_Order =
308
309     ## default value: 2
310     attribute val { dchrt_ST_Order }?
311 dchrt_ST_Period =
312     xsd:unsignedByte { minInclusive = "2" maxInclusive = "255" }
313 dchrt_CT_Period =
314
315     ## default value: 2
316     attribute val { dchrt_ST_Period }?
317 dchrt_CT_TrendlineLbl =
318     element layout { dchrt_CT_Layout }?,
319     element tx { dchrt_CT_Tx }?,
320     element numFmt { dchrt_CT_NumFmt }?,
321     element spPr { a_CT_ShapeProperties }?,
322     element txPr { a_CT_TextBody }?,
323     element extLst { dchrt_CT_ExtensionList }?
324 dchrt_CT_Trendline =
325     element name { xsd:string }?,
326     element spPr { a_CT_ShapeProperties }?,
327     element trendlineType { dchrt_CT_TrendlineType },
328     element order { dchrt_CT_Order }?,
329     element period { dchrt_CT_Period }?,
330     element forward { dchrt_CT_Double }?,
331     element backward { dchrt_CT_Double }?,
332     element intercept { dchrt_CT_Double }?,
333     element dispRSqr { dchrt_CT_Boolean }?,

```

```

334     element dispEq { dchrt_CT_Boolean }?,
335     element trendlineLbl { dchrt_CT_TrendlineLbl }?,
336     element extLst { dchrt_CT_ExtensionList }?
337 dchrt_ST_ErrDir = string "x" | string "y"
338 dchrt_CT_ErrDir = attribute val { dchrt_ST_ErrDir }
339 dchrt_ST_ErrBarType = string "both" | string "minus" | string "plus"
340 dchrt_CT_ErrBarType =
341
342     ## default value: both
343     attribute val { dchrt_ST_ErrBarType }?
344 dchrt_ST_ErrValType =
345     string "cust"
346     | string "fixedVal"
347     | string "percentage"
348     | string "stdDev"
349     | string "stdErr"
350 dchrt_CT_ErrValType =
351
352     ## default value: fixedVal
353     attribute val { dchrt_ST_ErrValType }?
354 dchrt_CT_ErrBars =
355     element errDir { dchrt_CT_ErrDir }?,
356     element errBarType { dchrt_CT_ErrBarType },
357     element errValType { dchrt_CT_ErrValType },
358     element noEndCap { dchrt_CT_Boolean }?,
359     element plus { dchrt_CT_NumDataSource }?,
360     element minus { dchrt_CT_NumDataSource }?,
361     element val { dchrt_CT_Double }?,
362     element spPr { a_CT_ShapeProperties }?,
363     element extLst { dchrt_CT_ExtensionList }?
364 dchrt_CT_UpDownBar = element spPr { a_CT_ShapeProperties }?
365 dchrt_CT_UpDownBars =
366     element gapWidth { dchrt_CT_GapAmount }?,
367     element upBars { dchrt_CT_UpDownBar }?,
368     element downBars { dchrt_CT_UpDownBar }?,
369     element extLst { dchrt_CT_ExtensionList }?
370 dchrt_EG_SerShared =
371     element idx { dchrt_CT_UnsignedInt },
372     element order { dchrt_CT_UnsignedInt },
373     element tx { dchrt_CT_SerTx }?,
374     element spPr { a_CT_ShapeProperties }?
375 dchrt_CT_LineSer =
376     dchrt_EG_SerShared,
377     element marker { dchrt_CT_Marker }?,
378     element dPt { dchrt_CT_DPt }*,
379     element dLbls { dchrt_CT_DLbls }?,
380     element trendline { dchrt_CT_Trendline }*,
381     element errBars { dchrt_CT_ErrBars }?,
382     element cat { dchrt_CT_AxDataSource }?,
383     element val { dchrt_CT_NumDataSource }?,
384     element smooth { dchrt_CT_Boolean }?,
385     element extLst { dchrt_CT_ExtensionList }?
386 dchrt_CT_ScatterSer =

```

```

387     dchart_EG_SerShared,
388     element marker { dchart_CT_Marker }?,
389     element dPt { dchart_CT_DPt }*,
390     element dLbls { dchart_CT_DLbls }?,
391     element trendline { dchart_CT_Trendline }*,
392     element errBars { dchart_CT_ErrBars }*,
393     element xVal { dchart_CT_AxDataSource }?,
394     element yVal { dchart_CT_NumDataSource }?,
395     element smooth { dchart_CT_Boolean }?,
396     element extLst { dchart_CT_ExtensionList }?
397 dchart_CT_RadarSer =
398     dchart_EG_SerShared,
399     element marker { dchart_CT_Marker }?,
400     element dPt { dchart_CT_DPt }*,
401     element dLbls { dchart_CT_DLbls }?,
402     element cat { dchart_CT_AxDataSource }?,
403     element val { dchart_CT_NumDataSource }?,
404     element extLst { dchart_CT_ExtensionList }?
405 dchart_CT_BarSer =
406     dchart_EG_SerShared,
407     element invertIfNegative { dchart_CT_Boolean }?,
408     element pictureOptions { dchart_CT_PictureOptions }?,
409     element dPt { dchart_CT_DPt }*,
410     element dLbls { dchart_CT_DLbls }?,
411     element trendline { dchart_CT_Trendline }*,
412     element errBars { dchart_CT_ErrBars }?,
413     element cat { dchart_CT_AxDataSource }?,
414     element val { dchart_CT_NumDataSource }?,
415     element shape { dchart_CT_Shape }?,
416     element extLst { dchart_CT_ExtensionList }?
417 dchart_CT_AreaSer =
418     dchart_EG_SerShared,
419     element pictureOptions { dchart_CT_PictureOptions }?,
420     element dPt { dchart_CT_DPt }*,
421     element dLbls { dchart_CT_DLbls }?,
422     element trendline { dchart_CT_Trendline }*,
423     element errBars { dchart_CT_ErrBars }*,
424     element cat { dchart_CT_AxDataSource }?,
425     element val { dchart_CT_NumDataSource }?,
426     element extLst { dchart_CT_ExtensionList }?
427 dchart_CT_PieSer =
428     dchart_EG_SerShared,
429     element explosion { dchart_CT_UnsignedInt }?,
430     element dPt { dchart_CT_DPt }*,
431     element dLbls { dchart_CT_DLbls }?,
432     element cat { dchart_CT_AxDataSource }?,
433     element val { dchart_CT_NumDataSource }?,
434     element extLst { dchart_CT_ExtensionList }?
435 dchart_CT_BubbleSer =
436     dchart_EG_SerShared,
437     element invertIfNegative { dchart_CT_Boolean }?,
438     element dPt { dchart_CT_DPt }*,
439     element dLbls { dchart_CT_DLbls }?,

```

```

440     element trendline { dchrt_CT_Trendline }*,
441     element errBars { dchrt_CT_ErrBars }*,
442     element xVal { dchrt_CT_AxDataSource }?,
443     element yVal { dchrt_CT_NumDataSource }?,
444     element bubbleSize { dchrt_CT_NumDataSource }?,
445     element bubble3D { dchrt_CT_Boolean }?,
446     element extLst { dchrt_CT_ExtensionList }?
447 dchrt_CT_SurfaceSer =
448     dchrt_EG_SerShared,
449     element cat { dchrt_CT_AxDataSource }?,
450     element val { dchrt_CT_NumDataSource }?,
451     element extLst { dchrt_CT_ExtensionList }?
452 dchrt_ST_Grouping =
453     string "percentStacked" | string "standard" | string "stacked"
454 dchrt_CT_Grouping =
455
456     ## default value: standard
457     attribute val { dchrt_ST_Grouping }?
458 dchrt_CT_ChartLines = element spPr { a_CT_ShapeProperties }?
459 dchrt_EG_LineChartShared =
460     element grouping { dchrt_CT_Grouping },
461     element varyColors { dchrt_CT_Boolean }?,
462     element ser { dchrt_CT_LineSer }*,
463     element dLbls { dchrt_CT_DLbls }?,
464     element dropLines { dchrt_CT_ChartLines }?
465 dchrt_CT_LineChart =
466     dchrt_EG_LineChartShared,
467     element hiLowLines { dchrt_CT_ChartLines }?,
468     element upDownBars { dchrt_CT_UpDownBars }?,
469     element marker { dchrt_CT_Boolean }?,
470     element smooth { dchrt_CT_Boolean }?,
471     element axId { dchrt_CT_UnsignedInt }+,
472     element extLst { dchrt_CT_ExtensionList }?
473 dchrt_CT_Line3DChart =
474     dchrt_EG_LineChartShared,
475     element gapDepth { dchrt_CT_GapAmount }?,
476     element axId { dchrt_CT_UnsignedInt }+,
477     element extLst { dchrt_CT_ExtensionList }?
478 dchrt_CT_StockChart =
479     element ser { dchrt_CT_LineSer }+,
480     element dLbls { dchrt_CT_DLbls }?,
481     element dropLines { dchrt_CT_ChartLines }?,
482     element hiLowLines { dchrt_CT_ChartLines }?,
483     element upDownBars { dchrt_CT_UpDownBars }?,
484     element axId { dchrt_CT_UnsignedInt }+,
485     element extLst { dchrt_CT_ExtensionList }?
486 dchrt_ST_ScatterStyle =
487     string "none"
488     | string "line"
489     | string "lineMarker"
490     | string "marker"
491     | string "smooth"
492     | string "smoothMarker"

```

```

493 dchrt_CT_ScatterStyle =
494
495     ## default value: marker
496     attribute val { dchrt_ST_ScatterStyle }?
497 dchrt_CT_ScatterChart =
498     element scatterStyle { dchrt_CT_ScatterStyle },
499     element varyColors { dchrt_CT_Boolean }?,
500     element ser { dchrt_CT_ScatterSer }*,
501     element dLbIs { dchrt_CT_DLbIs }?,
502     element axId { dchrt_CT_UnsignedInt }+,
503     element extLst { dchrt_CT_ExtensionList }?
504 dchrt_ST_RadarStyle =
505     string "standard" | string "marker" | string "filled"
506 dchrt_CT_RadarStyle =
507
508     ## default value: standard
509     attribute val { dchrt_ST_RadarStyle }?
510 dchrt_CT_RadarChart =
511     element radarStyle { dchrt_CT_RadarStyle },
512     element varyColors { dchrt_CT_Boolean }?,
513     element ser { dchrt_CT_RadarSer }*,
514     element dLbIs { dchrt_CT_DLbIs }?,
515     element axId { dchrt_CT_UnsignedInt }+,
516     element extLst { dchrt_CT_ExtensionList }?
517 dchrt_ST_BarGrouping =
518     string "percentStacked"
519     | string "clustered"
520     | string "standard"
521     | string "stacked"
522 dchrt_CT_BarGrouping =
523
524     ## default value: clustered
525     attribute val { dchrt_ST_BarGrouping }?
526 dchrt_ST_BarDir = string "bar" | string "col"
527 dchrt_CT_BarDir =
528
529     ## default value: col
530     attribute val { dchrt_ST_BarDir }?
531 dchrt_ST_Shape =
532     string "cone"
533     | string "coneToMax"
534     | string "box"
535     | string "cylinder"
536     | string "pyramid"
537     | string "pyramidToMax"
538 dchrt_CT_Shape =
539
540     ## default value: box
541     attribute val { dchrt_ST_Shape }?
542 dchrt_EG_BarChartShared =
543     element barDir { dchrt_CT_BarDir },
544     element grouping { dchrt_CT_BarGrouping }?,
545     element varyColors { dchrt_CT_Boolean }?,

```



```

546     element ser { dchrt_CT_BarSer }*,
547     element dLbIs { dchrt_CT_DLbIs }?
548 dchrt_CT_BarChart =
549     dchrt_EG_BarChartShared,
550     element gapWidth { dchrt_CT_GapAmount }?,
551     element overlap { dchrt_CT_Overlap }?,
552     element serLines { dchrt_CT_ChartLines }*,
553     element axId { dchrt_CT_UnsignedInt }+,
554     element extLst { dchrt_CT_ExtensionList }?
555 dchrt_CT_Bar3DChart =
556     dchrt_EG_BarChartShared,
557     element gapWidth { dchrt_CT_GapAmount }?,
558     element gapDepth { dchrt_CT_GapAmount }?,
559     element shape { dchrt_CT_Shape }?,
560     element axId { dchrt_CT_UnsignedInt }+,
561     element extLst { dchrt_CT_ExtensionList }?
562 dchrt_EG_AreaChartShared =
563     element grouping { dchrt_CT_Grouping }?,
564     element varyColors { dchrt_CT_Boolean }?,
565     element ser { dchrt_CT_AreaSer }*,
566     element dLbIs { dchrt_CT_DLbIs }?,
567     element dropLines { dchrt_CT_ChartLines }?
568 dchrt_CT_AreaChart =
569     dchrt_EG_AreaChartShared,
570     element axId { dchrt_CT_UnsignedInt }+,
571     element extLst { dchrt_CT_ExtensionList }?
572 dchrt_CT_Area3DChart =
573     dchrt_EG_AreaChartShared,
574     element gapDepth { dchrt_CT_GapAmount }?,
575     element axId { dchrt_CT_UnsignedInt }+,
576     element extLst { dchrt_CT_ExtensionList }?
577 dchrt_EG_PieChartShared =
578     element varyColors { dchrt_CT_Boolean }?,
579     element ser { dchrt_CT_PieSer }*,
580     element dLbIs { dchrt_CT_DLbIs }?
581 dchrt_CT_PieChart =
582     dchrt_EG_PieChartShared,
583     element firstSliceAng { dchrt_CT_FirstSliceAng }?,
584     element extLst { dchrt_CT_ExtensionList }?
585 dchrt_CT_Pie3DChart =
586     dchrt_EG_PieChartShared,
587     element extLst { dchrt_CT_ExtensionList }?
588 dchrt_CT_DoughnutChart =
589     dchrt_EG_PieChartShared,
590     element firstSliceAng { dchrt_CT_FirstSliceAng }?,
591     element holeSize { dchrt_CT_HoleSize }?,
592     element extLst { dchrt_CT_ExtensionList }?
593 dchrt_ST_OfPieType = string "pie" | string "bar"
594 dchrt_CT_OfPieType =
595
596     ## default value: pie
597     attribute val { dchrt_ST_OfPieType }?
598 dchrt_CT_OfPieChart =

```

```

599     element ofPieType { dchrt_CT_OfPieType },
600     dchrt_EG_PieChartShared,
601     element gapWidth { dchrt_CT_GapAmount }?,
602     element splitType { dchrt_CT_SplitType }?,
603     element splitPos { dchrt_CT_Double }?,
604     element custSplit { dchrt_CT_CustSplit }?,
605     element secondPieSize { dchrt_CT_SecondPieSize }?,
606     element serLines { dchrt_CT_ChartLines }*,
607     element extLst { dchrt_CT_ExtensionList }?
608 dchrt_CT_BubbleChart =
609     element varyColors { dchrt_CT_Boolean }?,
610     element ser { dchrt_CT_BubbleSer }*,
611     element dLbls { dchrt_CT_DLbls }?,
612     element bubble3D { dchrt_CT_Boolean }?,
613     element bubbleScale { dchrt_CT_BubbleScale }?,
614     element showNegBubbles { dchrt_CT_Boolean }?,
615     element sizeRepresents { dchrt_CT_SizeRepresents }?,
616     element axId { dchrt_CT_UnsignedInt }+,
617     element extLst { dchrt_CT_ExtensionList }?
618 dchrt_CT_BandFmt =
619     element idx { dchrt_CT_UnsignedInt },
620     element spPr { a_CT_ShapeProperties }?
621 dchrt_CT_BandFmts = element bandFmt { dchrt_CT_BandFmt }*
622 dchrt_EG_SurfaceChartShared =
623     element wireframe { dchrt_CT_Boolean }?,
624     element ser { dchrt_CT_SurfaceSer }*,
625     element bandFmts { dchrt_CT_BandFmts }?
626 dchrt_CT_SurfaceChart =
627     dchrt_EG_SurfaceChartShared,
628     element axId { dchrt_CT_UnsignedInt }+,
629     element extLst { dchrt_CT_ExtensionList }?
630 dchrt_CT_Surface3DChart =
631     dchrt_EG_SurfaceChartShared,
632     element axId { dchrt_CT_UnsignedInt }+,
633     element extLst { dchrt_CT_ExtensionList }?
634 dchrt_ST_AxPos = string "b" | string "l" | string "r" | string "t"
635 dchrt_CT_AxPos = attribute val { dchrt_ST_AxPos }
636 dchrt_ST_Crosses = string "autoZero" | string "max" | string "min"
637 dchrt_CT_Crosses = attribute val { dchrt_ST_Crosses }
638 dchrt_ST_CrossBetween = string "between" | string "midCat"
639 dchrt_CT_CrossBetween = attribute val { dchrt_ST_CrossBetween }
640 dchrt_ST_TickMark =
641     string "cross" | string "in" | string "none" | string "out"
642 dchrt_CT_TickMark =
643
644     ## default value: cross
645     attribute val { dchrt_ST_TickMark }?
646 dchrt_ST_TickLblPos =
647     string "high" | string "low" | string "nextTo" | string "none"
648 dchrt_CT_TickLblPos =
649
650     ## default value: nextTo
651     attribute val { dchrt_ST_TickLblPos }?

```

```

652 dchrt_ST_Skip = xsd:unsignedShort { minInclusive = "1" }
653 dchrt_CT_Skip = attribute val { dchrt_ST_Skip }
654 dchrt_ST_TimeUnit = string "days" | string "months" | string "years"
655 dchrt_CT_TimeUnit =
656
657     ## default value: days
658     attribute val { dchrt_ST_TimeUnit }?
659 dchrt_ST_AxisUnit = xsd:double { minExclusive = "0" }
660 dchrt_CT_AxisUnit = attribute val { dchrt_ST_AxisUnit }
661 dchrt_ST_BuiltInUnit =
662     string "hundreds"
663     | string "thousands"
664     | string "tenThousands"
665     | string "hundredThousands"
666     | string "millions"
667     | string "tenMillions"
668     | string "hundredMillions"
669     | string "billions"
670     | string "trillions"
671 dchrt_CT_BuiltInUnit =
672
673     ## default value: thousands
674     attribute val { dchrt_ST_BuiltInUnit }?
675 dchrt_ST_PictureFormat =
676     string "stretch" | string "stack" | string "stackScale"
677 dchrt_CT_PictureFormat = attribute val { dchrt_ST_PictureFormat }
678 dchrt_ST_PictureStackUnit = xsd:double { minExclusive = "0" }
679 dchrt_CT_PictureStackUnit = attribute val { dchrt_ST_PictureStackUnit }
680 dchrt_CT_PictureOptions =
681     element applyToFront { dchrt_CT_Boolean }?,
682     element applyToSides { dchrt_CT_Boolean }?,
683     element applyToEnd { dchrt_CT_Boolean }?,
684     element pictureFormat { dchrt_CT_PictureFormat }?,
685     element pictureStackUnit { dchrt_CT_PictureStackUnit }?
686 dchrt_CT_DispUnitsLbl =
687     element layout { dchrt_CT_Layout }?,
688     element tx { dchrt_CT_Tx }?,
689     element spPr { a_CT_ShapeProperties }?,
690     element txPr { a_CT_TextBody }?
691 dchrt_CT_DispUnits =
692     (element custUnit { dchrt_CT_Double }
693     | element builtInUnit { dchrt_CT_BuiltInUnit }),
694     element dispUnitsLbl { dchrt_CT_DispUnitsLbl }?,
695     element extLst { dchrt_CT_ExtensionList }?
696 dchrt_ST_Orientation = string "maxMin" | string "minMax"
697 dchrt_CT_Orientation =
698
699     ## default value: minMax
700     attribute val { dchrt_ST_Orientation }?
701 dchrt_ST_LogBase =
702     xsd:double { minInclusive = "2" maxInclusive = "1000" }
703 dchrt_CT_LogBase = attribute val { dchrt_ST_LogBase }
704 dchrt_CT_Scaling =

```

```

705     element logBase { dchrt_CT_LogBase }?,
706     element orientation { dchrt_CT_Orientation }?,
707     element max { dchrt_CT_Double }?,
708     element min { dchrt_CT_Double }?,
709     element extLst { dchrt_CT_ExtensionList }?
710 dchrt_ST_LblOffset =
711     xsd:unsignedShort { minInclusive = "0" maxInclusive = "1000" }
712 dchrt_CT_LblOffset =
713
714     ## default value: 100
715     attribute val { dchrt_ST_LblOffset }?
716 dchrt_EG_AxShared =
717     element axId { dchrt_CT_UnsignedInt },
718     element scaling { dchrt_CT_Scaling },
719     element delete { dchrt_CT_Boolean }?,
720     element axPos { dchrt_CT_AxPos },
721     element majorGridlines { dchrt_CT_ChartLines }?,
722     element minorGridlines { dchrt_CT_ChartLines }?,
723     element title { dchrt_CT_Title }?,
724     element numFmt { dchrt_CT_NumFmt }?,
725     element majorTickMark { dchrt_CT_TickMark }?,
726     element minorTickMark { dchrt_CT_TickMark }?,
727     element tickLblPos { dchrt_CT_TickLblPos }?,
728     element spPr { a_CT_ShapeProperties }?,
729     element txPr { a_CT_TextBody }?,
730     element crossAx { dchrt_CT_UnsignedInt },
731     (element crosses { dchrt_CT_Crosses }
732      | element crossesAt { dchrt_CT_Double })?
733 dchrt_CT_CatAx =
734     dchrt_EG_AxShared,
735     element auto { dchrt_CT_Boolean }?,
736     element lblAlign { dchrt_CT_LblAlign }?,
737     element lblOffset { dchrt_CT_LblOffset }?,
738     element tickLblSkip { dchrt_CT_Skip }?,
739     element tickMarkSkip { dchrt_CT_Skip }?,
740     element noMultiLvlLbl { dchrt_CT_Boolean }?,
741     element extLst { dchrt_CT_ExtensionList }?
742 dchrt_CT_DateAx =
743     dchrt_EG_AxShared,
744     element auto { dchrt_CT_Boolean }?,
745     element lblOffset { dchrt_CT_LblOffset }?,
746     element baseTimeUnit { dchrt_CT_TimeUnit }?,
747     element majorUnit { dchrt_CT_AxisUnit }?,
748     element majorTimeUnit { dchrt_CT_TimeUnit }?,
749     element minorUnit { dchrt_CT_AxisUnit }?,
750     element minorTimeUnit { dchrt_CT_TimeUnit }?,
751     element extLst { dchrt_CT_ExtensionList }?
752 dchrt_CT_SerAx =
753     dchrt_EG_AxShared,
754     element tickLblSkip { dchrt_CT_Skip }?,
755     element tickMarkSkip { dchrt_CT_Skip }?,
756     element extLst { dchrt_CT_ExtensionList }?
757 dchrt_CT_ValAx =

```

```

758 dchrt_EG_AxShared,
759 element crossBetween { dchrt_CT_CrossBetween }?,
760 element majorUnit { dchrt_CT_AxisUnit }?,
761 element minorUnit { dchrt_CT_AxisUnit }?,
762 element dispUnits { dchrt_CT_DispUnits }?,
763 element extLst { dchrt_CT_ExtensionList }?
764 dchrt_CT_PlotArea =
765 element layout { dchrt_CT_Layout }?,
766 (element areaChart { dchrt_CT_AreaChart }
767 | element area3DChart { dchrt_CT_Area3DChart }
768 | element lineChart { dchrt_CT_LineChart }
769 | element line3DChart { dchrt_CT_Line3DChart }
770 | element stockChart { dchrt_CT_StockChart }
771 | element radarChart { dchrt_CT_RadarChart }
772 | element scatterChart { dchrt_CT_ScatterChart }
773 | element pieChart { dchrt_CT_PieChart }
774 | element pie3DChart { dchrt_CT_Pie3DChart }
775 | element doughnutChart { dchrt_CT_DoughnutChart }
776 | element barChart { dchrt_CT_BarChart }
777 | element bar3DChart { dchrt_CT_Bar3DChart }
778 | element ofPieChart { dchrt_CT_OfPieChart }
779 | element surfaceChart { dchrt_CT_SurfaceChart }
780 | element surface3DChart { dchrt_CT_Surface3DChart }
781 | element bubbleChart { dchrt_CT_BubbleChart })+,
782 (element valAx { dchrt_CT_ValAx }
783 | element catAx { dchrt_CT_CatAx }
784 | element dateAx { dchrt_CT_DateAx }
785 | element serAx { dchrt_CT_SerAx })*,
786 element dTable { dchrt_CT_DTable }?,
787 element spPr { a_CT_ShapeProperties }?,
788 element extLst { dchrt_CT_ExtensionList }?
789 dchrt_CT_PivotFmt =
790 element idx { dchrt_CT_UnsignedInt },
791 element spPr { a_CT_ShapeProperties }?,
792 element txPr { a_CT_TextBody }?,
793 element marker { dchrt_CT_Marker }?,
794 element dLbl { dchrt_CT_DLbl }?,
795 element extLst { dchrt_CT_ExtensionList }?
796 dchrt_CT_PivotFmts = element pivotFmt { dchrt_CT_PivotFmt }*
797 dchrt_ST_LegendPos =
798 string "b" | string "tr" | string "l" | string "r" | string "t"
799 dchrt_CT_LegendPos =
800
801 ## default value: r
802 attribute val { dchrt_ST_LegendPos }?
803 dchrt_EG_LegendEntryData = element txPr { a_CT_TextBody }?
804 dchrt_CT_LegendEntry =
805 element idx { dchrt_CT_UnsignedInt },
806 (element delete { dchrt_CT_Boolean }
807 | dchrt_EG_LegendEntryData),
808 element extLst { dchrt_CT_ExtensionList }?
809 dchrt_CT_Legend =
810 element legendPos { dchrt_CT_LegendPos }?,

```

```

811     element legendEntry { dchrt_CT_LegendEntry }*,
812     element layout { dchrt_CT_Layout }?,
813     element overlay { dchrt_CT_Boolean }?,
814     element spPr { a_CT_ShapeProperties }?,
815     element txPr { a_CT_TextBody }?,
816     element extLst { dchrt_CT_ExtensionList }?
817 dchrt_ST_DisplanksAs = string "span" | string "gap" | string "zero"
818 dchrt_CT_DisplanksAs =
819
820     ## default value: zero
821     attribute val { dchrt_ST_DisplanksAs }?
822 dchrt_CT_Chart =
823     element title { dchrt_CT_Title }?,
824     element autoTitleDeleted { dchrt_CT_Boolean }?,
825     element pivotFmts { dchrt_CT_PivotFmts }?,
826     element view3D { dchrt_CT_View3D }?,
827     element floor { dchrt_CT_Surface }?,
828     element sideWall { dchrt_CT_Surface }?,
829     element backWall { dchrt_CT_Surface }?,
830     element plotArea { dchrt_CT_PlotArea },
831     element legend { dchrt_CT_Legend }?,
832     element plotVisOnly { dchrt_CT_Boolean }?,
833     element displanksAs { dchrt_CT_DisplanksAs }?,
834     element showDLblsOverMax { dchrt_CT_Boolean }?,
835     element extLst { dchrt_CT_ExtensionList }?
836 dchrt_ST_Style =
837     xsd:unsignedByte { minInclusive = "1" maxInclusive = "48" }
838 dchrt_CT_Style = attribute val { dchrt_ST_Style }
839 dchrt_CT_PivotSource =
840     element name { s_ST_Xstring },
841     element fmtId { dchrt_CT_UnsignedInt },
842     element extLst { dchrt_CT_ExtensionList }*
843 dchrt_CT_Protection =
844     element chartObject { dchrt_CT_Boolean }?,
845     element data { dchrt_CT_Boolean }?,
846     element formatting { dchrt_CT_Boolean }?,
847     element selection { dchrt_CT_Boolean }?,
848     element userInterface { dchrt_CT_Boolean }?
849 dchrt_CT_HeaderFooter =
850
851     ## default value: true
852     attribute alignWithMargins { xsd:boolean }?,
853
854     ## default value: false
855     attribute differentOddEven { xsd:boolean }?,
856
857     ## default value: false
858     attribute differentFirst { xsd:boolean }?,
859     element oddHeader { s_ST_Xstring }?,
860     element oddFooter { s_ST_Xstring }?,
861     element evenHeader { s_ST_Xstring }?,
862     element evenFooter { s_ST_Xstring }?,
863     element firstHeader { s_ST_Xstring }?,

```

```

864     element firstFooter { s_ST_Xstring }?
865 dchrt_CT_PageMargins =
866     attribute l { xsd:double },
867     attribute r { xsd:double },
868     attribute t { xsd:double },
869     attribute b { xsd:double },
870     attribute header { xsd:double },
871     attribute footer { xsd:double }
872 dchrt_ST_PageSetupOrientation =
873     string "default" | string "portrait" | string "landscape"
874 dchrt_CT_ExternalData =
875     r_id,
876     element autoUpdate { dchrt_CT_Boolean }?
877 dchrt_CT_PageSetup =
878
879     ## default value: 1
880     attribute paperSize { xsd:unsignedInt }?,
881     attribute paperHeight { s_ST_PositiveUniversalMeasure }?,
882     attribute paperWidth { s_ST_PositiveUniversalMeasure }?,
883
884     ## default value: 1
885     attribute firstPageNumber { xsd:unsignedInt }?,
886
887     ## default value: default
888     attribute orientation { dchrt_ST_PageSetupOrientation }?,
889
890     ## default value: false
891     attribute blackAndWhite { xsd:boolean }?,
892
893     ## default value: false
894     attribute draft { xsd:boolean }?,
895
896     ## default value: false
897     attribute useFirstPageNumber { xsd:boolean }?,
898
899     ## default value: 600
900     attribute horizontalDpi { xsd:int }?,
901
902     ## default value: 600
903     attribute verticalDpi { xsd:int }?,
904
905     ## default value: 1
906     attribute copies { xsd:unsignedInt }?
907 dchrt_CT_PrintSettings =
908     element headerFooter { dchrt_CT_HeaderFooter }?,
909     element pageMargins { dchrt_CT_PageMargins }?,
910     element pageSetup { dchrt_CT_PageSetup }?,
911     element legacyDrawingHT { dchrt_CT_RelId }?
912 dchrt_CT_ChartSpace =
913     element date1904 { dchrt_CT_Boolean }?,
914     element lang { dchrt_CT_TextLanguageID }?,
915     element roundedCorners { dchrt_CT_Boolean }?,
916     element style { dchrt_CT_Style }?,

```

```

917     element clrMapOvr { a_CT_ColorMapping }?,
918     element pivotSource { dchrt_CT_PivotSource }?,
919     element protection { dchrt_CT_Protection }?,
920     element chart { dchrt_CT_Chart },
921     element spPr { a_CT_ShapeProperties }?,
922     element txPr { a_CT_TextBody }?,
923     element externalData { dchrt_CT_ExternalData }?,
924     element printSettings { dchrt_CT_PrintSettings }?,
925     element userShapes { dchrt_CT_RelId }?,
926     element extLst { dchrt_CT_ExtensionList }?
927 dchrt_chartSpace = element chartSpace { dchrt_CT_ChartSpace }
928 dchrt_userShapes = element userShapes { cdr_CT_Drawing }
929 dchrt_chart = element chart { dchrt_CT_RelId }

```

B.5.1.1 Part Schemas

B.5.1.1.1 Chart Part

```

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-compatibility.rnc"
10 include "dml-chartDrawing.rnc"
11 start = dchrt_chartSpace

```

B.5.1.1.2 Chart Drawing Part

```

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-compatibility.rnc"
10 include "dml-chartDrawing.rnc"
11 start = dchrt_userShapes

```

B.5.2 DrawingML - Chart Drawing

```

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:vm1"
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

```

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     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
48     attribute minVer { xsd:string }?,
49     element title { ddgrm_CT_CTName }*,
50     element desc { ddgrm_CT_CTDescription }*,
51     element catLst { ddgrm_CT_CTCategories }?,
52     element styleLbl { ddgrm_CT_CTStyleLabel }*,
53     element extLst { a_CT_OfficeArtExtensionList }?
54 ddgrm_colorsDef = element colorsDef { ddgrm_CT_ColorTransform }
55 ddgrm_CT_ColorTransformHeader =
56     attribute uniqueId { xsd:string },
57
58     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
59     attribute minVer { xsd:string }?,
60
61     ## default value: 0
62     attribute resId { xsd:int }?,
63     element title { ddgrm_CT_CTName }+,
64     element desc { ddgrm_CT_CTDescription }+,
65     element catLst { ddgrm_CT_CTCategories }?,
66     element extLst { a_CT_OfficeArtExtensionList }?
67 ddgrm_colorsDefHdr =
68     element colorsDefHdr { ddgrm_CT_ColorTransformHeader }
69 ddgrm_CT_ColorTransformHeaderLst =
70     element colorsDefHdr { ddgrm_CT_ColorTransformHeader }*

```

```

71 ddgrm_colorsDefHdrLst =
72   element colorsDefHdrLst { ddgrm_CT_ColorTransformHeaderLst }
73 ddgrm_ST_PtType =
74   "node" | "asst" | "doc" | "pres" | "parTrans" | "sibTrans"
75 ddgrm_CT_Pt =
76   attribute modelId { ddgrm_ST_ModelId },
77
78   ## default value: node
79   attribute type { ddgrm_ST_PtType }?,
80
81   ## default value: 0
82   attribute cxnId { ddgrm_ST_ModelId }?,
83   element prSet { ddgrm_CT_ElemPropSet }?,
84   element spPr { a_CT_ShapeProperties }?,
85   element t { a_CT_TextBody }?,
86   element extLst { a_CT_OfficeArtExtensionList }?
87 ddgrm_CT_PtList = element pt { ddgrm_CT_Pt }*
88 ddgrm_ST_CxnType =
89   "parOf" | "presOf" | "presParOf" | "unknownRelationship"
90 ddgrm_CT_Cxn =
91   attribute modelId { ddgrm_ST_ModelId },
92
93   ## default value: parOf
94   attribute type { ddgrm_ST_CxnType }?,
95   attribute srcId { ddgrm_ST_ModelId },
96   attribute destId { ddgrm_ST_ModelId },
97   attribute srcOrd { xsd:unsignedInt },
98   attribute destOrd { xsd:unsignedInt },
99
100  ## default value: 0
101  attribute parTransId { ddgrm_ST_ModelId }?,
102
103  ## default value: 0
104  attribute sibTransId { ddgrm_ST_ModelId }?,
105  attribute presId { xsd:string }?,
106  element extLst { a_CT_OfficeArtExtensionList }?
107 ddgrm_CT_CxnList = element cxn { ddgrm_CT_Cxn }*
108 ddgrm_CT_DataModel =
109   element ptLst { ddgrm_CT_PtList },
110   element cxnLst { ddgrm_CT_CxnList }?,
111   element bg { a_CT_BackgroundFormatting }?,
112   element whole { a_CT_WholeE2oFormatting }?,
113   element extLst { a_CT_OfficeArtExtensionList }?
114 ddgrm_dataModel = element dataModel { ddgrm_CT_DataModel }
115 ddgrm_AG_IteratorAttributes =
116
117  ## default value: none
118  attribute axis { ddgrm_ST_AxisTypes }?,
119
120  ## default value: all
121  attribute ptType { ddgrm_ST_ElementTypes }?,
122
123  ## default value: true

```

```

124 attribute hideLastTrans { ddgrm_ST_Booleans }?,
125
126 ## default value: 1
127 attribute st { ddgrm_ST_Ints }?,
128
129 ## default value: 0
130 attribute cnt { ddgrm_ST_UnsignedInts }?,
131
132 ## default value: 1
133 attribute step { ddgrm_ST_Ints }?
134 ddgrm_AG_ConstraintAttributes =
135 attribute type { ddgrm_ST_ConstraintType },
136
137 ## default value: self
138 attribute for { ddgrm_ST_ConstraintRelationship }?,
139 attribute forName { xsd:string }?,
140
141 ## default value: all
142 attribute ptType { ddgrm_ST_ElementType }?
143 ddgrm_AG_ConstraintRefAttributes =
144
145 ## default value: none
146 attribute refType { ddgrm_ST_ConstraintType }?,
147
148 ## default value: self
149 attribute refFor { ddgrm_ST_ConstraintRelationship }?,
150 attribute refForName { xsd:string }?,
151
152 ## default value: all
153 attribute refPtType { ddgrm_ST_ElementType }?
154 ddgrm_CT_Constraint =
155 ddgrm_AG_ConstraintAttributes,
156 ddgrm_AG_ConstraintRefAttributes,
157
158 ## default value: none
159 attribute op { ddgrm_ST_BoolOperator }?,
160
161 ## default value: 0
162 attribute val { xsd:double }?,
163
164 ## default value: 1
165 attribute fact { xsd:double }?,
166 element extLst { a_CT_OfficeArtExtensionList }?
167 ddgrm_CT_Constraints = element constr { ddgrm_CT_Constraint }*
168 ddgrm_CT_NumericRule =
169 ddgrm_AG_ConstraintAttributes,
170
171 ## default value: NaN
172 attribute val { xsd:double }?,
173
174 ## default value: NaN
175 attribute fact { xsd:double }?,
176

```

```

177   ## default value: NaN
178   attribute max { xsd:double }?,
179   element extLst { a_CT_OfficeArtExtensionList }?
180 ddgrm_CT_Rules = element rule { ddgrm_CT_NumericRule }*
181 ddgrm_CT_PresentationOf =
182   ddgrm_AG_IteratorAttributes,
183   element extLst { a_CT_OfficeArtExtensionList }?
184 ddgrm_ST_LayoutShapeType = a_ST_ShapeType | ddgrm_ST_OutputShapeType
185 ddgrm_ST_Index1 = xsd:unsignedInt { minInclusive = "1" }
186 ddgrm_CT_Adj =
187   attribute idx { ddgrm_ST_Index1 },
188   attribute val { xsd:double }
189 ddgrm_CT_AdjLst = element adj { ddgrm_CT_Adj }*
190 ddgrm_CT_Shape =
191
192   ## default value: 0
193   attribute rot { xsd:double }?,
194
195   ## default value: none
196   attribute type { ddgrm_ST_LayoutShapeType }?,
197   r_blip?,
198
199   ## default value: 0
200   attribute zOrderOff { xsd:int }?,
201
202   ## default value: false
203   attribute hideGeom { xsd:boolean }?,
204
205   ## default value: false
206   attribute lkTxEntry { xsd:boolean }?,
207
208   ## default value: false
209   attribute blipPhldr { xsd:boolean }?,
210   element adjLst { ddgrm_CT_AdjLst }?,
211   element extLst { a_CT_OfficeArtExtensionList }?
212 ddgrm_CT_Parameter =
213   attribute type { ddgrm_ST_ParameterId },
214   attribute val { ddgrm_ST_ParameterVal }
215 ddgrm_CT_Algorithm =
216   attribute type { ddgrm_ST_AlgorithmType },
217
218   ## default value: 0
219   attribute rev { xsd:unsignedInt }?,
220   element param { ddgrm_CT_Parameter }*,
221   element extLst { a_CT_OfficeArtExtensionList }?
222 ddgrm_CT_LayoutNode =
223   attribute name { xsd:string }?,
224   attribute styleLbl { xsd:string }?,
225
226   ## default value: b
227   attribute chOrder { ddgrm_ST_ChildOrderType }?,
228   attribute moveWith { xsd:string }?,
229   (element alg { ddgrm_CT_Algorithm }?)

```

```

230 | element shape { ddgrm_CT_Shape }?
231 | element presOf { ddgrm_CT_PresentationOf }?
232 | element constrLst { ddgrm_CT_Constraints }?
233 | element ruleLst { ddgrm_CT_Rules }?
234 | element varLst { ddgrm_CT_LayoutVariablePropertySet }?
235 | element forEach { ddgrm_CT_ForEach }
236 | element layoutNode { ddgrm_CT_LayoutNode }
237 | element choose { ddgrm_CT_Choose }
238 | element extLst { a_CT_OfficeArtExtensionList }?)*
239 ddgrm_CT_ForEach =
240   attribute name { xsd:string }?,
241   attribute ref { xsd:string }?,
242   ddgrm_AG_IteratorAttributes,
243   (element alg { ddgrm_CT_Algorithm }?
244     | element shape { ddgrm_CT_Shape }?
245     | element presOf { ddgrm_CT_PresentationOf }?
246     | element constrLst { ddgrm_CT_Constraints }?
247     | element ruleLst { ddgrm_CT_Rules }?
248     | element forEach { ddgrm_CT_ForEach }
249     | element layoutNode { ddgrm_CT_LayoutNode }
250     | element choose { ddgrm_CT_Choose }
251     | element extLst { a_CT_OfficeArtExtensionList }?)*
252 ddgrm_CT_When =
253   attribute name { xsd:string }?,
254   ddgrm_AG_IteratorAttributes,
255   attribute func { ddgrm_ST_FunctionType },
256
257   ## default value: none
258   attribute arg { ddgrm_ST_FunctionArgument }?,
259   attribute op { ddgrm_ST_FunctionOperator },
260   attribute val { ddgrm_ST_FunctionValue },
261   (element alg { ddgrm_CT_Algorithm }?
262     | element shape { ddgrm_CT_Shape }?
263     | element presOf { ddgrm_CT_PresentationOf }?
264     | element constrLst { ddgrm_CT_Constraints }?
265     | element ruleLst { ddgrm_CT_Rules }?
266     | element forEach { ddgrm_CT_ForEach }
267     | element layoutNode { ddgrm_CT_LayoutNode }
268     | element choose { ddgrm_CT_Choose }
269     | element extLst { a_CT_OfficeArtExtensionList }?)*
270 ddgrm_CT_Otherwise =
271   attribute name { xsd:string }?,
272   (element alg { ddgrm_CT_Algorithm }?
273     | element shape { ddgrm_CT_Shape }?
274     | element presOf { ddgrm_CT_PresentationOf }?
275     | element constrLst { ddgrm_CT_Constraints }?
276     | element ruleLst { ddgrm_CT_Rules }?
277     | element forEach { ddgrm_CT_ForEach }
278     | element layoutNode { ddgrm_CT_LayoutNode }
279     | element choose { ddgrm_CT_Choose }
280     | element extLst { a_CT_OfficeArtExtensionList }?)*
281 ddgrm_CT_Choose =
282   attribute name { xsd:string }?,

```

```

283     element if { ddgrm_CT_When }+,
284     element else { ddgrm_CT_Otherwise }?
285 ddgrm_CT_SampleData =
286
287     ## default value: false
288     attribute useDef { xsd:boolean }?,
289     element dataModel { ddgrm_CT_DataModel }?
290 ddgrm_CT_Category =
291     attribute type { xsd:anyURI },
292     attribute pri { xsd:unsignedInt }
293 ddgrm_CT_Categories = element cat { ddgrm_CT_Category }*
294 ddgrm_CT_Name =
295     attribute lang { xsd:string }?,
296     attribute val { xsd:string }
297 ddgrm_CT_Description =
298     attribute lang { xsd:string }?,
299     attribute val { xsd:string }
300 ddgrm_CT_DiagramDefinition =
301     attribute uniqueId { xsd:string }?,
302
303     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
304     attribute minVer { xsd:string }?,
305     attribute defStyle { xsd:string }?,
306     element title { ddgrm_CT_Name }*,
307     element desc { ddgrm_CT_Description }*,
308     element catLst { ddgrm_CT_Categories }?,
309     element sampData { ddgrm_CT_SampleData }?,
310     element styleData { ddgrm_CT_SampleData }?,
311     element clrData { ddgrm_CT_SampleData }?,
312     element layoutNode { ddgrm_CT_LayoutNode },
313     element extLst { a_CT_OfficeArtExtensionList }?
314 ddgrm_layoutDef = element layoutDef { ddgrm_CT_DiagramDefinition }
315 ddgrm_CT_DiagramDefinitionHeader =
316     attribute uniqueId { xsd:string },
317
318     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
319     attribute minVer { xsd:string }?,
320     attribute defStyle { xsd:string }?,
321
322     ## default value: 0
323     attribute resId { xsd:int }?,
324     element title { ddgrm_CT_Name }+,
325     element desc { ddgrm_CT_Description }+,
326     element catLst { ddgrm_CT_Categories }?,
327     element extLst { a_CT_OfficeArtExtensionList }?
328 ddgrm_layoutDefHdr =
329     element layoutDefHdr { ddgrm_CT_DiagramDefinitionHeader }
330 ddgrm_CT_DiagramDefinitionHeaderLst =
331     element layoutDefHdr { ddgrm_CT_DiagramDefinitionHeader }*
332 ddgrm_layoutDefHdrLst =
333     element layoutDefHdrLst { ddgrm_CT_DiagramDefinitionHeaderLst }
334 ddgrm_CT_RelIds = r_dm, r_lo, r_qs, r_cs
335 ddgrm_relIds = element relIds { ddgrm_CT_RelIds }

```



```

336 ddgrm_ST_ParameterVal =
337     ddgrm_ST_DiagramHorizontalAlignment
338     | ddgrm_ST_VerticalAlignment
339     | ddgrm_ST_ChildDirection
340     | ddgrm_ST_ChildAlignment
341     | ddgrm_ST_SecondaryChildAlignment
342     | ddgrm_ST_LinearDirection
343     | ddgrm_ST_SecondaryLinearDirection
344     | ddgrm_ST_StartingElement
345     | ddgrm_ST_BendPoint
346     | ddgrm_ST_ConnectorRouting
347     | ddgrm_ST_ArrowheadStyle
348     | ddgrm_ST_ConnectorDimension
349     | ddgrm_ST_RotationPath
350     | ddgrm_ST_CenterShapeMapping
351     | ddgrm_ST_NodeHorizontalAlignment
352     | ddgrm_ST_NodeVerticalAlignment
353     | ddgrm_ST_FallbackDimension
354     | ddgrm_ST_TextDirection
355     | ddgrm_ST_PyramidAccentPosition
356     | ddgrm_ST_PyramidAccentTextMargin
357     | ddgrm_ST_TextBlockDirection
358     | ddgrm_ST_TextAnchorHorizontal
359     | ddgrm_ST_TextAnchorVertical
360     | ddgrm_ST_DiagramTextAlignment
361     | ddgrm_ST_AutoTextRotation
362     | ddgrm_ST_GrowDirection
363     | ddgrm_ST_FlowDirection
364     | ddgrm_ST_ContinueDirection
365     | ddgrm_ST_Breakpoint
366     | ddgrm_ST_Offset
367     | ddgrm_ST_HierarchyAlignment
368     | xsd:int
369     | xsd:double
370     | xsd:boolean
371     | xsd:string
372     | ddgrm_ST_ConnectorPoint
373 ddgrm_ST_ModelId = xsd:int | s_ST_Guid
374 ddgrm_CT_ElemPropSet =
375     attribute presAssocID { ddgrm_ST_ModelId }?,
376     attribute presName { xsd:string }?,
377     attribute presStyleLbl { xsd:string }?,
378     attribute presStyleIdx { xsd:int }?,
379     attribute presStyleCnt { xsd:int }?,
380     attribute loTypeId { xsd:string }?,
381     attribute loCatId { xsd:string }?,
382     attribute qsTypeId { xsd:string }?,
383     attribute qsCatId { xsd:string }?,
384     attribute csTypeId { xsd:string }?,
385     attribute csCatId { xsd:string }?,
386     attribute coherent3DOff { xsd:boolean }?,
387     attribute phldrT { xsd:string }?,
388     attribute phldr { xsd:boolean }?,

```

```

389     attribute custAng { xsd:int }?,
390     attribute custFlipVert { xsd:boolean }?,
391     attribute custFlipHor { xsd:boolean }?,
392     attribute custSzX { xsd:int }?,
393     attribute custSzY { xsd:int }?,
394     attribute custScaleX { xsd:int }?,
395     attribute custScaleY { xsd:int }?,
396     attribute custT { xsd:boolean }?,
397     attribute custLinFactX { xsd:int }?,
398     attribute custLinFactY { xsd:int }?,
399     attribute custLinFactNeighborX { xsd:int }?,
400     attribute custLinFactNeighborY { xsd:int }?,
401     attribute custRadScaleRad { xsd:int }?,
402     attribute custRadScaleInc { xsd:int }?,
403     element presLayoutVars { ddgrm_CT_LayoutVariablePropertySet }?,
404     element style { a_CT_ShapeStyle }?
ddgrm_ST_Direction = "norm" | "rev"
406 ddgrm_ST_HierBranchStyle = "l" | "r" | "hang" | "std" | "init"
407 ddgrm_ST_AnimOneStr = "none" | "one" | "branch"
408 ddgrm_ST_AnimLvlStr = "none" | "lvl" | "ctr"
409 ddgrm_CT_OrgChart =
410
411     ## default value: false
412     attribute val { xsd:boolean }?
413 ddgrm_ST_NodeCount = xsd:int { minInclusive = "-1" }
414 ddgrm_CT_ChildMax =
415
416     ## default value: -1
417     attribute val { ddgrm_ST_NodeCount }?
418 ddgrm_CT_ChildPref =
419
420     ## default value: -1
421     attribute val { ddgrm_ST_NodeCount }?
422 ddgrm_CT_BulletEnabled =
423
424     ## default value: false
425     attribute val { xsd:boolean }?
426 ddgrm_CT_Direction =
427
428     ## default value: norm
429     attribute val { ddgrm_ST_Direction }?
430 ddgrm_CT_HierBranchStyle =
431
432     ## default value: std
433     attribute val { ddgrm_ST_HierBranchStyle }?
434 ddgrm_CT_AnimOne =
435
436     ## default value: one
437     attribute val { ddgrm_ST_AnimOneStr }?
438 ddgrm_CT_AnimLvl =
439
440     ## default value: none
441     attribute val { ddgrm_ST_AnimLvlStr }?

```

```

442 ddgrm_ST_ResizeHandlesStr = "exact" | "rel"
443 ddgrm_CT_ResizeHandles =
444
445     ## default value: rel
446     attribute val { ddgrm_ST_ResizeHandlesStr }?
447 ddgrm_CT_LayoutVariablePropertySet =
448     element orgChart { ddgrm_CT_OrgChart }?,
449     element chMax { ddgrm_CT_ChildMax }?,
450     element chPref { ddgrm_CT_ChildPref }?,
451     element bulletEnabled { ddgrm_CT_BulletEnabled }?,
452     element dir { ddgrm_CT_Direction }?,
453     element hierBranch { ddgrm_CT_HierBranchStyle }?,
454     element animOne { ddgrm_CT_AnimOne }?,
455     element animLvl { ddgrm_CT_AnimLvl }?,
456     element resizeHandles { ddgrm_CT_ResizeHandles }?
457 ddgrm_CT_SDName =
458     attribute lang { xsd:string }?,
459     attribute val { xsd:string }
460 ddgrm_CT_SDDescription =
461     attribute lang { xsd:string }?,
462     attribute val { xsd:string }
463 ddgrm_CT_SDCategory =
464     attribute type { xsd:anyURI },
465     attribute pri { xsd:unsignedInt }
466 ddgrm_CT_SDCategories = element cat { ddgrm_CT_SDCategory }*
467 ddgrm_CT_TextProps = a_EG_Text3D?
468 ddgrm_CT_StyleLabel =
469     attribute name { xsd:string },
470     element scene3d { a_CT_Scene3D }?,
471     element sp3d { a_CT_Shape3D }?,
472     element txPr { ddgrm_CT_TextProps }?,
473     element style { a_CT_ShapeStyle }?,
474     element extLst { a_CT_OfficeArtExtensionList }?
475 ddgrm_CT_StyleDefinition =
476     attribute uniqueId { xsd:string }?,
477
478     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
479     attribute minVer { xsd:string }?,
480     element title { ddgrm_CT_SDName }*,
481     element desc { ddgrm_CT_SDDescription }*,
482     element catLst { ddgrm_CT_SDCategories }?,
483     element scene3d { a_CT_Scene3D }?,
484     element styleLbl { ddgrm_CT_StyleLabel }+,
485     element extLst { a_CT_OfficeArtExtensionList }?
486 ddgrm_styleDef = element styleDef { ddgrm_CT_StyleDefinition }
487 ddgrm_CT_StyleDefinitionHeader =
488     attribute uniqueId { xsd:string },
489
490     ## default value: http://schemas.openxmlformats.org/drawingml/2006/diagram
491     attribute minVer { xsd:string }?,
492
493     ## default value: 0
494     attribute resId { xsd:int }?,

```

```

495     element title { ddgrm_CT_SDName }+,
496     element desc { ddgrm_CT_SDDescription }+,
497     element catLst { ddgrm_CT_SDCategories }?,
498     element extLst { a_CT_OfficeArtExtensionList }?
499 ddgrm_styleDefHdr =
500     element styleDefHdr { ddgrm_CT_StyleDefinitionHeader }
501 ddgrm_CT_StyleDefinitionHeaderLst =
502     element styleDefHdr { ddgrm_CT_StyleDefinitionHeader }*
503 ddgrm_styleDefHdrLst =
504     element styleDefHdrLst { ddgrm_CT_StyleDefinitionHeaderLst }
505 ddgrm_ST_AlgorithmType =
506     "composite"
507     | "conn"
508     | "cycle"
509     | "hierChild"
510     | "hierRoot"
511     | "pyra"
512     | "lin"
513     | "sp"
514     | "tx"
515     | "snake"
516 ddgrm_ST_AxisType =
517     "self"
518     | "ch"
519     | "des"
520     | "desOrSelf"
521     | "par"
522     | "ancst"
523     | "ancstOrSelf"
524     | "followSib"
525     | "precedSib"
526     | "follow"
527     | "preced"
528     | "root"
529     | "none"
530 ddgrm_ST_AxisTypes = list { ddgrm_ST_AxisType* }
531 ddgrm_ST_BoolOperator = "none" | "equ" | "gte" | "lte"
532 ddgrm_ST_ChildOrderType = "b" | "t"
533 ddgrm_ST_ConstraintType =
534     "none"
535     | "alignOff"
536     | "begMarg"
537     | "bendDist"
538     | "begPad"
539     | "b"
540     | "bMarg"
541     | "bOff"
542     | "ctrX"
543     | "ctrXOff"
544     | "ctrY"
545     | "ctrYOff"
546     | "connDist"
547     | "diam"

```

```

548 | "endMarg"
549 | "endPad"
550 | "h"
551 | "hArH"
552 | "hOff"
553 | "l"
554 | "lMarg"
555 | "lOff"
556 | "r"
557 | "rMarg"
558 | "rOff"
559 | "primFontSz"
560 | "pyraAcctRatio"
561 | "secFontSz"
562 | "sibSp"
563 | "secSibSp"
564 | "sp"
565 | "stemThick"
566 | "t"
567 | "tMarg"
568 | "tOff"
569 | "userA"
570 | "userB"
571 | "userC"
572 | "userD"
573 | "userE"
574 | "userF"
575 | "userG"
576 | "userH"
577 | "userI"
578 | "userJ"
579 | "userK"
580 | "userL"
581 | "userM"
582 | "userN"
583 | "userO"
584 | "userP"
585 | "userQ"
586 | "userR"
587 | "userS"
588 | "userT"
589 | "userU"
590 | "userV"
591 | "userW"
592 | "userX"
593 | "userY"
594 | "userZ"
595 | "w"
596 | "wArH"
597 | "wOff"
598 ddgrm_ST_ConstraintRelationship = "self" | "ch" | "des"
599 ddgrm_ST_ElementType =
600 "all"

```

```

601 | "doc"
602 | "node"
603 | "norm"
604 | "nonNorm"
605 | "asst"
606 | "nonAsst"
607 | "parTrans"
608 | "pres"
609 | "sibTrans"
610 ddgrm_ST_ElementTypes = list { ddgrm_ST_ElementType* }
611 ddgrm_ST_ParameterId =
612     "horzAlign"
613     | "vertAlign"
614     | "chDir"
615     | "chAlign"
616     | "secChAlign"
617     | "linDir"
618     | "secLinDir"
619     | "stElem"
620     | "bendPt"
621     | "connRout"
622     | "begSty"
623     | "endSty"
624     | "dim"
625     | "rotPath"
626     | "ctrShpMap"
627     | "nodeHorzAlign"
628     | "nodeVertAlign"
629     | "fallback"
630     | "txDir"
631     | "pyraAcctPos"
632     | "pyraAcctTxMar"
633     | "txBldir"
634     | "txAnchorHorz"
635     | "txAnchorVert"
636     | "txAnchorHorzCh"
637     | "txAnchorVertCh"
638     | "parTxLTRAlign"
639     | "parTxRTLAlign"
640     | "shpTxLTRAlignCh"
641     | "shpTxRTLAlignCh"
642     | "autoTxRot"
643     | "grDir"
644     | "flowDir"
645     | "contDir"
646     | "bkpt"
647     | "off"
648     | "hierAlign"
649     | "bkPtFixedVal"
650     | "stBulletLvl"
651     | "stAng"
652     | "spanAng"
653     | "ar"

```

```

654 | "lnSpPar"
655 | "lnSpAfParP"
656 | "lnSpCh"
657 | "lnSpAfChP"
658 | "rtShortDist"
659 | "alignTx"
660 | "pyraLvlNode"
661 | "pyraAcctBkgdNode"
662 | "pyraAcctTxNode"
663 | "srcNode"
664 | "dstNode"
665 | "begPts"
666 | "endPts"
667 ddgrm_ST_Ints = list { xsd:int* }
668 ddgrm_ST_UnsignedInts = list { xsd:unsignedInt* }
669 ddgrm_ST_Booleans = list { xsd:boolean* }
670 ddgrm_ST_FunctionType =
671   "cnt"
672   | "pos"
673   | "revPos"
674   | "posEven"
675   | "posOdd"
676   | "var"
677   | "depth"
678   | "maxDepth"
679 ddgrm_ST_FunctionOperator = "equ" | "neq" | "gt" | "lt" | "gte" | "lte"
680 ddgrm_ST_DiagramHorizontalAlignment = "l" | "ctr" | "r" | "none"
681 ddgrm_ST_VerticalAlignment = "t" | "mid" | "b" | "none"
682 ddgrm_ST_ChildDirection = "horz" | "vert"
683 ddgrm_ST_ChildAlignment = "t" | "b" | "l" | "r"
684 ddgrm_ST_SecondaryChildAlignment = "none" | "t" | "b" | "l" | "r"
685 ddgrm_ST_LinearDirection = "fromL" | "fromR" | "fromT" | "fromB"
686 ddgrm_ST_SecondaryLinearDirection =
687   "none" | "fromL" | "fromR" | "fromT" | "fromB"
688 ddgrm_ST_StartingElement = "node" | "trans"
689 ddgrm_ST_RotationPath = "none" | "alongPath"
690 ddgrm_ST_CenterShapeMapping = "none" | "fNode"
691 ddgrm_ST_BendPoint = "beg" | "def" | "end"
692 ddgrm_ST_ConnectorRouting = "stra" | "bend" | "curve" | "longCurve"
693 ddgrm_ST_ArrowheadStyle = "auto" | "arr" | "noArr"
694 ddgrm_ST_ConnectorDimension = "1D" | "2D" | "cust"
695 ddgrm_ST_ConnectorPoint =
696   "auto"
697   | "bCtr"
698   | "ctr"
699   | "midL"
700   | "midR"
701   | "tCtr"
702   | "bL"
703   | "bR"
704   | "tL"
705   | "tR"
706   | "radial"

```

```

707 ddgrm_ST_NodeHorizontalAlignment = "l" | "ctr" | "r"
708 ddgrm_ST_NodeVerticalAlignment = "t" | "mid" | "b"
709 ddgrm_ST_FallbackDimension = "1D" | "2D"
710 ddgrm_ST_TextDirection = "fromT" | "fromB"
711 ddgrm_ST_PyramidAccentPosition = "bef" | "aft"
712 ddgrm_ST_PyramidAccentTextMargin = "step" | "stack"
713 ddgrm_ST_TextBlockDirection = "horz" | "vert"
714 ddgrm_ST_TextAnchorHorizontal = "none" | "ctr"
715 ddgrm_ST_TextAnchorVertical = "t" | "mid" | "b"
716 ddgrm_ST_DiagramTextAlignment = "l" | "ctr" | "r"
717 ddgrm_ST_AutoTextRotation = "none" | "upr" | "grav"
718 ddgrm_ST_GrowDirection = "tL" | "tR" | "bL" | "bR"
719 ddgrm_ST_FlowDirection = "row" | "col"
720 ddgrm_ST_ContinueDirection = "revDir" | "sameDir"
721 ddgrm_ST_Breakpoint = "endCnv" | "bal" | "fixed"
722 ddgrm_ST_Offset = "ctr" | "off"
723 ddgrm_ST_HierarchyAlignment =
724     "tL"
725     | "tR"
726     | "tCtrCh"
727     | "tCtrDes"
728     | "bL"
729     | "bR"
730     | "bCtrCh"
731     | "bCtrDes"
732     | "lT"
733     | "lB"
734     | "lCtrCh"
735     | "lCtrDes"
736     | "rT"
737     | "rB"
738     | "rCtrCh"
739     | "rCtrDes"
740 ddgrm_ST_FunctionValue =
741     xsd:int
742     | xsd:boolean
743     | ddgrm_ST_Direction
744     | ddgrm_ST_HierBranchStyle
745     | ddgrm_ST_AnimOneStr
746     | ddgrm_ST_AnimLvlStr
747     | ddgrm_ST_ResizeHandlesStr
748 ddgrm_ST_VariableType =
749     "none"
750     | "orgChart"
751     | "chMax"
752     | "chPref"
753     | "bulEnabled"
754     | "dir"
755     | "hierBranch"
756     | "animOne"
757     | "animLvl"
758     | "resizeHandles"
759 ddgrm_ST_FunctionArgument = ddgrm_ST_VariableType

```


760 `ddgrm_ST_OutputShapeType = "none" | "conn"`

B.5.3.1 Part Schemas

B.5.3.1.1 Diagram Colors Part

```

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 include "dml-compatibility.rnc"
11 start = ddgrm_colorsDef

```

B.5.3.1.2 Diagram Data Part

```

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 include "dml-compatibility.rnc"
11 start = ddgrm_dataModel

```

B.5.3.1.3 Diagram Layout Definitions Part

```

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 include "dml-compatibility.rnc"
11 start = ddgrm_layoutDef

```

B.5.3.1.4 Diagram Style Part

```

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"
include "dml-compatibility.rnc"
start = ddgrm_styleDef

```

B.6 VML

B.6.1 VML - Main

```

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_targetscreenSize?,
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?)
304 v_CT_TextPath =
305     v_AG_Id,
306     v_AG_Style,
307     attribute on { s_ST_TrueFalse }?,
308     attribute fitshape { s_ST_TrueFalse }?,
309     attribute fitpath { s_ST_TrueFalse }?,

```

```

310     attribute trim { s_ST_TrueFalse }?,
311     attribute xscale { s_ST_TrueFalse }?,
312     attribute string { xsd:string }?
313 v_arc = element arc { v_CT_Arc }
314 v_curve = element curve { v_CT_Curve }
315 v_image = element image { v_CT_Image }
316 v_line = element line { v_CT_Line }
317 v_oval = element oval { v_CT_Oval }
318 v_polyline = element polyline { v_CT_PolyLine }
319 v_rect = element rect { v_CT_Rect }
320 v_roundrect = element roundrect { v_CT_RoundRect }
321 v_CT_Arc =
322     v_AG_AllCoreAttributes,
323     v_AG_AllShapeAttributes,
324     attribute startAngle { xsd:decimal }?,
325     attribute endAngle { xsd:decimal }?,
326     v_EG_ShapeElements*
327 v_CT_Curve =
328     v_AG_AllCoreAttributes,
329     v_AG_AllShapeAttributes,
330     attribute from { xsd:string }?,
331     attribute control1 { xsd:string }?,
332     attribute control2 { xsd:string }?,
333     attribute to { xsd:string }?,
334     v_EG_ShapeElements*
335 v_CT_Image =
336     v_AG_AllCoreAttributes,
337     v_AG_AllShapeAttributes,
338     v_AG_ImageAttributes,
339     v_EG_ShapeElements*
340 v_CT_Line =
341     v_AG_AllCoreAttributes,
342     v_AG_AllShapeAttributes,
343     attribute from { xsd:string }?,
344     attribute to { xsd:string }?,
345     v_EG_ShapeElements*
346 v_CT_Oval =
347     v_AG_AllCoreAttributes,
348     v_AG_AllShapeAttributes,
349     (v_EG_ShapeElements)*
350 v_CT_PolyLine =
351     v_AG_AllCoreAttributes,
352     v_AG_AllShapeAttributes,
353     attribute points { xsd:string }?,
354     (v_EG_ShapeElements | o_ink)*
355 v_CT_Rect =
356     v_AG_AllCoreAttributes,
357     v_AG_AllShapeAttributes,
358     (v_EG_ShapeElements)*
359 v_CT_RoundRect =
360     v_AG_AllCoreAttributes,
361     v_AG_AllShapeAttributes,
362     attribute arcsize { xsd:string }?,

```



```

363 (v_EG_ShapeElements*)+
364 v_ST_Ext = string "view" | string "edit" | string "backwardCompatible"
365 v_ST_FillType =
366     string "solid"
367     | string "gradient"
368     | string "gradientRadial"
369     | string "tile"
370     | string "pattern"
371     | string "frame"
372 v_ST_FillMethod =
373     string "none"
374     | string "linear"
375     | string "sigma"
376     | string "any"
377     | string "linear sigma"
378 v_ST_ShadowType =
379     string "single"
380     | string "double"
381     | string "emboss"
382     | string "perspective"
383 v_ST_StrokeLineStyle =
384     string "single"
385     | string "thinThin"
386     | string "thinThick"
387     | string "thickThin"
388     | string "thickBetweenThin"
389 v_ST_StrokeJoinStyle = string "round" | string "bevel" | string "miter"
390 v_ST_StrokeEndCap = string "flat" | string "square" | string "round"
391 v_ST_StrokeArrowLength =
392     string "short" | string "medium" | string "long"
393 v_ST_StrokeArrowWidth =
394     string "narrow" | string "medium" | string "wide"
395 v_ST_StrokeArrowType =
396     string "none"
397     | string "block"
398     | string "classic"
399     | string "oval"
400     | string "diamond"
401     | string "open"
402 v_ST_ImageAspect = string "ignore" | string "atMost" | string "atLeast"
403 v_ST_EditAs =
404     string "canvas"
405     | string "orgchart"
406     | string "radial"
407     | string "cycle"
408     | string "stacked"
409     | string "venn"
410     | string "bullseye"

```

B.6.2 VML - Office Drawing

```

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_targetsize = attribute o:targetsize { 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_OLEDDrawAspect = 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

```

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

```

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

```

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.7 Shared MLs

B.7.1 Math

```

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_RunLevelElts
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

```

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

```

1 include "shared-documentPropertiesExtended.rnc"
2 include "shared-documentPropertiesVariantTypes.rnc"
3 include "shared-commonSimpleTypes.rnc"
4 start = shdDcEP_Properties

```

B.7.3 Custom Properties

```

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:vml"
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_ostream
25 | vt_oleobj
26 | vt_oleobj
27 | vt_oleobj
28 | vt_oleobj
29 | vt_oleobj
30 | vt_oleobj
31 | vt_oleobj
32 | vt_oleobj
33 | vt_oleobj
34 | vt_oleobj
35 | vt_oleobj
36 | vt_oleobj
37 | vt_oleobj
38 | vt_oleobj
39 | vt_oleobj
40 | vt_oleobj
41 | vt_oleobj
42 | vt_oleobj
43 | vt_oleobj
44 | vt_oleobj
45 | vt_oleobj
46 | vt_oleobj
47 | vt_oleobj
48 | vt_oleobj
49 | vt_oleobj
50 | vt_oleobj
51 | vt_oleobj
52 | vt_oleobj
53 | vt_oleobj

```

B.7.3.1 Part Schemas

B.7.3.1.1 Custom File Properties Part

```

1 include "shared-documentPropertiesCustom.rnc"
2 include "shared-documentPropertiesVariantTypes.rnc"
3 include "shared-commonSimpleTypes.rnc"
4 start = shdCstm_Properties

```

B.7.4 Variant Types

```

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

```

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:vml"
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

```

1 include "shared-customXmlDataProperties.rnc"
2 include "shared-commonSimpleTypes.rnc"
3 start = ds_datastoreItem

```

B.7.6 Bibliography

```

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:vml"
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

```

1 include "shared-bibliography.rnc"
2 include "shared-commonSimpleTypes.rnc"
3 start = shrdBib_Sources
```

B.7.7 Additional Characteristics

```

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

```

1 include "shared-additionalCharacteristics.rnc"
2 start = shrdChr_additionalCharacteristics

```

B.7.8 Office Document Relationships

```

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:vml"
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

```

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:vml"
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_AlgorithmClass = string "hash" | string "custom"
27 s_ST_CryptProv = string "rsaAES" | string "rsaFull" | string "custom"
28 s_ST_AlgorithmType = 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
36 s_ST_String = xsd:string
37 s_ST_TrueFalse =
38     string "t" | string "f" | string "true" | string "false"
39 s_ST_TrueFalseBlank =
40     string "t"
41     | string "f"
42     | string "true"
43     | string "false"
44     | string ""
45     | string "True"
46     | string "False"
47 s_ST_UnsignedDecimalNumber = xsd:unsignedLong
48 s_ST_TwipsMeasure =
49     s_ST_UnsignedDecimalNumber | s_ST_PositiveUniversalMeasure
50 s_ST_VerticalAlignRun =
51     string "baseline" | string "superscript" | string "subscript"
52 s_ST_Xstring = xsd:string
53 s_ST_XAlign =
54     string "left"
55     | string "center"
56     | string "right"

```

```

57 | string "inside"
58 | string "outside"
59 s_ST_YAlign =
60   string "inline"
61   | string "top"
62   | string "center"
63   | string "bottom"
64   | string "inside"
65   | string "outside"
66 s_ST_ConformanceClass = string "strict" | string "transitional"
67 s_ST_UniversalMeasure =
68   xsd:string { pattern = "-?[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)" }
69 s_ST_PositiveUniversalMeasure =
70   xsd:string {
71     pattern = "-?[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)"
72     pattern = "[0-9]+(\\.[0-9]+)?(mm|cm|in|pt|pc|pi)"
73   }
74 s_ST_Percentage = xsd:string { pattern = "-?[0-9]+(\\.[0-9]+)?%" }
75 s_ST_FixedPercentage =
76   xsd:string {
77     pattern = "-?[0-9]+(\\.[0-9]+)?%"
78     pattern = "-?((100)|([0-9][0-9]?))(\\.[0-9][0-9]?)?%"
79   }
80 s_ST_PositivePercentage =
81   xsd:string {
82     pattern = "-?[0-9]+(\\.[0-9]+)?%"
83     pattern = "[0-9]+(\\.[0-9]+)?%"
84   }
85 s_ST_PositiveFixedPercentage =
86   xsd:string {
87     pattern = "-?[0-9]+(\\.[0-9]+)?%"
88     pattern = "((100)|([0-9][0-9]?))(\\.[0-9][0-9]?)?%"
89   }

```

B.8 Custom XML Schema References

```

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:vml"
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

```
1 anyElement = element * { anyAttribute*, text?, anyElement* }  
2 anyAttribute = attribute * { text }
```

B.9.2 XML

```
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-376:2008 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-376:2008 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 (§9.10.8) 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 dateCompatibility attribute was added to the workbookPr element (Part 1, §18.2.28)
- 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 valIso and maxValIso 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 (§11.2.1.1) 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 (§15.1.2.6) 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 (§14.2.3.10) was added
- The equationxml element (§14.2.2.10) was added
- The contentType attribute was added to the ink element (§14.2.2.15)
- The ST_AlternateMathContentType simple type (§14.2.3.1) was added
- The ST_OLELinkType simple type (§14.2.3.19) was modified to allow any xsd:string
- The ST_TrueFalseBlank simple type (§15.1.2.6) 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 (§14.4.3.1) was modified to allow any xsd:string
- The ST_TrueFalseBlank simple type (§15.1.2.6) 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_Align simple type (§15.1.2.1) now uses an enumeration value of custom in place of invalid
- The ST_AlignType simple type (§15.1.2.2) now uses an enumeration value of custom in place of invalid
- The ST_CryptProv simple type (§15.1.2.4) 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.