IMS Simple Sequencing XML Binding

Version 1.0 Final Specification

IPR and Distribution Notices

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

IMS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on IMS's procedures with respect to rights in IMS specifications can be found at the IMS Intellectual Property Rights web page: http://www.imsglobal.org/ipr/imsipr_policyFinal.pdf.

If you wish to copy or distribute this document, you must complete a valid Registered User license registration with IMS and receive an email from IMS granting the license to distribute the specification. To register, follow the instructions on the IMS website: http://www.imsglobal.org/specificationdownload.cfm.

This document may be copied and furnished to others by Registered Users who have registered on the IMS website provided that the above copyright notice and this paragraph are included on all such copies. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to IMS, except as needed for the purpose of developing IMS specifications, under the auspices of a chartered IMS project group.

Use of this specification to develop products or services is governed by the license with IMS found on the IMS website: http://www.imsglobal.org/license.html.

The limited permissions granted above are perpetual and will not be revoked by IMS or its successors or assigns.

THIS SPECIFICATION IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NONINFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE CONSORTIUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS SPECIFICATION.

Document Name: IMS Simple Sequencing XML Binding

Date: 03 March 2003

Table of Contents

1. Introduction

1.1 Simple Sequencing Specification Overview

This document describes the XML Binding for the IMS Simple Sequencing Information Model. It specifies an XML Schema binding conformant with the W3C's XML Schema Recommendation of 02 May 2001.

An XML binding using the Document Type Definition (DTD) format is specifically excluded from this binding specification. Other XML binding types such as Resource Description Format (RDF) may be added in the future.

The prose copy of the Sequencing Definition Model and Tracking Information Model contained within the Simple Sequencing Information and Behavior Model document are the normative and definitive specification of IMS Simple Sequencing. All binding structures specified herein are not normative or definitive, with the exception of names for XSD structures. The XSD structure names as applied to the Sequencing Definition Model elements may not be redefined, nor may any XSD structures.

1.2 Scope and Context

This document is the IMS Simple Sequencing XML Binding v1.0 Final Specification document and is derived from the corresponding Information and Behavior Model [SS, 03a]. As such it will be used as the basis for the development of the following document:

- IMS Simple Sequencing Best Practice and Implementation Guide v1.0 [SS, 03c].

1.3 Structure of this Document

The structure of this document is:

2. XML Basics	A brief description of the key basics of XML. XML is the implementation language for the realization of the IMS Simple Sequencing Information and Behavioral Model;
3. Narrative Description of the XML Binding	A detailed description of the realization of the IMS Simple Sequencing Information and Behavioral Model in XML;
4. Sequencing Definition Element Mapping	The mapping of the XML binding elements and attributes to the information model data objects;
5. Physical Realization of the XML Binding	The mapping of the XML binding elements and attributes to the different XML schema definition files (XSDs);
6. Normative Points	Important normative statements that must be maintained by implementations of this XML binding.

1.4 Nomenclature

CP	Content Packaging
DTD	Document Type Definition
RDF	Resource Description Format
SS	Simple Sequencing
W3C	World Wide Web Consortium
XML	Extensible Mark-up Language

XSD	XML Schema

1.5 References

[SS, 03a] *IMS Simple Sequencing Information and Behavior Model v1.0 Final Specification*, March 2003.

[SS, 03c] *IMS Simple Sequencing Best Practice and Implementation Guide v1.0 Final Specification*, March 2003.

2. XML Basics

The conceptual model for expressing sequencing definitions is a hierarchy. Hierarchical models are convenient for representing data consisting of many elements and sub-elements. XML is perfectly suited for representing hierarchical models. An XML document is a hierarchy comprised of **elements** that have **contents** and **attributes**.

2.1 Elements

An element is a component of a document that has been identified in a way a computer can understand. Each element has a **tag name**. When a tag name is shown as "<TAGNAME>", with less-than and greater-than symbols before and after the tag name, it serves as the **start-tag** to mark the beginning of an element. When that same tag name has a forward slash "/" added, it serves as an **end-tag** such as "</TAGNAME>". An element may have contents between its start and end-tags, and may have one or more **attributes**. When an XML element has a start and end-tag (also called an **opening** and **closing** tag) with a common name, it is considered to be "well-formed" XML. The contents of an element are placed between the start and end-tags as shown below:

```
<TAGNAME>contents</TAGNAME>
```

2.1.1 Element Contents

An element may contain other elements, Parsed Character Data (PCDATA), Character Data (CDATA), or a mixture of PCDATA and elements. The allowable contents of an element are its **content model**. XML parsers treat PCDATA with their special or reserved meaning unless they are specifically marked (or "escaped"). In contrast, CDATA can use special or reserved characters without having to escape them, as CDATA is not read by XML parsers.

2.1.2 Element Attributes

An attribute provides additional information about an element. Attributes are a way of attaching characteristics or properties to the elements of a document. An element may have more than one attribute. Attributes are contained within the start tag of an element. Attributes are represented by an attribute name followed by an equal sign and the attribute value in quotation marks:

```
<timeframe>
   <begin restrict="1"> 1999-07-23 </begin>
</timeframe>
```

In this example, the `timeframe` element contains another element: the `begin` element. The `begin` element has one attribute "restrict", with the value "1". The value for the element `BEGIN` is `"1999-07-23"`. These two elements then make up a `timeframe begin` date.

2.1.3 Element Names

Each element has a unique name, referred to as the tag name. XML is case-sensitive in its processing of tag names. The IMS Simple Sequencing XML Binding Specification adheres to the following tag name rules:

- All tag names will conform to the rules for element naming as given within the XML 1.0 Specification;
- Names beginning in XML in any case or mix of cases are not permitted;
- The IMS binding will use camel case with initial lowercase tag and element names (such as "firstElementName");
- Element names may not include words reserved by the XML specification. These include:
  DOCTYPE
  ELEMENT
  ATTLIST
  ENTITY
- Tag names defined by this IMS Binding may not be redefined.

2.2 Document Type Definitions

The tag name, content model, and attributes of elements were historically defined in a **Document Type Definition** (DTD) statement. A DTD is a kind of schema. It may exist as an external file or a block of text internal to an XML document. However, the DTD schema was developed before object-oriented programming concepts became prevalent within software development communities and before the use of entities drawn from multiple namespaces became a requirement. Those communities began working on other schema representations to provide more object-like structures and procedures and to support the flexible use of multiple namespaces than could be realized within the constraints of the DTD schema for describing and structuring the contents of XML documents. One such schema language has become today's preferred representation language: XML Schema produced by the Worldwide Web Consortium (W3C).

2.3 XML Schemas

A schema is a formal specification of element names that indicates which elements are allowed in an XML instance, and in which combinations. New schema languages, such as those defined in the XML-Schemas Working Group, provide the same baseline functionality as a DTD. However, because these schema languages are extensible, developers can augment them with additional information, such as data types, inheritance, and presentation rules. This makes schema languages far more powerful than DTDs. For more information about XML schemas, visit http://www.w3.org/TR/smlschema-0/.

This specification defines a W3C XML Schema (imsss_bindv1p0.xsd) as a non-normative reference. Some XML editors may make use of these schemas to help guide the developer in creating the proper elements at the proper locations in an XML file. Other developers will make use of the schemas to validate their XML instances and/or to define extensions to the IMS Meta-Data Binding. Details of the construction of schemas are outside the scope of this document.

2.4 Valid Character Sets

An IMS Sequencing definition instance must use UTF-8 encoding of the character sets as defined in ISO 10646. See the XML Version 1.0 for more details on the specification of well-formed XML.

3. Narrative Description of the XML Binding

This section of the specification uses a series of graph segments, accompanied by simple narrative, to describe the XML format of the XML Schema definition (XSD) binding for the Sequencing Definition Model. XSD documents that implement this *abstract* format are referenced as non-normative parts of this specification.

These data elements and their relationships from the information model are expressed in XML Schema by means structuring entities called elements, attributes, and groups. These structuring entities are further typified as complex types or simple types. The XML Schema structuring entities comprise a binding of an information model when expressed in a special instance of an XML document – an XSD.

The Sequencing Definition Model also defines values or vocabulary terms required by certain data model elements. These controlled lists of terms or values can be expressed in XML Schema as restrictions placed on the values that may be declared for a given structuring element.

It is often necessary for bindings to create structuring entities with names that are not present in an information or data model that is being expressed in XML Schema. These special structuring entities enable the grouping and use of elements as defined in a data model.

The graphs below show individual elements encapsulated within larger structures, like complex types. Structures without child elements are not represented separately. The reader is referred to a binding document instance (.xsd) for the full lexical representation of all structures and values.

Key to graphical elements:

- Rectangle with square corners = element
- Rectangle with rounded corners, enclosed in red outline area = attribute
- Bold name in upper half of rectangle: element/attribute name
- Regular weight name in lower half of rectangle: type name
- Circled Multiplicity indicator to left of element, attribute, or group:
    - ? = 0..1
    - * = 0..n
    - + = 1..n
    - D = 0..1, with default value if absent (attributes only)
    - (no indicator) = 1

Red Branching lines = XML Schema grouping model connector

- Square bracket = “Sequence”
- Angle bracket = “Choice”

3.1 <sequencingCollection> Element

Description: This is a container for individual sequencing instructions. If used, references to the contained sequencing instructions are embedded in the IMS content package at the appropriate points. The actual sequencing details are then all collected and placed under the <sequencingCollection> element that is itself located at an agreed place within the content package.

Figure 3.1 - <sequencingCollection> element.

Multiplicity: Appears as a child of the <manifest> element within an IMS content package. Occurs zero or one times within any given manifest.

Attributes: None.

Elements:

- sequencing

Example:

The following minimal IMS CP manifest (stripped of namespace definitions for clarity) shows how an <organization> can reference sequencing information that is stored at the top level of the manifest.

```
<manifest identifier = "IMSSS.TestManifest.1">
   <organizations default = "IMSSS.TestManifest.1.Org.1">
      <organization identifier = "IMSSS.TestManifest.1.Org.1">
         <item identifier = "IMSSS.TestManifest.1.Item.1"/>
         <imsss:sequencing IDRef = "IMSSS.TestManifest.1.Seq.1"/>
      </organization>
   </organizations>
   <resources/>
   <imsss:sequencingCollection>
      <imsss:sequencing ID = "IMSSS.TestManifest.1.Seq.1">
         <!-- Sequencing info goes here -->
      </imsss:sequencing>
   </imsss:sequencingCollection>
</manifest>
```

The following minimal IMS CP manifest (stripped of namespace definitions for clarity) shows how locally defined information overrides referenced information when using both in the same <sequencing> element.

```
<manifest identifier = "IMSSS.TestManifest.1">
   <organizations default = "IMSSS.TestManifest.1.Org.1">
      <organization identifier = "IMSSS.TestManifest.1.Org.1">
         <item identifier = "IMSSS.TestManifest.1.Item.1"/>
         <imsss:sequencing IDRef = "IMSSS.TestManifest.1.Seq.1">
            <imsss:limitConditions attemptLimit="1"/>
            <!-- This element has a limit of 1 attempt, and the begin time limit is
                undefined, because the inline element overrides the referenced element.
                However, this element has choice and choiceExit both false, because of the
                referenced element. -->
      </organization>
   </organizations>
   <resources/>
   <imsss:sequencingCollection>
      <imsss:sequencing ID = "IMSSS.TestManifest.1.Seq.1">
         <imsss:controlMode choice="false" choiceExit="false"/>
         <imsss:limitConditions attemptLimit="2" beginTimeLimit="2002-10-16T12:00:00"/>
      </imsss:sequencing>
   </imsss:sequencingCollection>
</manifest>
```

3.2 <sequencing> Element

Description: Sequencing information is associated with items in a hierarchical structure by associating a single <sequencing> element with the hierarchical item. In the context of IMS CP, this is done by including the <sequencing> element within either an <item> element or an <organization> element.

Sequencing information can be reused across multiple items. This is done by using the “IDRef” attribute of the <sequencing> element, and assigning the ID of the <sequencing> element to use to the “IDRef” attribute. The <sequencing> elements that are referred to using this mechanism may be stored in a <sequencingCollection> element placed at the top level of the manifest.

Note the following restrictions on the use of “IDRef” attributes. <sequencing> elements that are children of the <sequencingCollection> element may not have an “IDRef” attribute. Further, the “IDRef” attribute, when used, must reference a <sequencing> element that is a child of the <sequencingCollection> element in the same manifest.

Note that the in-line definition of sequencing information (using the various elements contained within the <sequencing> element) and the referencing of sequencing information (using the “IDRef” attribute) are not mutually exclusive for any particular element in the hierarchical structure. If a <sequencing> element uses both the “IDRef” attribute and in-line definition, any top-level element defined in-line overrides any similar element defined in the referenced element. Note that this override is for the entire top-level element, and not for individual parts of the top-level element.

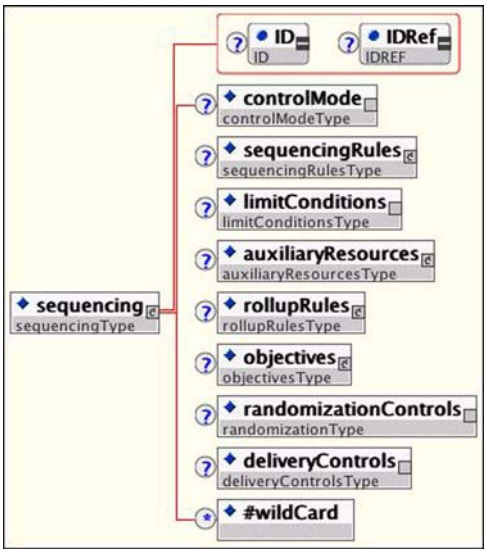

Figure 3.2 - <sequencing> element.

Multiplicity: Occurs once or more within the <sequencingCollection> element, if the <sequencingCollection> element is present. Occurs zero or once for each <item> or <organization> within an IMS content package.

Attributes:

- **ID (optional).** The unique identifier assigned to this sequencing set.
  Data type= ID (XML) string.
- **IDRef (optional).** Reference to the unique identifier assigned to this sequencing set. This is used to link this reference to the declared sequencing set that must be defined somewhere in the same XML document.
  Data type= IDREF (XML) string.

Elements:

- controlMode
- sequencingRules
- limitConditions
- auxiliaryResources
- rollupRules
- objectives

- randomizationControls
- deliveryControls
- #wildcard

Example:

```
<manifest identifier = "IMSSS.TestManifest.1">
   <organizations default = "IMSSS.TestManifest.1.Org.1">
      <organization identifier = "IMSSS.TestManifest.1.Org.1">
         <item identifier = "IMSSS.TestManifest.1.Item.1"/>
         <imsss:sequencing IDRef = "IMSSS.TestManifest.1.Seq.1">
            <imsss:limitConditions attemptLimit="1"/>
            <!-- This element has a limit of 1 attempt, and the begin time limit is
                 undefined, because the inline element overrides the referenced element.
                 However, this element has choice and choiceExit both false, because of the
                 referenced element. -->
      </organization>
   </organizations>
   <resources/>
   <imsss:sequencingCollection>
      <imsss:sequencing ID = "IMSSS.TestManifest.1.Seq.1">
         <imsss:controlMode choice="false" choiceExit="false"/>
         <imsss:limitConditions attemptLimit="2" beginTimeLimit="2002-10-16T12:00:00"/>
      </imsss:sequencing>
   </imsss:sequencingCollection>
</manifest>
```

3.3 <controlMode> Element

Description: This is the container for the sequencing control mode information including descriptions of the types of sequencing behaviors specified for an activity. Simple Sequencing processes may reference the sequencing control modes for any activity in the activity tree and the default values are used if the data is not defined for a given activity.

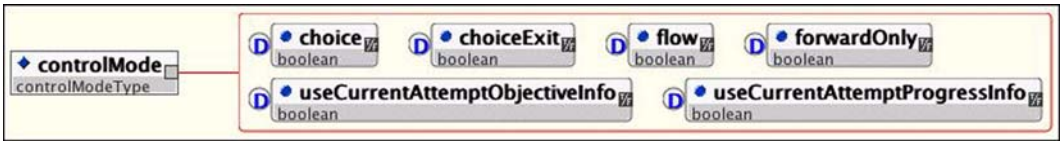

Figure 3.3 - <controlMode> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes:

- **choice (optional. Default = 'true').** Corresponds to SM.1:1 *Sequencing Control Choice.*
  Data type= Boolean.
- **choiceExit (optional. Default = 'true').** Corresponds to SM.1:2 *Sequencing Control Choice Exit.*
  Data type= Boolean
- **flow (optional. Default = 'false').** Corresponds to SM.1:3 *Sequencing Control Flow.*
  Data type= Boolean
- **forwardOnly (optional. Default = 'false').** Corresponds to SM.1:4 *Sequencing Control Forward Only.*
  Data type= Boolean
- **useCurrentAttemptObjectiveInfo (optional. Default = 'true').** Corresponds to SM.1:5 *Use Current Attempt Objective Information.*
  Data type= Boolean

- **useCurrentAttemptProgressInfo (optional. Default = 'true').** Corresponds to SM.1:6 *Use Current Attempt Progress Information.*
  Data type= Boolean

Elements: None.

3.4 <sequencingRules> Element

Description: The container for a sequencing rule description. Each rule describes the sequencing behavior for an activity. Each activity may have an unlimited number of sequencing rules and within any grouping the rules are evaluated in the order in which they are listed.

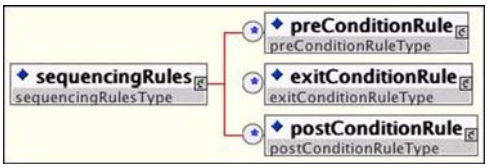

Figure 3.4 - <sequencingRules> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes: None.

Elements:

- preConditionRule
- exitConditionRule
- postConditionRule

3.4.1 <preConditionRule> Element

Description: The container for the description of actions that control sequencing decisions and delivery of a specific activity. Rules that include such actions are used to determine if the activity will be delivered.

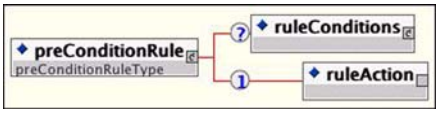

Figure 3.5 - <preConditionRule> element.

Multiplicity: Occurs zero or more times in the <sequencingRules> element.

Attributes: None.

Elements:

- ruleConditions
- ruleAction

Example:

The following example shows a precondition rule that causes the associated item to be disabled if it is either completed or satisfied:

```
<sequencing>
```

```
   <sequencingRules>
      <preConditionRule>
         <ruleConditions conditionCombination = "any">
            <ruleCondition condition = "completed"/>
            <ruleCondition condition = "satisfied"/>
         </ruleConditions>
         <ruleAction action = "disabled"/>
      </preConditionRule>
   </sequencingRules>
</sequencing>
```

3.4.1.1 <ruleAction> Element (within a <preConditionRule> element)

Description: The desired sequencing behavior if the precondition rule evaluates to ‘True’.

Figure 3.6 - <ruleAction> element.

Multiplicity: Occurs once and only once within a <preConditionRule> element.

Attributes:

- **action (required. An enumerated vocabulary of: ‘skip’, ‘disabled’, ‘hiddenFromChoice’, ‘stopForwardTraversal’).** Corresponds to SM.2:3 *Rule Action*.
  Data type= Token.

Elements: None.

3.4.2 <exitConditionRule> Element

Description: The container for the description of actions that terminate an activity. Rules that include such actions are applied when a descendent of an activity exits.

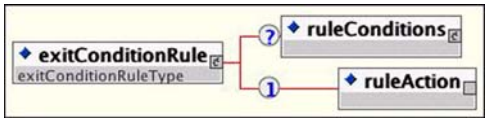

Figure 3.7 - <exitConditionRule> element.

Multiplicity: Occurs zero or more times in the <sequencingRules> element.

Attributes: None.

Elements:

- ruleConditions
- ruleAction

3.4.2.1 <ruleAction> Element (within a <exitConditionRule> element)

Description: The desired sequencing behavior if the exit rule evaluates to ‘True’.

Figure 3.8 - <ruleAction> element.

Multiplicity: Occurs once and only once within a <exitConditionRule> element.

Attributes:

- **action (required. An enumerated vocabulary of: 'exit').** Corresponds to SM.2:3 *Rule Action*.
  Data type= Token.

Elements: None.

3.4.3 <postConditionRule> Element

Description: The container for the description of actions that control sequencing flow by issuing sequencing requests. Rules that include such actions are applied when an activity exits.

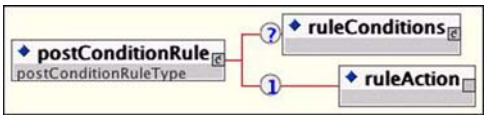

Figure 3.9 - <postConditionRule> element.

Multiplicity: Occurs zero or more times in the <sequencingRules> element.

Attributes: None.

Elements:

- ruleConditions
- ruleAction

3.4.3.1 <ruleAction> Element (within a <postConditionRule> element)

Description: The desired sequencing behavior if the postcondition rule evaluates to 'True'.

Figure 3.10 - <ruleAction> element.

Multiplicity: Occurs once and only once within a <postConditionRule> element.

Attributes:

- **action (required. An enumerated vocabulary of: 'exitParent', 'exitAll', 'retry', 'retryAll', 'continue', 'previous').** Corresponds to SM.2:3 *Rule Action*.
  Data type= Token.

Elements: None.

3.4.4 <ruleConditions> Element (for all rule types)

Description: The container for the set of conditions that are to be applied either the pre/post/exit condition rules.

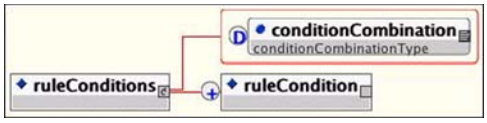

Figure 3.11 - <ruleConditions> element.

Multiplicity: Occurs zero or one times within the <preConditionRule>, <exitConditionRule>, and <postConditionRule> elements.

Attributes:

- **conditionCombination (optional. Non-enumerated vocabulary of: 'all', 'any'. Default = 'any').**
  Corresponds to SM.2:1 *Condition Combination*.
  Data type= Token.

Elements:

- ruleCondition

3.4.5 <ruleCondition> Element (for all rule types)

Description: Identification of a condition to be applied as part of the sequencing pre/post/exit rule.

Figure 3.12 - <ruleCondition> element.

Multiplicity: Occurs once or more times within the <ruleConditions> element.

Attributes:

- **referencedObjective (optional).** Corresponds to SM.2:2.2 *Rule Condition Referenced Objective*.
  Data type= anyURI.
- **measureThreshold (optional).** Corresponds to SM.2:2.3 *Rule Condition Measure Threshold*.
  Data type= Real number in the rage -1.0000 to 1.0000 with a precision of at least four decimal places and a default value of 0.000.
- **operator (optional. An enumerated vocabulary of: 'not', 'noOp'. Default = 'noOp').** Corresponds to SM.2:2.4 *Rule Condition Operator.*
  Data type= Token.
- **condition (required. An enumerated vocabulary of: 'satisfied', 'objectiveStatusKnown', 'objectiveMeasureKnown', 'objectiveMeasureGreaterThan', 'objectiveMeasureLessThan', 'completed', 'activityProgressKnown', 'attempted', 'attemptLimitExceeded', 'timeLimtExceeded', 'outsideAvailableTimeRange', 'always').** Corresponds to SM.2:2.1 *Rule Condition*.
  Data type= Token.

Elements: None.

3.5 <limitConditions> Element

Description: Defines the constraints on the access to an activity based on the time of day, time spent on the activity and number of attempts.

Figure 3.13 - <limitConditions> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes:

- **attemptLimit (optional).** The presence or absence of this attribute corresponds to SM.3:1 *Limit Condition Attempt Control*. If present, the value of this attribute corresponds to SM.3:2 *Limit Condition Attempt Limit*.
  Data type= nonNegativeInteger.
- **attemptAbsoluteDurationLimit (optional).** The presence or absence of this attribute corresponds to SM.3:3 *Limit Condition Attempt Absolute Duration Control*. If present, the value of this attribute corresponds to SM.3:4 *Limit Condition Attempt Absolute Duration Limit*.
  Data type= duration (accuracy = 0.1s).
- **attemptExperiencedDurationLimit (optional).** The presence or absence of this attribute corresponds to SM.3:5 *Limit Condition Attempt Experienced Duration Control*. If present, the value of this attribute corresponds to SM.3:6 *Limit Condition Attempt Experienced Duration Limit*.
  Data type= duration (accuracy = 0.1s).
- **activityAbsoluteDurationLimit (optional).** The presence or absence of this attribute corresponds to SM.3:7 *Limit Condition Activity Absolute Duration Control*. If present, the value of this attribute corresponds to SM.3:8 *Limit Condition Activity Absolute Duration Limit*.
  Data type= duration (accuracy = 0.1s).
- **activityExperiencedDurationLimit (optional).** The presence or absence of this attribute corresponds to SM.3:9 *Limit Condition Activity Experienced Duration Control*. If present, the value of this attribute corresponds to SM.3:10 *Limit Condition Activity Experienced Duration Limit*.
  Data type= duration (accuracy = 0.1s).
- **beginTimeLimit (optional).** The presence or absence of this attribute corresponds to SM.3:11 *Limit Condition Begin Time Limit Control*. If present, the value of this attribute corresponds to SM.3:12 *Limit Condition Begin Time Limit*.
  Data type= dateTime (accuracy = 0.1s).
- **endTimeLimit (optional).** The presence or absence of this attribute corresponds to SM.3:13 *Limit Condition End Time Limit Control*. If present, the value of this attribute corresponds to SM.3:14 *Limit Condition End Time Limit*.
  Data type= dateTime (accuracy = 0.1s).

Elements: None.

3.6 <auxiliaryResources> Element

Description: The container for the set of references to the set of auxiliary resources. When an activity is delivered to the learner the auxiliary resources are also made available. The order in which the auxiliary resources are described is not significant.

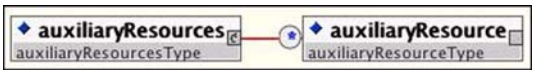

Figure 3.14 - <auxiliaryResources> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes: None.

Elements:

- auxiliaryResource

3.6.1 <auxiliaryResource> Element

Description: The container for the information describing a single auxiliary resource.

Figure 3.15 - <auxiliaryResource> element.

Multiplicity: Occurs zero or more times in the <auxiliaryResources> element.

Attributes:

- **auxiliaryResourceID (Required).** Corresponds to SM.4:1 *Resource ID*.
  Data type= anyURI.
- **purpose (Required).** Corresponds to SM.4:2 *Purpose*.
  Data type= String.

Elements: None.

3.7 <rollupRules> Element

Description: The container for the set of rollup rules that are to be applied to an activity. Each activity may have an unlimited number of rollup rules and the order in which they are defined is not significant.

Figure 3.16 - <rollupRules> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes:

- **rollupObjectiveSatisfied (optional. Default = 'true').** Corresponds to SM.8:1 *Rollup Objective Satisfied*.
  Data type= Boolean.
- **rollupProgressCompletion (optional. Default = 'true').** Corresponds to SM.8:3 *Rollup Progress Completion*.
  Data type= Boolean.

- **objectiveMeasureWeight (optional. Default = ‘1.0000’).** Corresponds to SM.8:2 *Rollup Objective Measure Weight*.
  Data type= Real number range 0.0000 to 1.0000 (precision to at least 4 significant decimal places).

Elements:

- rollupRule

3.7.1 <rollupRule> Element

Description: The container for each rollup rule that is to be applied to an activity. The general format for a rule can be expressed informally as ‘If child-activity set, condition set Then action’. Multiple conditions are permitted.

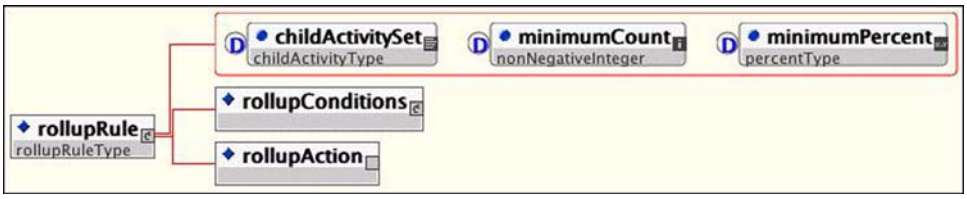

Figure 3.17 - <rollupRule> element.

Multiplicity: Occurs zero or more times in the <rollupRules> element.

Attributes:

- **childActivitySet (optional. Non-enumerated vocabulary of: ‘all’, ‘any’, ‘none’, ‘atLeastCount’, ‘atLeastPercent’. Default = ‘all’).** Corresponds to SM.5:1 *Rollup Child Activity Set*.
  Data type= String.
- **minimumCount (optional. Default = ‘0’).** Corresponds to SM.5:1.1 *Rollup Minimum Count*.
  Data type= nonNegativeInteger.
- **minimumPercent (optional. Default = ‘0.0000’).** Corresponds to SM.5:1.2 *Rollup Minimum Percent*.
  Data type= Real number range 0.0000 to 1.0000 (precision to at least 4 significant decimal places).

Elements:

- rollupConditions
- rollupAction

Example:

The following rollup rule, if attached to an item, states that the item is satisfied if at least three of its children are either satisfied or completed.

```
<sequencing>
   <rollupRules>
      <rollupRule childActivitySet = "atLeastCount" minimumCount = "3">
         <rollupConditions conditionCombination = "any">
            <rollupCondition condition = "satisfied"/>
            <rollupCondition condition = "completed"/>
         </rollupConditions>
         <rollupAction action = "satisfied"/>
      </rollupRule>
   </rollupRules>
</sequencing>
```

3.7.2 <rollupConditions> Element

Description: The container for the set of conditions that are applied within a single rollup rule.

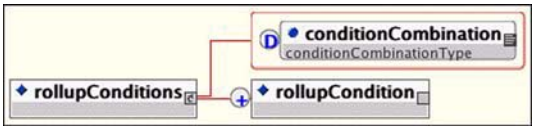

Figure 3.18 - <rollupConditions> element.

Multiplicity: Occurs once in the <rollupRule> element.

Attributes:

- **conditionCombination (optional. Non-enumerated vocabulary of: 'all', 'any'. Default = 'any').** Corresponds to SM.5:2 *Condition Combination*.
  Data type=Token.

Elements:

- rollupCondition

3.7.3 <rollupCondition> Element

Description: Identification of a condition to be applied in the rollup rule.

Figure 3.19 - <rollupCondition> element.

Multiplicity: Occurs once or more in the <rollupConditions> element.

Attributes:

- **operator (optional. Non-enumerated vocabulary of: 'not', 'noOp'. Default = 'noOp').** Corresponds to SM.5:3.2 *Rollup Condition Operator.*
  Data type=Token.
- **condition (required. Non-enumerated vocabulary of: 'satisfied', 'objectiveStatusKnown', 'objectiveMeasureKnown', 'completed', 'activityProgressKnown', 'attempted', 'attemptLimitExceeded', 'timeLimtExceeded', 'outsideAvailableTimeRange').** Corresponds to SM.5:3.1 *Rollup Condition*.
  Data type=Token.

Elements: None.

3.7.4 <rollupAction> Element

Description: The action that is to be undertaken if the rollup rule is declared as 'True'.

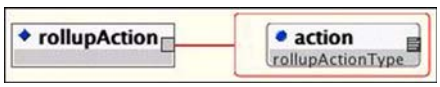

Figure 3.20 - <rollupAction> element.

Multiplicity: Occurs once in the <rollupRule> element.

Attributes:

- **action (required. Non-enumerated vocabulary of: ‘satisfied’, ‘notSatisfied’, ‘completed’, ‘incomplete’).** Corresponds to SM.5:4 *Rollup Action*.
  Data type=Token.

Elements: None.

3.8 <objectives> Element

Description: The container for the set of objectives that are to be associated with an activity. Each activity must have at least one primary objective and may have an unlimited number of objectives.

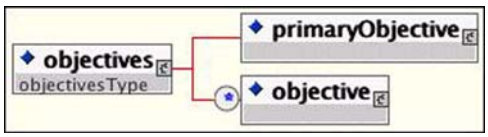

Figure 3.21 - <objectives> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes: None.

Elements:

- primaryObjective
- objective

3.8.1 <primaryObjective> Element

Description: Identifies the objective that contributes to the rollup associated with the activity.

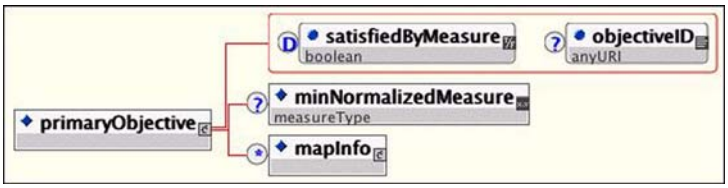

Figure 3.22 - <primaryObjective> element.

Multiplicity: Occurs once in the <objectives> element.

Attributes:

- **satisfiedByMeasure (optional. Default = ‘false’).** Corresponds to SM.6:2 *Objective Satisfied by Measure*.
  Data type= Boolean.
- **objectiveID (optional).** Corresponds to SM.6:1 *Objective ID*. If Objective Maps are defined for this objective, this also corresponds to SM.7:1 *Activity Objective ID*.
  Data type= anyURI.

Elements:

- minNormalizedMeasure
- mapInfo

3.8.2 <objective> Element

Description: Identifies objectives that do not contribute to rollup associated with the activity.

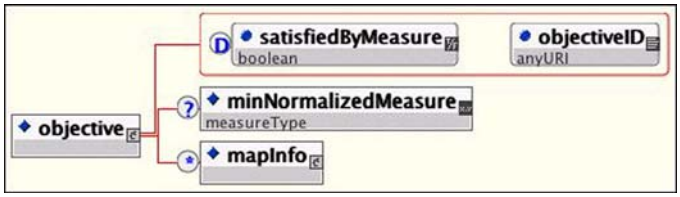

Figure 3.23 - <objective> element.

Multiplicity: Occurs zero or once in the <objectives> element.

Attributes:

- **satisfiedByMeasure (optional. Default = 'false').** Corresponds to SM.6:2 *Objective Satisfied by Measure*. Data type= Boolean.
- **objectiveID (required).** Corresponds to SM.6:1 *Objective ID*. If Objective Maps are defined for this objective, this also corresponds to SM.7:1 *Activity Objective ID*.
  Data type= anyURI.

Elements:

- minNormalizedMeasure
- mapInfo

3.8.3 <minNormalizedMeasure> Element

Description: Defines the minimum satisfaction measure for the objective normalized between -1 and 1. If the objective value of 'Objective Normalized Measure' exceeds this value, the objective value of 'Objective Data Status' is set to 'true' and the objective value of 'Objective Satisfied Status' is set to 'true'. It has a real number in the range -1.0000 to 1.000 with a precision of at least four decimal places and has a default value of 1.0000.

Multiplicity: Occurs zero or once on the <primaryObjective> and <objective> elements.

Attributes: None.

Elements: None.

3.8.4 <mapInfo> Element

Description: The container for the objective map description. This defines the mapping of an activity's local objective information to and from a shared global objective. Each activity may have an unlimited number of objective maps.

Note that the information model defines constraints on how objectives may be mapped to and from global objectives, to assure that there is never an indeterminate state on any objective. While the XSD document does not enforce these constraints, they are still normative on all binding instances.

Figure 3.24 - <mapInfo> element.

Multiplicity: Occurs zero, once, or an unlimited number of times on the <primaryObjective> and <objective> elements.

Attributes:

- **targetObjectiveID (required).** Corresponds to SM.7:2 *Target Objective ID*.
  Data type= anyURI.
- **readSatisfiedStatus (optonal. Default = 'true').** Corresponds to SM.7:3 *Read Objective Satisfied Status*.
  Data type= Boolean.
- **readNormalizedMeasure (optonal. Default = 'true').** Corresponds to SM.7:5 *Read Objective Normalized Measure*.
  Data type= Boolean.
- **writeSatisfiedStatus (optonal. Default = 'false').** Corresponds to SM.7:4 *Write Objective Satisfied Status*.
  Data type= Boolean.
- **writeNormalizedMeasure (optonal. Default = 'false').** Corresponds to SM.7:6 *Write Objective Normalized Measure*.
  Data type= Boolean.

Elements: None.

3.9 <randomizationControls> Element

Description: The container for the descriptions of how children of an activity should be ordered during the sequence process. Simple Sequencing processes may reference the randomization control data for any activity in the activity tree and when this is absent the default values should be used.

Figure 3.25 - <randomizationControls> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes:

- **randomizationTiming (optional. Non-enumerated vocabulary of: 'never', 'once', 'onEachNewAttempt'. Default = 'never').** Corresponds to SM.10:1 *Randomization Timing*.
  Data type= Token.
- **selectCount (optional).** The presence or absence of this attribute corresponds to SM.9:2 *Selection Count Status*. If present, the value of this attribute corresponds to SM.9:3 *Selection Count*.
  Data type= nonNegativeInteger.
- **reorderChildren (optional. Default = 'false').** Corresponds to SM.10:2 *Randomize Children*.
  Data type= Boolean.
- **selectionTiming (optional. Non-enumerated vocabulary of: 'never', 'once', 'onEachNewAttempt'. Default = 'never').** Corresponds to SM.9:1 *Selection Timing*.
  Data type= Token.

Elements: None.

3.10 <deliveryControls> Element

Description: This contains the description of the actions and controls used when an activity is delivered. Simple Sequencing processes may reference the delivery control data for any activity in the activity tree. The delivery controls are optional and if these are absent then the default values must be used.

Figure 3.26 - <deliveryControls> element.

Multiplicity: Occurs zero or once in the <sequencing> element.

Attributes:

- **tracked (optional. Default = 'true').** Corresponds to SM.11:1 *Tracked*.
  Data type= Boolean.
- **completionSetByContent (optional. Default = 'false').** Corresponds to SM.11:2 *Completion Set by Content*.
  Data type= Boolean.
- **objectiveSetByContent (optional. Default = 'false').** Corresponds to SM.11:3 *Objective Set by Content*.
  Data type= Boolean.

Elements: None.

3.11 #wildCard

Description: This is the Simple Sequencing Binding extension facility.

Multiplicity: Occurs zero or more times in the <sequencing> element.

Attributes: None.

Elements: None.

4. Sequencing Definition Element Mapping

This section contains several tables. Table 4.1 maps a Sequencing Definition element by its number and name as listed in the Simple Sequencing Information Model specification to the XML Schema entity used to represent it in the XML Schema binding. The Schema entity's name, kind of structure, and type are provided.

The Sequencing Definition and Tracking Definition Models identify data elements in a dot-delimited enumeration and by a name. The dot-delimited enumeration typifies the relationship of elements to each other, where elements with a numeral following a dot are subordinate to elements who share the same numeric value to the left of a dot. The enumeration within a table is separated from the enumeration of the table by a colon character; thus, element 2.1 in table SM.2 is identified as "SM.2:2.1". The enumeration typically does **not** imply a strict sequence or order of occurrence. It merely implies a group relationship.

In some cases, two Information Model elements map to the same XML Schema entity. Unless otherwise noted, the following rules apply:

- If one of the IM elements is of type boolean, and the other is not, the boolean IM element maps to the presence or absence of the XML Schema entity, and the other IM element maps to the contents of the XML Schema entity. In such cases, the boolean type element will be marked with an asterisk.
- If both of the IM elements are of type complexType, then the XML Schema entity represents the union of the two IM elements.

Table 4.1 also includes references to additional XML Schema structures used to properly express the relationships or controlled lists of Sequencing Definition element values. These additional XML Schema structures are inserted into Table 4.1 in proximity to the Sequencing Definition elements they describe.

XSD elements of type *complexType* and *simpleType* may be nested. Table 4.1 does not show this nesting. Separate tables in this section provide the further details for nested elements defined as *complexType* or *simpleType*. In those tables, the element number from the Sequencing Definition or Tracking Definition model and the XML Schema element name are used to associate a nested Schema structuring entity with the appropriate element from the Sequencing Definition model.

Table 4.1 - Sequencing Definition elements mapped to XML Schema (XSD) structures.

No.	SD/TD Element Name	XSD Entity Name	XSD Structure	XSD Type
		sequencingCollection	element	complexType
SM.12	Sequencing Description	sequencing	element	complexType
SM.12	Sequencing Description	sequencingRef	element	complexType
SM.1	Sequencing Control Modes	controlMode	element	complexType
:1	Sequencing Control Choice	choice	attribute	boolean
:2	Sequencing Control Choice Exit	choiceExit	attribute	boolean
:3	Sequencing Control Flow	flow	attribute	boolean
:4	Sequencing Control Forward Only	forwardOnly	attribute	boolean
:5	Use Current Attempt Objective Progress Information	useCurrentAttemptObjectiveInfo	attribute	boolean
:6	Use Current Attempt Progress Information	useCurrentAttemptProgressInfo	attribute	boolean
		sequencingRules	element	complexType
SM.2	Sequencing Rule Description	preConditionRule postConditionRule exitConditionRule	element	complexType
:1	Condition Combination	conditionCombination	element	complexType
:2	Rule Conditions	ruleConditions	element	complexType

No.	SD/TD Element Name	XSD Entity Name	XSD Structure	XSD Type
		ruleCondition	element	complexType
:2.1	Rule Condition	condition	attribute	simpleType
:2.2	Rule Condition Referenced Objective	referencedObjective	attribute	string
:2.3	Rule Condition Measure Threshold	measureThreshold	attribute	simpleType
:2.4	Rule Condition Operator	operator	attribute	simpleType
		ruleAction	element	complexType
:3	Rule Action	action	attribute	simpleType
SM.3	Limit Conditions Description	limitConditions	element	complexType
:1	Limit Condition Attempt Control	attemptLimit	element	boolean *
:2	Limit Condition Attempt Limit	attemptLimit	element	nonNegativeInteger
:3	Limit Condition Attempt Absolute Duration Control	attemptAbsoluteDurationLimit	element	boolean *
:4	Limit Condition Attempt Absolute Duration Limit	attemptAbsoluteDurationLimit	element	duration
:5	Limit Condition Attempt Experienced Duration Control	attemptExperiencedDurationLimit	element	boolean *
:6	Limit Condition Attempt Experienced Duration Limit	attemptExperiencedDurationLimit	element	duration
:7	Limit Condition Activity Absolute Duration Control	activityAbsoluteDurationLimit	element	boolean *
:8	Limit Condition Activity Absolute Duration Limit	activityAbsoluteDurationLimit	element	duration
:9	Limit Condition Activity Experienced Duration Control	activityExperiencedDurationLimit	element	boolean *
:10	Limit Condition Activity Experienced Duration Limit	activityExperiencedDurationLimit	element	duration
:11	Limit Condition Begin Time Limit Control	beginTimeLimit	element	boolean *
:12	Limit Condition Begin Time Limit	beginTimeLimit	element	dateTime
:13	Limit Condition End Time Limit Control	endTimeLimit	element	boolean *
:14	Limit Condition End Time Limit Limit	endTimeLimit	element	dateTime
		auxiliaryResources	element	complexType
SM.4	Auxiliary Resource Description	auxiliaryResource	element	complexType
:1	Resource ID	auxiliaryResourceID	attribute	string
:2	Purpose	purpose	attribute	string
		rollupRules	element	complexType
SM.5	Rollup Rule Description	rollupRule	element	complexType
:1	Rollup Child Activity Set	childActivitySet	attribute	simpleType
:1.1	Rollup Minimum Count	minimumCount	attribute	integer
:1.2	Rollup Minimum Percent	minimumPercent	attribute	decimal
:2	Condition Combination	conditionCombination	attribute	simpleType
:3	Rollup Conditions	rollupConditions	element	complexType
		rollupCondition	element	complexType
:3.1	Rollup Condition	condition	attribute	simpleType
:3.2	Rollup Condition Operator	operator	attribute	simpleType

No.	SD/TD Element Name	XSD Entity Name	XSD Structure	XSD Type
		rollupAction	element	complexType
:4	Rollup Action	action	attribute	simpleType
		objectives	element	complexType
SM.6	Objective Description	primaryObjective objective	element	complexType
:1	Objective ID	objectiveID	attribute	string
:2	Objective Satisfied By Measure	satisfiedByMeasure	attribute	boolean
:3	Objective Minimum Satisfied Normalized Measure	minNormalizedMeasure	element	simpleType
:4	Objective Contributes To Rollup	primaryObjective	element	boolean *
		objectives	element	complexType
SM.7	Objective Map	mapInfo	element	complexType
:1	Activity Objective ID	objectiveID	attribute	string
:2	Target Objective ID	targetObjectiveID	attribute	string
:3	Read Objective Satisfied Status	readSatisfiedStatus	attribute	boolean
:4	Write Objective Satisfied Status	writeSatisfiedStatus	attribute	boolean
:5	Read Objective Normalized Measure	readNormalizedMeasure	attribute	boolean
:6	Write Objective Normalized Measure	writeNormalizedMeasure	attribute	boolean
SM.8	Rollup Controls	rollupRules	element	complexType
:1	Rollup Objective Satisfied	rollupObjectiveSatisfied	attribute	boolean
:2	Rollup Objective Measure Weight	objectiveMeasureWeight	attribute	simpleType
:3	Rollup Progress Completion	rollupProgressCompletion	attribute	boolean
SM.9	Selection Controls	randomizationControls	element	complexType
:1	Selection Timing	selectionTiming	attribute	simpleType
:2	Selection Count Status	selectCount	attribute	boolean *
:3	Selection Count	selectCount	attribute	nonNegativeInteger
SM.10	Randomization Controls	randomizationControls	element	complexType
:1	Randomization Timing	randomizationTiming	attribute	simpleType
:2	Randomize Children	reorderChildren	attribute	boolean
SM.11	Delivery Controls	deliveryControls	element	complexType
:1	Tracked	tracked	attribute	boolean
:2	Completion Set By Content	completionSetByContent	attribute	boolean
:3	Objective Set By Content	objectiveSetByContent	attribute	boolean

4.1 Mapping Sequencing Definition Elements to XSD Entities of simpleType

Table 4.2 - Simple Types with Restricted Enumerations.

No.	XSD Name	XSD Enumeration Value	XSD Data Type	Sequencing Definition Model Values
SM.2:1	conditionCombination	all	token	All
		any	token	Any
SM.2:2.1	condition	satisfied	token	Satisfied
		objectiveStatusKnown	token	Objective Status Known

No.	XSD Name	XSD Enumeration Value	XSD Data Type	Sequencing Definition Model Values
		objectiveMeasureKnown	token	Objective Measure Known
		objectiveMeasureGreaterThan	token	Objective Measure Greater Than
		objectiveMeasureLessThan	token	Objective Measure Less Than
		completed	token	Completed
		activityProgressKnown	token	Activity Progress Known
		attempted	token	Attempted
		attemptLimitExceeded	token	Attempt Limit Exceeded
		timeLimitExceeded	token	Time Limit Exceeded
		outsideAvailableTimeRange	token	Outside Available Time Range
		always	token	Always
SM.2:2.4	operator	not	token	Not
		noOp	token	NO-OP
SM.2:3	action	skip	token	Skip
		disabled	token	Disabled
		hiddenFromChoice	token	Hidden from Choice
		stopForwardTraversal	token	Stop Forward Traversal
		exitParent	token	Exit Parent
		exitAll	token	Exit All
		retry	token	Retry
		retryAll	token	Retry All
		continue	token	Continue
		previous	token	Previous
		exit	token	Exit
SM.5:1	childActivitySet	all	token	All
		any	token	Any
		none	token	None
		atLeastCount	token	At Least Count
		atLeastPercent	token	At Least Percent
SM.5:2	conditionCombination	all	token	All
		any	token	Any
SM.5:3.1	condition	satisfied	token	Satisfied
		objectiveStatusKnown	token	Objective Status Known
		objectiveMeasureKnown	token	Objective Measure Known
		completed	token	Completed
		activityProgressKnown	token	Activity Progress Known
		attempted	token	Attempted
		attemptLimitExceeded	token	Attempt Limit Exceeded
		timeLimitExceeded	token	Time Limit Exceeded
		outsideAvailableTimeRange	token	Outside Available Time Range
SM.5:3.2	operator	not	token	Not
		noOp	token	NO-OP
SM.5:4	action	satisfied	token	Satisfied
		notSatisfied	token	Not Satisfied
		completed	token	Completed
		incomplete	token	Incomplete
SM.9:1	selectionTiming	never	token	Never

No.	XSD Name	XSD Enumeration Value	XSD Data Type	Sequencing Definition Model Values
		once	token	Once
		onEachNewAttempt	token	On Each New Attempt
SM.10:1	randomTiming	never	token	Never
		once	token	Once
		onEachNewAttempt	token	On Each New Attempt

Table 4.3 - Simple Types with Restricted Numerical Types.

No.	XSD Name	XSD Base Type	MinInclusive	MaxInclusive
SM.2:2.3	measureThreshold	decimal	-1	1
SM.5:1.2	minimumPercent	decimal	0	1
SM.6:3	minNormalizedMeasure	decimal	-1	1
SM.8:2	objectiveMeasureWeight	decimal	0	1

5. Physical Realization of the XML Binding

The XML Binding has been released as a set of XSDs. The binding is implemented as a series of files which are linked using the XML XSD 'include' statement; the hierarchical relationship is shown in Figure 5.1. The XSD files created to realize the binding are:

- The bindings that are defined for each of the core data structures are:
  - 'imsss_v1p0.xsd' – the SS root XML structure
  - 'imsss_v1p0auxresource.xsd' – the auxiliary resources core data structure
  - 'imsss_v1p0control.xsd' – the 'controlMode' core data structure
  - 'imsss_v1p0delivery.xsd' – the 'deliveryControls' core data structure
  - 'imsss_v1p0limit.xsd' – the 'limitConditions' core data structure
  - 'imsss_v1p0objective.xsd' – the 'objectives' core data structure
  - 'imsss_v1p0random.xsd' – the 'randomizationControls' core data structure
  - 'imsss_v1p0rollup.xsd' – the rollup rules core data structure
  - 'imsss_v1p0seqrule.xsd' – the sequencing rules core data structure
  - 'imsss_v1p0util.xsd' – the set of utility declarations used by other XSD files

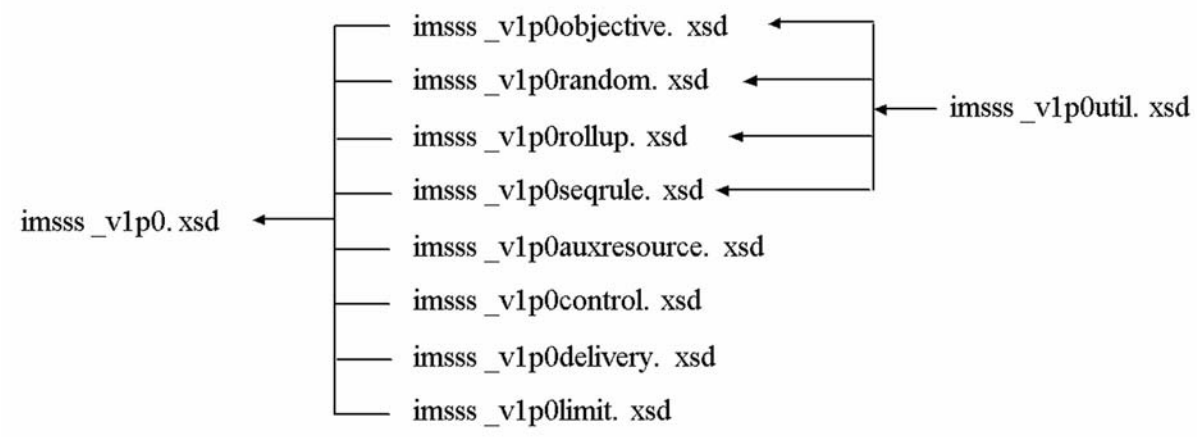

Figure 5.1 - The hierarchical relationship of the SS XSD files.

6. Normative Points

6.1 XML Schema File Naming Convention

Conformant with IMS file naming conventions, an IMS Simple Sequencing XML Schema binding instance shall be named according to this syntactical model: imsss_v*mpn*[p*r*[p*s*]].xsd, where

- "imsss_" signifies that the file pertains to the IMS Simple Sequencing (IMS SS) Specification,
- "vm" is a major version of the IMS SS Specification defined by a binding instance, where "m" is the major version number, and is always followed by
- "pn", which signifies a minor version of the IMS SS Specification defined by a binding instance, where "n" is minor version number, and where
- [pr[ps]] is an optional set of major release and minor release numbers (as indicated by the brackets which are not part of the file name), depending on changes to schema instances themselves that are not driven by a change to a normative instance of the IMS SS Specification, with
- "pr" representing a major release component without an additional minor release part, where "r" is the major release number, and with
- "ps" representing a minor release component modifying an major release component, where "s" is the minor release number.

Non-normative examples of XML Schema file names for IMS SS:

- "imsss_v1p0.xsd" is a binding tied to IMS Simple Sequencing v1.0;
- "imsss_v1p0p1.xsd" is a 'major' revision of the imsss_v1p0.xsd binding instance;
- "imsss_v1p0p1p1.xsd" is 'minor' revision of imsss_v1p0p1.xsd binding instance.

6.2 Namespace URI (Namespace Identifier)

The namespace URI for IMS Simple Sequencing shall be: http://www.imsglobal.org/xsd/imsss/

6.2.1 Location of Current IMS Sequencing Binding Instance

The .xsd file found at http://www.imsglobal.org/xsd/imsss/ shall be the current XML Schema binding instance of the IMS Simple Sequencing Information Model normative elements and values. This file will change.

6.2.2 Location of Versioned IMS Sequencing Binding Instances

Versioned instances of each binding for IMS Simple Sequencing, including all major or minor release instances of that version's binding, can be found at http://www.imsglobal.org/xsd/imsss/vmpn/, where "*vmpn*" is a folder named for a versioned instance of the specification. These instances will not change.

For example, the non-normative examples of XML Schema files named for various binding instances of IMS Simple Sequencing v1.0 would be held at this location: http://www.imsglobal.org/xsd/imsss/v1p0/ .

6.3 Namespace Name

The namespace Name used as the prefix of an XML Namespace Qualified Name instance of an IMS Simple Sequencing binding instance used in other XML documents shall be *imsss*.

For example, when declaring the IMS SS namespace URI in an IMS manifest, this attribute representation would be used:

```
xmlns:imsss="http://www.imsglobal.org/xsd/imsss"
```

An element of the IMS Simple Sequencing declared in an XML document where the imsss namespace is declared would appear like this:

```
<imsss:sequencing></imsss:sequencing>
```

6.4 Versioning a Binding Instance

A binding version of the IMS Simple Sequencing Specification shall always reflect the current version number of the IMS Simple Sequencing specification, whether or not binding structures change across specification instances. That is, the file name of the current binding instance shall share the same major and minor *version* number of the IMS Simple Sequencing specification.

A change to binding structures without a related change to a specification version shall result in an increment of the major or minor revision level component of a binding's file name.

Changes to binding structures without a related change in a specification shall result in an increment of the *major revision* level.

Changes to structure names or values without changes in structures themselves or the meaning associated with a value shall result in an increment of the *minor revision* level.

All versioning and revisioning shall be reflected in a binding's file name and complete version information. This new binding shall become the current binding found at http://www.imsglobal.org/xsd/imsss/. The obsolete binding shall be moved to its place within the imsss file path structure based on the value of its version level component, excluding the revision level component.

6.5 Localization

6.5.1 Localizing Binding Structure Names and Values

An IMS Simple Sequencing binding's structure names and values shall be represented as depicted in the Tables within Section 3 of this document. This will ensure machine-level interoperability of sequencing information.

US English annotations and comment strings within an IMS binding shall occur in all binding instances of an IMS Simple Sequencing binding. Localized annotations and comments may be *added* to the US English annotations and comment strings within an IMS binding as deemed necessary by IMS stakeholders. Such localization activity in a binding shall promote a binding instance to the next increment in *minor* revision level.

6.5.2 Localizing this Document

This binding document may be rendered into different lingual representations except for the US English representations of:

- Binding structure names;
- Binding structure values;
- File naming syntax and examples;
- Binding namespace URI;
- Binding file path representation;
- Binding namespace name.

Localization of this document shall *not* cause any change in specification version numbering or binding revision instance numbering.

6.6 Extensibility

The IMS Simple Sequencing Binding provides one and only one mechanism for extending binding instances: the “wildcard” element contained in the top level of the “sequencing” element. All other extensions to binding instances are prohibited. This means:

- The IMS Simple Sequencing XML Schema Definition File shall not be modified;
- Elements within the IMS Simple Sequencing namespace shall not be modified;
- All elements in an instance document that are defined to be within the IMS Simple Sequencing namespace shall conform exactly to the definition of that element provided in this binding description document;
- Elements in another namespace shall not be substituted for elements in the IMS Simple Sequencing namespace (via the XML Schema substitution group mechanism, for example, or via any other mechanism);
- The vocabularies defined for elements and attributes within the IMS Simple Sequencing namespace shall not be extended or restricted within the IMS Simple Sequencing namespace.

The IMS Simple Sequencing XML Schema Definition file enforces these restrictions by defining block=”#all” on all elements in the IMS Simple Sequencing namespace.

Further, extensions to the IMS Simple Sequencing namespace are prohibited. This means:

- All extensions shall be in a namespace other than the IMS Simple Sequencing namespace.

The IMS Simple Sequencing XML Schema Definition file enforces this restriction by defining namespace=”##other” on the wildcard element defined within the sequencing element in the IMS Simple Sequencing namespace.

As a potential aid to implementers, the types defined in the IMS Simple Sequencing XML Schema Definition file may be imported into other namespaces and reused within those other namespaces. However, any such reuse shall be consistent with the restrictions defined in the IMS Simple Sequencing Specification document.

About this Document

Title	IMS Simple Sequencing XML Binding
Editors	Mark Norton (IMS), Brendon Towle (Thomson NETg), Colin Smythe (IMS)
Team Co-Leads	Ron Ball (ADL), Brandt Dargue (Boeing)
Version	1.0
Version Date	03 March 2003
Status	**Final Specification**
Summary	This document describes the XML binding definition model for the IMS Simple Sequencing Specification.
Purpose	Defines the IMS Simple Sequencing XML Binding.
Document Location	http://www.imsglobal.org/simplesequencing/ssv1p0/imsss_bindv1p0.html

List of Contributors

The following individuals contributed to the development of this document:

Name	Organization
Thor Anderson	IMS Global Learning Consortium, Inc.
Bill Blackmon	Carnegie Mellon University
Fabrizio Cardinali	Giunti Interactive Labs
Travis Carlton	Boeing
Thomas Dinger	IBM
Philip Dodds	Advanced Distributed Learning (ADL)
Leonard Greenberg	Pathlore
Michael Korcuska	DigitalThink
Jon McCarty	Docent
Boyd Nielsen	Thomson NETg
Land Ormiston	Saba
Claude Ostyn	Click2learn
Angelo Panar	Advanced Distributed Learning (ADL)
Dan Rehak	Carnegie Mellon University
GianLuca Rolandelli	Giunti Interactive Labs
Brian Taliesin	Microsoft
Schawn Thropp	Advanced Distributed Learning (ADL)
Robert Todd	DigitalThink
Ed Walker	IMS Global Learning Consortium, Inc.
Michael Vax	WebCT
Jonathan Zempel	IBM

Revision History

Version No.	Release Date	Comments
0.7.5 Public Draft	12 April 2002	First Public Draft released to the public.
1.0 Public Draft	17 October 2002	Numerous editorial updates and additional clarifications. Updated pseudo code. Added introduction section and several diagrams.
1.0 Final	03 March 2003	Made many changes to resolve various technical issues. See the Issues List document for a detailed description of the changes. Added Normative section.

Index