SmarterApp Assessment Item Format Specification

V 0.80: PROJECT DRAFT

Daniel Rehak
2014-08-12

This is a draft of the SmarterApp *Assessment Item Format Specification* (SAAIF). The document has not been finalized as a Smarter Balanced Assessment Consortium interoperability specification and is subject to revision.

Produced by:
Smarter Balanced Assessment Consortium
SBAC
Contact address
Phone +1 000-000-0000
Fax +1 000-000-0000
http://www.smarterapp.org/

Once completed the Specification will be available on the World Wide Web at:
http://www.smarterapp.org/specifications.html

The contents of the Specification were developed under a grant from the U.S. Department of Education. However, its contents do not necessarily represent the policy of the U.S. Department of Education and the reader should not assume endorsement by the Federal government.

Project Draft Specification

Note: The draft specification status below to be removed upon SmarterApp publication.

The Specification is an unapproved project draft Smarter Balanced Assessment Consortium interoperability specification. The Specification is subject to change. Use at your own risk! The unapproved project draft Specification must not be used for any Smarter Balanced Assessment Consortium acquisition, conformance or compliance processes.

Specification Maintenance

The Specification is maintained and updated by the Smarter Balanced Assessment Consortium. The Specification may be superseded by new versions, new editions or may be amended through published errata.

The official Specification consists of the most recent version or edition along with all published amendments and errata. The Specification is available on the World Wide Web at:
http://www.smarterapp.org/specifications.html
Users are encouraged to check this URL for the most recent version of the Specification.

Requests for revision of the Specification are welcome from any interested party, regardless of membership affiliation with the Smarter Balanced Assessment Consortium. Suggestions for revision should be in the form of a proposed change to the text, together with appropriate supporting rationale. Requests for revision to the Specification should be submitted to the following address:
http://www.smarterapp.org/specifications.html

Use

Use of the Specification by third parties is wholly voluntary. The Smarter Balanced Assessment Consortium disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon the Specification.

Any person using the Specification should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of the Specification.

Users of the Specification should consult all applicable laws and regulations. Compliance with the provisions of the Specification does not imply compliance to any applicable regulatory requirements. Implementers of the Specification are responsible for observing or referring to the applicable regulatory requirements.

Patents

Implementation of the Specification may require use of subject matter covered by patent rights. The Smarter Balanced Assessment Consortium takes no position with respect to the existence or validity of any patent rights connected to the Specification. Users of the Specification are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility.

Copyright

Trademarks

License

Disclaimers

Contents

List of Tables

List of Figures

Code Listings

This page intentionally left blank

Introduction

Note: This section is informative.

This document (the *Specification*) defines an XML document structure for the encoding and representation of assessment items – the SmarterApp *Assessment Item Format Specification* (SAAIF).

The XML representation of an assessment item contained in the Specification may be used to publish assessment items for purposes such as exchange between producers such as item authoring tools and consumers such as item banks and test delivery engines. The Specification does not limit how the XML document structure and elements may be used.

The XML document structure is defined in the narrative of the Specification. The specified XML document structure may be defined, in part, in XML through XML schemata [XSD 1], XML DTDs [XML] or in descriptions encoded in other XML modeling languages. Additional requirements specified in the narrative cannot be modeled in XML Schemata or XML DTDs.

The Specification is based on the AIR Item Representation Format and is used by the Smarter Balanced Assessment Consortium (SBAC) to represent the SBAC assessment items.

The main audience for the Specification is developers who are producing tools and systems to create, process or consume XML documents that conform to the Specification. The Specification is not targeted at users such as item developers or assessment administrators. The Specification does not include guidance on how to use design, model or create assessment items that are encoded in the XML document format specified herein.

The Specification only describes the structure of the XML documents for assessment items. It does not address how to store or exchange these documents. The Specification does not describe how to produce, transform, process or consume the documents except for describing conforming documents that a conforming processor produces or consumes.

The Specification includes:

- **Notation** – Definitions of normative terms and conventions used in the Specification.
- **Informal Model** – The model for the digital representation of an assessment item and related items (informative).
- **XML Document Elements** – The XML element definitions for assessment item documents:
    - *Assessment Item Release XML Elements* – the XML elements defining the *release* of an assessment item XML document.
    - *Assessment Item XML Elements* – the XML elements defining an assessment item XML document. An assessment item XML document can be embedded in an assessment item release document.
    - *Passage Item XML Elements* – the XML elements defining a passage item XML document. A passage item XML document can be embedded in an assessment item release document.
    - *Tutorial XML Elements* – the XML elements defining a tutorial used in an assessment item. A tutorial XML document can be embedded in an assessment item release document.
    - *Wordlist XML Elements* – the XML elements defining a *wordlist* resource (glossary or thesaurus) used in an assessment item. A wordlist XML document can be embedded in an assessment item release document.

  - *Assessment Item Accessibility XML Elements* – the XML elements defining an item accessibility XML document. An assessment item accessibility document is embedded in an assessment item (including a tutorial) or passage item document.
  - *Grid Item Rendering Specification XML Elements* – the XML elements defining a grid interaction item rendering specification XML document. These elements are normally embedded in a grid interaction assessment item.
  - *Equation Editor Configuration XML Elements* – the XML elements defining an equation editor configuration XML document. An equation editor configuration XML document is normally referenced from an equation assessment item.
  - *Assessment Item Usage Statistics XML Elements* – the XML elements holding statistical data about assessment item usage. These elements are normally embedded in an assessment item. The details of these elements are not documented in the Specification.
  - *Assessment Item Machine Rubric XML Elements* – the XML elements defining a machine rubric XML document. The machine rubric XML document is normally referenced from an assessment item. The details of these elements are not documented in the Specification.
- **XML Document Criteria** – General characteristics for all XML documents for assessment items.
  - *Document Criteria* – descriptions of document element content that cannot be specified at the XML element level.
  - *Semantic Constraints* – constraints on the XML documents that cannot be specified at the XML element level.
  - *Specification Versioning* – criteria for identifying the specific version of the Specification in XML documents describing an assessment item.
  - *IANA Considerations* – recommendations for Internet media type names for XML documents conforming to the Specification.
  - *Implementation Considerations* – best practices on how to represent or use XML documents describing an assessment item.
- **Conformance** – Criteria for an XML document and an XML document processor to conform to the Specification.
- **Security Considerations** – Security considerations for creating, producing or consuming XML documents for assessment items that conform to the Specification.
- **Normative References** – Normative references to other specifications used in the Specification.
- **Definitions** – Definitions of terms used in the Specification.
- **Acronyms** – Acronyms used in the Specification.
- **Informative References** – References to other documents used in the Specification (informative).
- **Annex: Examples** – Sample assessment items represented as XML documents that conform to the Specification (informative).
- **Annex: Standard Equation Editor Input Key Panel Configurations** – Description of the different equation editor input key configurations (informative).
- **Annex: XML Representation Design Decisions** – Choices and rationale in designing the XML document structure and sample schemata design (informative).
- **Annex: XML Schemata** – Description of XML Schemata (XSDs) that conform to the Specification (informative).
- **Annex: XML DTDs** – Description of XML DTDs that conform to the Specification (informative).

Notation

Keywords

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in the Specification are to be interpreted as described in [RFC 2119].

The key word “IS DEPRECATED” in the Specification designates a feature that MAY be removed in a future version of the Specification. The feature is maintained for backward compatibility. The feature SHOULD NOT be used when creating new assessment items. All conforming processors MUST support the feature.

The key word “TO BE DEPRECATED” in the Specification designates a feature that MAY be removed in a future version of the Specification. The feature is maintained for backward compatibility. If REQUIRED, the feature SHALL be used when creating new assessment items. All conforming processors MUST support the feature.

The key word “NOT SPECIFIED” in the Specification designates a behavior that is not described. Different implementations (typically consumers) MAY behave differently. Interoperability between different implementations is not insured.

The key word “IMPLEMENTATION DEFINED” in the Specification designates a behavior that SHALL be defined by a test client. The implementation MAY be test client specific.

The key word “IMPLEMENTATION DEPENDENT” in the Specification designates a behavior MAY not be specified in any implementation.

The key word “NON CONFORMING” in the Specification designates XML structure or data that violates the Specification. An implementation SHOULD indicate an error when non-conforming XML or data is found. A *conforming* implementation SHALL indicate an error when non-conforming data is found. Conformance criteria for a conforming implementation are detailed in the *XML Document Conformance* section of the Specification.

A key word that appears in normal font (e.g., may, shall) are to be interpreted in their normal sense.

Normative Text

Unless otherwise noted, all sections in the Specification are normative.

Within a normative section, all notes and illustrations are informative unless indicated otherwise.

Presentation of Elements

The Specification describes XML elements and XML documents in both tabular and graphical form.

The Specification uses the tabular structure shown in Table 1, Table 2 and Table 3 for the description of an XML element. Each element is described in a table using the format defined in Table 1 for Complex Elements or Table 2 for Simple Elements (no subelements or attributes). If an

element has attributes, these are described in a related table using the format defined in Table 3. In the Specification, the table of attributes will immediately follow the table defining the element.

Table 1: XML Element Descriptions – Complex Elements

Element	*The XML Element Name*		
Description	A narrative description of the XML element, its semantics and its behavior. The description contains the information that a user needs to produce or consume the element.		
Element Type	Description of the type of subelements of the element. Value is one of:		
	Empty	There are no subelements. There are attributes.	
	Any	Any number or type of subelements is permitted.	
	sequence	The elements in the list MUST appear in the XML document in the sequence shown.	
	mixed	The elements in the list MUST appear in the XML document in the sequence shown. Arbitrary text MAY appear around the elements.	
	choice	Only one of the elements in the list MAY appear in the XML document.	
	HTML	The element contains [XHTML 1.1] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	
	QTI	The element contains [QTI 2.1] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	
	MathML	The element contains [MathML] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	
	xsd:<type>	The element contains content that conforms to a specific XML datatype [XML 2] denoted by <type>.	
			For string types, the number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.
			For string types, constraints on the string value are defined via a regular expression.
			For token types, the enumerated list of valid tokens is defined.
			For numeric types, the conditional or range in braces ({≥}) is the inclusive range of the value.
Elements	The list of subelements of the element, each in a separate row in the table. There are two entries (Name, Multiplicity) for each element.		
	This entry is omitted if the element is a specific XML datatype and if there are no subelements and attributes.		
	Name	The name of the element. If there are no elements, the value is *None*.	
	Multiplicity	The multiplicity of the element in an XML document:	
		[0]	Element occurs 0 times.
		[0..1]	Element occurs 0 or 1 times.
		[0..*]	Element occurs 0 or more times.
			The number in braces ({NN}) is the minimum number of element instances that a conforming consumer MUST accept.

Element	*The XML Element Name*		
		[1]	Element occurs 1 time.
		[1..*]	Element occurs 1 or more times.
			The number in braces ({NN}) is the minimum number of element instances that a conforming consumer MUST accept.
		A ☒ indicates the element is a candidate TO BE DEPRECATED.	
		A ☒ indicates the element IS DEPRECATED.	
Attributes	The list of attributes of the element, each in a separate row in the table. There are four entries (Name, Required, Data Type, Default) for each attribute.		
	This entry is omitted if the element is a specific XML datatype and if there are no subelements and attributes.		
	Name	The name of the attribute.	
		If there are no attributes, the value is *None*.	
	Required	A ☑ indicates the attribute is REQUIRED.	
		A ☐ indicates the attribute is OPTIONAL.	
		A ☒ indicates the element is a candidate TO BE DEPRECATED.	
		A ☒ indicates the element IS DEPRECATED.	
	Data Type	The XSD [XSD 2] data type of the attribute.	
	Default	The default value for an OPTIONAL attribute that is omitted from the XML document.	
		The entry is empty for any REQUIRED attribute that does not have a default value.	
		The entry is *None* for any OPTIONAL attribute that does not have a default value.	
Extensions	A ☑ indicates that the element MAY include XML namespaced extensions. A ☒ indicates that the element MAY NOT include XML namespaced extensions.		
Conformance	Any additional semantics and conformance requirements not represented elsewhere.		
	This entry is omitted if there are no additional conformance requirements.		
Notes	Any additional notes about the XML element.		
	The entry is empty if there are no additional notes.		
	All notes are informative.		

Table 2: XML Element Descriptions – Simple Elements

Element	*The XML Element Name*	
Description	A narrative description of the XML element, its semantics and its behavior. The description contains the information that a user needs to produce or consume the element.	
Element Type	The name of the specific XML datatype [XML 2]. There are no subelements.	
		For string types, the number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.
		For string types, constraints on the string value are defined via a regular expression.
		For token types, the enumerated list of valid tokens is defined.
	HTML indicates that the element contains [XHTML 1.1] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	
	QTI indicates that the element contains [QTI 2.1] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	

Element	*The XML Element Name*	
	MathML indicates that the element contains [MathML] content. The number in braces ({NN}) is the minimum string length that a conforming consumer MUST accept.	
	The name of the specific XML element from another namespace. Subelements and attributes of the element are not presented.	
Value	The description of the value space for the element. The description MAY include constraints on acceptable data values for the attribute within the specified data type and value space.	
	For xsd:boolean or xsd:token (a vocabulary) there are two entries (Value, Description) for each value in the value space.	
	Value	A value for the element within the value space.
	Description	A description of the meaning of the value.
Default	The default value for an OPTIONAL element that is omitted from the XML document.	
	The entry is empty for any REQUIRED element that does not have a default value.	
Extensions	A ☑ indicates that the element MAY include XML namespaced extensions. A ☒ indicates that the element MAY NOT include XML namespaced extensions.	
Conformance	Any additional semantics and conformance requirements not represented elsewhere.	
	This entry is omitted if there are no additional conformance requirements.	
Notes	Any additional notes about the XML element.	
	The entry is empty if there are no additional notes.	
	All notes are informative.	

Table 3: XML Element Attribute Descriptions

Attributes	*The XML Element Name*	
The XML Attribute Name	A narrative description of the XML attribute, its semantics and its behavior. The description contains the information that a user needs to produce or consume the attribute for the element. Each attribute is described by two entries (Value, Description).	
	Value	The value space for the attribute.
	Description	Constraints on acceptable data values for the attribute within the specified data type and value space.
		If the attribute is from a constrained vocabulary or value space, there is one table row per vocabulary value or value space indicating the value and describing the meaning of the vocabulary item or value space.

The Specification uses graphics generated by Altova XML Spy® software to illustrate the XML structure of a document. The illustrations use the graphical conventions shown in Figure 1.

Illustrations of XML document structure and elements are informative.

Note: The use of XML Spy does not constitute an endorsement by the SBAC of the product. Other products may be used to produce similar illustrations of the XML document structure.

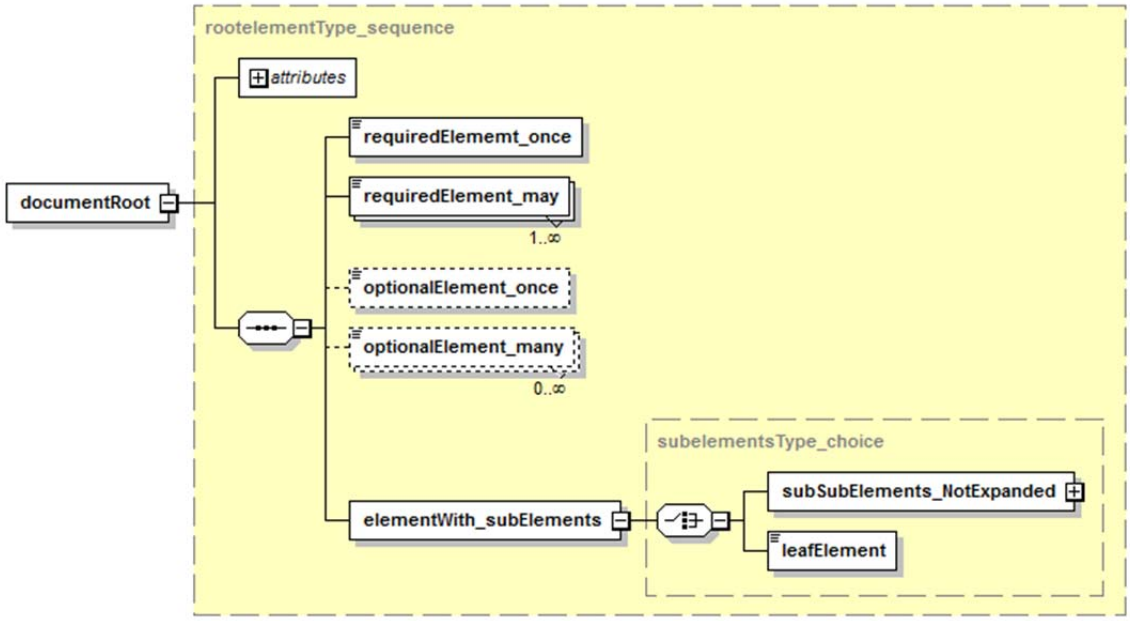


Figure 1: XML Graphical Conventions

Element Presentation Order

Within the description of an XML document, the root element of the document or element tree is described first, followed by subelements in depth-first order. If there are multiple root elements, each and its subelements are described independently.

Common subelements used by multiple elements are collected in a separate section designated *Shared Elements*.

Namespaces

The Specification uses the XML namespace prefixes shown in Table 4. Use of these prefixes in schemata or instance documents is NOT REQUIRED.

Table 4: XML Namespace Prefixes

Document Type/Element	Prefix	Namespace
XSD	xsd:	http://www.w3.org/2001/XMLSchema
Instance	xsi:	http://www.w3.org/2001/XMLSchema-Instance
QTI Assessment Item	qti:	http://www.imsglobal.org/xsd/imsqti_v2p1
MathML Content (Math Element)	mml:	http://www.w3.org/1998/Math/MathML
xHTML	xhtml:	http://www.w3.org/1999/xhtml

Special Characters

Special characters in strings are described with informal character name, followed in parentheses by the character itself, the [ISO 8859-1] *character entity* and the *entity name* for the character, e.g., comma if defined (, , ,).

Typographic Conventions

The Specification uses the typographic conventions shown in Table 5 for XML element and attribute descriptions within the element description tables and code examples.

Table 5: XML Typographic Conventions

Convention	Description
Bold Text	Descriptive metatag used as part of the element description format.
Normal Text	Description of an XML element, XML attribute or attribute value.
Italics *Italics San Serif*	A special value for an XML element, XML attribute or attribute value that is not encoded in XML. Examples include *None* and *Any*. Typically a semantic constraint.
San Serif	Sample XML tags, name, code, values, schemata, or portion thereof.

Informal Document Model

Note: This section is informative.

The entire information model consists of:

- An (optional) *Assessment Item Release* XML document.
- An *Assessment Item* XML document.
- A *Passage Item* XML document.
- A *Tutorial* XML document.
- A *Wordlist* XML document.
- An *Assessment Item Accessibility* XML document.
- A *Grid Item Rendering Specification* XML document.
- An *Equation Editor Configuration* XML document.
- An *Assessment Item Usage Statistics* XML document (not documented in the Specification).
- The *Assessment Item Machine Rubric* XML documents (not documented in the Specification).

The parts of the information model are illustrated in Figure 2. The core of the information model is the *Assessment Item* XML document, with an item being identified by an item number. The *Assessment Item* XML document contains or links to all of the parts of an assessment item.

- An assessment item may include a passage item, stored separately from the assessment item. A passage is a separate type of XML document but it uses some elements that are similar to those of an assessment item. The structure of a passage is defined by the *Passage Item* XML document elements. The assessment item references the passage item through the passage item number.
- An assessment item may contain resources, e.g., a wordlist, stored separately from the assessment item. Each different type of resource is modeled as a type of an assessment item. The assessment item references the item number of the assessment item that contains the resource, e.g., the assessment item references an assessment item of type wordlist. A resource has a unique XML element structure within the assessment item XML element. The structure of a wordlist resource is defined by the *Wordlist* XML document elements. Other resources are treated in the same manner, defined by specialized assessment item XML elements. No other resources are currently defined in the Specification.
- An assessment item may contain a tutorial, stored separately from the assessment item. A tutorial is modeled as an assessment item – some of the XML elements within the assessment item model are not used in a tutorial item. The assessment item references the item number of the assessment item that contains the tutorial content.
- An assessment item may include a rendering specification used to control how the item is displayed. The rendering specification is generally stored separately from the assessment item. Different types of assessments have their own unique *Rendering Specification* XML document elements. There are currently two defined rendering specifications: one for a grid item and one for the equation editor configuration for an equation item. In the current implementation, the grid item rendering specification is stored within in the assessment item instead of being stored separately.
- An assessment item may include a machine rubric used to control how the item is automatically graded. Machine rubrics are present only for assessment items that are automatically graded. The machine rubric is stored separately from the assessment item and each of the different types of machine rubrics is defined by its own *Assessment Item Machine Rubric* XML document elements.
- An assessment item may incorporate usage statistics within the item. The structure of the usage statistics elements is defined by the *Assessment Item Usage Statistics* XML document elements. These elements are documented independently.
- An assessment item may contain file references to attachment files, stored separately from the assessment item. Attachments typically hold accessibility content.

- An assessment item may incorporate accessibility XML elements within the item. The structure of the accessibility elements is defined by the *Assessment Item Accessibility* XML document element. These elements are documented independently as they are used for both assessment (including a tutorial) and passage items. The accessibility XML document is stored within the assessment item.

A *Passage Item* XML document is similar to an *Assessment Item* XML document. The passage is identified by an item number. The *Passage Item* XML document contains or links to all of the parts of the passage item. The passage item information model incorporates a subset of the components of an assessment item: resources, attachments and accessibility elements.

A *Tutorial* XML document is a specialized type of *Assessment Item* XML document. A *Tutorial* XML document constrains which elements of an *Assessment Item* XML document are used to describe the tutorial content.

A *Wordlist* XML (or any other type of resource) document is a variant of an *Assessment Item* XML document. While the top-level XML element is the same as an *Assessment Item* XML document, the structure of a *Wordlist* XML document is specific to a wordlist. Figure 2 illustrates a resource in an assessment item or passage linking to a specific type of resource, a wordlist.

An *Assessment Item Release* XML container document is used to hold an *Assessment Item* XML document or a *Passage Item* XML document inline in the item release document. The *Assessment Item Release* XML document may hold any of the different types of *Assessment Item* XML documents, i.e., a *Wordlist* or *Tutorial* XML document. The *Assessment Item Release* XML container document is an optional wrapper.

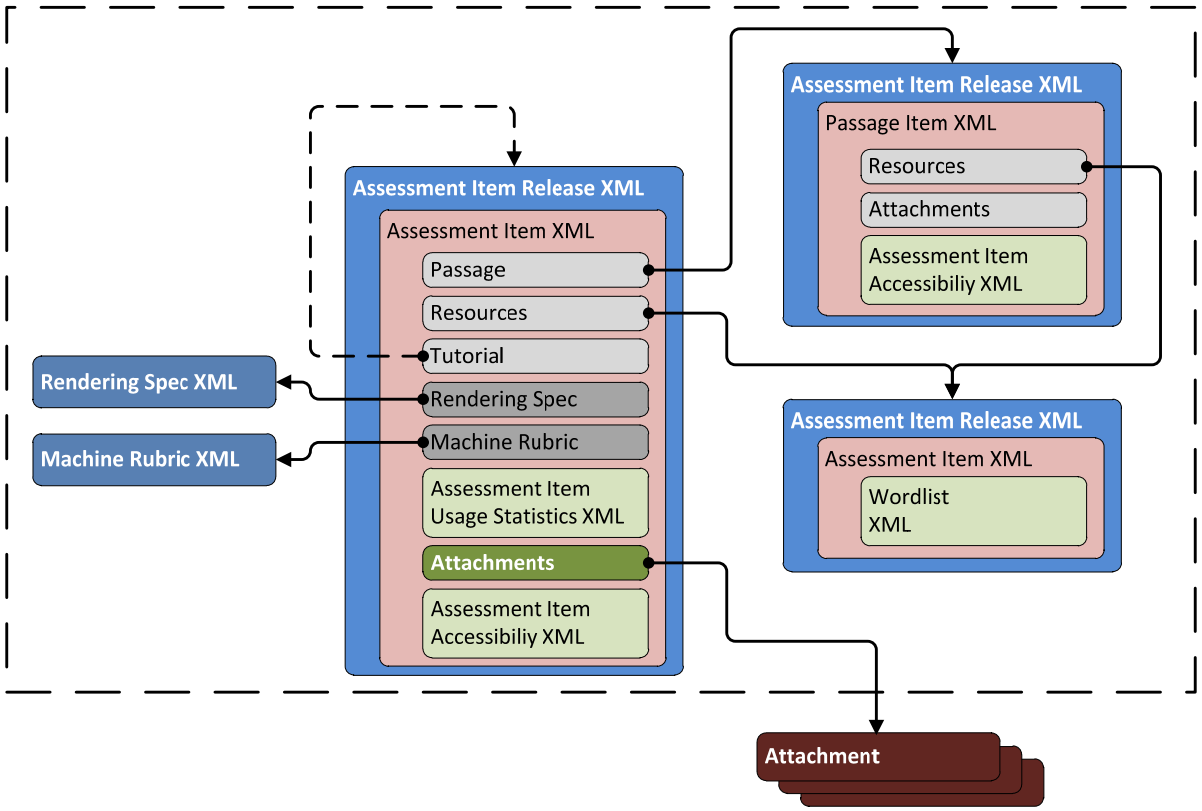


Figure 2: Overall XML Document Model (Informative)

For the purpose of item exchange, an assessment item and its associated files may be organized into a file folder hierarchy and packaged into an exchange format such as a ZIP file for transport. For example, SBAC uses a custom profile of the IMS APIP Profile of IMS Content Packaging [APIP Tech 1.0] that defines the specific file hierarchy and naming structure used to package and exchange SBAC assessment items and associated asset files [SBAC Packaging 1.4].

The Specification allows any attachment to be stored anywhere, fully independent of an assessment system, i.e., the attachment may be a web-accessible document held and maintained by a 3rd party.

The Specification does NOT REQUIRE the use of specific item storage, file name, folder structure, packaging or exchange representations.

Assessment Item Release XML Document Information Model

An *Assessment Item Release* XML document is a container for distribution and exchange of any type of assessment item or passage item. The container is required for the AIR implementation and AIR workflow processes. The container is not required for other purposes. The *Assessment Item Release* XML document contains no information about the assessment item.

The information model for an *Assessment Item Release* XML document includes either:

- An *Assessment Item* XML document (including documents that describe assessment items, tutorials as a type of assessment item, and wordlist resources as a type of assessment item).
- A *Passage Item* XML document.

Assessment Item XML Document Information Model

An *Assessment Item* XML document holds the content for an assessment item that is presented to the student. It may be either the container for the entire assessment item, or a container for a specialized subset of content used in an assessment item, including standalone tutorial content or a standalone resource used by the assessment item, such as a wordlist.

The information model for an *Assessment Item* XML document includes:

- A list of assessment item attributes.
- The assessment item content.
- An associated passage for the assessment item, e.g., a stimulus. The actual passage content is stored in an independent *Passage Item* XML document.
- An item tutorial that is stored is an independent *Tutorial* XML document modeled as an *Assessment Item* XML document.
- A list of item resources, e.g., other independent *Assessment Item* XML documents that have a special format. The resource is stored as an independent XML document modeled as a special type of *Assessment Item* XML document.
- A collection of assessment item usage statistics modeled as an inline *Assessment Item Usage Statistics* XML document.
- A machine scoreable rubric modeled as an independent *Machine Rubric* XML document.
- A rendering specification modeled as an independent *rendering specification* XML document.

The assessment item content includes:

- IMS QTI XML content.
- A list of rationale options describing the expected assessment item response.

- An illustration for an assessment item modeled as a block of HTML, i.e., an HTML wrapper for media.
- A stem modeled as a block of HTML.
- A list of grading rubrics.
- A list of response choice options for the item.
- A list of *attachments* containing accessibility content for the item; each attachment is a separate file.
- Assessment item accessibility information modeled inline as an *Assessment Item Accessibility* XML document.

The assessment item is identified by an item number and version. The item number is unique across all XML documents that include an item number.

Passage Item XML Document Information Model

A *Passage Item* XML document holds the content for a stimulus passage that is presented to the student. A passage item is used with an assessment item and the *Passage Item* XML document is referenced from the associatedpassage element within the assessment item.

The information model for a *Passage Item* XML document includes:

- A list of item attributes.
- The passage item content.
- An item tutorial that is stored is an independent *Tutorial* XML document modeled as an *Assessment Item* XML document.
- A list of item resources, e.g., other independent *Assessment Item* XML documents that have a special format. The resource is stored as an independent XML document modeled as a special type of *Assessment Item* XML document.

The passage item content includes:

- A *stem* modeled as a block of HTML.
- A list of *attachments* containing accessibility content for the item; each attachment is a separate file.
- Passage item accessibility information modeled inline as an *Assessment Item Accessibility* XML document.

The passage item is identified by an item number and version. The item number is unique across all XML documents that include an item number.

Tutorial XML Document Information Model

A *Tutorial* XML document holds the content for a tutorial that is presented to the student. A tutorial is used with an assessment item and the *Tutorial* XML document is referenced from the tutorial element within the assessment item.

A *Tutorial* XML document is modeled as an *Assessment Item* XML document where the value of the format attribute of the item element is tut. Most features of an *Assessment Item* XML document may be used in a *Tutorial* XML document. Additional conformance constraints apply to some of the elements.

The information model for a *Tutorial* XML document mirrors the information model for an *Assessment Item* XML.

The tutorial as an assessment item is identified by an item number and version. The item number is unique across all XML documents that include an item number.

Wordlist XML Document Information Model

A *Wordlist* XML document holds the content for a wordlist (thesaurus and glossary definitions) type of resource. A wordlist is used in with an assessment item and the *Wordlist* XML document is referenced from the resource element within the assessment item.

A wordlist is modeled as a special type of assessment item, i.e., it contains a different set of sublelements.

The information model for a *Wordlist* XML document includes:

- A list of keywords containing individual multi-lingual keyword descriptions, each keyword description modeled as a block of HTML.

The wordlist is identified by an item number and version. The item number is unique across all XML documents that include an item number.

Assessment Item Accessibility XML Document Information Model

An *Assessment Item Accessibility* XML document holds the description of accessibility content (e.g., Braille alternative content). Accessibility content is associated with an assessment item (including a tutorial) or a passage item and is referenced from the apipAccessibility element within the assessment item or passage item.

Accessibility information is stored inline in an *Assessment Item* XML document or *Passage Item* XML document in the apipAccessibility element.

.

The information model for an *Assessment Item Accessibility* XML document includes:

- Accessibility elements containing text-to-speech pronunciation information and Braille or American Sign Language (ASL) alternative text.

Grid Item Rendering Specification XML Document Information Model

A *Grid Item Rendering Specification* XML document holds the rendering configuration settings for a grid type of assessment item. The grid item rendering specification is a type of rendering specification. The grid item rendering specification is referenced from the gridanswerspace element within a grid type of assessment item and is stored inline in the gridanswerspace element.

The information model for a *Grid Item Rendering Specification* XML document includes:

- A description of the display canvas, it rendering and behavior of UI interactions.
- A description of the initial elements placed on the display canvas.

Equation Editor Configuration XML Document Information Model

An *Equation Editor Configuration* XML document holds the configuration settings for the equation editor that is presented to the student for an equation type of assessment item. The equation editor configuration is a type of rendering specification. The equation editor configuration is stored as an independent XML document and referenced from the RendererSpec element within an equation type of assessment item.

The information model for an *Equation Editor Configuration* XML document includes:

- The overall configuration settings for the equation editor.
- Details of panels containing the layout of input keys used to enter equations.
- MathML elements.

Assessment Item Usage Statistics XML Document Information Model

An *Assessment Item Usage Statistics* XML document captures data about the use of an assessment item. The assessment item usage statistics are represented as a collection of subelements of the statistic element. The statistic element is the root of the subtree of elements. The statistic element is one of the subelements of an assessment item.

The complete information model for an *Assessment Item Usage Statistics* XML document will be included in a future version of the Specification.

Assessment Item Machine Rubric XML Document Information Model

An *Assessment Item Machine Rubric* XML document contains the rubric rules for automated item grading. An assessment item may include a *machine rubric*. The machine rubric is contained in an external XML document that is referenced from the assessment item through the filename attribute of the item MachineRubric element in the assessment item. Different types of assessment items use different machine rubrics.

The details of the machine rubric XML elements for the different types of assessment items are not documented in the Specification.

XML Document Elements

Details of the elements used to describe the assessment item XML documents are presented in individual sections, each section describing one of the XML documents. The element details are presented using the notation described.

- An *Assessment Item Release* XML document.
- An *Assessment Item* XML document.
- A *Passage Item* XML document.
- A *Tutorial* XML document.
- A *Wordlist* XML document.
- An *Assessment Item Accessibility* XML document.
- A *Grid Item Rendering Specification* XML document.
- An *Equation Editor Configuration* XML document.
- An *Assessment Item Usage Statistics* XML document (not documented in the Specification).
- The *Assessment Item Machine Rubric* XML documents (not documented in the Specification).

Assessment Item Release XML Document Elements

An *Assessment Item Release* XML document is a container for the distribution and exchange of any type of assessment item or passage item. The container is required for the AIR implementation and AIR workflow processes. The container is not required for other purposes. The *Assessment Item Release* XML document contains no information about the assessment item.

An *Assessment Item Release* document consists of a single root XML element. The element describes release information that is part of the element development workflow. The root element of a document instance contains a single subelement. Different types of subelements are used to contain different types of assessment item documents (assessment items [which includes tutorial and resource items] and passage items). Each of the assessment item element document formats are documented separately in the Specification.

The *Assessment Item Release* XML document is a candidate TO BE DEPRECATED and removed in a future version of the Specification. The different types of assessment item documents will be standalone documents; workflow processes will be documented separately from the assessment item specification.

The element hierarchy within an *Assessment Item Release* XML document is illustrated in Figure 3 (informative).

- The item and passage elements are not expanded in the diagram.
- Consistent with the XML schema design, there is no schema for an *Assessment Item Release* XML document. The *Assessment Item* XML document, *Passage Item* XML document, *Tutorial* XML document and *Wordlist* XML document schemata each contain two root elements, one for the standalone item document and one for the item release document.
- The diagram illustrates the itemrelease element within an *Assessment Item* XML document or a *Passage Item* XML document.
- The form of the *Tutorial* XML document and *Wordlist* XML document follow that of the *Assessment Item* XML document.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.

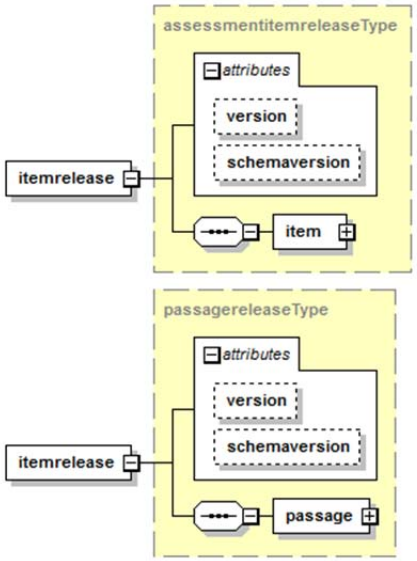


Figure 3: Assessment Item Release XML Document Structure (Informative)

Assessment Item Release Elements

Element	itemrelease			
Description	Container element for the release of an *Assessment Item*.			
Element Type	choice			
Elements	**Name**	**Multiplicity**		
	item	[1]		
	passage	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	version	☑ ☒	xsd:string {100}	*None*
Extensions	☒			
Notes	The item and passage elements are documented individually below.			
	The itemrelease element contains either a single item (including a tutorial or a wordlist modeled as an item) or a single passage.			
	The itemrelease element is a candidate TO BE DEPRECATED and removed. The different types of *Assessment Item* documents will be standalone documents.			

Attributes	itemrelease
version	Version identifier for the item as part of the release.
	There are no constraints on the value of the attribute.
	A value is REQUIRED but not used. Any non null string MAY be used.
	The version attribute is a candidate TO BE DEPRECATED and removed. The attribute is REQUIRED.

Assessment Item XML Document Elements

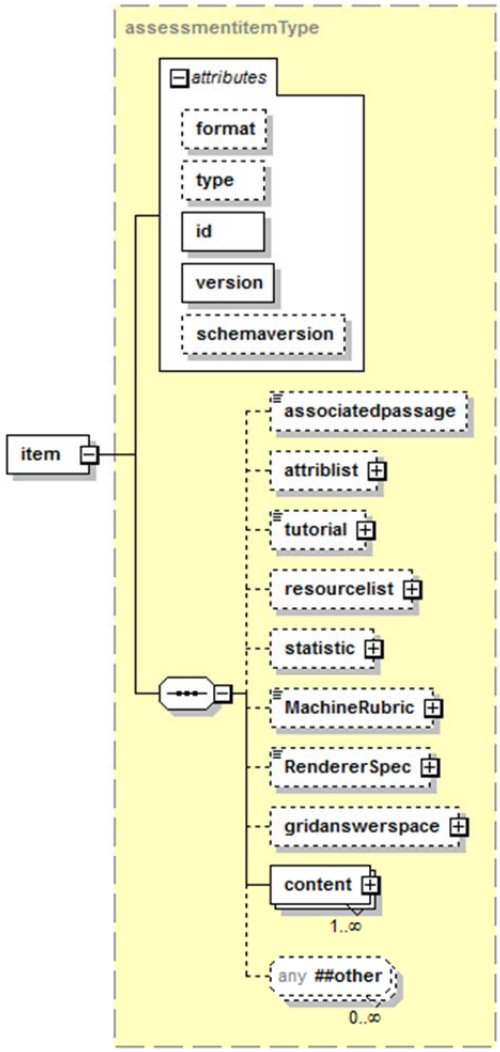
An *Assessment Item* XML document holds the content for an assessment item that is presented to the student. It MAY be either the container for the entire assessment item, or a container for a specialized subset of content used in an assessment item, including standalone tutorial content or standalone resources used by the assessment item, such as a wordlist.

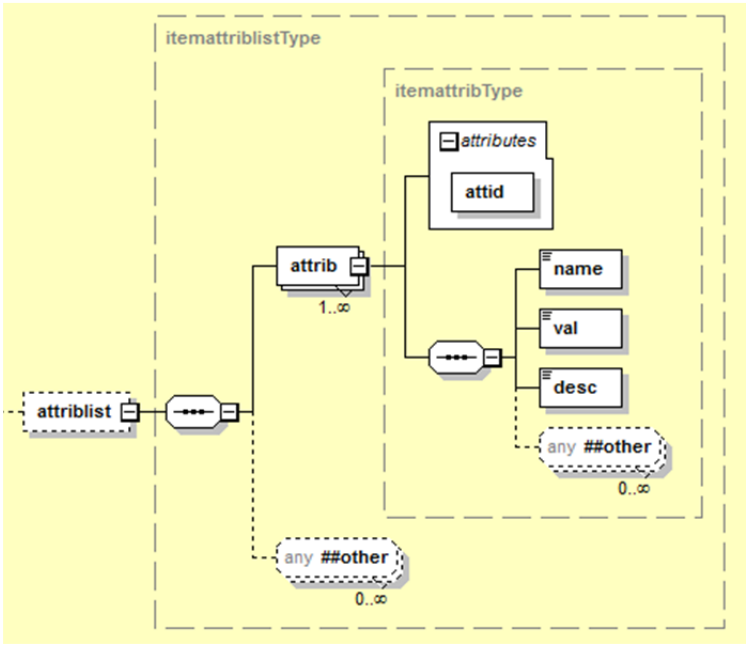
The XML elements for an *Assessment Item* XML document are detailed in four groups:

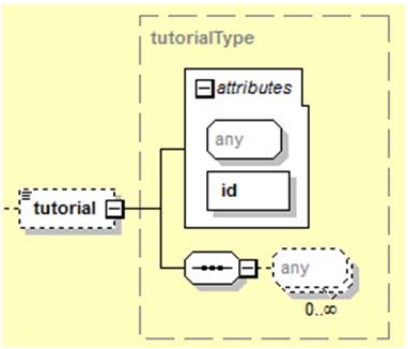
- *Assessment item elements* – the definition of elements used to describe the item as a whole. A single item element is the root element of the element tree. The item element MAY be embedded in an itemrelease element.
- *Content elements* – the definition of elements used to describe the content of the item. Content elements are rooted through a set of content subelements within the item element.
- *Accessibility elements* – the definition of elements used to describe accessibility features for the item. Accessibility elements are rooted through a single apipAccessibility subelement within any set of item content elements. Definitions of accessibility elements are shared with other types of items and are documented separately in the Specification.
- *Shared elements* – the definition of simple, common XML elements that are subelements of various other elements (i.e., name, val, desc, annotation) and have common usage throughout the assessment item. Shared elements with the same names are used in other XML documents. Their definition MAY be XML-document specific.

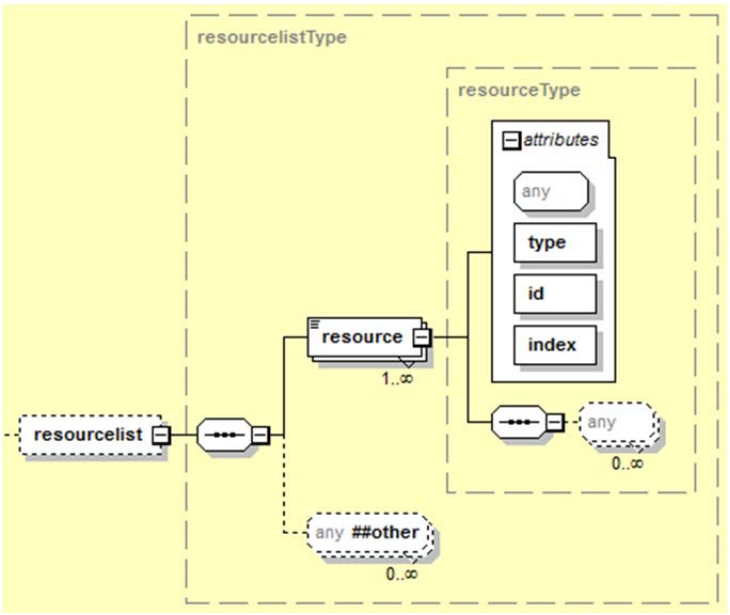
The element hierarchy within an *Assessment Item* XML document is illustrated in Figure 4 (informative).

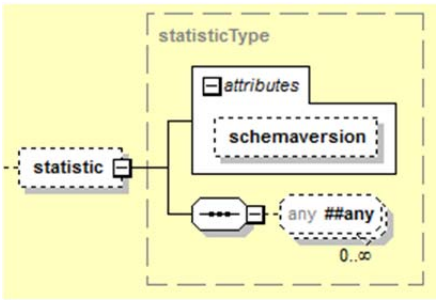
- For clarity, the first part of the element hierarchy show at the top of the diagram does not expand any of the elements. Each is expanded individually.
- Each of the elements within the content element are also expended individually.
- The xHTML content of the illustration, stem, samplecontent, feedback and any val elements are not expanded in the diagram.
- The gridanswerspace element (Grid Rendering) is not expanded in the diagram.
- Accessibility content within the apipAccessibility element is not expanded in the diagram.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.

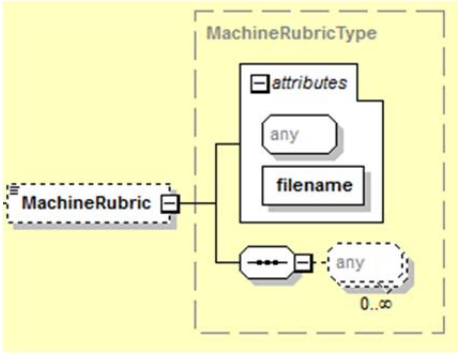

assessmentitemType
attributes
format
type
id
version
schemaversion
item
associatedpassage
attriblist
tutorial
resourcelist
statistic
MachineRubric
RendererSpec
gridanswerspace
content
1..∞
any ##other
0..∞

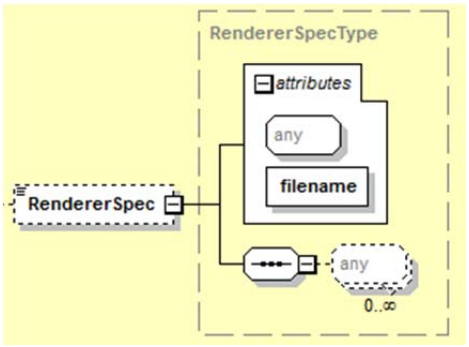

itemattributlistType
itemattributType
attributes
attid
attrib
1..∞
name
val
desc
any ##other
0..∞
attriblist
any ##other
0..∞

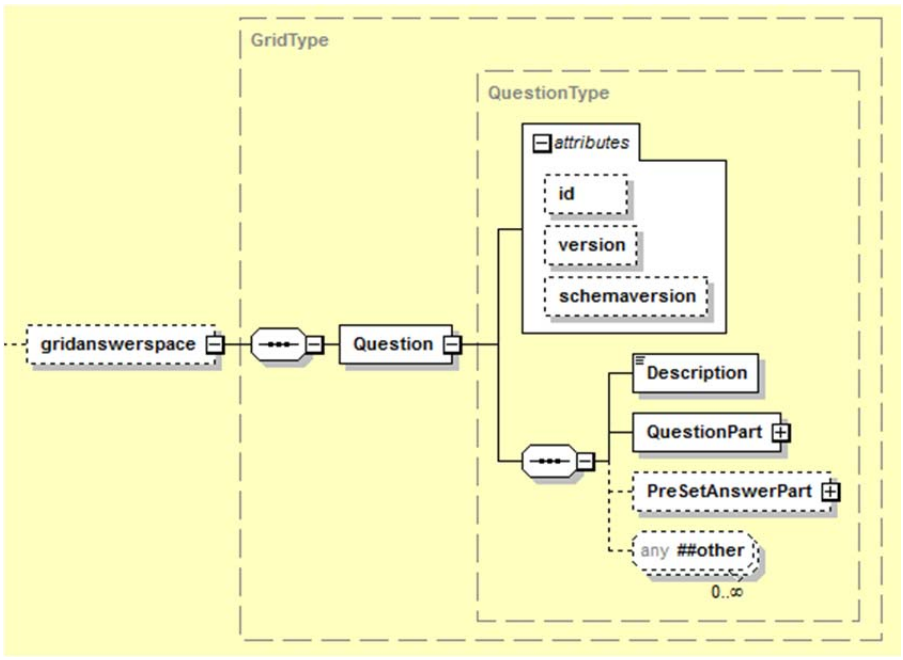

tutorialType
attributes
any
id
tutorial
any
0..∞

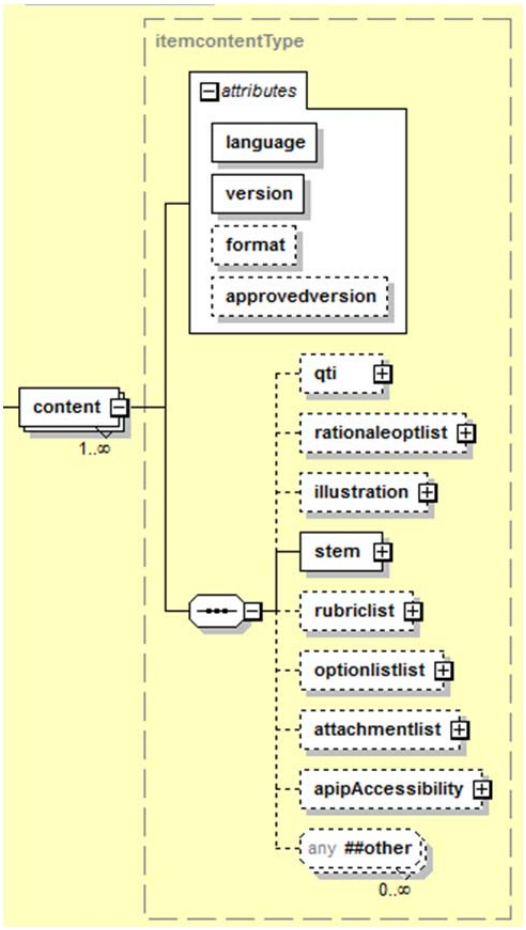

resourcelistType
resourceType
attributes
any
type
id
index
resource
1..∞
resourcelist
any
0..∞
any ##other
0..∞

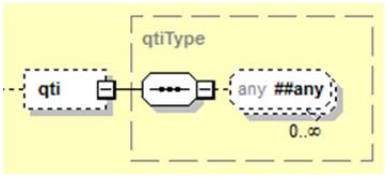

statisticType
attributes
schemaversion
statistic
any ##any
0..∞

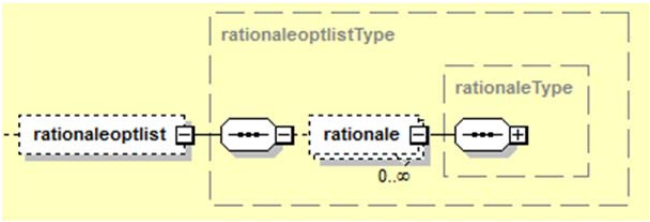

MachineRubricType
attributes
any
filename
MachineRubric
any
0..∞

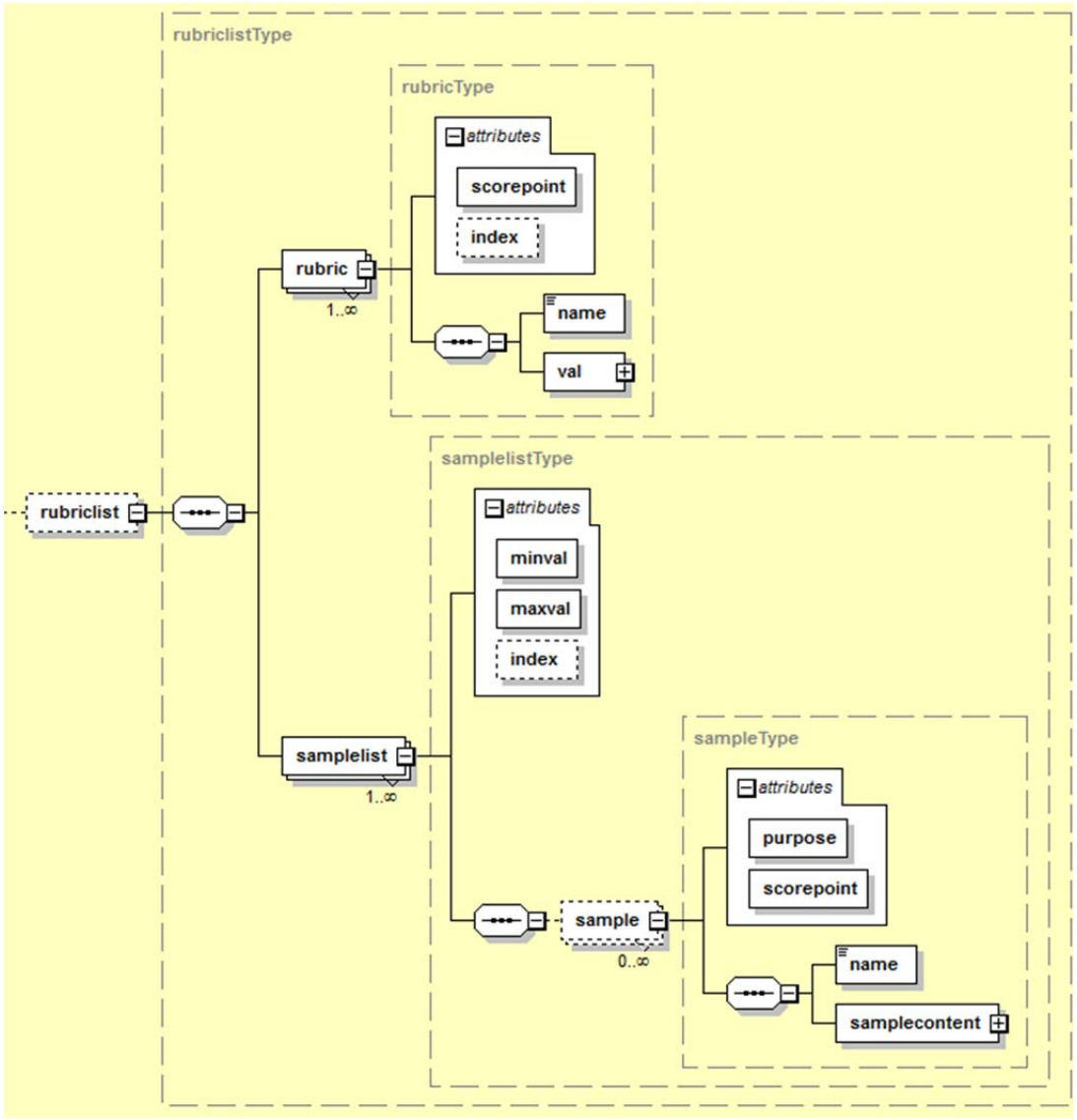

RendererSpecType
attributes
any
filename
RendererSpec
any
0..∞

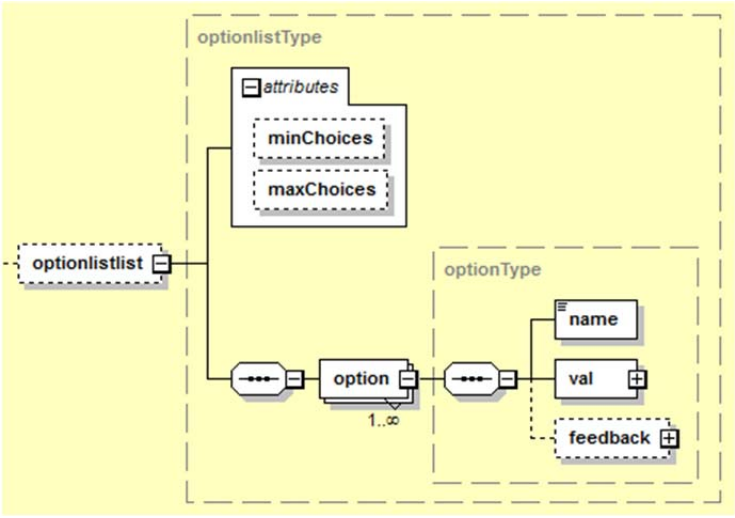

GridType
QuestionType
attributes
id
version
schemaversion
gridanswerspace
Question
Description
QuestionPart
PreSetAnswerPart
any ##other
0..∞


itemcontentType
attributes
language
version
format
approvedversion
content
1..∞
qti
rationaleoptlist
illustration
stem
rubriclist
optionlistlist
attachmentlist
apipAccessibility
any ##other
0..∞


qtiType
qti
any ##any
0..∞


rationaleoptlistType
rationaleType
rationaleoptlist
rationale
0..∞


rubriclistType
rubricType
attributes
scorepoint
index
rubric
1..∞
name
val
samplelistType
attributes
minval
maxval
index
rubriclist
samplelist
1..∞
sampleType
attributes
purpose
scorepoint
sample
0..∞
name
samplecontent


optionlistType
attributes
minChoices
maxChoices
optionlistlist
optionType
name
option
1..∞
val
feedback

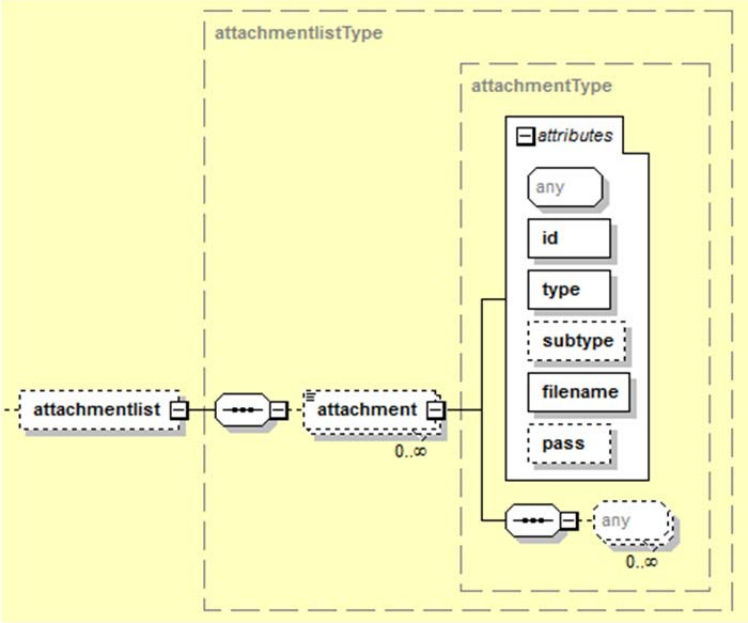


Figure 4: Assessment Item XML Document Structure (Informative)

Assessment Item Elements

Element	item			
Description	An assessment item.			
Element Type	sequence			
Elements	Name	**Multiplicity**		
	associatedpassage	[0..1]		
	attriblist	[0..1]		
	tutorial	[0..1]		
	resourcelist	[0..1]		
	statistic	[0..1]		
	MachineRubric	[0..1]		
	RendererSpec	[0..1]		
	gridanswerspace	[0..1]		
	content	[1..*] {10}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	format	☐	xsd:token	
	type	☐ ☒	xsd:token	
	id	☑	xsd:int {>0}	
	version	☑	xsd:string {100}	
Extensions	☑			
Conformance	Either the format or the type attribute SHALL be present.			
	An element that contains both the format and type attributes SHALL be NON CONFORMING.			
	The attriblist element SHALL be present only if the format attribute value or type attribute value is not wordlist (wordlist).			
	An element that contains the attriblist element with a format attribute value or type attribute value of wordlist (wordlist) SHALL be NON CONFORMING.			

Element	item
	The tutorial element SHALL be present only if the format attribute value is not tut (tutorial).
	An element that contains the tutorial element with a format attribute value of tut (tutorial) SHALL be NON CONFORMING.
	The gridanswerspace element SHALL be present only if the format attribute value is gi (grid item).
	An element that contains the gridanswerspace element with a format attribute value other than gi (grid item) SHALL be NON CONFORMING.
	The RendererSpec element SHALL be present only if the format attribute value is eq (equation item).
	An element that contains the RendererSpec element with a format attribute value other than eq (equation item) SHALL be NON CONFORMING.
Notes	All types of items except *Wordlist* items indicate the item type with the format attribute. *Wordlist* items use the type attribute instead of the format attribute to indicate the item type.
	The type attribute is a candidate TO BE DEPRECATED and replaced by the format attribute.
	If present, the associatedpassage element is ignored for *Passage* or *Tutorial* items.
	Wordlist items use the item element but use a different set of subelements. *Wordlist* items are documented separately in the Specification.
	How or when the *Assessment Item* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Attributes	item	
format	The type of the item. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	EBSR	Evidence-Based Selected Response item.
	eq	Equation item.
	er	Extended Response item.
	gi	Grid item.
	ht	Hot Text item.
	mc	Multiple Choice item.
	mi	Match Interaction item.
	ms	Multi-Select item.
	nl	Natural Language item.
	pass	Passage item.
	sa	Short Answer item.
	SIM	Simulation item.
	ti	Table Interaction item.
	tut	Tutorial item.
	wer	Writing Extended Response item.
	wordlist	Wordlist resource.
	All types of items except *Wordlist* items SHALL indicate the item type with the format attribute. wordlist is included in the vocabulary but SHALL NOT be used as a value of format. It is included to permit the type attribute TO BE DEPRECATED and replaced by the format attribute.	
	The value pass is reserved and SHALL NOT be used.	

Attributes	item			
	The list of values MAY be extended in a future version of the Specification.			
type	The type of the item. A vocabulary of values.			
	The value SHALL be one of the vocabulary values listed.			
	Value	**Description**		
	wordlist	Wordlist resource.		
	The type attribute SHALL be used only for *Wordlist* items.			
	The type attribute is a candidate TO BE DEPRECATED and replaced by the format attribute.			
	The list of values SHALL NOT be extended.			
id	Unique item number for the item.			
	The value of the item number SHALL be unique within the context of all items.			
	The value of the item number SHALL be $< 2^{31}-1$.			
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.			
version	Version identifier for the item.			
	The value SHOULD match the regular expression: \d+(\.\d+)?(\.\d+)?			

Element	associatedpassage			
Description	Item number for the stimulus or passage for an item.			
Element Type	xsd:int {>0}			
Value	*Any*			
Default	*None*			
Extensions	☒			
Conformance	The value of the element SHALL match the id of the corresponding *Passage Item*.			
	An element that contains an item number that references an item that is not a *Passage Item* SHALL be NON CONFORMING.			
Notes	How the element value item number is converted into the file name of the XML document holding the corresponding *Passage Item* is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	There MAY be multiple *Passage Items* with the same item number but with different version numbers. How to determine the version of the passage that is referenced is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The file location and naming convention for the *Passage Item* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	How or when the *Passage Item* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Element	attriblist			
Description	Attributes of an item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	attrib	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			

Element	attriblist
Notes	How or when the attributes are used is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Element	attrib			
Description	Attribute of an item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	val	[1]		
	desc	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	attid	☑	xsd:token	
Extensions	☑			
Conformance	The value of the name element and the value and value space of the value element SHALL align with the value of the attid attribute as shown in Table 6.			
	An element that contains a name element or a value or value space of the value element that does not align with the value of the attid attribute as shown in Table 6 SHALL be NON CONFORMING.			
Notes				

Attributes	attrib	
attid	The identifier for the attribute.	
	Value	**Description**
	itm_att_Answer Key	The item rubric.
	itm_att_Cloze Answers	The item rubric for *Cloze* or *Word Builder* items.
	itm_att_Grade	Grade level for the item.
	itm_att_Item Format	Item format.
	itm_att_Item Point	The maximum number of points for the item.
	itm_att_Page Layout	The layout file used to render the item.
	itm_att_Response Type	The rendering of the item.
	itm_att_Strand	The content standard that the item is aligned to.
	itm_FTUse	A description of the item's use on a field test form.
	itm_OPUse	A description of the item's use on an operational test form.
	itm_item_desc	A description of the item.
	itm_item_id	The item number of the item.
	itm_item_subject	The subject of the item.
	stm_pass_id	The item number of the associated stimulus passage.

Additional details of each of the attributes are shown in Table 6. The table includes:

- The attribute id attid.
- An indication if the attribute is REQUIRED (☑) or OPTIONAL (☐).
- The value of the name element that corresponds to the attribute id.
- The value of the value element that corresponds to the attribute id.
- The value space of the value element.
- Notes and conformance criteria. The conformance criteria (indicated with upper case letters) and notes (indicated with lower case roman numerals) are detailed after the table.

Table 6: Assessment Item Attributes

attid	☑/☐	name	value	value space	Notes
itm_att_Answer Key	☑	Item: Answer Key	Item rubric	xsd:string	A,B,C,D,i
itm_att_Cloze Answers	☐	Item: Cloze Answers	Item rubric	xsd:string	E.F,ii
itm_att_Grade	☑	Item: Grade	Item grade level	xsd:token	G,H,I,J
itm_att_Item Format	☑	Item: Item Format	Item format	xsd:token	K,L
itm_att_Item Point	☑	Item: Item Point	Any	xsd:string	iii
itm_att_Page Layout	☑	Item: Page Layout	Layout file	xsd:int {>0}	M,N.O,iv,v
itm_att_Response Type	☑	Item: Response Type	Rendering code	xsd:token	P,Q,vi
itm_att_Strand	☐	Item: Strand	Standard	xsd:string	
itm_FTUse	☐	Field test Use	Any	xsd:string	vii
itm_OPUse	☐	Operational Use	Any	xsd:string	viii
itm_item_desc	☑	Item: Item Description	Any	xsd:string	
itm_item_id	☑	Item: ITS ID	*Assessment Item* number	xsd:int {>0}	R,S
itm_item_subject	☑	Item: Subject	Subject classifier: MATH, ELA or STUDENT HELP	xsd:token	ix
stm_pass_id	☐	Stim: ITS ID	Associated *Passage Item* number	xsd:int {>0}	T,U

General conformance criteria are:

- A name element or a value or value space of the value element that does not align with the value of the attid attribute as shown SHALL be NON CONFORMING.
- An attid element that contains a name element or a value or value space of the value element that does not align with the value of the attid attribute as shown SHALL be NON CONFORMING.

Specific conformance criteria referenced from Table 6 are:

A. For an attid value of itm_att_Answer Key the value of the value element SHALL match the value of the format attribute of the item element converted to upper case (as shown in Table 7) if the value of the format attribute of the item element is not mc or ms.
B. For an attid value of itm_att_Answer Key a value of the value element that does not match the value of the format attribute of the item element converted to upper case (as shown in Table 7) SHALL be NON CONFORMING if the value of the format attribute of the item element is not mc or ms.
C. For an attid value of itm_att_Answer Key the value of the value element SHALL match the regular expression \a+(, \a+)* if the value of the format attribute of the item element is mc or ms.
D. For an attid value of itm_att_Answer Key the value of the value element that does not match the regular expression \a+(, \a+)* SHALL be NON CONFORMING if the value of the format attribute of the item element is mc or ms.
E. For an attid value of itm_att_Cloze Answers the value of the value element SHALL match the regular expression \d+(, \d+)*
F. For an attid value of itm_att_Cloze Answers the value of the value element that does not match the regular expression \d+(, \d+)* SHALL be NON CONFORMING.
G. For an attid value of itm_att_Grade the value of the value element SHALL match the regular expression (KG|01|1|02|2|03|3|04|4|05|5|06|6|07|7|08|8|09|9|10|11|12|NA).

H. For an attid value of itm_att_Grade the value of the value element that does not match the regular expression (KG|01|1|02|2|03|3|04|4|05|5|06|6|07|7|08|8|09|9|10|11|12|NA) SHALL be NON CONFORMING.
I. For an attid value of itm_att_Grade a value of the value element of NA SHALL be only used if the value of the format attribute of the item element is tut.
J. For an attid value of itm_att_Grade a value of the value element of NA if the value of the format attribute of the item element is not tut SHALL be NON CONFORMING.
K. For an attid value of itm_att_Item Format the value of the value element SHALL match the value of the format attribute of the item element converted to upper case (as shown in Table 7).
L. For an attid value of itm_att_Item Format a value of the value element that does not match the value of the format attribute of the item element converted to upper case SHALL be NON CONFORMING (as shown in Table 7).
M. For an attid value of itm_att_Page Layout the value of the value element SHALL match the values shown in Table 8 for each item type.
N. For an attid value of itm_att_Page Layout a value of the value element that does not match the values shown in Table 8 for each item type SHALL be NON CONFORMING.
O. For an attid value of itm_att_Page Layout a value of the value element that does not reference a rendering layout file SHALL be NON CONFORMING.
P. For an attid value of itm_att_Response Type the value of the value element SHALL match the values shown in Table 9 for each item type.
Q. For an attid value of itm_att_Response Type a value of the value element that does not match the values shown in Table 9 for each item type SHALL be NON CONFORMING.
R. For an attid value of itm_item_id the value of the value element SHALL match the value of the id attribute of the item element.
S. For an attid value of itm_item_id a value of the value element that does not match the value of the id attribute of the item element SHALL be NON CONFORMING.
T. For an attid value of stm_pass_id the value of the value element SHALL match the value of the id attribute of the associatedpassage element.
U. For an attid value of stm_pass_id a value of the value element that does not match the value of the id attribute of the associatedpassage item element SHALL be NON CONFORMING.

Additional notes referenced from Table 6 are:

i. The value of the value attribute for the itm_att_Answer Key attribute is a rubric for a multiple choice item (the item format attribute value is MC) or a multiple select item (the item format attribute value is MS). The rubric is a comma separated list of letters. The meaning of the rubric is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
ii. The rubric is a comma separated list of numbers. The meaning of the rubric is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
iii. The form and meaning of the value of the value attribute for the itm_att_Item Point are NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
iv. How an itm_att_Page Layout value of the value element is converted to the name of the rendering layout file is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
v. The list of values of the value attribute of itm_att_Page Layout MAY be extended in a future version of the Specification.
vi. The list of values of the value attribute of itm_att_Response Type MAY be extended in a future version of the Specification.
vii. The form and meaning of the value of the value attribute for the itm_FTUse attribute are NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
viii. The form and meaning of the value of the value attribute for the itm_OPUse attribute are NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
ix. The list of values of the value attribute of the itm_item_subject attribute MAY be extended in a future version of the Specification.

The mapping of the item format to the corresponding value of itm_item_Format is show in Table 7.

Table 7: Item Format to itm_item_Format Value Mapping

Item Format	itm_item_Format Value
EBSR	EBSR
eq	EQ
er	ER
gi	GI
ht	HT
mc	MC
mi	MI
ms	MS
nl	NL
pass	PASS
sa	SA
SIM	SIM
ti	TI
tut	TUT
wer	WER
wordlist	WORDLIST

The mapping of the item format to the corresponding value of itm_att_Page Layout is show in Table 8. The meaning of the itm_att_Page Layout value is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Table 8: Item Format to itm_att_Page Layout Value Mapping

Item Format	itm_att_Page Layout Values
EBSR	21
eq	8, 21
er	1, 8, 21, 29
gi	8, 21, 22
ht	8, 21
mc	8, 21
mi	1, 8, 21
ms	8, 21
nl	8, 21
pass	
sa	8, 21
SIM	
ti	13
tut	
wer	21
wordlist	

The mapping of the item format to the corresponding value of itm_att_Response Type is show in Table 9. The meaning of the itm_att_Response Type value is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Table 9: Item Format to itm_att_Response Type Value Mapping

Item Format	itm_att_Response Type Value
EBSR	EBSR
eq	EquationEditor
er	PlainText
gi	Grid
ht	HotText
mc	Vertical
	Stacked
mi	TableMatch
	MatchItem
ms	Vertical MS
nl	PlainText
pass	NA
sa	PlainText
SIM	NA
ti	TableInput
tut	NA
wer	HTMLEditor
wordlist	NA

Element	tutorial			
Description	Item number for the *Tutorial* for an item.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	id	☑	xsd:int {>0}	
Extensions	☑			
Conformance	The value for the id attribute SHALL match the id of the corresponding *Tutorial*.			
	An element that contains a value for the id attribute that references an item that is not a *Tutorial* SHALL be NON CONFORMING.			
Notes	How the id attribute item number is converted into the file name of the XML document holding the corresponding *Tutorial* is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	There MAY be multiple *Tutorial Items* with the same item number but with different version numbers. How to determine the version of the *Tutorial* that is referenced is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The file location and naming convention for the *Tutorial* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	How or when the *Tutorial* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	tutorial
id	Item number for the *Tutorial* for an item.
	The value of the item number SHALL be unique within the context of all items.
	The value of the item number SHALL be $< 2^{31}-1$.

Attributes	tutorial
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Element	resourcelist			
Description	Additional resources for an item. The resource is described in an XML document specific to the type of resource.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	resource	[1..*] {10}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	Most items use only one resource.			
	Tools such as a calculator, protractor, or ruler used by an assessment item are specified on the test form and not with the item.			

Element	resource			
Description	A resource for an item. The resource is described in an XML document type that is specific to the type of resource.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	type	☑	xsd:token	
	id	☑	xsd:int {>0}	
	index	☑	xsd:int {>0}	
Extensions	☑			
Conformance	The value of the id attribute SHALL match the id of the corresponding resource that has an item type attribute that matches the type attribute.			
	An element that contains a value for the id attribute that references a resource that has an item type attribute that does not match the type attribute SHALL be NON CONFORMING.			
Notes	How the id attribute value item number is converted into the file name of the XML document holding the corresponding resource is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	There MAY be multiple resources with the same item number but with different version numbers. How to determine the version of the resource that is referenced is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The file location and naming convention for the resource XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	How or when the *Resource* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	resource	
type	Type of the resource. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	tutorial	The resource is a *Tutorial* item. The *Tutorial* XML document structure is described separately.
	wordlist	The resource is a *Wordlist* item. The *Wordlist* XML document structure is described separately.
	The value tutorial is reserved and SHALL NOT be used.	
id	Item number for the resource for an item.	
	The value of the item number SHALL be unique within the context of all items.	
	The value of the item number SHALL be $< 2^{31}-1$.	
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
index	The presentation order of the resource in the list of resources. The values need not be contiguous.	

Element	statistic			
Description	The container for the *Assessment Item Usage Statistics* XML document elements.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	The set of elements are documented separately in the *Assessment Item Usage Statistics* XML document.			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	The content of a non-empty statistic element SHALL be ignored.			
Notes	The element is normally empty (<statistic />) when authoring an item or before the item has been used.			
	An empty statistic element SHOULD NOT be interpreted to mean that there are no usage statistics for the item.			

Element	MachineRubric			
Description	The file name for the rubric used to machine score the item. The rubric format and structure is specific to the format of the *Assessment Item*.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	filename	☑	xsd:string {4000}	
Extensions	☑			
Conformance	The value of the filename attribute SHALL reference an *Assessment Item Machine Rubric* XML document whose content is appropriate for the assessment item format.			
	An element that contains a value of the filename attribute that references an *Assessment Item Machine Rubric* XML document whose content is not appropriate for the assessment item format SHALL be NON CONFORMING.			

Element	MachineRubric
	The MachineRubric element SHOULD be omitted for any format of an assessment item that is not automatically graded.
	The MachineRubric element SHALL be omitted for any format of an assessment item that is not automatically graded.
Notes	The *Assessment Item Machine Rubric* XML Document structure is defined separately.
	Different *Assessment Item Machine Rubric* XML document structures are used for different assessment item formats.
	The file location and naming convention for the *Machine Rubric* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].

Attributes	MachineRubric
filename	File name of the file containing the rubric.
	The file location and naming convention *Machine Rubric* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Element	RendererSpec			
Description	The file name for the rendering specification used to render the item on the test client device. The rendering specification is specific to the format of the *Assessment Item*.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	filename	☑	xsd:string {4000}	
Extensions	☑			
Conformance	The value of the filename attribute SHALL reference a rendering specification XML document whose content is appropriate for the assessment item format.			
	An element that contains a value of the filename attribute that references a rendering specification XML document whose content is not appropriate for the assessment item format SHALL be NON CONFORMING.			
	The RendererSpec element SHALL be present only if the format attribute value is eq (equation item).			
	An element that contains the RendererSpec element with a format attribute value other than eq (equation item) SHALL be NON CONFORMING.			
	An item with a format attribute value of eq (equation item) that does not include a RendererSpec element SHALL be NON CONFORMING.			
Notes	The rendering specifications are defined separately for the different assessment item formats.			
	Different rendering specifications are used for different assessment item formats.			
	The file location and naming convention for the *Rendering Specification* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			

Attributes	RendererSpec
filename	File name of the file containing the rendering specification.
	The file location and naming convention for the *Rendering Specification* XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Element	gridanswerspace			
Description	The container for the *Grid Item Rendering Specification* XML document elements.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	The set of elements are documented separately in the *Grid Item Rendering Specification* XML document.			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	The gridanswerspace element SHALL be present only if the format attribute value is gi (grid item).			
	An element that contains the gridanswerspace element with a format attribute value other than gi (grid item) SHALL be NON CONFORMING.			
	An item with a format attribute value of gi (grid item) that does not include a gridanswerspace element SHALL be NON CONFORMING.			
Notes				

Content Elements

Element	content			
Description	Content of an assessment item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	qti	[0..1]		
	rationaleoptlist	[0..1]		
	illustration	[0..1]		
	stem	[1]		
	rubriclist	[0..1]		
	optionlist	[0..1]		
	attachmentlist	[0..1]		
	apipAccessibility	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	language	☑ ☒	xsd:language	
	version	☑ ☒	xsd:string {100}	
	format	☐	xsd:token	
	approvedversion	☐	xsd:string {100}	
Extensions	☑			
Conformance	An XML document with two or more content elements with the same value of the language attribute or of xml:lang SHALL be NON CONFORMING.			

Element	content
	The value of the version attribute SHOULD match the value of the version attribute of the itemrelease element in the *Assessment Item Release* XML container document.
	The behavior if the value of the version attribute does not match the value of the version attribute of the itemrelease element is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
	The value of the format attribute SHOULD match the value of the format attribute of the item element.
	An element that contains a value for the format attribute that does not match the value of the format attribute of the item element SHALL be NON CONFORMING.
	The value of the approvedversion attribute SHOULD match the value of the version attribute of the item element.
	The behavior if the element that contains a value for the approvedversion attribute that does not match the value of the version attribute of the item element is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
Notes	There is one content element instance for each language variant of the item.
	The language variant only applies to the non media (text) content. Animation media elements in xHTML content referenced through an href attribute of an anchor element MAY have multiple language variant files or multiple format variant files even though the element only specifies one file in one format. How the test client selects the file to delivery is NOT SPECIFIED and is IMPLEMENTATION DEFINED. Specific rules used to name the media files are described in *Document Criteria*.

Attributes	content	
language	Language of the content.	
	The value SHALL conform to [RFC 5646].	
	The language attribute is a candidate TO BE DEPRECATED and replaced by xml:lang. The language attribute is REQUIRED. xml:lang MAY be used in addition to the language attribute.	
version	The version identifier for the content of the item release.	
	There are no constraints on the value of the attribute.	
	A value is REQUIRED but not used. Any non null string MAY be used.	
	The version attribute is a candidate TO BE DEPRECATED and removed. The attribute is REQUIRED.	
format	The type of the item. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	EBSR	Evidence-Based Selected Response item.
	eq	Equation item.
	er	Extended Response item.
	gi	Grid item.
	ht	Hot Text item.
	mc	Multiple Choice item.
	mi	Match Interaction item.
	ms	Multi-Select item.
	pass	Passage item.
	nl	Natural Language item.
	sa	Short Answer item.
	SIM	Simulation item.

Attributes	content	
	ti	Table Interaction item.
	tut	Tutorial item.
	wer	Writing Extended Response item.
	wordlist	Wordlist resource.
	The value pass is reserved and SHALL NOT be used.	
approvedversion	Version identifier for the item content.	
	The value SHOULD match the regular expression: \d+(\.\d+)?(\.\d+)?	

Element	qti			
Description	The container for IMS QTI XML document elements describing an *Assessment Item*.			
Element Type	QTI			
Elements	**Name**	**Multiplicity**		
	Any			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	spec	☑	xsd:token	
Extensions	☒			
Conformance	The content of the subelements SHOULD conform to the requirements for the itemBody element of [QTI 2.1 XML].			
	The behavior if the subelements do not conform to the requirements for the itemBody element of [QTI 2.1 XML] is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.			
Notes	All QTI itemBody subelements and attributes are permitted as defined in [QTI XML 2.1]. All QTI interactions are supported.			
	Existing item MAY include non-conforming extensions to QTI2.1. These items will not validate as conforming to QTI2.1.			
	Extensions to QTI 2.1 are currently under consideration to produce QTI 2.2. A future version of the Speciation MAY specific the use of QTI 2.2. The qti element content MAY fully conform to QTI 2.2 (conformance will be "SHALL conform to the requirements for the itemBody element of [QTI 2.2 XML]").			

Attributes	qti	
spec	Type of QTI content.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	itemBody	The QTI content is any QTI itemBody elements.

Element	rationaleoptlist			
Description	Rationales for each response option.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	rationale	[0..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	This element SHALL be used only for a multiple choice (the item format attribute value is MC) or a multiple select (the item format attribute value is MS) item.			

Element	rationaleoptlist
	An item that is not a multiple choice item (the item format attribute value is MC) or a multiple select item (the item format attribute value is MS) and includes the rationaloptlist element SHALL be NON CONFORMING.
Notes	

Element	rationale			
Description	Justification for a response.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	val	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	This element SHALL be used only for a multiple choice (the item format attribute value is MC) or a multiple select (the item format attribute value is MS) item.			
	An item that is not a multiple choice item (the item format attribute value is MC) or a multiple select item (the item format attribute value is MS) and includes the rationale element SHALL be NON CONFORMING.			
Notes				

Element	illustration
Description	Illustration associated with an item.
Element Type	HTML {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The illustration content SHOULD conform to [XHTML 1.1].
Notes	How or when the *illustration* is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
	The illustration content MAY include references to external content files. The file location and naming convention for the content file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].

Element	stem
Description	Directions to the student to provide a response to the item.
Element Type	HTML {64000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The stem content SHOULD conform to [XHTML 1.1].
Notes	

Element	rubriclist			
Description	Rubrics associated with the item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	rubric	[1..*] {100}		
	samplelist	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	This element SHALL be used only for an extend response (the item format attribute value is er), an equation (the item format attribute value is eq) item, a grid (the item format attribute value is gi), a natural language (the item format attribute value is nl) or a short answer (the item format attribute value is sa) item.			
	An item that is not an extend response item (the item format attribute value is er), an equation item (the item format attribute value is eq) item, a grid item (the item format attribute value is gi), a natural language item (the item format attribute value is nl) or a short answer item (the item format attribute value is sa) and includes the rubriclist element SHALL be NON CONFORMING.			
	The number of elements in rubric SHALL equal the number of elements in samplelist.			
	A rubriclist element that does not have the same number of elements for rubric and samplelist SHALL be NON CONFORMING.			
Notes	The structure is a collection of pairs of rubric and samplelist elements. There is a one-to-one correspondence between each item in rubric and samplelist.			
	An optional index attribute has been added to the rubric and samplelist elements to align the pairs.			

Element	rubric			
Description	The scoring rubric.			
Element Type	*sequence*			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	val	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	scorepoint	☑	xsd:string {100}	
	index	☐	xsd:int {>0}	
Extensions	☑			
Conformance	The value of the index attribute SHALL match the value of the index attribute of a samplelist element.			
	An element that contains a value of the index attribute that does not match the value of the index attribute of a samplelist element SHALL be NON CONFORMING.			
Notes	The name element holds the name of the rubric.			
	The val element holds the rubric text in [XHTML 1.1].			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	rubric
scorepoint	The number of points that can be awarded to the student.
	The value MAY be an empty string.

Attributes	rubric
	If the value MAY only be interpreted as an integer only if the value is nonnegative.
Index	The order of the rubric element within the rubriclist. The values need not be contiguous.
	The index attribute is used to align the ordering or rubric and samplelist elements.

Element	samplelist			
Description	Examples of applications of the rubric.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	sample	[0..*]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	minval	☑	xsd:int {>0}	
	maxval	☑	xsd:int {>0}	
	index	☐	xsd:int {>0}	
Extensions	☑			
Conformance	The value of minval SHALL equal the value of maxval.			
	The value of minval SHALL equal the value of scorepoint in the rubric element.			
	The value of the index attribute SHALL match the value of the index attribute of a rubric element.			
	An element that contains a value of the index attribute that does not match the value of the index attribute of a rubric element SHALL be NON CONFORMING.			
Notes	There is typically only zero instances of the sample element, one instance with a value of Exemplar for the purpose attribute or two instances, one with the value of Exemplar and the other with the value of OtherExemplar for the purpose attribute.			
	There is typically zero instances of the sample element for an equation item (the item format attribute value is eq) or for a grid item (the item format attribute value is gi.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	samplelist
minval	Minimum number of points awarded.
maxval	Maximum number of points awarded.
index	The order of the samplelist element within the rubriclist. The values need not be contiguous.
	The index attribute is used to align the ordering or rubric and samplelist elements.

Element	sample			
Description	Example of application of the rubric.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	samplecontent	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	purpose	☑	xsd:token	
	scorepoint	☑	xsd:string {100}	

Element	sample
Extensions	☑
Notes	

Attributes	sample	
purpose	The purpose of the sample. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	Exemplar	The sample is an exemplar rubric.
	OtherExemplar	The sample is an alternative exemplar rubric.
scorepoint	The number of points that can be awarded to the student.	
	The value MAY be an empty string.	
	If the value MAY only be interpreted as an integer only if the value is nonnegative.	

Element	samplecontent
Description	Description of how to apply the rubric for the specified purpose.
Element Type	HTML {16000}
Value	*Any*
Default	*None*
Extensions	☑
Conformance	The samplecontent content SHOULD conform to [XHTML 1.1].
Notes	

Element	optionlist			
Description	Response choices associated with the item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	option	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	minChoices	☐	xsd:int {≥0}	
	maxChoices	☐	xsd:int {≥0}	
Extensions	☑			
Conformance	This element SHALL be used only for a multiple choice (the item format attribute value is MC) or a multiple select (the item format attribute value is MS) item.			
	An item that is not a multiple choice item (the item format attribute value is MC) or a multiple select item (the item format attribute value is MS) and includes the optionlist element SHALL be NON CONFORMING.			
	The value of minChoices SHALL not exceed the number of items in option.			
	An element that contains a value of minChoices that exceeds the number of items in option SHALL be NON CONFORMING.			
	The value of maxChoices SHALL be greater than or equal to the value of minChoices.			
	An element that contains a value of maxChoices that is less than the value of minChoices SHALL be NON CONFORMING.			
	The value of maxChoices SHALL not exceed the number of items in option.			
	An element that contains a value of maxChoices that exceeds the number of items in option SHALL be NON CONFORMING.			

Element	optionlist
Notes	A value of 0 for minChoices or maxChoices indicates any number of options is valid.

Attributes	option
minChoices	The minimum number of options that MUST be present.
maxChoices	The maximum number of options that MUST be present.

Element	option			
Description	Response choice description.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	val	[1]		
	feedback	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	This element SHALL be used only for a multiple choice (the item format attribute value is MC) or a multiple select (the item format attribute value is MS) item.			
	An item that is not a multiple choice item (the item format attribute value is MC) or a multiple select item (the item format attribute value is MS) and includes the option element SHALL be NON CONFORMING.			
Notes				

Element	feedback
Description	Content presented to the student when the associated option is selected.
Element Type	HTML {16000}
Value	*Any*
Default	*None*
Extensions	☑
Conformance	The feedback content SHOULD conform to [XHTML 1.1].
Notes	How or when the *feedback* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Element	attachmentlist			
Description	Accessibility attachments associated with the item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	attachment	[0..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	Attachments are used to associate ASL or Braille files with an item.			

Element	attachment			
Description	URI of an attachment included with an item.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	id	☑	xsd:string {4000}	
	type	☑	xsd:token	
	subtype	☐	xsd:token	
	filename	☑	xsd:string {4000}	
	pass	☐ ☒	xsd:boolean	
Extensions	☑			
Conformance	The value of the subtype attribute SHALL be limited to a subset of the values based on the value of the type attribute.			
	An item that has a value of the subtype attribute that does not correspond to one of valid values based on the value of the type attribute SHALL be NON CONFORMING.			
	The pass attribute is REQUIRED if the value of the type attribute is ASL. The attribute value SHALL be true.			
	An item that has a value of the type attribute of ASL and does not have a value of the pass attribute of true SHALL be NON CONFORMING.			
Notes	The file location and naming convention for the attachment file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	The pass attribute is a candidate TO BE DEPRECATED and removed. It is equivalent to using the value of ASL for the type attribute with STEM for the value of the subtype attribute.			

Attributes	attachment	
id	An identifier that uniquely identifies the attachment.	
type	Type of attachment. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	ASL	The attachment is American Sign Language (ASL).
	BRF	The attachment is a Braille Ready File (BRF).
	PRN	The attachment is a Print Ready Document (PRN).
subtype	Subtype of the attachment. A vocabulary of values. The value is dependent on the value of type.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	contracted	Braille contracted – the value is valid only for type BRF or PRN.
	uncontracted	Braille uncontracted – the value is valid only for type BRF or PRN.
	nemeth	Braille nemeth [Nemeth] – the value is valid only type BRF or PRN.
	STEM	ASL Stem – the value is valid only for type ASL.
filename	File name of the file containing the attachment.	
	The file location and naming convention for the attachment file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].	

Attributes	attachment	
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
pass	The attachment is ASL for the item stem.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	true	The attachment is ASL for the item stem.
	false	The value is NON CONFORMING.

Shared Elements

Element	name
Description	Human readable name of an attribute, rationale, rubric, scorepoint or option.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
	For an attribute element, the value of name SHALL correspond to the value of attid as shown in Table 6.
	An attribute element that contains a value of name that does not correspond to the value of attid as shown in Table 6 SHALL be NON CONFORMING.
Notes	name is a subelement of several other elements (attribute, rationale, rubric, scorepoint, option). The definition is the same for all uses.

Element	val
Description	Value of an attribute, rationale, rubric, scorepoint or option.
Element Type	HTML {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	For an attribute element HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
	For a rationale, rubric, scorepoint or option element content SHOULD conform to [XHTML 1.1].
	For an attribute element, the value of val SHALL correspond to the value of attid as shown in Table 6.
	An attribute element that contains a value of name that does not correspond to the value of attid as shown in Table 6 SHALL be NON CONFORMING.
Notes	name is a subelement of several other elements (attribute, rationale, rubric, scorepoint, option). Except as noted in Conformance, the definition is the same for all uses.

Element	desc
Description	Human readable description of an attribute.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	

Passage Item XML Document Elements

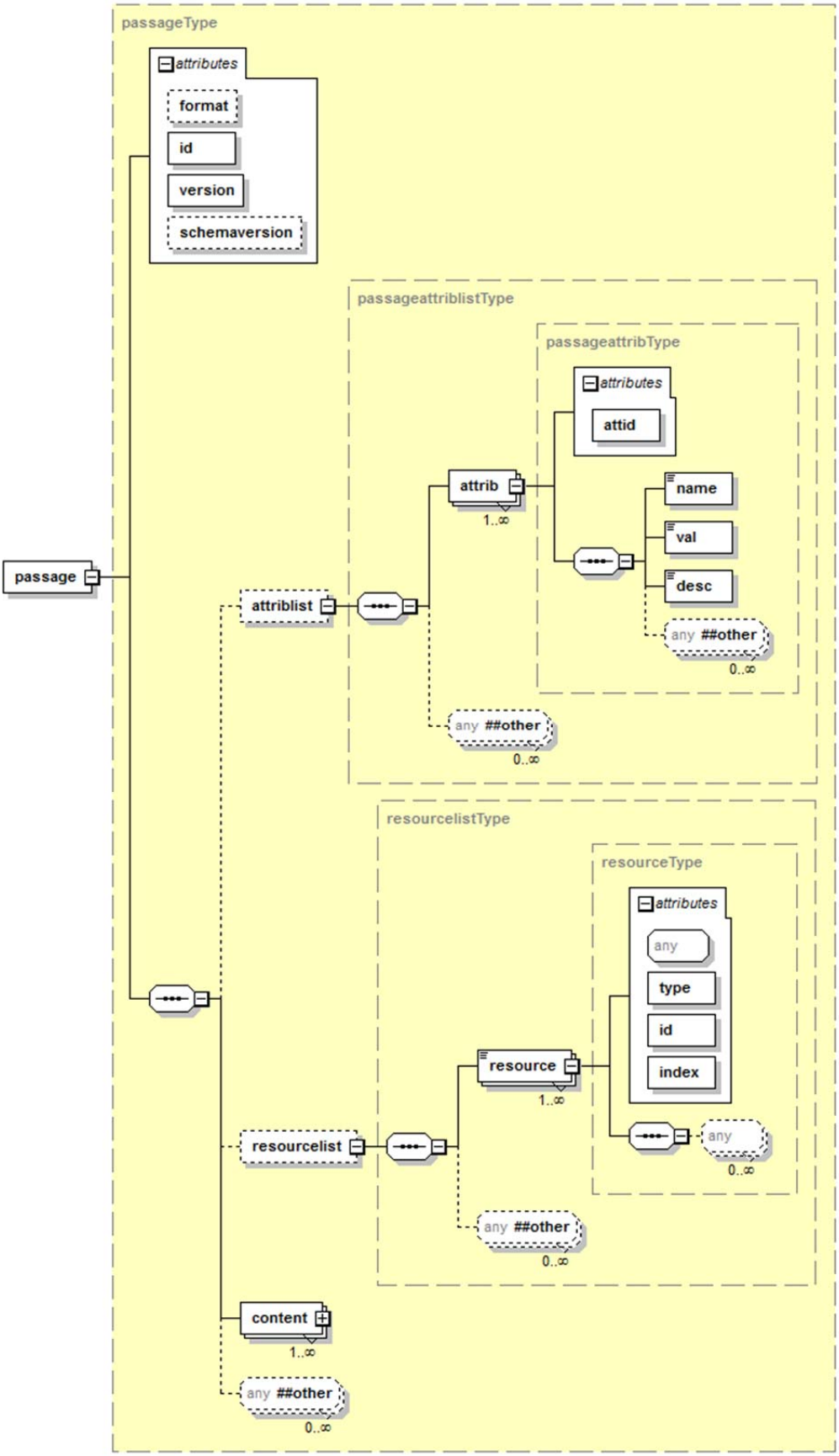
A *Passage Item* XML document holds the content for a stimulus passage that is presented to the student. A passage item is used with an assessment item and the *Passage Item* XML document is referenced from the associatedpassage element within the assessment item.

The XML elements for a *Passage Item* XML document are detailed in four groups:

- *Passage item elements* – the definition of elements used to describe the passage item as a whole. A single passage element is the root element of the element tree. The passage element MAY be embedded in an itemrelease element.
- *Content elements* – the definition of elements used to describe the content of the passage, i.e., the stimulus, of the passage. Content elements are rooted through a set of content subelements within the passage element.
- *Accessibility elements* – the definition of elements used to describe accessibility features for the passage. Accessibility elements are rooted through a single apipAccessibility subelement within any set of passage content elements. Definitions of accessibility elements are shared with other types of items and are documented separately in the Specification.
- *Shared elements* – the definition of simple, common XML elements that are subelements of various other elements (i.e., name, val, desc) and have common usage throughout the passage item. Shared elements with the same names are used in other XML documents. Their definition MAY be XML-document specific.

The element hierarchy within a *Passage Item* XML document is illustrated in Figure 5 (informative).

- For clarity, the first part of the element hierarchy show at the top of the diagram does not expand the content element. It is expanded in the lower part of the diagram.
- The xHTML content of the title, author and stem elements is not expanded in the diagram.
- Accessibility content within the apipAccessibility element is not expanded in the diagram.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.


passageType
attributes
format
id
version
schemaversion
passageattriblistType
passageattribType
attributes
attid
attrib
1..∞
name
val
desc
any ##other
0..∞
passage
attriblist
any ##other
0..∞
resourcelistType
resourceType
attributes
any
type
id
index
resource
1..∞
any
0..∞
resourcelist
any ##other
0..∞
content
1..∞
any ##other
0..∞

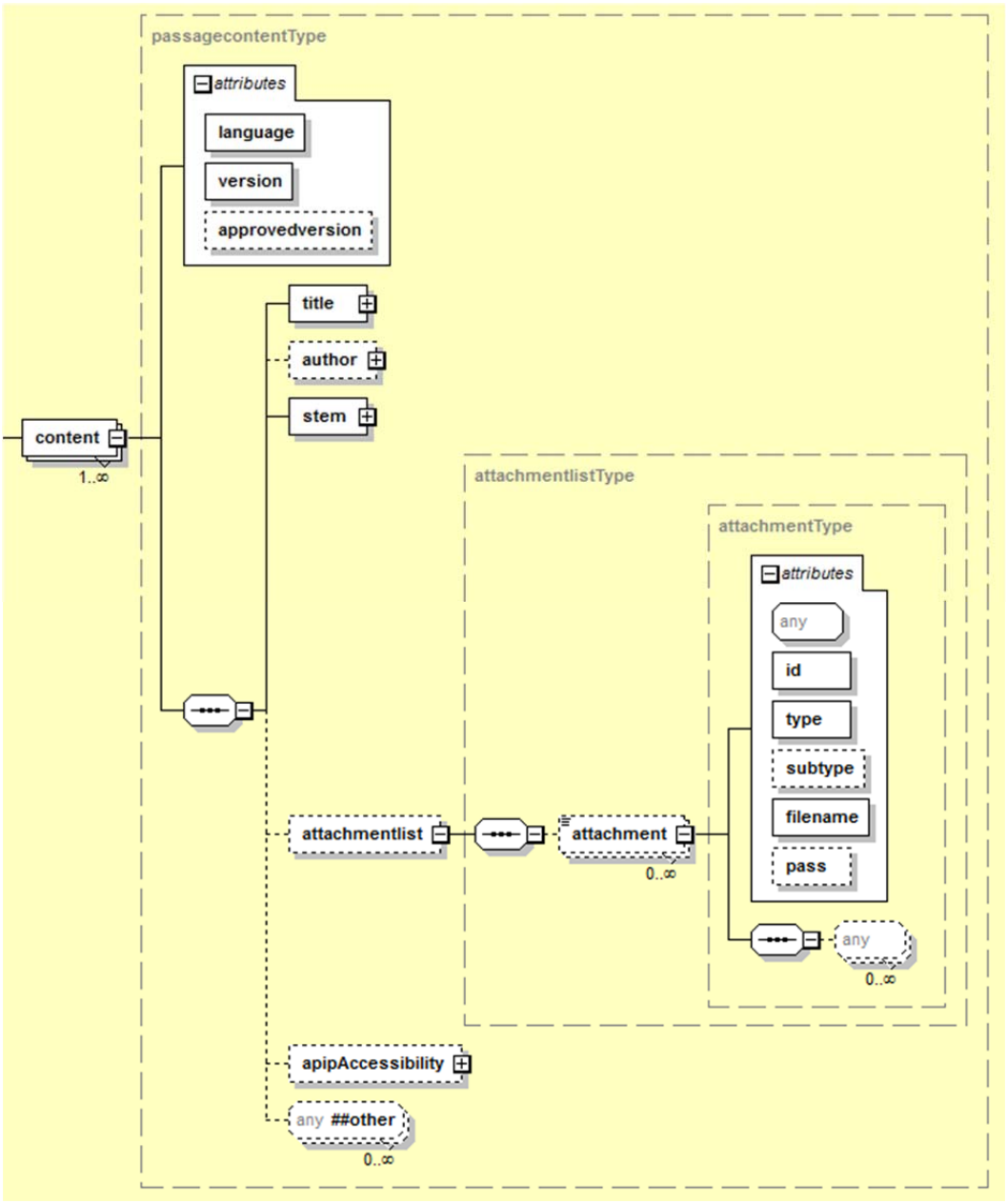


Figure 5: Passage Item XML Document Structure (Informative)

Passage Item Elements

Element	passage			
Description	An assessment item passage.			
Element Type	sequence			
Elements	Name	Multiplicity		
	attriblist	[0..1]		
	resourcelist	[0..1]		
	content	[1..*] {10}		

Element	passage			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	format	☐	xsd:token	
	id	☑	xsd:int {>0}	
	version	☑	xsd:string {100}	
Extensions	☑			
Notes	The format attribute has been added for alignment with the attributes of an *Assessment Item*. This permits a *Passage Item* to be treated as type of Assessment Item in a future version of the Specification.			
	How or when the *Passage Item* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	passage	
format	The type of the passage item. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	pass	Passage item.
	The format attribute SHOULD NOT be used.	
id	Unique item number for the *Passage Item*.	
	The value of the item number SHALL be unique within the context of all items.	
	The value of the item number SHALL be $< 2^{31}-1$.	
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
version	Version identifier for the passage.	
	The value SHOULD match the regular expression: \d+(\.\d+)?(\.\d+)?	

Element	attriblist			
Description	Attributes of a passage.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	attrib	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	How or when the attributes are used is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.			

Element	attrib			
Description	Attribute of a passage.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	name	[1]		
	val	[1]		
	desc	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	attid	☑	xsd:token	
Extensions	☑			

Element	attrib
Conformance	The value of the name element and the value and value space of the value element SHALL align with the value of the attid attribute as shown in Table 10.
	An element that contains a name element or a value or value space of the value element that does not align with the value of the attid attribute as shown in Table 10 SHALL be NON CONFORMING.
Notes	

Attributes	attrib	
attid	The identifier for the attribute.	
	Value	**Description**
	stm_pass_desc	A description of the passage.
	stm_pass_id	The item number of the passage.
	stm_pass_subject	The subject of the passage.

Additional details of each of the attributes are shown in Table 10. The table includes:

- The attribute id attid.
- An indication if the attribute is REQUIRED (☑) or OPTIONAL (☐).
- The value of the name element that corresponds to the attribute id.
- The value of the value element that corresponds to the attribute id.
- The value space of the value element.
- Notes and conformance criteria. The conformance criteria (indicated with upper case letters) and notes (indicated with lower case roman numerals) are detailed after the table.

Table 10: Passage Item Attributes

attid	☑/☐	name	value	value space	Notes
stm_pass_desc	☑	Stim: Description	Any	xsd:string	
stm_pass_id	☑	Stim: ITS ID	*Passage Item* number	xsd:int {>0}	A,B
stm_pass_subject	☑	Stim: Subject	Subject classifier: MATH, ELA or STUDENT HELP	xsd:token	C,D,i

General conformance criteria are:

- A name element or a value or value space of the value element that does not align with the value of the attid attribute as shown SHALL be NON CONFORMING.
- An attid element that contains a name element or a value or value space of the value element that does not align with the value of the attid attribute as shown SHALL be NON CONFORMING.

Specific conformance criteria referenced from Table 10 are:

A. For an attid value of stm_pass_id the value of the value element SHALL match the id attribute of the passage element.
B. For an attid value of stm_pass_id a value of the value element that does not match the id attribute of the passage element SHALL be NON CONFORMING.
C. For an attid value of stm_pass_subject the value of the value element SHALL be the token MATH, ELA or STUDENT HELP.
D. An element with an attid value of stm_pass_subject that has a value of the value element that is not MATH, ELA or STUDENT HELP SHALL be NON CONFORMING.

Additional notes referenced from Table 10 are:

i. The list of values of the value attribute of stm_pass_subject MAY be extended in a future version of the Specification.

Element	resourcelist			
Description	Additional resources for a passage. The resource is described in an XML document specific to the type of resource.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	resource	[1..*] {10}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	Most passages use only one resource.			

Element	resource			
Description	A resource for a passage. The resource is described in an XML document type that is specific to the type of resource.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	type	☑	xsd:token	
	id	☑	xsd:int {>0}	
	index	☑	xsd:int {>0}	
Extensions	☑			
Conformance	The value of the id attribute SHALL match the id of the corresponding resource that has an item type attribute that matches the type attribute.			
	An element that contains a value for the id attribute that references a resource that has an item type attribute that does not match the type attribute SHALL be NON CONFORMING.			
Notes	How the id attribute value item number is converted into the file name of the XML document holding the corresponding resource NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	There MAY be multiple resources with the same item number but with different version numbers. How to determine the version of the resource that is referenced is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The list of values of the type attribute MAY be extended in a future version of the Specification.			
	The value of tutorial for the type attribute has been added to permit tutorials to be modeled as resources in a future version of the Specification.			
	The file location and naming convention for the resource XML document are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	How or when the *Resource* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	resource	
type	Type of the resource. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	tutorial	The resource is a *Tutorial* item. The *Tutorial* XML document structure is described separately.
	wordlist	The resource is a word list. The *Wordlist* XML document structure is described separately.
	The value tutorial is reserved and SHALL NOT be used.	
id	Item number for the resource for an item.	
	The value of the item number SHALL be unique within the context of all items.	
	The value of the item number SHALL be $< 2^{31}-1$.	
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
index	The presentation order of the resource in the resource list. The values need not be contiguous.	

Content Elements

Element	content			
Description	Content of a passage.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	author	[0..1]		
	stem	[1]		
	attachmentlist	[0..1]		
	apipAccessibility	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	language	☑ ☒	xsd:language	
	version	☑ ☒	xsd:string {100}	
	approvedversion	☐	xsd:string {100}	
Extensions	☑			
Conformance	An XML document with two or more content elements with the same value of language or of xml:lang SHALL be NON CONFORMING.			
	The value of the version attribute SHOULD match the value of the version attribute of the itemrelease element in the *Assessment Item Release* XML container document.			
	The behavior if the value of the version attribute does not match the value of the version attribute of the itemrelease element is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The value of the approvedversion attribute SHOULD match the value of the version attribute of the passage element.			
	The behavior if an element that contains a value for the approvedversion attribute that does not match the value of the version attribute of the passage element is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
Notes	There is one content element instance for each language variant of the passage.			

Element	content
	The language variant only applies to the non media (text) content. Animation media elements in xHTML content referenced through an href attribute of an anchor element MAY have multiple language variant files or multiple format variant files even though the element only specifies one file in one format. How the test client selects the file to delivery is NOT SPECIFIED and is IMPLEMENTATION DEFINED. Specific rules used to name the media files are described in *Document Criteria*.

Attributes	content
language	Language of the content.
	The value SHALL conform to [RFC 5646].
	The language attribute is a candidate TO BE DEPRECATED and replaced by xml:lang. The language attribute is REQUIRED. xml:lang MAY be used in addition to the language attribute.
version	The version identifier for the content of the passage item release.
	There are no constraints on the value of the attribute.
	A value is REQUIRED but not used. Any non null string MAY be used.
	The version attribute is a candidate TO BE DEPRECATED and removed. The attribute is REQUIRED.
approvedversion	Version identifier for the passage content.
	The value SHOULD match the regular expression: \d+(\.\d+)?(\.\d+)?

Element	title
Description	The description of the passage.
Element Type	HTML {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The title content elements SHOULD conform to [XHTML 1.1].
Notes	How or when the title element content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Element	author
Description	The author of the passage.
Element Type	HTML {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The author content elements SHOULD conform to [XHTML 1.1].
Notes	The element MAY be empty.
	How or when the author element content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Element	stem
Description	Directions to the student to provide a response to the item.
Element Type	HTML {64000}

Element	stem
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The stem content SHOULD conform to [XHTML 1.1].
Notes	How or when the stem element content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Element	attachmentlist			
Description	Accessibility attachments associated with the passage.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	attachment	[0..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	Attachments are used to associate ASL or Braille files with a passage.			

Element	attachment			
Description	URI of an attachment included with the passage.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	id	☑	xsd:string {4000}	
	type	☑	xsd:token	
	subtype	☐	xsd:token	
	filename	☑	xsd:string {4000}	
	pass	☐ ☒	xsd:boolean	
Extensions	☑			
Conformance	The value of the subtype attribute SHALL be limited to a subset of the values based on the value of the type attribute.			
	An item that has a value of the subtype attribute that does not correspond to one of valid values based on the value of the type attribute SHALL be NON CONFORMING.			
	The pass attribute is REQUIRED if the value of the type attribute is ASL. The attribute value SHALL be true.			
	An item that has a value of the type attribute of ASL and does not have a value of the pass attribute of true SHALL be NON CONFORMING.			
Notes	The file location and naming convention for the attachment file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	The pass attribute is a candidate TO BE DEPRECATED and removed. It is equivalent to using the value of ASL for the type attribute with STEM for the value of the subtype attribute.			

Attributes	attachment
id	An identifier that uniquely identifies the attachment.
type	Type of attachment. A vocabulary of values.

Attributes	attachment	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	ASL	The attachment is American Sign Language (ASL).
	BRF	The attachment is a Braille Ready File (BRF).
	PRN	The attachment is a Print Ready Document (PRN).
subtype	Subtype of the attachment. A vocabulary of values. The value is dependent on the value of type.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	contracted	Braille contracted – the value is valid only for type BRF or PRN.
	uncontracted	Braille uncontracted – the value is valid only for type BRF or PRN.
	nemeth	Braille nemeth [Nemeth] – the value is valid only for type BRF or PRN.
	STEM	ASL Stem – the value is valid only for type ASL.
filename	File name of the file containing the attachment.	
	The file location and naming convention for the attachment file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].	
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
pass	The attachment is ASL for the item stem.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	true	The attachment is ASL for the item stem.
	false	The value is NON CONFORMING.

Shared Elements

Element	name
Description	Human readable name of an attribute.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
	For an attribute element, the value of name SHALL correspond to the value of attid as shown in Table 10.
	For an attribute element, a value of name that does not correspond to the value of attid as shown in Table 10 SHALL be NON CONFORMING.
Notes	

Element	val
Description	Value of an attribute.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	For an attribute element, the value of val SHALL correspond to the value of attid as shown in Table 10.
	For an attribute element, a value of val that does not correspond to the value of attid as shown in Table 10 SHALL be NON CONFORMING.
Notes	

Element	desc
Description	Human readable description of an attribute.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	

Tutorial XML Document Elements

A *Tutorial* XML document holds the content for a tutorial that is presented to the student. A tutorial is used with an assessment item and the *Tutorial* XML document is referenced from the tutorial element within the assessment item.

A *Tutorial* XML document is modeled as an *Assessment Item* XML document where the value of the format attribute of the item element is tut. Most features of an *Assessment Item* XML document MAY be used in a *Tutorial* XML document. Additional conformance constraints apply to some of REQUIRED the elements. The item element MAY be embedded in an itemrelease element.

To use a tutorial in an assessment item, the assessment item will contain a reference to the tutorial by including a tutorial element containing an id attribute with a value of an item number that corresponds to the id attribute of the item element in the *Tutorial* XML document.

How the id attribute item number is converted into the file name of the *Tutorial* XML document is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

The information model for a *Tutorial* XML document mirrors the information model for an *Assessment Item* XML document. Table 11 indicates which of the assessment item elements are used for a tutorial item.

- A ☑ indicates that the element SHALL be present in a tutorial. A *Tutorial* XML document that does not contain the element SHALL be NON CONFORMING.
- A ☒ indicates that the element SHALL NOT be present in a tutorial. A *Tutorial* XML document that contains the element SHALL be NON CONFORMING.
- A ☐ indicates that the element is OPTIONAL in a tutorial.
- The Notes describe additional behavior and conformance constraints.

Table 11: Assessment Item Elements Used in a Tutorial

Element	Req'd	Notes
item	☑	The value of the format attribute SHALL be tut.
associatedpassage	☐	Typically omitted. Ignored if present.
attriblist	☐	The *Tutorial Item* MAY use the same attributes as an *Assessment Item*.
tutorial	☒	A *Tutorial Item* SHALL NOT include a tutorial element.
resourcelist	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
statistic	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
MachineRubric	☐	If present the MachineRubric element SHALL be ignored.
RendererSpec	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
gridanswerspace	☒	A *Tutorial Item* SHALL NOT include a gridanswerspace element.
content	☑	
qti	☐	Typically omitted. Ignored if present.
rationaloptlist	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
illustration	☐	
stem	☑	
rubriclist	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
optionlist	☐	Typically an empty element in a *Tutorial Item*. Ignored if present.
attachmentlist	☐	
apipAccessibility	☐	

Tutorial Elements

The tutorial elements are identical to the assessment item elements. Additional constraints are as described in Table 11.

Refer to the assessment item elements for the full description of the elements.

The xHTML content within the stem element of a tutorial item often includes an href attribute of an anchor element to an animation or media file. The content will contain only a reference to a single file. Multiple language variant files or multiple format variant files MAY exist and be included with the element. How the test client selects the file to delivery is NOT SPECIFIED and is IMPLEMENTATION DEFINED. Specific rules used to name the media files are described in *Document Criteria*.

Wordlist XML Document Elements

A *Wordlist* XML document holds the content for a wordlist (thesaurus and glossary definitions) type of resource. A wordlist is used with an assessment item and the *Wordlist* XML document is referenced from the resource element within the assessment item.

The *Wordlist* XML document elements contain a list of thesaurus and multi-lingua glossary definitions. A single item element is the root element of the element tree. The item element MAY be embedded in an itemrelease element.

The item element contains a single keywordList element. The keywordList element contains multiple keywords. Each keyword includes the description of the keyword, potentially including multiple glosses in different languages. Note, there is no accessibility content for a keyword.

The element hierarchy within the *Wordlist* XML document elements is illustrated in Figure 6 (informative).

- The xHTML content of the html element is partially expanded in the diagram showing the Wordlist specific attributes.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.

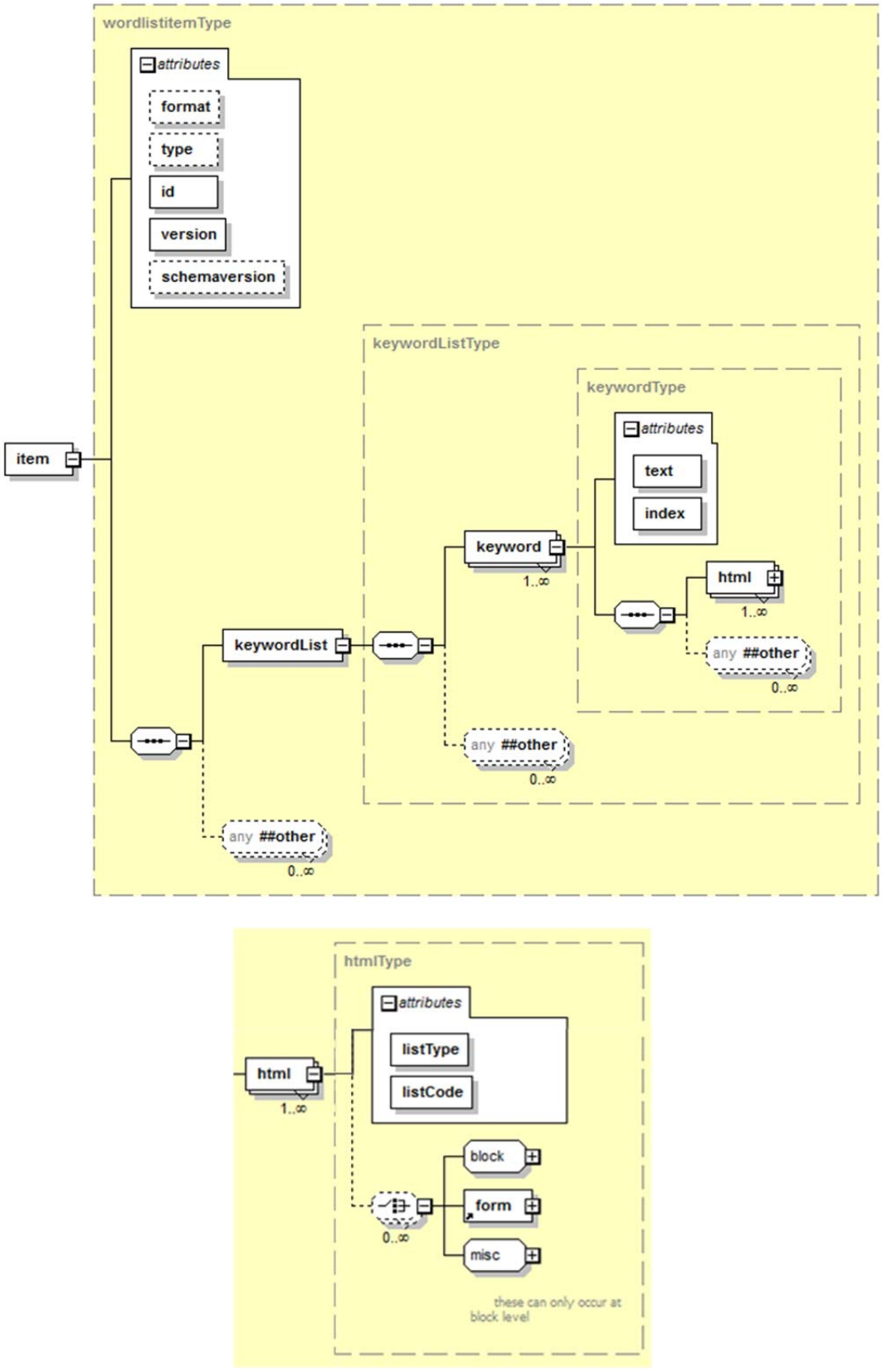


Figure 6: Wordlist XML Document Structure (Informative)

Wordlist Elements

Element	item			
Description	An item container for a *Wordlist* item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	keywordList	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	format	☐	xsd:token	
	type	☐ ☒	xsd:token	
	id	☑	xsd:int {>0}	
	version	☑	xsd:string {100}	
Extensions	☑			
Conformance	Either the format or type attribute SHALL be present.			
	An element that contains both the format and type attributes SHALL be NON CONFORMING.			
Notes	The type attribute is a candidate TO BE DEPRECATED and replaced by the format attribute. Until the type attribute IS DEPRECATED the type attribute SHOULD be used instead of using the format attribute.			
	How or when the *Wordlist* content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	item	
format	The type of the item. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	wordlist	Wordlist resource.
type	The type of the item. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	wordlist	Wordlist resource.
	The type attribute is a candidate TO BE DEPRECATED and replaced by the format attribute.	
id	Unique item number for the item/wordlist.	
	The value of the item number SHALL be unique within the context of all items.	
	The value of the item number SHALL be $< 2^{31}-1$.	
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
version	Version identifier for the item/wordlist.	
	The value SHOULD match the regular expression: \d+(\.\d+)?(\.\d+)?	

Element	keywordList			
Description	The list of keywords in a *Wordlist*.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	keyword	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			

Element	keywordList
Extensions	☑
Notes	

Element	keyword			
Description	A keyword in a *Wordlist*.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	html	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	text	☑	xsd:string {1000}	
	index	☑	xsd:int {>0}	
Extensions	☑			
Notes	The keyword elements do not need to have contiguous or ordered values for the index attribute.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	keyword
text	The text of the keyword.
index	A sort key used to order the keywords. The values need not be contiguous.

Element	html			
Description	The definition for the keyword. The listType and listCode attributes define the type of definition.			
Element Type	HTML {4000}			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	listType	☑	xsd:token	
	listCode	☑	xsd:token	
Extensions	☒			
Conformance	The keyword text SHOULD conform to [XHTML 1.1].			
	The value of the listType attribute and the value of the listCode attribute SHALL correspond one-to-one as shown in Table 12.			
	An element that contains a listType attribute and a listCode attribute that do not align as shown in Table 12 SHALL be NON CONFORMING.			
	Each listType/listCode attribute value pair SHALL appear only once per keyword. A keyword that repeats the listType/listCode attribute value pair SHALL be NON CONFORMING.			
	An element that includes an xml:lang attribute that does not correspond to the language of the listType/listCode attribute value pair SHALL be NON CONFORMING.			
Notes	The valid values for listType and listCode MAY be extended in a future version of the Specification.			
	Table 12 lists the language codes (used for xml:lang) for the listType/listCode attribute value pairs.			

Attributes	html
listType	The type of entry for the keyword.
	The value SHALL be one of the vocabulary values listed in Table 12.
	The value SHALL correspond to the associated value of the listType attribute as shown in Table 12.
listCode	The code of the type of entry for the keyword.
	The value SHALL be one of the vocabulary values listed in Table 12.
	The value SHALL correspond to the associated value of the listCode attribute as shown in Table 12.

Table 12: Glossary Entry Types and Codes

Keyword Entry Type (listType)	Keyword Entry Code (listCode)	xml:lang
glossary	TDS_WL_Glossary	
thesaurus	TDS_WL_THES	
arabicGlossary	TDS_WL_ArabicGloss	ar
cantoneseGlossary	TDS_WL_CantoneseGloss	zh-HK
esnGlossary	TDS_WL_ESNGlossary	es
koreanGlossary	TDS_WL_KoreanGloss	ko
mandarinGlossary	TDS_WL_MandarinGloss	zh, zh-CN, zh-TW
punjabiGlossary	TDS_WL_PunjabiGloss	pa
russianGlossary	TDS_WL_RussianGloss	ru
tagalGlossary	TDS_WL_TagalGloss	tl
ukrainianGlossary	TDS_WL_UkrainianGloss	uk
vietnameseGlossary	TDS_WL_VietnameseGloss	vi

Assessment Item Accessibility XML Document Elements

An *Assessment Item Accessibility* XML document holds accessibility content (e.g., Braille alternative content). Accessibility content is used with an assessment item (including a tutorial) or a passage item and is referenced from the apipAccessibility element within the assessment item or passage item.

The assessment item accessibility elements consist of a single root XML element, the apipAccessibility element. The element and its subelements describe the accessibility features of an assessment item. The accessibility elements appear inline within the content elements of an assessment item (including a tutorial) or a passage item.

The element hierarchy within the *Assessment Item Accessibility* XML document elements is illustrated in Figure 7 (informative).

- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.

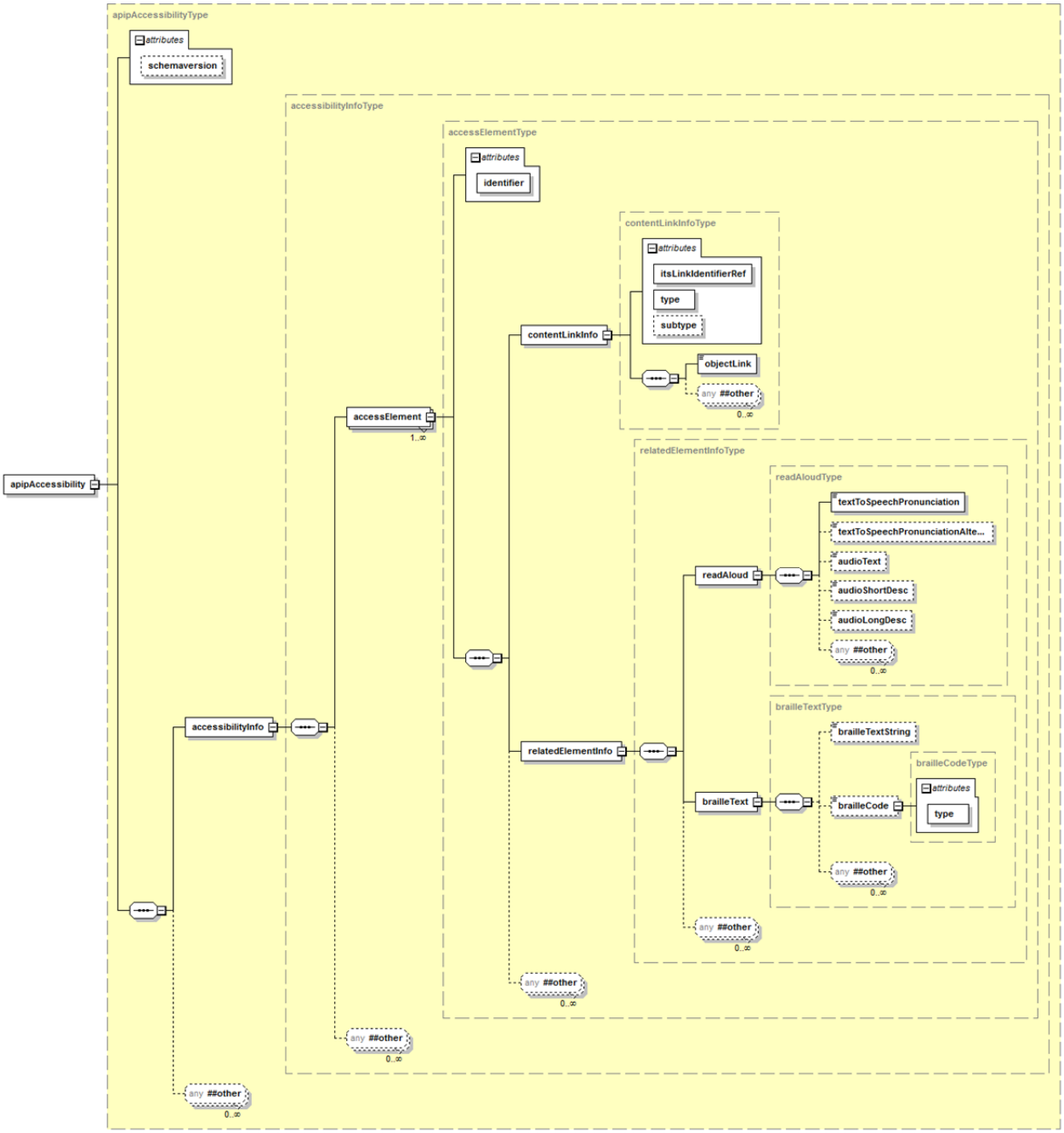


Figure 7: Assessment Item Accessibility XML Document Structure (Informative)

Accessibility Elements

Element	apipAccessibility			
Description	[APIP] accessibility information for the item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	accessibilityInfo	[1]		

Element	apipAccessibility			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	While the element name includes "APIP", the content does not conform to the [APIP] profile.			
	How or when the item accessibility content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Element	accessibilityInfo			
Description	Accessibility information for an item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	accessElement	[1..*]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes				

Element	accessElement			
Description	Accessibility information for an item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	contentLinkInfo	[1]		
	relatedElementInfo	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	identifier	☑	xsd:string {4000}	
Extensions	☑			
Notes				

Attributes	accessElement
identifier	Unique identifier for the accessibility information.
	The value of the identifier SHALL be unique within the context of all items.

Element	contentLinkInfo			
Description	Link to accessibility content.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	objectLink	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	itsLinkIdentifierRef	☑	xsd:string {4000}	
	type	☑	xsd:token	
	subtype	☐	xsd:token	
Extensions	☑			
Conformance	The subtype attribute SHALL be present if the type attribute has the value Graphic.			
	An element that does not contain a subtype attribute when the type attribute has the value Graphic SHALL be NON CONFORMING.			

Element	contentLinkInfo
Notes	The subtype attribute is ignored if the type attribute does not have the value Graphic.
	Additional values for the subtype attribute MAY be added in a future version of the Specification.

Attributes	contentLinkInfo	
itsLinkIdentifierRef	Reference to the id attribute of an xHTML element within the stem element of the item content. The link ties the accessibility content to the content of the item.	
type	The type of element for the accessibility content.	
	Value	**Description**
	Equation	The accessibility content is for an equation.
	Graphic	The accessibility content is for an image (<img>) element.
	Table	The accessibility content is for a table (<table>) element.
	Text	The accessibility content is for an HTML text element, typically an anchor (<a>), paragraph (<p>) or span () element.
subtype	Additional attribibute for specific values of the type attribute.	
	Value	**Description**
	Data	The table accessibility content is for table data. The attribute is present only if the value of the type attribute is Table.
	Formatting	The table accessibility content is for table formatting. The attribute is present only if the value of the type attribute is Table.
	Generic	The image accessibility content is not specialized. The attribute is present only if the value of the type attribute is Graphic.

Element	objectLink
Description	Link to the accessibility content.
Element Type	xsd:anyURI
Value	*Any*
Default	*None*
Extensions	☒
Notes	The element is REQUIRED but MAY be an empty element if there is no accessibility content.

Element	relatedElementInfo			
Description	Additional accessibility information.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	readAloud	[1]		
	brailleText	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			

Element	relatedElementInfo
Notes	

Element	readAloud			
Description	Pronunciation text for text-to-speech.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	textToSpeechPronunciation	[1]		
	textToSpeechPronunciationAlternate	[0..1]		
	audioText	[0..1]		
	audioShortDesc	[0..1]		
	audioLongDesc	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The accessibility profile will determine which of the alternatives will be used in which situation.			

Element	textToSpeechPronunciation
Description	String containing pronunciation directives.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	Phonetic spelling, if present, SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
	An element that contains phonetic spelling that is not valid IPA SHALL be NON CONFORMING.
Notes	The string MAY contain both plain text and phoneme spelling.

Element	textToSpeechPronunciationAlternative
Description	Alternative string containing pronunciation directives.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	Phonetic spelling, if present, SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
	An element that contains phonetic spelling that is not valid IPA SHALL be NON CONFORMING.
Notes	The string MAY contain both plain text and phoneme spelling.

Element	audioText
Description	String containing pronunciation text.
Element Type	xsd:string {16000}
Value	*Any*

Element	audioText
Default	*None*
Extensions	☒
Conformance	Phonetic spelling, if present, SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
	An element that contains phonetic spelling that is not valid IPA SHALL be NON CONFORMING.
Notes	The string MAY contain both plain text and phoneme spelling.

Element	audioShortDesc
Description	String containing pronunciation text.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	Phonetic spelling, if present, SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
	An element that contains phonetic spelling that is not valid IPA SHALL be NON CONFORMING.
Notes	The string MAY contain both plain text and phoneme spelling.

Element	audioLongDesc
Description	String containing pronunciation text.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	Phonetic spelling, if present, SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
	An element that contains phonetic spelling that is not valid IPA SHALL be NON CONFORMING.
Notes	The string MAY contain both plain text and phoneme spelling.

Element	brailleText			
Description	Braille text for Braille transcription.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	brailleTextString	[0..1]		
	brailleCode	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes				

Element	brailleTextString
Description	Modified text for Braille display or embossers.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Notes	Avoid lexical and structural elements that are known to cause issues with faithful transcription to Braille.

Element	brailleCode			
Description	Code for Braille display or embossers.			
Element Type	xsd:string {16000}			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	type	☑	xsd:token	
Extensions	☑			
Notes	Avoid lexical and structural elements that are known to cause issues with faithful transcription to Braille.			
	Braille code is typically used only for Braille nemeth for mathematics content. Other types of braille code are not used.			

Attributes	brailleCode	
type	The type of Braille code.	
	Value	**Description**
	Nemeth	Braille code is Braille nemeth [Nemeth] code.

Grid Item Rendering Specification XML Document Elements

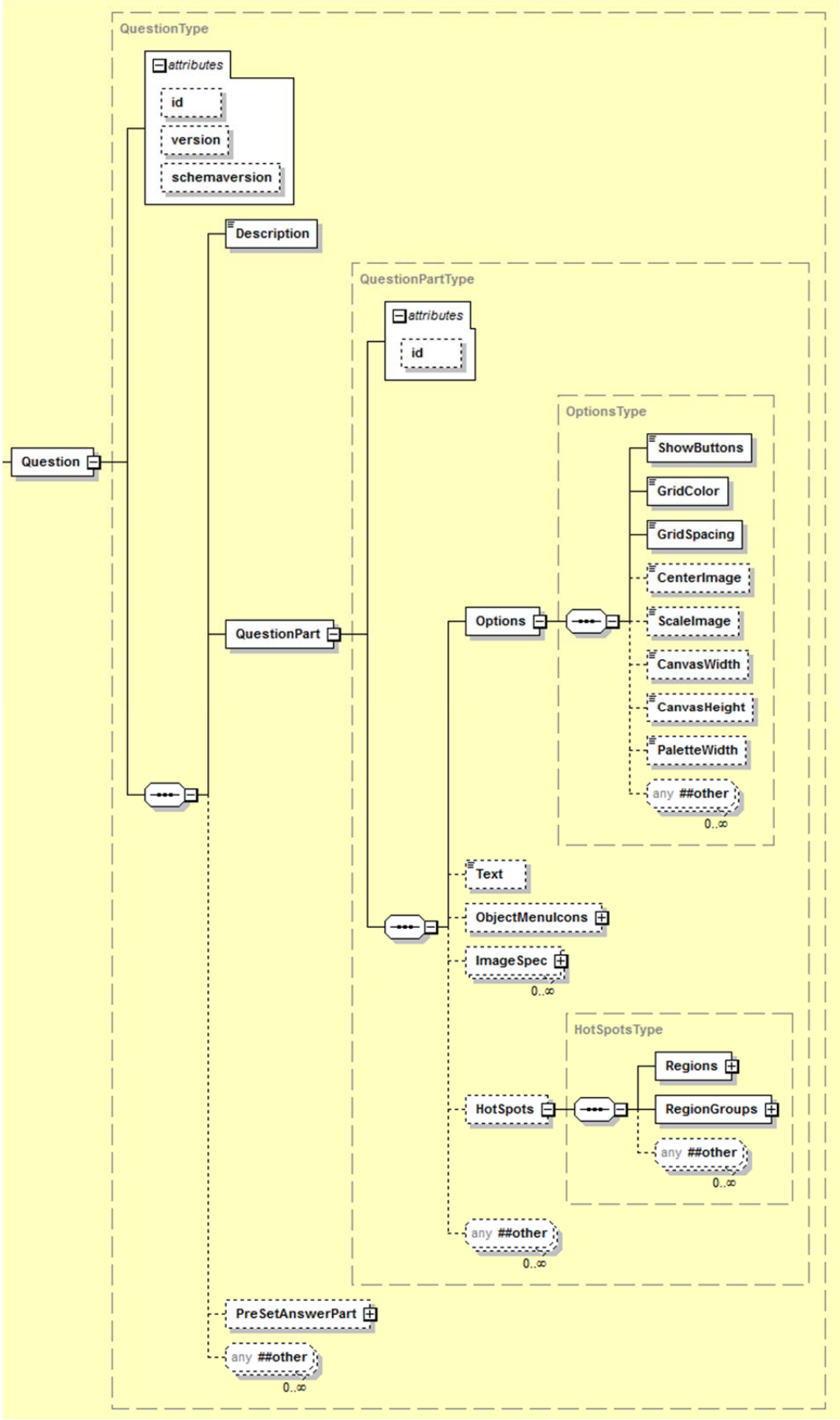
A *Grid Item Rendering Specification* XML document holds the rendering configuration settings for a grid type of assessment item. The grid item rendering specification is a type of rendering specification. The grid item rendering specification is referenced from the gridanswerspace element within a grid type of assessment item and is stored inline in the gridanswerspace element.

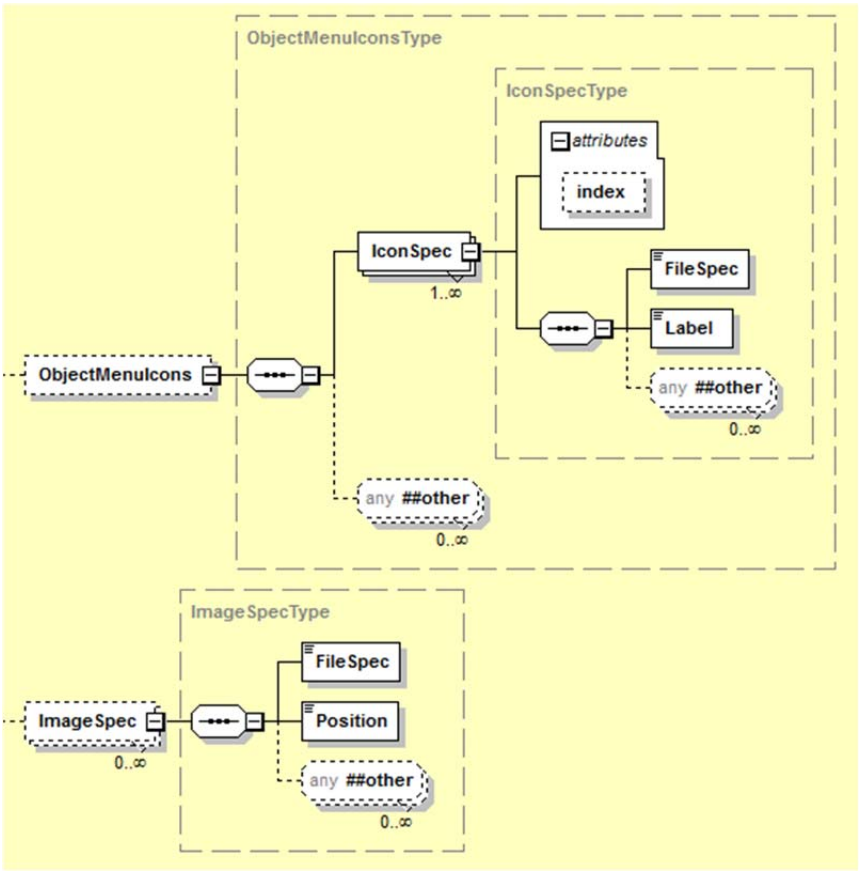
The XML elements for a *Grid Item Rendering Specification* XML document are detailed in three groups:

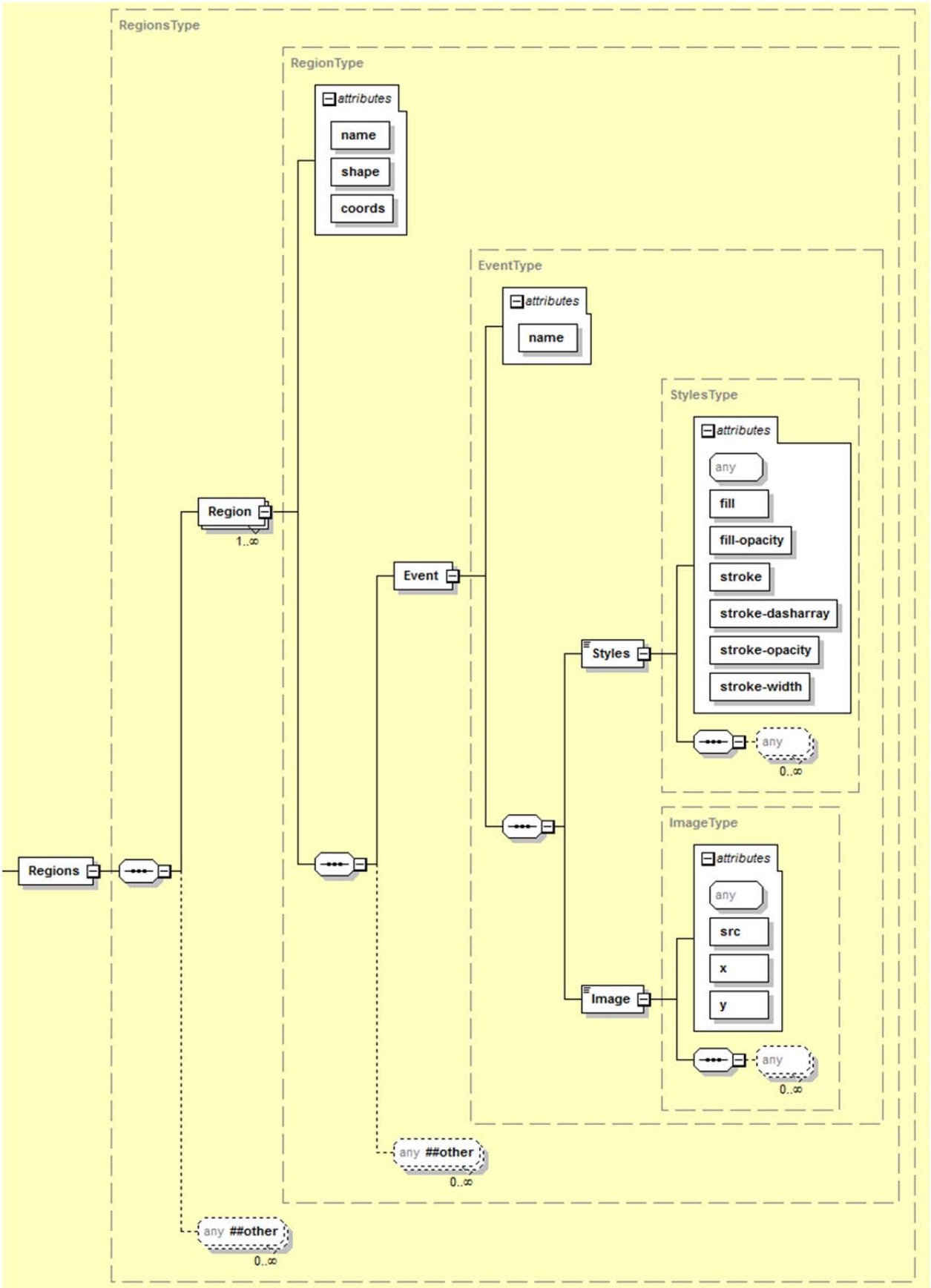
- *Grid Question elements* – the definition of elements used to describe the grid item rendering. A single Question element is the top-level element in the document. The Question element is embedded in a gridanswerspace element.
- *Preset Answer elements* – the definition of elements used to describe the initial or preset answer component on the display canvas. A single PreSetAnswerPart element is embedded as a top-level element in the Question element.
- *Shared elements* – the definition of simple, common XML elements that are subelements of various other elements (i.e., FileSpec, Label) and have common usage throughout the rendering specification. Shared elements with the same names are used in other XML documents. Their definition MAY be XML-document specific.

The element hierarchy within a *Grid Item Rendering Specification* XML document is illustrated in Figure 8 (informative).

- For clarity, the first part of the element hierarchy show at the top of the diagram does not expand the ObjectMenuIcons, ImageSpec, Regions, Regiongroups or PreSetAnswerPart elements.
- Each of these are expanded individually in other parts of the illustration.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.


QuestionType
attributes
id
version
schemaversion
Description
QuestionPartType
attributes
id
OptionsType
ShowButtons
GridColor
GridSpacing
CenterImage
ScaleImage
CanvasWidth
CanvasHeight
PaletteWidth
any ##other
0..∞
Question
QuestionPart
Options
Text
ObjectMenuIcons
ImageSpec
0..∞
HotSpotsType
Regions
HotSpots
RegionGroups
any ##other
0..∞
any ##other
0..∞
PreSetAnswerPart
any ##other
0..∞


ObjectMenuIconsType
IconSpecType
attributes
index
IconSpec
1..∞
FileSpec
Label
any ##other
0..∞
ObjectMenuIcons
any ##other
0..∞
ImageSpecType
FileSpec
ImageSpec
Position
0..∞
any ##other
0..∞


RegionsType
RegionType
attributes
name
shape
coords
EventType
attributes
name
StylesType
attributes
any
fill
fill-opacity
stroke
stroke-dasharray
stroke-opacity
stroke-width
any
0..∞
Region
1..∞
Event
Styles
ImageType
attributes
any
src
x
y
Image
any
0..∞
Regions
any ##other
0..∞
any ##other
0..∞

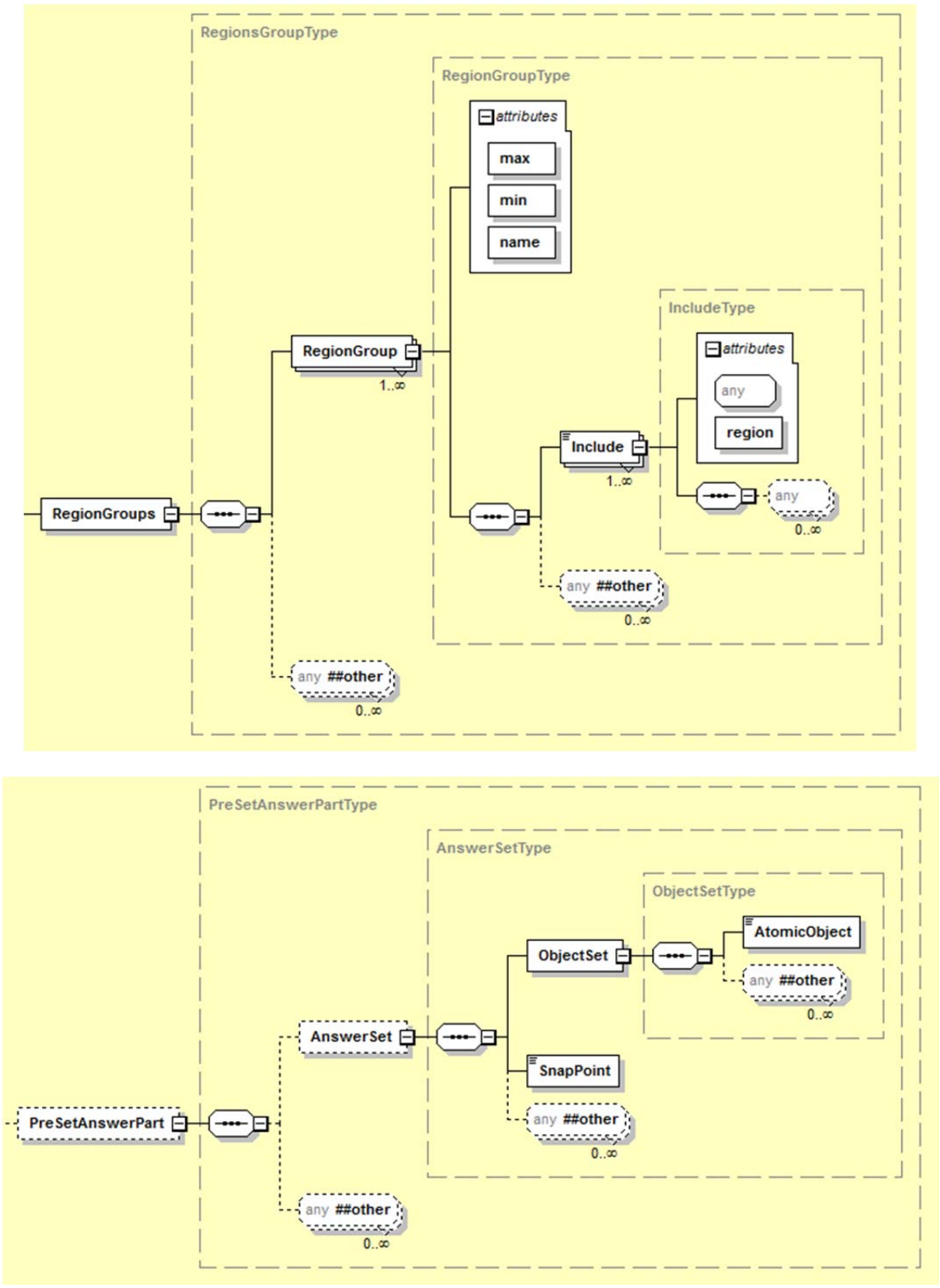


Figure 8: Grid Item Rendering Specification XML Document Structure (Informative)

The rendering space conventions are illustrated in Figure 9 (normative).

- Items are displayed on the canvas.
- The canvas has a defined size.
- The canvas may be covered by a set of grid lines, with equal spacing the in X and Y directions.
- The origin of display is the upper left-hand corner of the canvas.
- The preset answer data uses a coordinate system with an origin at the lower left-hand corner of the canvas.
- The grid layout starts at the upper left-hand corner. If the canvas size is not modulo the grid size, the right and bottom grid cells will have a different size.
- The palette of icons (not illustrated) that can be placed on the image is normally rendered vertically on the left side of the display canvas.
- The palette has a size; the size is the width if the palette is rendered vertically.
- The length of the palette accommodates the icons in a single column.
- Icons MAY be centered within the width of the palette, and scaled to fit.
- The toolbar (not illustrated) is normally displayed above the canvas.

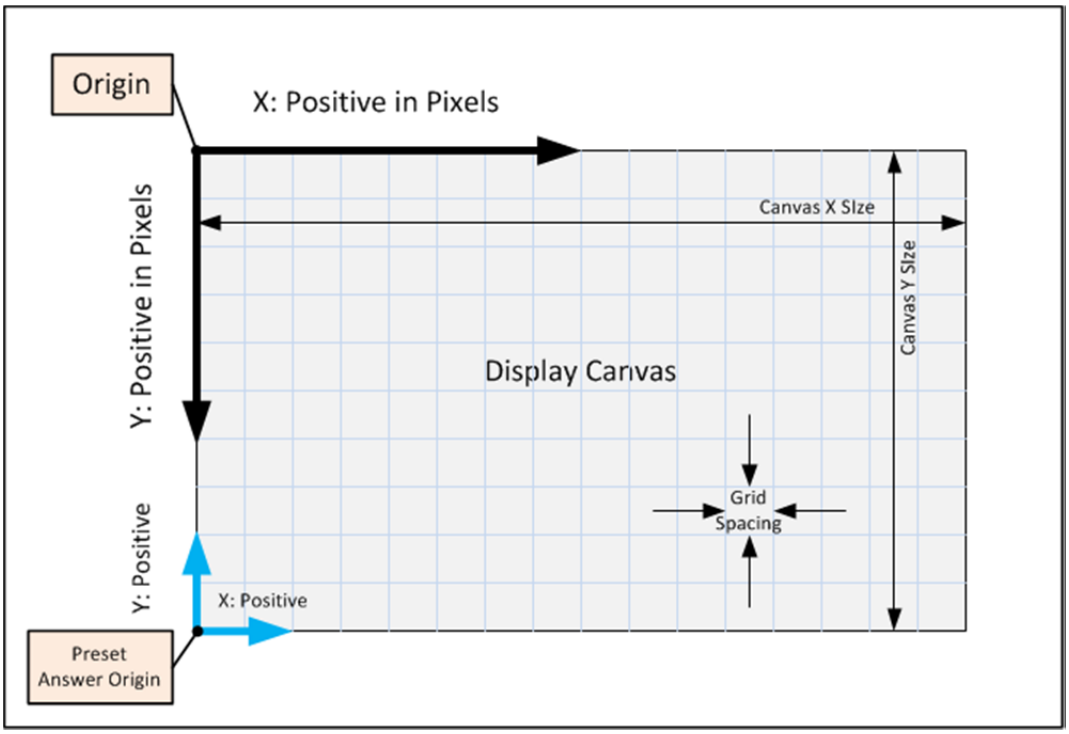


Figure 9: Grid Rendering Space Conventions (Normative)

Grid Question Elements

Element	Question			
Description	Rendering description for a graphic response item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Description	[1] ☒		
	QuestionPart	[1]		

Element	Question			
	PreSetAnswerPart	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	id	☐	xsd:int {>0}	
	version	☑ ☒	xsd:string {100}	
Extensions	☑			
Conformance	The value of the id attribute SHALL match the value of the id attribute of the item.			
	The behavior if the element contains a value for the id attribute that does not match the value of the id attribute of the item SHALL be NON CONFORMING.			
	The value of the version attribute SHOULD match the value of the version attribute of the itemrelease element in the *Assessment Item Release* XML container document.			
	The behavior if the value of the version attribute does not match the value of the version attribute of the itemrelease element is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
Notes				

Attributes	Question
id	Unique item number for the item.
	The value SHALL match the id of the item.
	The value of the item number SHALL be < 2^{31}-1.
	How a producing system insures uniqueness or the behavior of a consuming system when different items have the same id is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
version	The version identifier for the content of the item release.
	There are no constraints on the value of the attribute.
	A value is REQUIRED but not used. Any non null string MAY be used.
	The version attribute is a candidate TO BE DEPRECATED and removed. The attribute is REQUIRED.

Element	Description
Description	Human readable description of the question.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	The content is metadata and is only used in item development. The element is REQUIRED but not used.
	The Description element is a candidate TO BE DEPRECATED and removed

Element	QuestionPart			
Description	Rendering description for a graphic response item.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Options	[1]		
	Text	[0..1] ☒		
	ObjectMenuIcons	[0..1]		

Element	QuestionPart			
	ImageSpec	[0..*]		
	HotSpots	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	id	☑	xsd:string {4000}	
Extensions	☑			
Notes	The text element is a candidate TO BE DEPRECATED and removed.			

Attributes	QuestionPart
id	Unique identifier for the question part.
	The value of the identifier SHALL be unique within the context of the individual items.

Element	Options			
Description	Options that control grid item appearance.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	ShowButtons	[1]		
	GridColor	[1]		
	GridSpacing	[1]		
	CenterImage	[0:1]		
	ScaleImage	[0:1]		
	CanvasWidth	[0:1]		
	CanvasHeight	[0:1]		
	PaletteWidth	[0:1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes				

Element	ShowButtons	
Description	List of buttons to be displayed on the toolbar.	
Element Type	xsd:string {4000}	
Value	The value SHALL be zero or more of the vocabulary values listed.	
	Value	Description
	arrow	Add an arrow between points, pointing forward.
	arrw2	Add a double headed arrow between points.
	circle	Add a circle at the point.
	connect	Add a line between points.
	dash	Add a dashed line between points.
	delete	Delete an object from the canvas.
	move	Move an object on the canvas.
	point	Add a point to the canvas.
	The value MAY BE empty.	
	If the value is not empty, the value SHALL match the regular expression (arrow|arrw2|circle|connect|dash|delete|move|point)(,(arrow|arrw2|circle|connect|dash|delete|move|point))*	
Default	*None*	

Element	ShowButtons
Extensions	☒
Notes	The value is a comma separated set of button names. Any of the button names may appear in any order. Valid button names are: move, delete, point, connect, arrow, arrw2, dash, circle
	If the value is empty, the toolbar is hidden and only the move option is available.
	Move is a *modal* behavior – move may be selected or unselected. Once move is selected, any object on the canvas can be selected and moved until move is unselected.

Element	GridColor	
Description	Color of grid lines.	
Element Type	xsd:token – A vocabulary of values.	
Value	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	LightBlue	Grid lines SHALL be displayed in LightBlue, RGB Hex code #ADD8E6
	None	Grid lines SHALL NOT be displayed.
Default	*None*	
Extensions	☑	
Notes	The width of the grid line is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	
	The gridline color is not altered for accessibility.	
	A valid value is the character literal None. This is not the metavalue *None* (in italics) used in the Specification when a value does not exist, i.e., the default value is not None – there is no default value.	

Element	GridSpacing
Description	Spacing between grid lines, in pixels.
Element Type	xsd:string {100}
Value	The value SHALL match the regular expression \d+(,(Y|N))?
Default	*None*
Extensions	☒
Notes	The grid spacing is the same for both the X and Y directions.
	The value is an integer for the grid spacing, followed by an OPTIONAL comma (, , ,) and the character Y or N.
	If the character Y is present, display objects SHALL snap to the grid.
	If the character Y is not present, display objects SHALL NOT snap to the grid.

Element	CenterImage	
Description	Controls centering of the icon images on the palette.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	Images SHALL be centered on the palette.
	false	There is no constraint on image placement.
Default	*None*	
Extensions	☒	

Element	CenterImage
Notes	The palette is normally vertical and placed to the left of the canvas. If the value is true, the icon image is placed on the palette and centered horizontally on the palette.
	If the element is omitted, the test client SHALL behave as if the value is false.

Element	ScaleImage	
Description	Controls scaling of the icon image to fit the palette.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	The image SHALL be scaled to fit the palette.
	false	There is no constraint on image scaling.
Default	*None*	
Extensions	☒	
Notes	The palette is normally vertical and placed to the left of the canvas. If the value is true, the icon image is placed on the palette and scaled to fit within the width of the palette.	
	If the element is omitted, the test client SHALL behave as if the value is false.	

Element	CanvasWidth
Description	Controls the width of the canvas.
Element Type	xsd:int {>0}
Default	*None*
Extensions	☒
Notes	The maximum size of the canvas is NOT SPECIFIED and the behavior if the CanvasWidth exceeds the maximum size is IMPLEMENTATION DEPENDENT.
	If the element is omitted, the test client SHALL behave as if the value is 500 pixels.

Element	CanvasHeight
Description	Controls the height of the canvas.
Element Type	xsd:int {>0}
Default	*None*
Extensions	☒
Notes	The maximum size of the canvas is NOT SPECIFIED and the behavior if the CanvasHeight exceeds the maximum size is IMPLEMENTATION DEPENDENT.
	If the element is omitted, the test client SHALL behave as if the value is 410 pixels.

Element	PaletteWidth
Description	Controls the width of the icon palette.
Element Type	x xsd:int {>0}
Default	*None*
Extensions	☒
Notes	The maximum size of the palette is NOT SPECIFIED and the behavior if the PaletteWidth exceeds the maximum size is IMPLEMENTATION DEPENDENT.

Element	PaletteWidth
	If the element is omitted, the test client SHALL behave as if the value is 75 pixels.

Element	Text
Description	Text on the canvas.
Element Type	xsd:string {100}
Value	*Any*
Default	*None*
Extensions	☒
Notes	The element SHOULD be empty. If a value is present, it is not used.
	The Text element is a candidate TO BE DEPRECATED and removed.

Element	ObjectMenuIcons			
Description	Icon images to appear on the palette.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	IconSpec	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The icon images are placed on the palette in the order specified.			

Element	IconSpec			
Description	Image to be rendered on the palette.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	FileSpec	[1]		
	Label	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	index	☐	xsd:int {>0}	
Extensions	☑			
Notes	The file location and naming convention for the image file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	There are no constraints on an icon.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	IconSpec
index	The display order of the icons on the palette. The values need not be contiguous.

Element	ImageSpec			
Description	Background graphic to be rendered on the canvas.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	FileSpec	[1]		

Element	ImageSpec			
	Position	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The file location and naming convention for the image file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			

Element	Position
Description	Background image position on the canvas.
Element Type	xsd:string {100}
Value	Pair of comma delimited coordinates (X and Y).
	The value SHALL match the regular expression: \d+,\d+
Default	*None*
Extensions	☒
Conformance	The value of the x coordinate of the position SHALL NOT exceed the width of the canvas.
	The value of the y coordinate of the position SHALL NOT exceed the height of the canvas.
Notes	The value is a nonnegative integer for the *x* position, followed by a comma (, , ,) and a nonnegative integer for the *y* position.
	The origin of the coordinate system is the upper left hand corner of the canvas. The position places the background image relative to origin of the canvas. X and Y denote the position of the upper left hand corner of the background image relative to the origin of the canvas. X is positive to the left. Y is positive downward.
	The behavior if the position places all or part of the image outside of the canvas is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Element	SnapPoint
Description	Points on the display canvas that exhibit snap behavior.
Element Type	xsd:string {4000}
Value	The value SHALL match the regular expression \d+@(\d+,\d+)(;\d+,\d+)*
Value	*None*
Extensions	☒
Conformance	The value of the x coordinate of the snap point SHALL NOT exceed the width of the canvas.
	The value of the y coordinate of the snap point SHALL NOT exceed the height of the canvas.
Notes	The value is the snap radius (pixels) followed by the at character (@ @ @) followed by a list of coordinate pairs. The coordinate pairs are delimited by a semi colon (; ; ;). The coordinate pair consists of an x and y coordinate, each a non negative integer values, delimited by a comma (, , ,).
	The behavior if all or part of a snap point is outside of the canvas is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
	Both grid snap and individual snap points MAY be specified.

Element	HotSpots			
Description	Canvas areas that responds to mouse events.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Regions	[1]		
	RegionGroups	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes				

Element	Regions			
Description	A set of hotspot areas.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Region	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	A region is a single hotspot.			

Element	Region			
Description	Shape and position of a hotspot.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Event	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	name	☑	xsd:string {4000}	
	shape	☑	xsd:token	
	coords	☑	xsd:string {4000}	
Extensions	☑			
Conformance	The value of any x coordinate of the shape SHALL NOT exceed the width of the canvas.			
	The value of any y coordinate of the shape SHALL NOT exceed the height of the canvas.			
Notes	The shape and coords attributes are patterned after the shape and coords attributes of the HTML area element.			
	The coords regular expression is a subset of the complete pattern defined in the xHTML schema: [-+]?(\d+|\d+(\.\d+)?%)(,\s*[-+]?(\d+|\d+(\.\d+)?%))*			

Attributes	Region	
name	Name of the region or hotspot.	
	The name MUST be unique within the context of the item.	
shape	Shape of the region. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	rect	Rectangular region.
	circle	Circular region.

Attributes	Region	
	poly	Polygon region.
coords	List of coordinate values or dimensions that define the region.	
	Shape	**Description**
	rect	The coordinate pairs of the corners of the rectangle. 4 values: x coordinate of the upper left-hand corner, y coordinate of the upper left-hand corner, x coordinate of the lower right-hand corner, y coordinate of the lower right-hand corner. The values SHALL be nonnegative. The values SHALL NOT exceed the size of the canvas. The lower right-hand corner values SHALL NOT be less than the upper left-hand corner values.
		The value SHALL match the regular expression: \d+,\d+,\d+,\d+
	circle	The coordinates pair of the origin plus the radius. 3 values: x coordinate of the origin of the circle, y coordinate of the origin of the circle, radius of the circle. The values SHALL be nonnegative. The values SHALL NOT exceed the size of the canvas. The values SHALL NOT cause any part of the circle to fall outside of the canvas.
		The value SHALL match the regular expression: \d+,\d+,\d+
	poly	An arbitrarily long ordered list of coordinate pairs of the vertices of the polygon, each pair consisting of the x coordinate of a vertex of the polygon and the y coordinate of the vertex of the polygon. The values SHALL be nonnegative. The values SHALL NOT exceed the size of the canvas. There SHALL be at least 3 pairs of values. The maximum number of pair is NOT SPECIFIED and is IMPLEMENTATION DEFINED. The behavior if the number of pairs exceeds the maximum number of pairs is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
		The value SHALL match the regular expression: \d+,\d+,\d+,\d+,\d+,\d+(,\d+,\d+)*

Element	Event			
Description	Visual effects applied to the region when a specific mouse event occurs.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Styles	[1]		
	Image	[1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	name	☑	xsd:token	
Extensions	☑			
Notes	When the specified event (defined by the value of the name attribute) occurs in the region, the style and image are applied to the region.			

Attributes	Event	
name	Type of the event. A vocabulary of values.	
	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	select	Region gets user focus.
	unselect	Region loses focus.
	hover	Mouse is dwelling over the region.
	The minimum time for the input cursor to be stationary to consider the event to be *dwell* is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	

Element	Styles			
Description	Display attributes applied to a region when an event occurs.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	fill	☑	xsd:string {100}	
	fill-opacity	☑	xsd:float	
	stroke	☑	xsd:string {100}	
	stroke-dasharray	☑	xsd:string {1000}	
	stroke-opacity	☑	xsd:float	
	stroke-width	☑	xsd:int {>0}	
Extensions	☒			
Conformance	Fill and stroke colors SHALL conform to [SVG 1.1] and [CSS 2].			
	An element that contains fill and stroke colors that is not valid [SVG 1.1] and [CSS 2] SHALL be NON CONFORMING.			
Notes	Colors MAY be expressed as 6 hex digits, 3 hex digits or a CSS color name.			

Attributes	Styles
fill	The interior fill color of the region.
	The value SHALL match the regular expression #([0-9a-fA-F]{3}|[0-9a-fA-F]{6}) or it SHALL match one of the 140 named CSS colors.
fill-opacity	The opacity of the region.
	The value SHALL be between 0.0 and 1.0, inclusive. 0.0 is transparent. 1.0 is opaque.
	The opacity applies only to the interior of the region. It does not apply to the outline of the region.
stroke	Line color for all lines in the region, including the border.
	The value SHALL match the regular expression #([0-9a-fA-F]{3}|[0-9a-fA-F]{6}) or it SHALL match one of the 140 named CSS colors.
stroke-dasharray	Comma delimited list of integers describing the stroke and space pattern of the dash pattern of lines including the border in pixels. A comma delimited list of pairs of integers describing the stroke and space pattern of a dashed line. The first number of each pair is the length of the stroke in pixels. The second number of each pair is the length of the space in pixels.
	The pattern is repeated as needed, including partial lengths of stroke or space.
	The value SHALL match the regular expression (\d+,\d+(,\d+,\d+)*)?
	The value MAY be empty.

Attributes	Styles
stoke-opacity	The opacity of the outline of region.
	The value SHALL be between 0.0 and 1.0, inclusive. 0.0 is transparent. 1.0 is opaque.
	The value MAY be empty.
stoke-width	The width of the outline of the region in pixels.

Element	Image			
Description	Graphic displayed in the region when an event occurs.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	src	☑ ☒	xsd:string {4000}	
	filename	☒	xsd:string {4000}	
	x	☑	xsd:int {≥0}	
	y	☑	xsd:int {≥0}	
Extensions	☑			
Conformance	The value of x SHALL NOT exceed the width of the canvas.			
	The value of y SHALL NOT exceed the height of the canvas.			
Notes	The file location and naming convention for the image file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].			
	The behavior if all or part of the image is outside of the canvas is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.			

Attributes	Image		
src	Link (filename) for an image.		
	The file location and naming convention for the image file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].		
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.		
	The test client SHALL support the following media types:		
	Name	**Media Type**	**Default File Extension**
	GIF	image/gif	.gif
	JPEG	image/jpeg	.jpg
	PNG	image/png	.png
	The src attribute is a candidate TO BE DEPRECATED and replaced by a filename attribute for consistency with other elements and attributes. The attribute is REQUIRED.		
filename	Link (filename) for an image.		
	The attribute name is reserved and SHALL NOT be used.		
x	X coordinate where the origin of the image is placed on the canvas.		
y	Y coordinate where the origin of the image is placed on the canvas.		

Element	RegionGroups			
Description	A set of groups of hotspot areas.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	RegionGroup	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	A region group is a collection of hotspots.			

Element	RegionGroup			
Description	Regions processed as a group.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Include	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	max	☑	xsd:int {≥0}	
	min	☑	xsd:int {≥0}	
	name	☑	xsd:string {4000}	
Extensions	☑			
Notes	The region group collects a set of hotspot. When multiple hotspots within the region group are selected at the same time (using standard user interface controls for selecting multiple items), the event associated with the group is triggered and the visual effect for the event is applied to all of the hotspots in the group.			

Attributes	RegionGroup
max	The maximum number of regions in the group that can be selected to trigger the event.
	The value SHALL NOT exceed the number of regions in the group.
	A value of 0 implies that no regions are required to be selected.
min	The minimum number of regions in the group that MUST be selected to trigger the event.
	The value SHALL NOT exceed max.
	A value of 0 implies that no regions are required to be selected.
name	The name of the region group.
	The name MUST be unique within the context of the item.

Element	Include			
Description	A reference to a region included in the region group.			
Element Type	*Empty*			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	region	☑	xsd:string {4000}	
Extensions	☒			
Conformance	The value for the region attribute SHALL match a name attribute of a Region element for the item.			

Element	Include
	An element that contains a value for the region attribute that does not match a name attribute of a region element for the item SHALL be NON CONFORMING.
Notes	The include element identifies the hotspot in the region group.

Attributes	Include
region	Name of the region included in the region group.
	The name SHALL match the name attribute of a Region element for the item.

Preset Answer Elements

Element	PreSetAnswerPart			
Description	Answer components included in the original question rendering.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	AnswerSet	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The element MAY be empty.			

Element	AnswerSet			
Description	Response to an item included in the original question rendering.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	ObjectSet	[1]		
	SnapPoint	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes				

Element	ObjectSet			
Description	Object on the canvas.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	AtomicObject	[0..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The element MAY be empty.			

Element	AtomicObject
Description	An object on the canvas.
Element Type	xsd:string {4000}
	The value is {objectname(xposition,yposition)}

Element	AtomicObject			
	The value SHALL match the regular expression \{.+\(\d+,\d+\)\}$			
Elements	**Name**	**Multiplicity**		
	None			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	The value of objectname in the string SHALL match the value of a Label element of an IconSpec element.			
	A value of objectname in the string that does not match the value of a Label element of an IconSpec element SHALL be NON CONFORMING.			
Notes	The Y coordinate is measured upward from the lower left hand corner of the canvas. This is not the same coordinate system used to place other objects on the canvas.			
	This alternative coordinate system is only used for placing objects and in scoring. It is not used for any other purpose.			
	This alternative coordinate system does not apply to snap points.			

Element	SnapPoint
Description	Points on the object that exhibit snap behavior.
Element Type	xsd:string {4000}
Value	The value SHALL match the regular expression \d+@(\d+,\d+)(;\d+,\d+)*
Value	*None*
Extensions	☒
Notes	The value is the snap radius (pixels) followed by the at character (@ @ @) followed by a list of coordinate pairs. The coordinate pairs are delimited by a semi colon (; ; ;). The coordinate pair consists of an x and y coordinate, each a non negative integer values, delimited by a comma (, , ,).
	The element MAY be empty.
	The behavior if all or part of a snap point is outside of the canvas is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
	Both grid snap and individual snap points MAY be specified.
	The Y coordinate is measured downward from the lower left hand corner of the canvas.

Shared Elements

Element	FileSpec
Description	Link (filename) for an image.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Notes	The file location and naming convention for the image file are NOT SPECIFIED and are IMPLEMENTATION DEPENDENT. Details MAY be provided in an item packaging profile, e.g., [SBAC Packaging 1.4].

Element	FileSpec		
	The behavior if the file does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.		
	The test client SHALL support the following media types:		
	Name	**Media Type**	**Default File Extension**
	GIF	image/gif	.gif
	JPEG	image/jpeg	.jpg
	PNG	image/png	.png

Element	Label
Description	Label for an image.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	The label is used as an identifier for the image in the rubric.

Equation Editor Configuration XML Document Elements

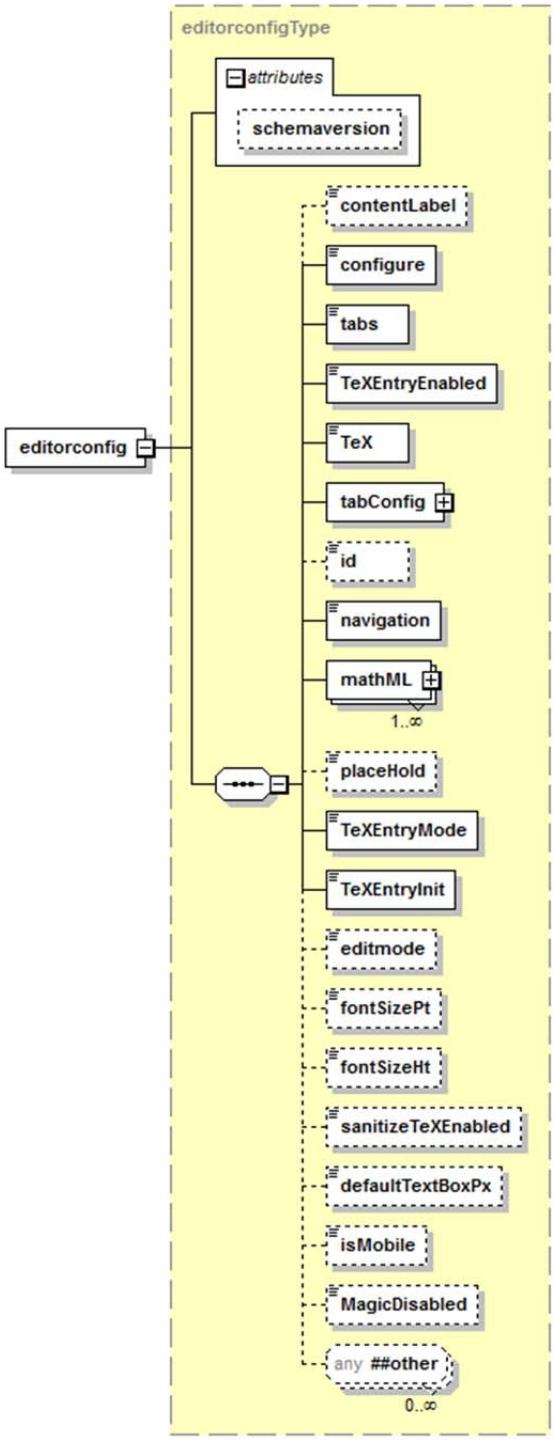
An *Equation Editor Configuration* XML document holds the configuration settings for the equation editor that is presented to the student with an equation type of assessment item. The equation editor configuration is a type of rendering specification. The equation editor configuration is referenced from the RendererSpec element within an equation type of assessment item.

The XML elements for an equation editor configuration document are detailed in three groups:

- *Equation Editor Configuration elements* – the definition of elements used to describe the equation editor configuration. A single editorconfig element is the root element of the element tree.
- *Input Keys Panel Tab Configuration elements* – the definition of elements used to describe the configuration and layout of input keys panel tabs; each tab containing multiple input *keys* or symbols that the user can select to enter an equation. The top-level elements that detail the parts of a tab all share a common set of elements.
- *MathML elements* – the definition of elements used to describe MathML content in the equation editor.

The element hierarchy within an *Equation Editor Configuration* XML document is illustrated in Figure 10 (informative).

- For clarity, the first part of the element hierarchy show at the top of the diagram does not expand the tabConfig element. It is expanded in the lower part of the diagram.
- Within the tabConfig element, only the SBAC6 element is expanded. The expansion of the Algebra, Basic, SBAC3, SBAC4, SBAC5, SBAC7, SBAC8, SBAC9, SBAC10, and SBAC11 elements are identical to the expansion of the SBAC6 element.
- The diagrams are generated from the sample XML Schemata. Alternative schemata would produce alternative schemata structure and diagrams.


editorconfigType
attributes
schemaversion
editorconfig
contentLabel
configure
tabs
TeXEntryEnabled
TeX
tabConfig
id
navigation
mathML
1..∞
placeHold
TeXEntryMode
TeXEntryInit
editmode
fontSizePt
fontSizeHt
sanitizeTeXEnabled
defaultTextBoxPx
isMobile
MagicDisabled
any ##other
0..∞

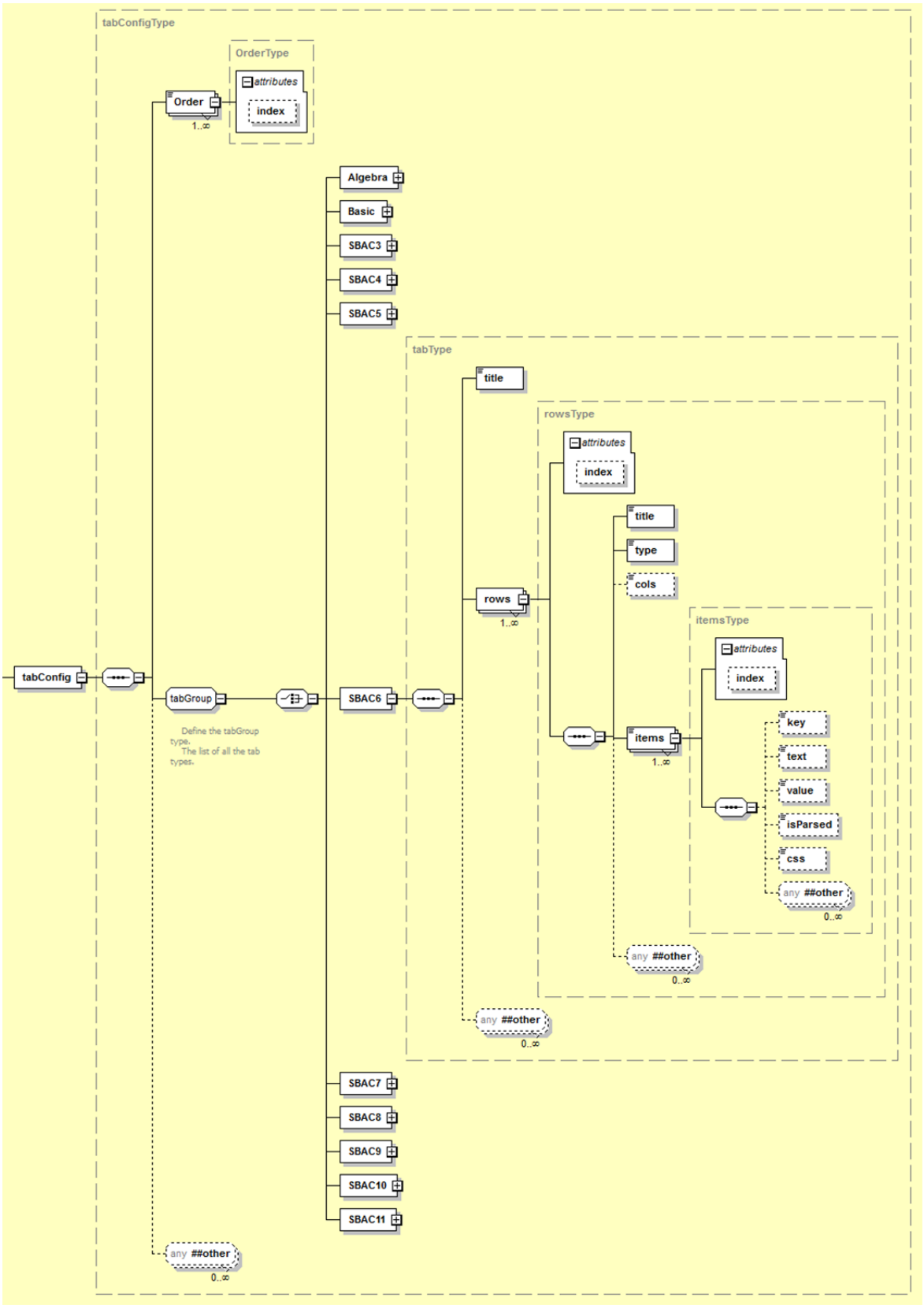


Figure 10: Equation Editor Configuration XML Document Structure (Informative)

Equation Editor Configuration Elements

Element	editorconfig			
Description	Description of the settings used to configure and render the equation editor in the test client.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	contentLabel	[0..1]		
	configure	[1]		
	tabs	[1]		
	TeXEntryEnabled	[1]		
	TeX	[1]		
	tabConfig	[1]		
	id	[0..1] ☒		
	navigation	[1]		
	mathML	[1..*]		
	placeHold	[0..1] ☒		
	TeXEntryMode	[1]		
	TeXEntryInit	[1]		
	editMode	[0..1] ☒		
	fontSizePt	[0..1] ☒		
	fontSizeHt	[0..1] ☒		
	sanitizeTeXEnabled	[0..1]		
	defaultTextBoxPx	[0..1]		
	isMobile	[0..1]		
	MagicDisabled	[0..1] ☒		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	The id element is OPTIONAL. It is a candidate TO BE DEPRECATED and removed. The element MAY be used.			
	The fontSizePt element is OPTIONAL. It is a candidate TO BE DEPRECATED and removed. The element MAY be used.			
	The fontSizeHt element is OPTIONAL. It is a candidate TO BE DEPRECATED and removed. The element MAY be used.			

Element	contentLabel
Description	Text to be displayed above the equation editor input box.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	

Element	configure	
Description	Control of item configuration data display when the item is viewed in the authoring tool or the test client.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	The item configuration data SHALL be displayed.
	false	The item configuration data SHALL NOT be displayed.
Default	*None*	
Extensions	☒	
Notes	A value of true is ignored by the test client. Configuration data is not displayed by the test client.	

Element	tabs	
Description	Control of display of the input keys panel tabs.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	Input keys panel tabs SHALL be displayed.
	false	Input keys panel tabs SHALL NOT be displayed.
Default	*None*	
Extensions	☒	
Notes	Equation items can have multiple input key panel tabs. Each tab displays a set of input keys organized in rows and columns.	

Element	TeXEntryEnabled	
Description	Control of permitted [TeX] entry into the equation editor.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	TeX input is permitted.
	false	TeX input is not permitted.
Default	*None*	
Extensions	☒	
Notes		

Element	TeX	
Description	TeX element to be displayed in the input box.	
Element Type	xsd:string {16000}	
Value	*Any*	
Default	*None*	
Extensions	☒	
Conformance	The value SHALL be valid [TeX] content.	
	A value that is not valid [TeX] content SHALL be NON CONFORMING.	
Notes		

Element	id
Description	Identifier of the equation editor item.
Element Type	xsd:int {>0}

Element	id
Value	*Any*
Default	*None*
Extensions	☒
Notes	The element is not used by the test client. If present, the element is ignored.
	The id element is a candidate TO BE DEPRECATED and removed.

Element	navigation	
Description	Controls the display of the equation input cursor navigation buttons.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	The navigation buttons SHALL be displayed.
	false	The navigation buttons SHALL NOT be displayed.
Default	*None*	
Extensions	☒	
Notes		

Element	placeHold
Description	TeX content to be used as a placeholder for user input to equations elements.
Element Type	xsd:string {16000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The value SHALL be valid [TeX] content.
	A value that is not valid [TeX] content SHALL be NON CONFORMING.
Notes	The element is not used. If present, the element is ignored.
	The placeHold element is a candidate TO BE DEPRECATED and removed.

Element	TeXEntryMode	
Description	User entered [TeX] input allowed.	
Element Type	xsd:token	
Value	**Value**	**Description**
	Allow	Direct TeX input SHALL be permitted.
	None	Direct TeX input SHALL NOT be permitted. Only input keys MAY be used.
Default	*None*	
Extensions	☒	
Notes	A valid value is the character literal None. This is not the metavalue *None* (in italics) used in the Specification when a value does not exist, i.e., the default value is not None – there is no default value.	

Element	TeXEntryInit	
Description	Initial [TeX] entry mode.	
Element Type	xsd:token	
Value	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**

Element	TeXEntryInit	
	Allow	Direct TeX input permitted.
	None	Direct TeX input not permitted. Only input keys may be used.
	Vim	VI Editor commands [Vim] may be used.
Default	*None*	
Extensions	☒	
Notes	A valid value is the character literal None. This is not the metavalue *None* (in italics) used in the Specification when a value does not exist, i.e., the default value is not None – there is no default value.	

Element	editMode	
Description	Supported editing mode.	
Element Type	xsd:token	
Value	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	APPEND	Append input to the end of the preset equation text. The preset equation text is not changed.
	INSERT	Insert text. Ignore any text selection.
	REPLACE	Replace selected text with input.
Default	*None*	
Extensions	☒	
Notes	The element is not used. If present, the element is ignored.	
	The editMode element is a candidate TO BE DEPRECATED and removed.	
	The default behavior of the test client allows inserting of text.	

Element	fontSizePt
Description	Font size in points.
Element Type	xsd:double, minExclusive=0
Value	*Any*
Default	*None*
Extensions	☒
Notes	The element is not used. If present, the element is ignored.
	The fontSizePt element is a candidate TO BE DEPRECATED and removed.

Element	fontSizeHt
Description	Font size.
Element Type	xsd:float, minExclusive=0
Value	*Any*
Default	*None*
Extensions	☒
Notes	The element is not used. If present, the element is ignored.
	The fontSizeHt element is a candidate TO BE DEPRECATED and removed.

Element	sanitizeTeXEnabled	
Description	Controls the addition of placeholder elements when certain symbols are entered.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	Additional placeholder elements are added when certain symbols are entered.
	false	Additional placeholder elements are not added when certain symbols are entered.
Default	*None*	
Extensions	☒	
Notes	The control is applied if the item element contains mixed content and the value of the content defines a fraction (fraction), absolute value (| |), parenthesis, square root (sqrt) or n^{th} root (nrt).	
	A placeholder can be added to any key that requires a placeholder by specifying the details of the key and placeholder in the value subelement of the item element.	
	If the element is omitted, the test client SHALL behave as if the value is true.	

Element	defaultTextBoxPx
Description	Default width of the editor input box in pixels.
Element Type	xsd:int {>0}
Value	*Any*
Default	*None*
Extensions	☒
Notes	The implementation MAY impose constraints on the maximum and minimum value of defaultTexBoxPx. The behavior when the value is smaller than the minimum or larger than the maximum is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.
	If the element is omitted, the test client SHALL behave as if the value is 5.

Element	isMobile	
Description	Controls if the item will be optimized for display on a mobile device when rendered on a mobile device.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	The item will be optimized for display on a mobile device.
	false	The item will not be optimized for display on a mobile device.
Default	*None*	
Extensions	☒	
Notes	If the element is omitted, the test client SHALL behave as if the value is false.	

Element	MagicDisabled	
Description	Controls how the sequence of input keys is processed.	
Element Type	xsd:boolean	
Value	**Value**	**Description**

Element	MagicDisabled	
	true	When a input key with a placeholder is selected (e.g., \sqrt{\ph\}), the equation editor will take the pervious key value and automatically insert if that operation is meaningful. The previous input token value SHALL NOT be used as a value for a placeholder in the next input token.
	false	The user input sequence is entered unchanged.
Default	*None*	
Extensions	☒	
Notes	As an example, if the value of magicDisabled is true and if the user selects the "x" key and then selects the "√" and the items element that specified the "√" includes a placeholder (i.e., the value of the value subelement of items is \sqrt{\PH}), the user input will be "√ x". If the value of magicDisabled is false, the user input will be"x √" and the placeholder for the "√" will be displayed.	
	The logic of using true to mean *is not us*ed and false to mean *is used* is a potential source of error. The default value of false means that magic is enabled by default.	
	The magicDisabled element is a candidate TO BE DEPRECATED and replaced by an element with a more meaningful name. The element MAY be present.	
	If the element is omitted, the test client SHALL behave as if the value is false.	

Input Keys Panel Tab Configuration Elements

The input keys panel tab layout structure is illustrated in Figure 11 (informative).

- An input box is shown at the top of the illustration.
- The input navigation controls are shown below the illustration.
- A single tab input keys panel is shown at the bottom of the illustration
  - The tab/panel contains a grid of 3 columns by 4 rows on the left showing a numeric keypad.
  - The tab/panel contains 4 rows of keys (arithmetic operators, comparison operators, layout/grouping, trigonometric functions).

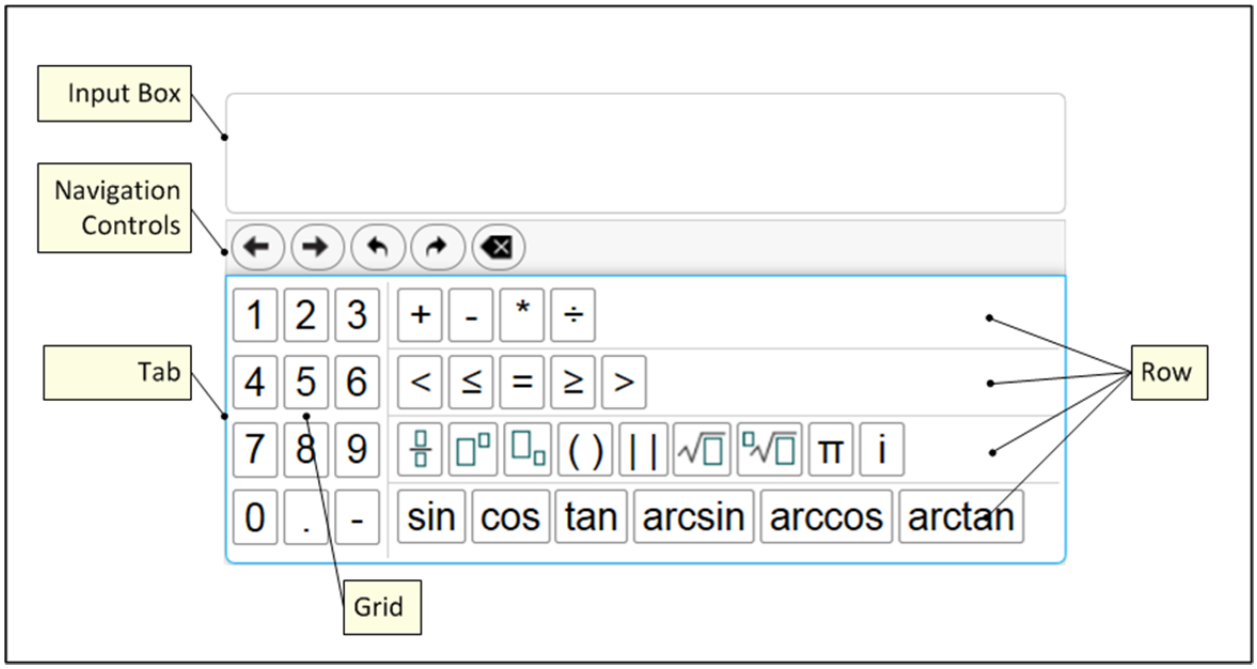


Figure 11: Input Keys Panel Layout (Informative)

Element	tabConfig			
Description	Layout of input keys for each input keys panel tab.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	Order	[1..*]		
	Algebra	[0..1]		
	Basic	[0..1]		
	SBAC3	[0..1]		
	SBAC4	[0..1]		
	SBAC5	[0..1]		
	SBAC6	[0..1]		
	SBAC7	[0..1]		
	SBAC8	[0..1]		
	SBAC9	[0..1]		
	SBAC10	[0..1]		
	SBAC11	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Notes	Each of the subelements has an associated default layout. If the subelement is empty, the default layout is rendered.			
	The tabs are rendered left to right.			

Element	Order			
Description	Tab order. The description of one of the tabs.			
Element Type	xsd:token			
Value	The value SHALL be one of the vocabulary values listed.			
	Value	**Description**		
	Basic	If the Basic element is not present, render the default Basic panel/tab configuration. Otherwise render the configuration defined in the Basic element.		
	Algebra	If the Algebra element is not present, render the default Algebra panel/tab configuration. Otherwise render the configuration defined in the Algebra element.		
	SBAC3	If the SBAC3 element is not present, render the default SBAC3 panel/tab configuration. Otherwise render the configuration defined in the SBAC3 element.		
	SBAC4	If the SBAC4 element is not present, render the default SBAC4 panel/tab configuration. Otherwise render the configuration defined in the SBAC4 element.		
	SBAC5	If the SBAC5 element is not present, render the default SBAC5 panel/tab configuration. Otherwise render the configuration defined in the SBAC5 element.		
	SBAC6	If the SBAC6 element is not present, render the default SBAC6 panel/tab configuration. Otherwise render the configuration defined in the SBAC6 element.		
	SBAC7	If the SBAC7 element is not present, render the default SBAC7 panel/tab configuration. Otherwise render the configuration defined in the SBAC7 element.		
	SBAC8	If the SBAC8 element is not present, render the default SBAC8 panel/tab configuration. Otherwise render the configuration defined in the SBAC8 element.		
	SBAC9	If the SBAC9 element is not present, render the default SBAC9 panel/tab configuration. Otherwise render the configuration defined in the SBAC9 element.		
	SBAC10	If the SBAC10 element is not present, render the default SBAC19 panel/tab configuration. Otherwise render the configuration defined in the SBAC10 element.		
	SBAC11	If the SBAC11 element is not present, render the default SBAC11 panel/tab configuration. Otherwise render the configuration defined in the SBAC11 element.		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	index	☐	xsd:int {>0}	
Extensions	☑			
Notes	The tabs are rendered left to right.			
	The name of the element has no relation to its purpose.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	Order
index	The display order of the tabs. The values need not be contiguous.

Element	Algebra			
Description	Specification of the Algebra tab layout if the default Algebra tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE Algebra.			
	If the element is present a tabConfig element with a value of Order element that is not Algebra SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	Basic			
Description	Specification of the Basic tab layout if the default Basic tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE Basic.			
	If the element is present a tabConfig element with a value of Order element that is not Basic SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC3			
Description	Specification of the SBAC3 tab layout if the default SBAC3 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC3.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC3 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC4			
Description	Specification of the SBAC4 tab layout if the default SBAC4 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC4.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC4 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC5			
Description	Specification of the SBAC5 tab layout if the default SBAC5 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC5.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC5 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC6			
Description	Specification of the SBAC6 tab layout if the default SBAC6 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC6.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC6 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC7			
Description	Specification of the SBAC7 tab layout if the default SBAC7 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC7.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC7 SHALL be NON CONFORMING			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC8			
Description	Specification of the SBAC8 tab layout if the default SBAC8 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC8.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC8 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC9			
Description	Specification of the SBAC9 tab layout if the default SBAC9 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC9.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC9 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC10			
Description	Specification of the SBAC10 tab layout if the default SBAC10 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC10.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC10 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	SBAC11			
Description	Specification of the SBAC11 tab layout if the default SBAC11 tab configuration is not used.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	title	[1]		
	rows	[1..*] {1000}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	If the element is present, the value of the Order element SHALL BE SBAC11.			
	If the element is present a tabConfig element with a value of Order element that is not SBAC11 SHALL be NON CONFORMING.			
Notes	The default input key layout configuration is specified in the *Standard Equation Editor Input Key Panel Configurations* Annex.			

Element	title
Description	Title of the input area to be displayed for the item.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	How or when the title element content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.

Element	rows			
Description	Individual *row* definition in the input keys panel tab area.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		

Element	rows			
	title	[1]		
	type	[1]		
	cols	[0..1]		
	items	[1..*] {100}		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	index	☐	xsd:int {>0}	
Extensions	☑			
Conformance	If the value of the type element is grid, the cols element is REQUIRED.			
	An element that contains a type element with a value of grid and does not contain the cols element SHALL be NON CONFORMING.			
Notes	The element name row is a misnomer; the area is either a grid (rows and columns) of items or a row of items.			
	The individual rows are stacked and rendered from top to bottom in the order specified.			
	If the row type is grid, the items in the grid span all rows vertically.			
	A row type of grid is normally the first row specified. Normally there is only one grid.			
	If there are multiple grids, each collection of rows before, between or after the grids are stacked in a column and each stack of rows and each grid are rendered left to right.			
	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			

Attributes	rows
index	The display order of the rows. The values need not be contiguous.

Element	title
Description	Title of the *row*.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	HTML or other markup embedded in the string SHALL be ignored and treated as plain text.
Notes	How or when the title element content is presented to the student is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
	This title element is different from the title element of the container element.

Element	type	
Description	Type of layout of the *rows* in the input keys panel.	
Element Type	xsd:token– A vocabulary of values.	
Value	The value SHALL be one of the vocabulary values listed.	
	Value	**Description**
	grid	The items are placed in the grid, from left to right, in the order specified. Each row in the grid contains cols number of items. As many rows are used as needed to accommodate all the items in the *row*.

Element	type	
	row	The items are placed in the row, from left to right, in the order specified. The row is as long as needed to accommodate all the items in the row.
Default	*None*	
Extensions	☒	
Notes	The behavior if the space required for the layout exceeds the available space is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.	

Element	cols
Description	Number of columns in a grid if the value of the type element is grid.
Element Type	xsd:int {>0}
Value	*Any*
Default	*None*
Extensions	☒
Notes	The maximum number of columns is NOT SPECIFIED and is IMPLEMENTATION DEFINED.
	The element is ignored if the value of the type element is row.

Element	items			
Description	Description of the individual input keys.			
Element Type	mixed			
Elements	**Name**	**Multiplicity**		
	key	[1]		
	text	[0..1]		
	value	[1]		
	isParsed	[0..1]		
	css	[0..1]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	index	☐	xsd:int {>0}	
Extensions	☑			
Conformance	An element that contains both mixed content and subelements SHALL be NON CONFORMING.			
	The value for an item element that has mixed content SHALL be a single, simple string whose value is defined in Table 13.			
	A value for an item element that has mixed content that is not a simple string whose value is defined in Table 13 SHALL be NON CONFORMING.			
Notes	The behavior when there are duplicate values for the index attribute is NOT SPECIFIED and is IMPLEMENTATION DEFINED.			
	The key that is displayed for an item element that has mixed content is defined in Table 13.			

Attributes	items
index	The display order of the items. The values need not be contiguous.

Values for an item element containing mixed content are listed in Table 13. The corresponding value to display on the key is shown in the table. An implementation of the test client that renders the equation editor content SHALL support the core values indicated. The implementation SHOULD

support all of the MathJax LaTeX macros listed [MathJax]. The implementation MAY support any other MathJax LaTeX macros and AMS macros. The implementation MAY support other values. For a macro value, the item value is the name of the macro (the leading \ is deleted).

Table 13: Item Element Values for Mixed Content

Item Value	Display	Conformance
Digits: [0-9], .	0 1 2 ... 9 .	SHALL
Letters/Variables: [a-zA-Z]	a b c ... z A B C ... Z	SHALL
Operators: + ,- , * , / ,times, div	+ - * / ÷ ×	SHALL
Relations: lt, le, =, ge, gt	< ≤ = ≥ >	SHALL
Functions: sin, cos, tan, arcsin, arccos, arctan	sin cos tan arcsin arccos arctan	SHALL
Greek Letters/Variables: pi	π	SHALL
Miscellaneous: fraction, ( ), | |	□/□ (□) |□|	SHALL
Any MathJax LaTeX BIN Class Macro	Corresponding LaTeX	SHOULD
Any MathJax LaTeX OP Class Macro	Corresponding LaTeX	SHOULD
Any MathJax LaTeX ORD Class Macro	Corresponding LaTeX	SHOULD
Any MathJax LaTeX REL Class Macro	Corresponding LaTeX	SHOULD
Other MathJax LaTeX Macros	Corresponding LaTeX	MAY
Any MathJax LaTeX AMSsymbol Class Macro	Corresponding LaTeX	MAY
Any MathJax LaTeX AMSmath Class Macro	Corresponding LaTeX	MAY

Element	key
Description	Identifier for an input key.
Element Type	xsd:string {1000}
Value	*Any*
Default	*None*
Extensions	☒
Notes	

Element	text
Description	Label that is displayed on the input key.
Element Type	xsd:string {1000}
Value	*Any*
Default	*None*
Extensions	☒
Notes	

Element	value
Description	Value that is entered into the input box when the key is pressed.
Element Type	xsd:string {1000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The value SHALL be valid [TeX] content.
	A value that is not valid [TeX] content SHALL be NON CONFORMING.

Element	value
Notes	Within the TeX content, the value\ph is used to render a placeholder symbol. The student can enter values into the placeholder.

Element	isParsed	
Description	Specification of how the key is displayed by the test client.	
Element Type	xsd:boolean	
Value	**Value**	**Description**
	true	The value of the text element is displayed on the button.
	false	The value of the value element is converted to Unicode that is displayed on the button.
Default	*None*	
Extensions	☒	
Notes	If the element is omitted, the test client SHALL behave as if the value is false.	

Element	css
Description	CSS directives for the visual appearance of the key.
Element Type	xsd:string {4000}
Value	*Any*
Default	*None*
Extensions	☒
Conformance	The value SHALL conform to [CSS 2].
	A value that is not a valid [CSS 2] SHALL be NON CONFORMING.
Notes	

MathML Elements

Element	mathML			
Description	MathML [MathML] to be displayed in the input box.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	math	[1..*]		
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☒			
Conformance	The content of the math element SHALL conform to [MathML].			
	A value that is not a valid [MathML] SHALL be NON CONFORMING.			
Notes	There is one math element for each input box.			
	All of [MathML] is supported.			
	For validation, the XML schema MAY import the [MathML] schema from schemaLocation http://www.w3.org/Math/XMLSchema/mathml3/mathml3.xsd			

Assessment Item Usage Statistics XML Document Elements

An *Assessment Item Usage Statistics* XML document captures data about the use of an assessment item. The assessment item usage statistics are represented as a collection of subelements of the item's statistic element. The statistic element is the root of the subtree of elements. The statistic element is one of the subelements of an assessment item.

Assessment Item Usage Statistics Elements

Note: Details of the statistic elements MAY be included in a future version of the Specification.

Element	statistic			
Description	Assessment item usage statistics.			
Element Type	sequence			
Elements	**Name**	**Multiplicity**		
	To be defined in a future version of the Specification.			
Attributes	**Name**	**Required**	**Data Type**	**Default**
	None			
Extensions	☑			
Conformance	The content of a non-empty statistic element SHALL be ignored.			
Notes	The element is normally empty (<statistic />) when authoring an item or before the item has been used.			
	An empty statistic element SHOULD NOT be interpreted to mean that there are no usage statistics for the item.			

Assessment Item Machine Rubric XML Document Elements

An *Assessment Item Machine Rubric* XML document contains the rubric rules for automated item grading. An assessment item MAY include a *machine rubric*. Different types of assessment items use different machine rubrics as shown in Table 14.

The machine rubric is contained in an external XML document that is referenced from the assessment item through the filename attribute of the item MachineRubric element in the assessment item.

For any item type listed in the table, a rubric contained in the external XML document that is not appropriate for the item type SHALL be NON CONFORMING. How to determine the type of rubric contained in the external XML document is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

For any item type not listed in the table, the MachineRubric element SHOULD be omitted from the assessment item. If the MachineRubric element is included in an assessment item for any item type not listed in the table, the file reference to the external XML document containing the rubric SHALL be ignored.

The behavior if the external referenced XML document does not exist is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

Table 14: Assessment Item Machine Rubrics

Assessment Item Type (Format)	Machine Rubric Type
Equation (eq)	Equation Rubric (erx)
Grid (gi)	Grid Rubric (grx)
Hot Text (ht)	Hot Text Rubric (hrx)
Natural Language (nl)	Natural Language Rubric (nlx)
Match Interaction (mi)	Match Interaction Rubric (mrx)
Evidence-Based Selected Response (EBSR)	Evidence-Based Selected Response Rubric (qrx)
Simulation (SIM)	Simulation Rubric (srx)
Table Interaction (ti)	Table Interaction Rubric (trx)

Note: Details machine rubric XML elements and the structure of the XML document for the machine rubrics for the different types of assessment items MAY be included in a future version of the Specification.

XML Schemata and Document Criteria

In addition to the XML document element definitions, all XML documents for assessment items SHALL satisfy the following criteria:

- *Document Criteria* – descriptions of document element content that cannot be specified at the XML element level.
- *Semantic Constraints* – constraints on the XML documents that cannot be specified at the XML element level.
- *Specification Versioning* – criteria for identifying the specific version of the Specification in XML schemata describing an assessment item.
- *IANA Considerations* – criteria for Internet media type names for XML documents conforming to the Specification.
- *Implementation Considerations* – best practices on how to represent or use XML documents describing an assessment item.

Document Criteria

xHTML [xHTML] element content MAY contain references to media files (animations) through href attributes in anchor element references. The href attribute will only contain a reference to a single file. Multiple files not referenced from the content in different formats MAY exist, e.g., an xHTML file (.xhtml file extension) or a Flash animation (.swf file extension). The collection of file formats supported is IMPLEMENTATION DEFINED. How the test client selects the file format to display is IMPLEMENTATION DEFINED.

Multiple files for different language variants MAY exist for each file format. The file name form is specific to the language of the content of the file. A language variant is specified by adding an underscore character (_ _) to the end of the file name (before the file extension) followed by the ISO language code for the language of the content. For example, a Japanese (ja) variant of the file animation.swf would be animation_ja.swf. The collection of language variants supported is IMPLEMENTATION DEFINED. How the test client selects the language variant file to display is IMPLEMENTATION DEFINED.

Semantic Constraints

The XML elements listed include element specific or attribute semantic constraints. There are additional document semantic constraints between elements that are documented as part of the individual elements. Constraints between two items are included in the conformance clause for each of the items.

An implementation MAY impose additional semantic constraints on an instance of a conforming XML document.

Specification Versioning

Any significant change to the Specification SHALL be indicated by an update to the specification version number and an update to the corresponding XML schemata specification or DTD version number. A change that is backward compatible SHALL be indicated by an update to the minor part of the version number. A change that is not backward compatible SHALL be indicated by an update to the major part of the version number.

Modifications to the Specification narrative that do not impact the schemata or DTDs SHALL be indicated by an update to the subpart of the minor part of the specification version number, but the XML schemata version number SHALL NOT change.

XML schemata and DTDs conforming to the Specification SHALL include a version number indicating the version of the Specification. The XML schemata SHALL use the xsd:schema version attribute to specify the schema version number. The specification version number for XML schemata are listed in Table 15.

Table 15: XML Schemata Specification Versions

Schemata	Specification Version	XML Schema Version
Assessment Item XML document	1.0.0	SAAIF AI v1p0
Passage Item XML document	1.0.0	SAAIF PI v1p0
Tutorial XML document	1.0.0	SAAIF TUT v1p0
Wordlist XML document	1.0.0	SAAIF WL v1p0
Assessment Item Accessibility XML document	1.0.0	SAAIF APIP v1p0
Grid Item Rendering Specification XML document	1.0.0	SAAIF GR v1p0
Equation Editor Configuration XML document	1.0.0	SAAIF EE v1p0
Assessment Item Usage Statistics XML document	1.0.0	SAAIF STAT v1p0
Assessment Item Machine Rubric XML document	1.0.0	SAAIF MR v1p0

XML documents SHOULD include an attribute on each root element that indicates the specification version number. Since the Specification does not mandate specific schemata or DTDs, the attribute is not included with the description of the root XML elements.

IANA Considerations

XML documents for assessment items SHALL be given a media type (MIME Type) and file extension. The MIME media type for the XML serialization of different assessment item documents is listed in Table 16. Corresponding file extensions are also shown in the table.

Table 16: XML Document Media Types

Document	Name	Subtype	Ext
Assessment Item Release XML document	application	vnd.smarterapp.assessmentitemrelease+XML	.xml
Assessment Item XML document	application	vnd.smarterapp.assessmentitem+XML	.xml
Passage Item XML document	application	vnd.smarterapp.passageitem+XML	.xml
Tutorial XML document	application	vnd.smarterapp.tutorial+XML	.xml
Wordlist XML document	application	vnd.smarterapp.assessmentitem+XML	.xml
Assessment Item Accessibility XML document	application	vnd.smarterapp.apip+XML	.xml
Grid Item Rendering Specification XML document	application	vnd.smarterapp.griditemrenderingspec+XML	.xml

Document	Name	Subtype	Ext
Equation Editor Configuration XML document	application	vnd.smarterapp.equationeditorconfig+XML	.xml
Assessment Item Usage Statistics XML document	application	vnd.smarterapp.useagestatistics+XML	.xml
Assessment Item Machine Rubric XML documents	application	vnd.smarterapp.machinerubric+XML	.xml

The media type SHALL conform to [RFC 4288] and SHOULD be registered with IANA [http://www.iana.org/cgi-bin/mediatypes.pl] in accordance with [RFC 4289].

In lieu of the media types listed in Table 16, an XML document MAY use MIME type application/xml

Documents for *attachments* for assessment items or passage items SHALL be given a media type (MIME Type) and file extension. The MIME media type for different attachment file types are listed in Table 17. Corresponding file extensions are also shown in the table.

Table 17: Attachment Media Types

Document	Name	Subtype	Ext
ASL STEM	video	mp4	.mp4
Braille Contracted Braille Ready File	application	vnd.smarterapp.braille.2.brf	.brf
Braille Contracted Printer File	application	vnd.smarterapp.braille.2.prn	.prn
Braille Nemeth Braille Ready File	application	vnd.smarterapp.braille.nemeth.brf	.brf
Braille Nemeth Printer File	application	vnd.smarterapp.braille.nemeth.prn	.prn
Braille Uncontracted Braille Ready File	application	vnd.smarterapp.braille.1.brf	.brf
Braille Uncontracted Printer File	application	vnd.smarterapp.braille.1.prn	.prn

The media type SHALL conform to [RFC 4288] and SHOULD be registered with IANA [http://www.iana.org/cgi-bin/mediatypes.pl] in accordance with [RFC 4289].

In lieu of the media types listed in Table 17, a Braille document MAY use MIME type text or text/plain.

Documents for content media assets and resources (images, multimedia) SHALL be given a media type (MIME Type) and file extension. The MIME media type for common attachment file types are listed in Table 18. Corresponding default file extensions are also shown in the table.

Table 18: Content Media Types

Document	Name	Subtype	Ext
Audio – MPEG-4 (MP4)	audio	ma4	.ma4
Audio – OGG Vorbis	audio	ogg	.ogg
Binary	application	octet-stream	
xHTML	text	html	.html
Image – GIF	image	gif	.gif
Image – JPEG	image	jpeg	.jpg
Image – PNG	image	png	.png
Image – PNG	image	svg+xml	.svg
JSON	application	json	.json
PDF	application	pdf	.pdf
Text	text	plain	.txt

Document	Name	Subtype	Ext
Video – Flash	application	vnd.adobe.flash-movie	.swf
Video – MPEG-4 (MP4)	video	mp4	.mp4
Video – OGG Vorbis	video	ogg	.ogg
XML	application	xml	.xml

The media type SHALL conform to [RFC 4288] and SHOULD be registered with IANA [http://www.iana.org/cgi-bin/mediatypes.pl] in accordance with [RFC 4289].

Implementation Considerations

Assessment item identifiers are integer *item numbers*. There is no mechanism to insure that different item producers do not use the same item number for different items; item numbers are not globally unique. Consumers need to be aware that items from different producers MAY use the same item number for different items.

Assessment items include elements that refer to other assessment items by their item number. Consumers need to be aware that the Specification does not define a mechanism to convert the item number into a file name or any other mechanism that can be used to access the referenced item.

Assessment items that refer to other assessment items refer to only the item number, not the combination of item plus version. Consumers need to be aware that the Specification does not define a mechanism to determine which of the multiple versions of an item is being referenced.

Assessment items include elements that refer to other files. Consumers need to be aware that the Specification does not define the behavior if the referenced file does not exist

The assessment item subelements of the rubriclist element are ordered pairs. The grid element IconSpec and the equation editor configuration item elements are also ordered. Within the XML Post Schema Validation Infoset (PSVI), the order of elements MAY NOT be preserved. Additional optional index attributes have been added to these elements to define order. Consumers need to be aware that the order of values cannot be determined in the PSVI if the index attributes are not present.

XML Document Conformance

To conform to the Specification, an XML document holding an *assessment item release*, an *assessment item*, a *passage item*, a *tutorial*, a *wordlist*, a *grid item rendering specification,* an *equation editor configuration, assessment item usage statistics* or an *assessment item machine rubric*:

- SHALL be a valid, well-formed XML 1.0 [XML] document.
- SHALL conform to all REQUIRED structural constraints defined herein.
- SHALL conform to all REQUIRED semantic constraints defined herein.
- SHALL conform to all REQUIRED XML document criteria defined herein.

The qti element of an *assessment item*:

- SHOULD conform to the itemBody element of [QTI 2.1 XML].
- XML schema Validation MAY use the IMS APIP QTI XSD.

The mathML element of an equation editor configuration:

- SHALL conform to [MathML].
- XML schema Validation MAY use the MathML XSD.

Any XML element that contains Cascading Style Sheet content:

- SHALL conform to [CSS 2].
- Validation MUST be performed external to XML schema validation.

Any XML element that contains HTML content:

- SHOULD conform to [XHTML 1.1].
- XML schema validation MAY use the xHTML XSD.

Any XML element that contains phonetic spelling:

- SHALL conform to the International Phonetic Alphabet (IPA) [IPA].
- Validation MUST be performed external to XML schema validation.

Any XML element that contains TeX content:

- SHALL conform to [TeX].
- Validation MUST be performed external to XML schema validation.

If an XML document is treated as a file, the document SHALL be given a media type and file extension as defined herein.

Conformance to or use of specific XML Schemata or XML DTDs is NOT REQUIRED. Schema validation is NOT REQUIRED.

Inclusion of an XML Schema definition or XML DTD reference in an XML document is RECOMMENDED. Use of an XML Schema definition is preferred. Schema validation is RECOMMENDED.

If an XML Schema definition is used, the XSD SHALL be versioned as defined herein.

If an XML Schema definition is used, the XSD SHOULD be the XSD defined in the Annex of the Specification.

If an XML DTD is used, the DTD SHOULD be the DTD defined in the Annex of the Specification.

XML Document Producer Conformance

To produce an XML document that conforms to the Specification, a document producer:

- SHALL produce a conforming XML 1.0 document as described above.
- SHALL include all REQUIRED XML elements in the XML document (all elements with multiplicity [1] or [1..*]).
- SHALL include all REQUIRED XML element attributes in the XML document.
- SHOULD NOT include any IS DEPRECATED XML elements in the XML document.
- SHOULD NOT include any IS DEPRECATED XML element attributes in the XML document.
- SHALL include XML document version information.
- SHALL use the media type and file extension defined if the document is treated as a file.
- MAY include any OPTIONAL XML elements in the XML document (any elements with multiplicity [0] or [0..*]).
- MAY include any OPTIONAL XML elements attributes in the XML document.
- MAY include any number of element instances for an element with unbounded multiplicity ([0..*] or [1..*]), including more element instances than the minimum number that a conforming consumer will accept.
- MAY include extension elements in the XML document only for those elements that permit extensions and only if the elements are namespace qualified to be within a separate XML namespace.
- SHOULD include references for all XML Schemata or XML DTDs used in the XML document.
- SHOULD include references to the schemata location for all XML Schemata used in the XML document.
- SHOULD include a schema version attribute on all root elements.

XML Document Consumer Conformance

An application that processes or consumes an XML document that conforms to the Specification:

- SHALL indicate with an error if an XML document is not well formed. The application SHALL reject the entire document.
- SHALL accept and process an XML document with the REQUIRED XML elements.
- SHALL indicate with an error if an XML document does not include any REQUIRED XML elements. The application SHALL reject the entire document.
- SHALL accept and process an XML document with the REQUIRED XML element attributes.
- SHALL indicate with an error if an XML document does not include any REQUIRED XML elements attributes. The application SHALL reject the entire document.
- SHALL accept and process an XML document with any OPTIONAL XML elements.
- SHALL accept and process an XML document with any OPTIONAL XML element attributes.
- SHALL accept an XML document with extensions elements only for those elements that permit extensions and only if the elements are namespace qualified to be within a separate XML namespace.
- SHALL indicate with an error if an XML document includes extension elements for those elements that do not permit extensions. The application MAY either reject the entire document or reject or ignore the extension elements.
- SHALL indicate with an error if an XML document includes extension elements that are not within a separate XML namespace. The application MAY either reject the entire document or reject or ignore the extension elements.
- SHALL accept and process an XML document with any IS DEPRECATED or TO BE DEPRECATED XML elements.
- SHALL accept and process an XML document with any IS DEPRECATED or TO BE DEPRECATED XML element attributes.

- SHALL indicate with an error if the schema version attribute on a root element does not correspond to the schema version if a schema is used.
- MAY process extension elements only for those elements that permit extensions and only if the elements are namespace qualified to be within a separate XML namespace.
- SHALL provide the specified default value for any OPTIONAL element not included in the XML document.
- SHALL accept and process the minimum number of element instances for an element with unbounded multiplicity ([0..*] or [1..*]).
- SHALL indicate with an error or warning if the number of element instances for an element with unbounded multiplicity ([0..*] or [1..*]) exceeds the number of instances that the application can process (this number that the application processes SHALL be equal to or greater than the minimum number of element instances specified herein). The application SHOULD accept and process the minimum REQUIRED number of element instances.
- SHALL indicate with an error or warning if the length of an XML string element exceeds the length of a string that the application can process (this length that the application processes SHALL be equal to or greater than the minimum element string length specified herein). The application SHOULD accept and process the minimum element string length.
- SHALL NOT rely on any XML file naming conventions or file extensions to infer the type of content in the file.

How errors are indicated is NOT SPECIFIED and is IMPLEMENTATION DEPENDENT.

A document that conforms to the Specification MAY include URIs from different web origins. The application that processes the document SHOULD be aware of the issues when attempting to access documents from different web origins as outlined in [RFC 6454].

XML Document Security Considerations

Assessment item XML documents MAY include arbitrary text strings and structured markup.

Producers including assessment item elements in an XML document or consumers accessing assessment items MAY want to consider the potential for unsolicited or malicious content and SHOULD take preventive measures to recognize such content and either identify it or not include it in their document.

Producers SHOULD take reasonable measures to make sure potentially malicious user input such as cross-site scripting attacks are not included in the assessment item XML documents.

Services that provide assessment item XML documents to consumers and other services MUST take reasonable measures to make sure potentially malicious ingested input is not distributed or emitted.

Consumers SHOULD be aware of the potential for malicious content where the attacker publishes documents with falsified property values with the intent of injecting malicious content, hiding or corrupting legitimate content, or misleading users.

Consumers that make assessment items available for crawling by search engines SHOULD take reasonable measures to limit any use of their site as a Search Engine Optimization loophole. This may include converting un-trusted hyperlinks to text or including a rel="nofollow" attribute.

The XML documents MAY include URIs; see [RFC 3986] for security considerations.

The XML documents MAY include IRIs; see [RFC 3987] for security considerations.

The XML documents MAY include URIs from different origins; see [RFC 6454] for security considerations.

Producers and consumers SHOULD be aware that the list of security considerations is not exhaustive.

Producers and consumers SHOULD take reasonable measures to address other potential security issues.

Normative References

Note: For dated references, the edition cited applies. A more recent edition that is backward compatible with respect to features used in the Specification or that corrects errors MAY be used. For undated references, the most recent edition applies.

[CSS 2] Bos, B, et al. (Eds.), "Cascading Style Sheets, Level 2, CSS Specification", *W3C Recommendation*, World Wide Web Consortium (W3C), April 2008.
[http://www.w3.org/TR/CSS2]

[IPA] *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*, Cambridge University Press, June 1999.

[ISO 8859-1] ISO/IEC 8859-1:1998, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*, International Standards Organization (ISO), 1998.

[MathML] Carlisle, D., Ion, P., and Mine, R., (Eds.), "Mathematical Markup Language (MathML)", Version 3.0, *W3C Recommendation*, World Wide Web Consortium (W3C), October 2010.
[http://www.w3.org/TR/MathML3/]

[Nemeth] *The Nemeth Braille Code for Mathematics and Science Notation*, 1972 Revision, American Association of Workers for the Blind, Association for Education of the Visually Handicapped, and National Braille Association, January 1972.

[QTI 2.1 XML] *Question & Test Interoperability (QTI) XSD Binding*, Version 2.1, Lay, S., Gorissen, P., and Kraan, W., Final Release, IMS Global Learning Consortium Inc., August 2012.
[http://www.imsglobal.org/question/qtiv2p1/imsqti_bindv2p1.html]

[RFC 2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", IETF RFC 2119, Internet Engineering Task Force (IETF), March 1997.
[http://tools.ietf.org/html/rfc2119]

[RFC 3339] Klyne, G., "Date and Time on the Internet: Timestamps", IETF RFC 3339, Internet Engineering Task Force (IETF), July 2002.
[http://tools.ietf.org/html/rfc3339]

[RFC 3986] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifier (URI)", IETF RFC 3986, Internet Engineering Task Force (IETF), January 2005.
[http://tools.ietf.org/html/rfc3986]

[RFC 3987] Duerst, M. and Suignard, M., "Internationalized Resource Identifiers (IRIs)", IETF RFC 3987, Internet Engineering Task Force (IETF), January 2005.
[http://tools.ietf.org/html/rfc3987]

[RFC 4288] Freed, N., and Klensin, J., "Media Type Specifications and Registration Procedures", IETF RFC 4288, Internet Engineering Task Force (IETF), December 2005.
[http://tools.ietf.org/html/rfc4288]

[RFC 4289] Freed, N., and Klensin, J., "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", IETF RFC 4288, Internet Engineering Task Force (IETF), December 2005.
[http://tools.ietf.org/html/rfc4289]

[RFC 5646] Phillips, A., and Davis, M. (Eds). "Tags for Identifying Languages", IETF RFC 5646, Internet Engineering Task Force (IETF), September 2009.
[http://tools.ietf.org/html/rfc5646]

[RFC 6454] Barth, A., "The Web Origin Concept", IETF RFC 6454, Internet Engineering Task Force (IETF), September 2011.
[http://tools.ietf.org/html/rfc6454]

[SVG 1.1] Dahlström, E., et al. (Eds.), "Scalable Vector Graphics (SVG 1.1)", Second Edition, *W3C Recommendation*, World Wide Web Consortium (W3C), August 2011.
[http://www.w3.org/TR/SVG11/]

[TeX] Knuth, D., *The TeX Book*, Addison Wesley, 1984.

[XHTML 1.1] Altheim, M., and McCarron, S. (Eds.), "XHTML 1.1 – Module-based XHTML", Second Edition, *W3C Recommendation*, World Wide Web Consortium (W3C), November 2010.
[http://www.w3.org/TR/xhtml11/]

[XML] Bray, T., et al., "Extensible Markup Language (XML) 1.0", Fifth Edition, *W3C Recommendation*, World Wide Web Consortium (W3C), November 2008.
[http://www.w3.org/TR/REC-xml/]

[XSD 1] Thompson, H., et al., "XML Schema Definition Language (XSD) 1.1 Part 1: Structures", *W3C Recommendation*, World Wide Web Consortium (W3C), April 2012.
[http://www.w3.org/TR/xmlschema11-1/]

[XSD 2] Peterson, D., et al., "XML Schema Definition Language (XSD) 1.1 Part 2: Data Types", *W3C Recommendation*, World Wide Web Consortium (W3C), April 2012.
[http://www.w3.org/TR/xmlschema11-2/]

Definitions

Associated Passage: The stimulus content for an assessment item. The associated passage content is stored in a passage item. See also: *Passage Item*.

Attachment: Accessibility content associated with an item.

Canvas: Display area for images and user input for a grid type of assessment item.

Consumer: A person or computer system that reads, processes, examines or uses an XML document.

Dwell: The input cursor remains stationary over an item for a minimum period of time.

Equation Editor: A tool provided by the *test client* that the student can use to enter an equation. The equation editor incorporates an equation text *input box*, navigation buttons and *input keys panel tabs*.

Equation Editor Configuration: Attributes and values that specify the configuration of the *equation editor*. An *Equation Editor Configuration* XML document describes the equation editor configuration.

Equation Item: A type of assessment item where the student enters an equation.

Grid: Grid lines overlaid on the *canvas*.

Grid Item Rendering Specification: The rendering configuration settings for a grid type of assessment item.

Input Box: An area provided by the *equation editor* consisting of one or more text areas that display the student input.

Input Keys Panel Tab: A panel displayed by the *equation editor* that contains one or more tabs; each tab containing a set of input keys that the user can select to enter equation elements in the input box.

Machine Rubric: The definition and rules describing how an assessment item is automatically graded and scored. Only certain types of assessment items MAY be automatically graded. A *Machine Rubric* XML document describes the rubric.

Navigation Buttons: Buttons provided by the *equation editor* that allow the student to navigate in the equation while building and edit it.

Palette: Area of a grid type of assessment item containing icons or images that the student can place on the *canvas*.

Passage Item: An assessment item that contains a stimulus.

Producer: A person or computer system that creates or originates an XML document.

Rendering Specification: A collection of attributes and values that describe how a particular type of assessment item is displayed or rendered by the *test client* when the item is presented to the student. A *Rendering Specification* XML document describes the rendering.

Resource: A type of auxiliary content for an assessment item, such as a *wordlist*. A specific variant of an *Assessment Item* XML document is defined for each type of resource.

Rubric: The definition and rules describing how an assessment item is graded and scored.

Snap Behavior: When snap behavior is enabled, positioning the input cursor within a specified radius of pixels close to predefined *snap point* results in the cursor moving (snapping) to the specified predefined *snap point*.

Snap Point: A point on the *canvas* where the *snap behavior* is enabled.

Stem: Directions to the student for an assessment item.

Test Client: Software agent that presents and renders a test form including assessment items to the student and handles student interactions.

Tutorial: A *Tutorial* XML document holds the tutorial. The *Tutorial* XML document is a type of *Assessment Item* XML document.

User Focus: Point or input item where the user input device (mouse) is pointing.

Wordlist: A type of *resource* containing a list of thesaurus and multi-lingua glossary definitions. A *Wordlist* XML document holds the content. The *Wordlist* XML document is a variant of an *Assessment Item* XML document.

Acronyms

AIF	Assessment Interoperability Framework
AIR	American Institutes for Research
APIP	Accessible Portable Item Protocol
ASL	American Sign Language
DTD	Document Type Definition
GUID	Globally Unique Identifier
HTML	Hypertext Markup Language
IANA	Internet Corporation for Assigned Names and Numbers
IPA	International Phonetic Alphabet
PSVI	Post Schema Validation Infoset
SAAIF	SmarterApp Assessment Item Format
SBAC	Smarter Balanced Assessment Consortium
xHTML	Extensible Hypertext Markup Language
XML	eXtensible Markup Language
XSD	XML Schema Definition

Informative References

Note: This section is informative.

[APIP] *The Accessible Portable Item Protocol (APIP)*, IMS Global Learning Consortium Inc.
[http://www.imsglobal.org/apip/]

[APIP BP 1.0] *Accessible Portable Item Protocol (APIP): Best Practices and Implementation Guide*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_Best_v1p0cf.html]

[APIP Conformance 1.0] *Accessible Portable Item Protocol (APIP) Conformance and Certification*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_Conf_v1p0cf.html]

[APIP Overview 1.0] *Accessible Portable Item Protocol (APIP) Overview, Candidate Final Release*, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_Oview_v1p0cf.html]

[APIP PNP 1.0] *Accessible Portable Item Protocol (APIP): Technical Specification for AfA PNPv2.0 Features*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_PNP_v1p0cf.html]

[APIP QTI 1.0] *Accessible Portable Item Protocol (APIP): Technical Specification for QTIv2.1 Features*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_QTI_v1p0cf.html]

[APIP Tech 1.0] *Accessible Portable Item Protocol (APIP): Technical Specification*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_Profile_v1p0cf.html]

[APIP Terms 1.0] *Accessible Portable Item Protocol (APIP) Terms and Definitions*, Candidate Final Release, Version 1.0, Driscoll, G., et al., IMS Global Learning Consortium Inc., March 2012.
[http://www.imsglobal.org/apip/apipv1p0cf/APIPv1p0_Terms_v1p0cf.html]

[APIP Validator] *IMS Assessment Conformance and Certification Validator*, IMS Global Learning Consortium Inc.
[http://validator.imsglobal.org/assessment/]

[MathJax] MathJaX TeX and LaTeX Support.
[http://docs.mathjax.org/en/latest/tex.html]

[SBAC Packaging 1.4] *Item Package Specification for Smarter Balanced Assessment Consortium*, Version 1.4, Publisher, January 2014.

[Vim] Vim Editor.
[http://www.vim.org/]

Annex: XML Document Examples

Note: This section is informative.

Note: The examples are for illustrative purposes only.

Examples include:

- An *Assessment Item* example (including *Assessment Item Accessibility* elements).
- A *Passage Item* example (including *Assessment Item Accessibility* elements).
- A *Tutorial* example.
- A *Wordlist* example.
- An *Assessment Item Release* example.
- A *Grid Item Rendering Specification* example embedded in a Grid Assessment Item.
- An *Equation Editor Configuration* example.

The examples use the sample schemata provided.

A standalone accessibility document and a standalone usage statistics document are not shown. Both of these document formats are incorporated into other examples and standalone forms of both documents are not used.

Assessment Item Example

The example shows the XML document for an Assessment Item. The examples show an assessment item with all optional elements for the item type (the RendererSpec, gridanswerspace, rationaleoptlist and optionlist are not appropriate for a natural language item). Only a select set of the potential item attributes are shown. The example includes a tutorial, wordlist, passage and machine rubric external links. It also includes accessibility content and a usage statistics placeholder. The Assessment Item document is shown as a standalone document, not a part of an Assessment Item Release document.

The example XML document is shown in Code Listing A.1. The item rendering is shown in Figure A.1.

Note: The example was created to illustrate the Assessment Item XML document model and the associated schemata. It does not represent a real assessment item.

Code Listing A.1: Assessment Item Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of an Assessment Item XML document -->
02 <item
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0"
04   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
05   xmlns:xhtml="http://www.w3.org/1999/xhtml"
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd"
08   format="nl"
09   id="504890245"
10   version="1.0"
11   schemaversion="1.0">
12
13   <associatedpassage>237047993</associatedpassage>
14   <attriblist>
```

```
    <attrib attid="itm_item_desc">
      <name>Item: Item Description</name>
      <val>A description</val>
      <desc></desc>
    </attrib>
    <attrib attid="itm_item_id">
      <name>Item: ITS ID</name>
      <val>504890245</val>
      <desc></desc>
    </attrib>
    <attrib attid="itm_att_Item Format">
      <name>Item:Item Format</name>
      <val>nl</val>
      <desc></desc>
    </attrib>
    <attrib attid="stm_pass_id">
      <name>Stim: ITS ID</name>
      <val>237047993</val>
      <desc></desc>
    </attrib>
  </attriblist>
  <tutorial id="670360095"/>
  <resourcelist>
    <resource type="wordlist" id="1278" index="1"/>
  </resourcelist>
  <statistic />
  <MachineRubric filename="rubric_file.nlx"/>
  <content language="en-US" format="nl" version="1.0" approvedversion="1.0">
    <illustration>
      <xhtml:p>Item illustration content <xhtml:img src="item_illustration.png" alt="Description"/></xhtml:p>
    </illustration>
    <stem>
      <xhtml:p id="access_tag">Content of the item</xhtml:p>
    </stem>
    <rubriclist>
      <rubric scorepoint="3" index="1">
        <name>Rubric name</name>
        <val>
          <xhtml:p>Rubric value</xhtml:p>
        </val>
      </rubric>
      <samplelist minval="1" maxval="1" index="1">
        <sample purpose="Exemplar" scorepoint="1">
          <name>Sample Name</name>
          <samplecontent>
            <xhtml:p>Sample Content</xhtml:p>
          </samplecontent>
        </sample>
      </samplelist>
    </rubriclist>
    <attachmentlist>
      <attachment id="ASL content" type="ASL" subtype="STEM" filename="ASL_file.mpeg" pass="true"/>
    </attachmentlist>
    <apipAccessibility>
      <accessibilityInfo>
        <accessElement identifier="access 1">
          <contentLinkInfo itsLinkIdentifierRef="access_tag" type="Text">
            <objectLink>uri:scheme/path/path</objectLink>
          </contentLinkInfo>
          <relatedElementInfo>
            <readAloud>
              <textToSpeechPronunciation>Text to read aloud</textToSpeechPronunciation>
            </readAloud>
```

```
78            <brailleText/>
79          </relatedElementInfo>
80        </accessElement>
81      </accessibilityInfo>
82    </apipAccessibility>
83  </content>
84 </item>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-11: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: item element.
  - Lines 03-07: XML namespace information.
  - Lines 08-11: item element attributes:
    - Line 08: Item format (Natural Language: nl).
    - Line 09: Item number.
    - Line 10: The item version number.
    - Line 11: schemaversion indicating which schema version is used (optional).

- Line 13: Passage number for the passage for the item.
- Lines 14-35: Attributes of the item:
  - Lines 15-19: Item description attribute:
    - Line 16: Attribute name.
    - Line 17: Attribute value.
    - Line 18: Attribute description.
  - Lines 20-24: Item number attribute (details as above).
  - Lines 25-29: Item format attribute (details as above).
  - Lines 30-34: Passage number for the item attribute (details as above).
- Line 36: Item number of tutorial for the item.
- Lines 37-39: Resources used with the passage:
  - Line 38: Wordlist resource reference (wordlist XML document item number).
- Line 40: Item sage statistics (empty).
- Line 41: Machine rubric reference (external file reference).

- Lines 42-83: Item content:
  - Lines 43-45: Item illustration (in xHTML).
  - Lines 46-48: Item stem (in xHTML).
  - Lines 49-64: Rubrics:
    - Lines 50-55: Rubric:
      - Line 51: Rubric name.
      - Lines 52-54: Rubric value (in xHTML).
    - Lines 56-63: Samples:
      - Lines 57-62: Sample:
        - Line 58: Sample name.
        - Line 59-61: Sample content (in xHTML).
  - Lines 65-67: Item accessibility attachments:
    - Line 66: ASL content for the passage (external file reference).
  - Lines 68-82: Passage accessibility content:
    - Lines 69-81: Accessibility information:
      - Lines 70-80: An access element:
        - Lines 71-73: A content link:
          - Line 72: An object link.

- Lines 74-79: Related elements:
  - Lines 75-77: Real aloud information.
    - Line 76: Read aloud passage.
  - Line 78: Braille text (empty).

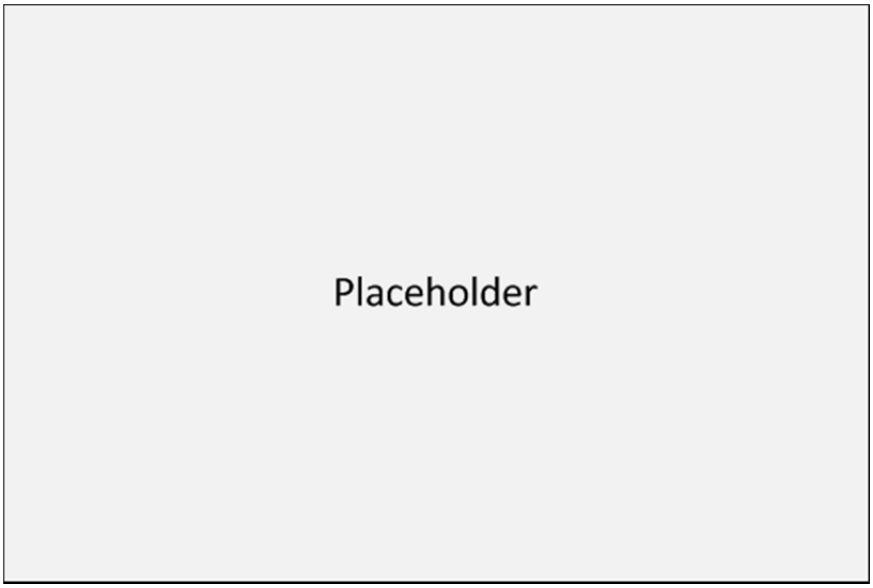


Figure A.1: Assessment Item Example Rendering

Passage Item Example

The example shows the XML document for a Passage Item. The examples show a passage item with all optional elements. The example includes accessibility content. The Passage item document is shown as a standalone document, not a part of an Assessment Item Release document.

The example XML document is shown in Code Listing A.2. The item rendering is shown in Figure A.2.

Note: The example was created to illustrate the Passage Item XML document model and the associated schemata. It does not represent a real passage item.

Code Listing A.2: Passage Item Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of a Passage XML document -->
02 <passage
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0"
04   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
05   xmlns:xhtml="http://www.w3.org/1999/xhtml"
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0.xsd"
08   id="256031931"
09   version="1.0"
10   schemaversion="1.0">
11
12   <attriblist>
```

```
13    <attrib attid="stm_pass_desc">
14      <name>Stim: Descrption</name>
15      <val>A description</val>
16      <desc></desc>
17    </attrib>
18    <attrib attid="stm_pass_id">
19      <name>Stim: ITS ID</name>
20      <val>256031931</val>
21      <desc></desc>
22    </attrib>
23    <attrib attid="stm_pass_subject">
24      <name>Stim: Subject</name>
25      <val>ELA</val>
26      <desc></desc>
27    </attrib>
28  </attriblist>
29  <resourcelist>
30    <resource type="wordlist" id="7962786560" index="1"/>
31  </resourcelist>
32  <content language="en-US" version="1.0" approvedversion="1.0">
33    <title>
34      <xhtml:p>Title of the passage</xhtml:p>
35    </title>
36    <author>
37      <xhtml:p>Author of the passage</xhtml:p>
38    </author>
39    <stem>
40      <xhtml:p id="access_tag">Content of the passage</xhtml:p>
41    </stem>
42    <attachmentlist>
43      <attachment id="ASL content" type="ASL" subtype="STEM" filename="ASL_file.mpeg" pass="true"/>
44      <attachment id="Braille content" type="BRF" subtype="nemeth" filename="nemeth_file.prn"/>
45    </attachmentlist>
46    <apipAccessibility>
47      <accessibilityInfo>
48        <accessElement identifier="access 1">
49          <contentLinkInfo itsLinkIdentifierRef="access_tag" type="Text">
50            <objectLink>uri:scheme/path/path</objectLink>
51          </contentLinkInfo>
52          <relatedElementInfo>
53            <readAloud>
54              <textToSpeechPronunciation>Text to read aloud</textToSpeechPronunciation>
55            </readAloud>
56            <brailleText/>
57          </relatedElementInfo>
58        </accessElement>
59      </accessibilityInfo>
60    </apipAccessibility>
61  </content>
62</passage>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-09: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: passage element.
  - Lines 03-07: XML namespace information.
  - Lines 08-10: passage element attributes:
    - Line 08: Item number.
    - Line 09: The item version number.

        - Line 10: schemaversion indicating which schema version is used (optional).

- Lines 12-28: Attributes of the passage:
    - Lines 13-17: Passage description attribute:
        - Line 14: Attribute name.
        - Line 15: Attribute value.
        - Line 16: Attribute description.
    - Lines 18-22: Passage item number attribute (details as above).
    - Lines 23-27: Passage subject attribute (details as above).
- Lines 29-31: Resources used with the passage:
    - Line 31: Wordlist resource reference (wordlist XML document item number).

- Lines 32-61: Passage content:
    - Lines 33-35: Passage title (in xHTML).
    - Lines 36-38: Passage author (in xHTML).
    - Lines 39-41: Passage stem (in xHTML).
    - Lines 42-45: Passage accessibility attachments:
        - Line 43: ASL content for the passage (external file reference).
        - Line 44: Nemeth braille content for the passage (external file reference).
    - Lines 46-60: Passage accessibility content:
        - Lines 47-59: Accessibility information:
            - Lines 48-58: An access element:
                - Lines 49-51: A content link:
                    - Line 50: An object link.
                - Lines 52-57: Related elements:
                    - Lines 53-55: Real aloud information:
                        - Line 54: Read aloud passage.
                    - Line 56: Braille text (empty).

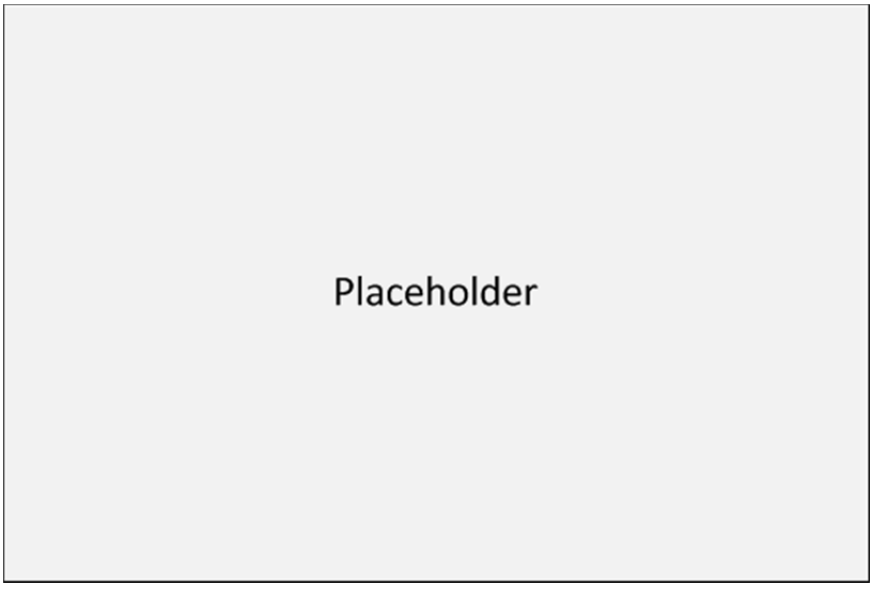


Figure A.2: Passage Item Example Rendering

Tutorial Example

The example shows the XML document for a Tutorial. The example shows a tutorial item that includes all optional elements that are used; optional elements that are ignored are not included. Only one item attribute is shown. The example omits accessibility content. The Tutorial is shown as a standalone document, not a part of an Assessment Item Release document.

The example XML document is shown in Code Listing A.3. The item rendering is shown in Figure A.3.

Note: The example was created to illustrate the Tutorial XML document model and the associated schemata. It does not represent a real tutorial.

Code Listing A.3: Tutorial Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of a Tutorial XML document -->
02 <item
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/tutorial_v1p0"
04   xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
05   xmlns:xhtml="http://www.w3.org/1999/xhtml"
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/tutorial_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/tutorial_v1p0.xsd"
08   format="tut"
09   id="902451852"
10   version="1.0"
11   schemaversion="1.0">
12
13   <attriblist>
14     <attrib attid="itm_item_id">
15       <name>Item: ITS ID</name>
16       <val>504890245</val>
17       <desc></desc>
18     </attrib>
19   </attriblist>
20   <content language="en-US" format="tut" version="1.0" approvedversion="1.0">
21     <illustration>
22       <xhtml:p>Tutorial illustration content <xhtml:img src="tutorial_illustration.png" alt="Description"/></xhtml:p>
23     </illustration>
24     <stem>
25       <xhtml:p>Content of the tutorial</xhtml:p>
26     </stem>
27   </content>
28 </item>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-11: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: item element.
  - Lines 03-07: XML namespace information.
  - Lines 08-11: item element attributes:
    - Line 08: Item format (Tutorial: tut).
    - Line 09: Item number.
    - Line 10: The item version number.

        - Line 11: schemaversion indicating which schema version is used (optional).

- Lines 13-19: Attributes of the item:
    - Lines 14-18: Item number attribute:
        - Line 15: Attribute name.
        - Line 16: Attribute value.
        - Line 17: Attribute description.

- Lines 20-27: Tutorial content:
    - Lines 21-23: Tutorial illustration (in xHTML).
    - Lines 24-26: Tutorial stem (in xHTML).

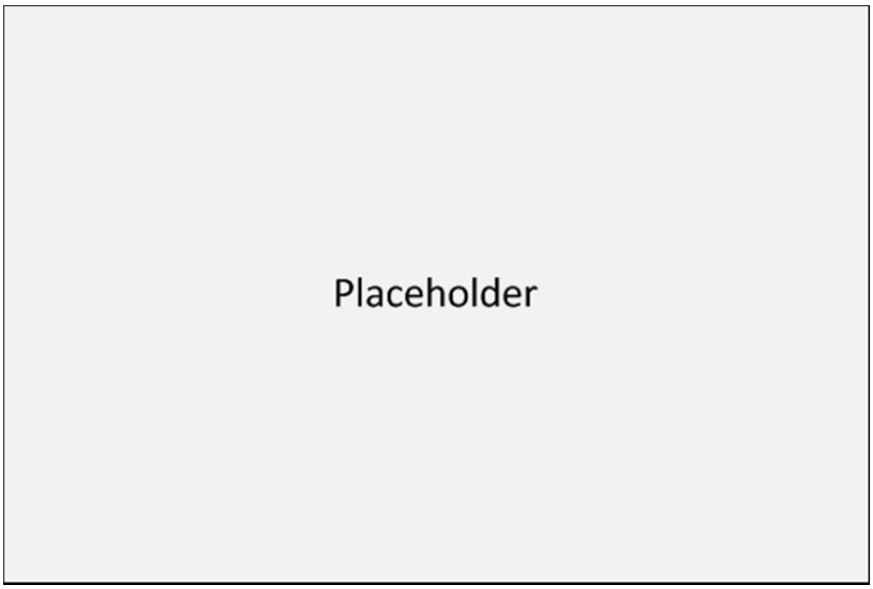


Figure A.3: Tutorial Example Rendering

Wordlist Example

The example shows the XML document for a wordlist. The example illustrates a Wordlist document with two items. Each of the items includes a glossary entry and a thesaurus entry. The Wordlist document is shown as a standalone document, not a part of an Assessment Item Release document.

The example XML document is shown in Code Listing A.4. The item rendering is shown in Figure A.4.

Note: The example was created to illustrate the Wordlist XML document model and the associated schemata. It does not represent a real wordlist item.

Code Listing A.4: Wordlist Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of a Wordlist XML document -->
02 <item
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/wordlist_v1p0"
04   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
05    xmlns:xhtml="http://www.w3.org/1999/xhtml"
06    xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/wordlist_v1p0
07  http://www.smarterapp.org/xsd/saaif/v1p0/wordlist_v1p0.xsd"
08    format="wordlist"
09    id="241852256"
10    version="1.0"
11    schemaversion="1.0">
12
13    <keywordList>
14      <keyword text="A term" index="1">
15        <html listType="glossary" listCode="TDS_WL_Glossary">
16          <xhtml:p>Text of "A term" Glossary Entry</xhtml:p>
17        </html>
18        <html listType="thesaurus" listCode="TDS_WL_THES">
19          <xhtml:p>Text of "A term" Thesaurus Entry</xhtml:p>
20        </html>
21      </keyword>
22      <keyword text="Another term" index="2">
23        <html listType="glossary" listCode="TDS_WL_Glossary">
24          <xhtml:p>Text of "Another term" Glossary Entry</xhtml:p>
25        </html>
26        <html listType="thesaurus" listCode="TDS_WL_THES">
27          <xhtml:p>Text of "Another term" Thesaurus Entry</xhtml:p>
28        </html>
29      </keyword>
30    </keywordList>
31  </item>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-11: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: item element.
  - Lines 03-07: XML namespace information.
  - Lines 08-11: item element attributes:
    - Line 08: Item type/format.
    - Line 09: Item number.
    - Line 10: Item version number.
    - Line 11: schemaversion indicating which schema version is used (optional).

- Lines 13-30: List of keywords:
  - Lines 14-21: A keyword:
    - Line 14: Keyword name and sort index.
    - Lines 15-17: Glossary entry for the keyword (in xHTML).
    - Lines 18-20: Thesaurus entry for the keyword (in xHTML).
  - Lines 22-29: A keyword:
    - Line 22: Keyword name and sort index.
    - Lines 23-25: Glossary entry for the keyword (in xHTML).
    - Lines 26-28: Thesaurus entry for the keyword (in xHTML).

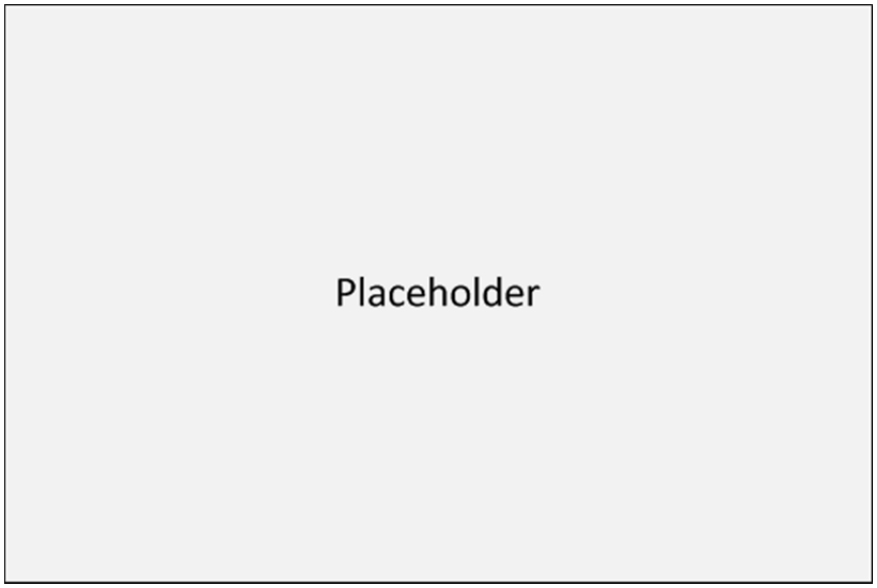


Figure A.4: Wordlist Example Rendering

Assessment Item Release Example

The example shows the XML document for an Assessment Item Release. The item release document wraps a passage item. For illustration, the passage is minimal – only the required passage elements are shown. The passage item schema includes the itemrelease element; the document uses the passage item schema. An assessment item release for an assessment item, tutorial or wordlist would use the appropriate schema for the type of item included in the item release.

The example XML document is shown in Code Listing A.5.

Note: The example was created to illustrate the Assessment Item Release XML document model and the associated schemata. It does not represent a real tutorial.

Code Listing A.5: Assessment Item Release Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of an Item Release Document wrapping a Passage XML document -->
02 <itemrelease
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0"
04   xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
05   xmlns:xhtml="http://www.w3.org/1999/xhtml"
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0.xsd"
08   version="1.0"
09   schemaversion="1.0">
10
11   <passage
12     id="69509844"
13     version="1.0">
14     <content language="en-US" version="1.0" approvedVersion="1.0">
15       <title>
16         <xhtml:p>Title of the passage</xhtml:p>
17       </title>
18       <stem>
```

```
      <xhtml:p>Content of the passage</xhtml:p>
    </stem>
  </content>
 </passage>
</itemrelease>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-09: Standard XML document header and namespace information:
  - Line 00: XML header
  - Line 02: itemrelease element
  - Lines 03-07: XML namespace information.
  - Lines 08-09: itemrelease element attributes:
    - Line 08: Item release version number.
    - Line 09: schemaversion indicating which schema version is used (optional).

- Lines 11-122: Passage item included in the item release:
  - Lines 12-13: passage element attributes:
    - Line 12: Passage item number.
    - Line 13: Passage version number.
  - Lines 14-21: Passage content:
    - Lines 15-17: Passage title (in xHTML).
    - Lines 18-21: Passage stem (in xHTML).

Grid Item Rendering Specification Example

The example shows the XML document for a Grid Item Rendering Specification embedded within a grid assessment item. For illustration, the assessment item is minimal – is does not include an optional elements (including accessibility content). The Assessment item document is shown as a standalone document, not a part of an Assessment Item Release document.

The example XML document is shown in Code Listing A.6. The item rendering is shown in Figure A.5.

Note: The example was created to illustrate the Grid Item Specification Rendering XML document model, how it is embedded in an Assessment Item XML document and the associated schemata. It does not represent a real grid assessment item or a real grid rendering specification.

Code Listing A.6: Grid Item Rendering Specification Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of a Grid Rendering Specification in an Assessment Item XML document -->
02 <item
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0"
04   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
05   xmlns:xhtml="http://www.w3.org/1999/xhtml"
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd"
08   format="gi"
09   id="677871574"
10   version="1.0"
11   schemaversion="1.0">
```

```
12
13  <attriblist>
14   <attrib attid="itm_item_id">
15    <name>Item: ITS ID</name>
16    <val>677871574</val>
17    <desc></desc>
18   </attrib>
19   <attrib attid="itm_att_Item Format">
20    <name>Item:Item Format</name>
21    <val>gi</val>
22    <desc></desc>
23   </attrib>
24  </attriblist>
25  <gridanswerspace>
26   <Question id="677871574" version="1">
27    <Description></Description>
28    <QuestionPart id="1">
29     <Options>
30      <ShowButtons>arrow,circle,point</ShowButtons>
31      <GridColor>LightBlue</GridColor>
32      <GridSpacing>50</GridSpacing>
33      <CenterImage>true</CenterImage>
34      <ScaleImage>true</ScaleImage>
35     </Options>
36     <Text/>
37     <ObjectMenuIcons>
38      <IconSpec index="1">
39       <FileSpec>icon1_file.jpg</FileSpec>
40       <Label>icon1</Label>
41      </IconSpec>
42      <IconSpec index="2">
43       <FileSpec>icon2_file.jpg</FileSpec>
44       <Label>icon2</Label>
45      </IconSpec>
46     </ObjectMenuIcons>
47     <ImageSpec>
48       <FileSpec>backgroundimage_file.gif</FileSpec>
49       <Position>20,10</Position>
50     </ImageSpec>
51     <HotSpots>
52      <Regions>
53       <Region name="region1" shape="rect" coords="0,0,50,50">
54        <Event name="select">
55         <Styles fill="#000" fill-opacity=".5" stroke="#00F" stroke-dasharray=""
56          stroke-opacity="1.0" stroke-width="5"/>
57         <Image src="eventimage1_file.png" x="30" y="30"/>
58        </Event>
59       </Region>
60       <Region name="region2" shape="rect" coords="50,50,100,100">
61        <Event name="select">
62         <Styles fill="#000" fill-opacity=".5" stroke="#00F" stroke-dasharray=""
63          stroke-opacity="1.0" stroke-width="5"/>
64         <Image src="eventimage2_file.png" x="80" y="80"/>
65        </Event>
66       </Region>
67       <Region name="region3" shape="rect" coords="100,1000,150,150">
68        <Event name="select">
69         <Styles fill="#000" fill-opacity=".5" stroke="#00F" stroke-dasharray=""
70          stroke-opacity="1.0" stroke-width="5"/>
71         <Image src="eventimage3_file.png" x="130" y="130"/>
72        </Event>
73       </Region>
74       <Region name="region4" shape="rect" coords="150,150,200,200">
```

```
75            <Event name="select">
76              <Styles fill="#000" fill-opacity=".5" stroke="#00F" stroke-dasharray=""
77                stroke-opacity="1.0" stroke-width="5"/>
78              <Image src="eventimage4_file.png" x="180" y="180"/>
79            </Event>
80          </Region>
81        </Regions>
82        <RegionGroups>
83          <RegionGroup max="1" min="0" name="group1">
84            <Include region="region1"/>
85            <Include region="region2"/>
86          </RegionGroup>
87          <RegionGroup max="1" min="0" name="group2">
88            <Include region="region3"/>
89            <Include region="region4"/>
90          </RegionGroup>
91        </RegionGroups>
92      </HotSpots>
93    </QuestionPart>
94    <PreSetAnswerPart>
95      <AnswerSet>
96        <ObjectSet></ObjectSet>
97        <SnapPoint>10@50,50;50,200;200,50;200,200</SnapPoint>
98      </AnswerSet>
99    </PreSetAnswerPart>
100  </Question>
101 </gridanswerspace>
102 <content language="en-US" format="gi" version="1.0" approvedversion="1.0">
103   <stem>
104     <xhtml:p id="access_tag">Content of the item</xhtml:p>
105   </stem>
106 </content>
107 </item>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-11: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: item element.
  - Lines 03-07: XML namespace information.
  - Lines 08-11: item element attributes:
    - Line 08: Item format (Grid: gi)
    - Line 09: Item number.
    - Line 10: The item version number.
    - Line 11: schemaversion indicating which schema version is used (optional).

- Lines 13-24: Attributes of the item:
  - Lines 14-18 Item number attribute
    - Line 15: Attribute name.
    - Line 16: Attribute value.
    - Line 17: Attribute description.
  - Lines 19-23: Item format attribute (details as above).

- Lines 25-101: Grid rendering specification:
  - Lines 26-93: Question element for grid and present answers:
    - Line 27: Grid question description (not used).
    - Lines 28-93: Grid description:

- Lines 29-35: Grid display options:
  - Line 30: Buttons to display.
  - Line 31: Grid color.
  - Line 32: Grid spacing.
  - Line 33: Icon image centering.
  - Line 34: Icon image scaling.
- Line 36: Description (ignored).
- Lines 37-46: Icons:
  - Lines 38-41: Icon:
    - Line 39: Icon file name.
    - Line 40: Icon label.
  - Lines 42-45: Icon (as above).
- Lines 47-50: Background image.
  - Line 48: Background image file name.
  - Line 49: Background image position.
- Lines 51-92: Hot spots
  - Lines 52-81: Hot spot regions:
    - Lines 53-59: Hot spot region:
      - Lines 54-58: Hot spot event:
        - Lines 55-56: Region styling for event.
        - Line 57: Image displayed for event.
  - Lines 60-66: Hot spot region (as above).
  - Lines 67-73: Hot spot region (as above).
  - Lines 74-80: Hot spot region (as above):
  - Lines 82-91: Hot spot region groups.
    - Lines 83-86: Hot spot region group:
      - Line 84: Region in group.
      - Line 85: Region in group.
    - Lines 87-90: Hot spot region group (as above).
- Lines 94-99: Preset answer description.
  - Lines 95-28: Answer set.
    - Line 96: Object set (empty).
    - Line 97: Snap Points

- Line 102-106: Item content:
  - Lines 103-105: Item stem (in xHTML).

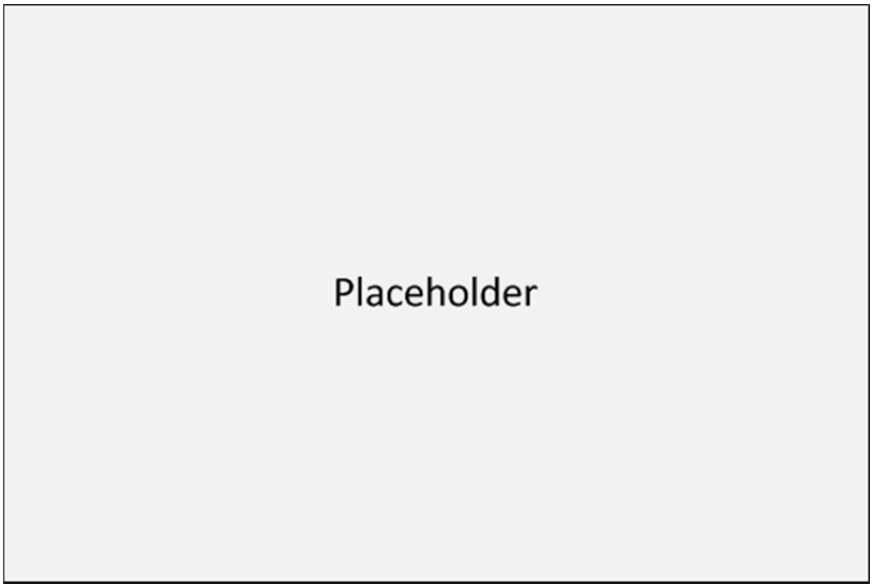


Figure A.5: Grid Item Rendering Specification Example Rendering

Equation Editor Configuration Example

The example shows the XML document for an equation editor configuration. The example shows an equation editor configuration that includes all optional document elements that are used; optional elements that are ignored are not included. The example shows a simple customized input key panel consisting of a grid of keys. The corresponding Equation type assessment item that uses the equation editor is not shown; the item would include a RendererSpec element containing the link (file name) of this equation editor configuration XML document.

The example XML document is shown in Code Listing A.7. The equation editor rendering is shown in Figure A.6.

Note: The example was created to illustrate the Equation Editor Configuration XML document model and the associated schemata. It does not represent an equation editor configuration.

Code Listing A.7: Equation Editor Configuration Example XML Document

```
00 <?xml version="1.0" encoding="utf-8"?>
01 <!-- Example of an Equation Editor Configuration XML document -->
02 <editorconfig
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfig_v1p0"
04   xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
05   xmlns:mml="http://www.w3.org/1998/Math/MathML
06   xsi:schemaLocation="http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfig_v1p0
07 http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfig_v1p0.xsd"
08   schemaversion="1.0">
09
10   <contentLabel>Label text</contentLabel>
11   <configure>false</configure>
12   <tabs>true</tabs>
13   <TeXEntryEnabled>true</TeXEntryEnabled>
14   <TeX>\displaystyle {}</TeX>
15   <tabConfig>
```

```
16    <Order>SBAC3</Order>
17    <SBAC3>
18     <title>SBAC3</title>
19     <rows>
20      <title>Numbers</title>
21      <type>grid</type>
22      <cols>3</cols>
23      <items>1</items>
24      <items>2</items>
25      <items>3</items>
26      <items>4</items>
27      <items>5</items>
28      <items>6</items>
29      <items>7</items>
30      <items>8</items>
31      <items>9</items>
32      <items>0</items>
33      <items>.</items>
34      <items>-</items>
35     </rows>
36    </SBAC3>
37   </tabConfig>
38   <navigation>true</navigation>
39   <mathML>
40    <mml:math/>
41   </mathML>
42   <TeXEntryMode>None</TeXEntryMode>
43   <TeXEntryInit>None</TeXEntryInit>
44   <sanitizeTeXEnabled>true</sanitizeTeXEnabled>
45   <defaultTextBoxPx>5</defaultTextBoxPx>
46   <isMobile>false</isMobile>
47   <MagicDisabled>false</MagicDisabled>
48  </editorconfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

In the example:

- Lines 00-08: Standard XML document header and namespace information:
  - Line 00: XML header.
  - Line 02: editorconfig element.
  - Lines 03-07: XML namespace information.
  - Line 08: editorconfig element attribute:
    - Line 08: The schemaversion indicating which schema version is used (optional).

- Line 10: Display text.
- Line 11: Control configuration settings (off).
- Line 12: Control input keys panel display (displayed).
- Line 13: Control TeX input (permitted).
- Line 14: Initial TeX displayed in input box (in TeX).

- Lines 15-37: Input tab configuration:
  - Line 16: Specified tab configuration.
  - Line 17: Title of the tab configuration.
  - Lines 18-36: Elements of specified tab configuration:
    - Lines 19-35: Definition of a row of input keys:
      - Line 20: Title of row.
      - Line 21: Type of row (grid).

- Line 22: Input key grid size.
- Lines 23-34: Individual input key definitions.

- Line 38: Control navigation button display (displayed).
- Lines 39-41: MathML content for the input box:
  - Line 40: MathML content (in MathML).
- Line 42: TeX placeholder content (in TeX).
- Line 43: Control TeX input (not allowed).
- Line 44: Add placeholders (yes).
- Line 45: Default text box size (5 pixels).
- Line 46: Mobile optimization (off).
- Line 47: Input sequence (magic) control (off).

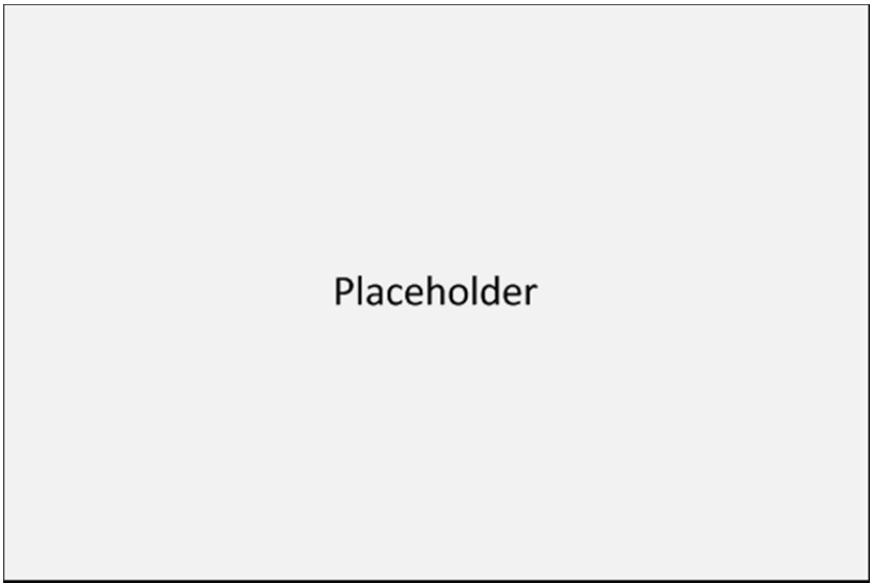


Figure A.6: Equation Editor Configuration Example Rendering

Annex: Standard Equation Editor Input Key Panel Configurations

Note: This section is informative.

Algebra Equation Editor Input Key Panel Configuration

The standard Algebra input key panel configuration is shown in Code Listing A.8. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-25) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and a fraction.
- Four rows of keys on the right:
  - A row (lines 26-31) with 2 variables: x, y
  - A row (lines 32-42) with 4 basic arithmetic operators: + − ∗ ÷×
  - A row (lines 43-51) with 5 basic comparison operators: < ≤ = ≥ >
  - A row (lines 52-63) with absolute value (| |) and parenthesis.

Code Listing A.8: Standard Algebra Input Key Panel Configuration

```
01 <tabConfig>
02   <Order>Algebra</Order>
03   <Algebra>
04    <title>Algebra</title>
05    <rows>
06     <title>Numbers</title>
07     <type>grid</type>
08     <cols>3</cols>
09     <items>1</items>
10     <items>2</items>
11     <items>3</items>
12     <items>4</items>
13     <items>5</items>
14     <items>6</items>
15     <items>7</items>
16     <items>8</items>
17     <items>9</items>
18     <items>0</items>
19     <items>.</items>
20     <items>
21      <key>fraction</key>
22      <text>fraction</text>
23      <value>\frac{\PH}{\PH}</value>
24     </items>
25    </rows>
26    <rows>
27     <title>Variables</title>
28     <type>row</type>
29     <items>x</items>
30     <items>y</items>
31    </rows>
32    <rows>
33     <title>Operations After Grade 6</title>
34     <type>row</type>
35     <items>+</items>
36     <items>-</items>
37     <items>
```

```
38          <key>*</key>
39          <text>*</text>
40        </items>
41        <items>div</items>
42      </rows>
43      <rows>
44        <title>Signs</title>
45        <type>row</type>
46        <items>lt</items>
47        <items>le</items>
48        <items>=</items>
49        <items>ge</items>
50        <items>gt</items>
51      </rows>
52      <rows>
53        <title>Other</title>
54        <type>row</type>
55        <items>
56          <key>| |</key>
57          <value>|\PH|</value>
58        </items>
59        <items>
60          <key>( )</key>
61          <value>(\PH)</value>
62        </items>
63      </rows>
64    </Algebra>
65  </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

Basic Equation Editor Input Key Panel Configuration

The standard Basic input key panel configuration is shown in Code Listing A.9. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-25) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and a fraction.
- Three rows of keys on the right:
  - A row (lines 26-33) with 4 basic arithmetic operators: + − * /
  - A row (lines 34-40) with 3 basic comparison operators: < = >
  - A row (lines 41-48) with a fraction.

Code Listing A.9: Standard Basic Input Key Panel Configuration

```
01  <tabConfig>
02    <Order>Basic</Order>
03    <Basic>
04      <title>Basic</title>
05      <rows>
06        <title>Numbers</title>
07        <type>grid</type>
08        <cols>3</cols>
09        <items>1</items>
10        <items>2</items>
11        <items>3</items>
```

```
12        <items>4</items>
13        <items>5</items>
14        <items>6</items>
15        <items>7</items>
16        <items>8</items>
17        <items>9</items>
18        <items>0</items>
19        <items>.</items>
20        <items>
21          <key>fraction</key>
22          <text>fraction</text>
23          <value>\frac{\PH}{\PH}</value>
24        </items>
25      </rows>
26      <rows>
27        <title>Operations</title>
28        <type>row</type>
29        <items>+</items>
30        <items>-</items>
31        <items>times</items>
32        <items>div</items>
33      </rows>
34      <rows>
35        <title>Signs</title>
36        <type>row</type>
37        <items>lt</items>
38        <items>=</items>
39        <items>gt</items>
40      </rows>
41      <rows>
42        <title>Fraction</title>
43        <type>row</type>
44        <items>
45          <key>fraction</key>
46          <value>\frac{\PH}{\PH}</value>
47        </items>
48      </rows>
49    </Basic>
50  </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC3 Equation Editor Input Key Panel Configuration

The standard SBAC3 input key panel configuration is shown in Code Listing A.10. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-25) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and a fraction.
- Three rows of keys on the right:
  - A row (lines 26-33) with 4 basic arithmetic operators: + – * /
  - A row (lines 34-40) with 3 basic comparison operators: < = >
  - A row (lines 41-48) with parenthesis.

Code Listing A.10: Standard SBAC3 Input Key Panel Configuration

```
<tabConfig>
  <Order>SBAC3</Order>
  <SBAC3>
   <title>SBAC3</title>
   <rows>
    <title>Numbers</title>
    <type>grid</type>
    <cols>3</cols>
    <items>1</items>
    <items>2</items>
    <items>3</items>
    <items>4</items>
    <items>5</items>
    <items>6</items>
    <items>7</items>
    <items>8</items>
    <items>9</items>
    <items>0</items>
    <items>.</items>
    <items>
     <key>fraction</key>
     <text>fraction</text>
     <value>\frac{\PH}{\PH}</value>
    </items>
   </rows>
   <rows>
    <title>Operations</title>
    <type>row</type>
    <items>+</items>
    <items>-</items>
    <items>times</items>
    <items>div</items>
   </rows>
   <rows>
    <title>Signs</title>
    <type>row</type>
    <items>lt</items>
    <items>=</items>
    <items>gt</items>
   </rows>
   <rows>
    <title>Other</title>
    <type>row</type>
    <items>
     <key>( )</key>
     <value>(\PH)</value>
    </items>
   </rows>
  </SBAC3>
 </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC4 Equation Editor Input Key Panel Configuration

The standard SBAC4 input key panel configuration is shown in Code Listing A.11. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-25) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and a fraction.
- Three rows of keys on the right:
  - A row (lines 26-33) with 4 basic arithmetic operators: + − ∗ /
  - A row (lines 34-40) with 3 basic comparison operators: < = >
  - A row (lines 41-48) with parenthesis.

Code Listing A.11: Standard SBAC4 Input Key Panel Configuration

```
01  <tabConfig>
02   <Order>SBAC4</Order>
03   <SBAC4>
04    <title>SBAC4</title>
05    <rows>
06     <title>Numbers</title>
07     <type>grid</type>
08     <cols>3</cols>
09     <items>1</items>
10     <items>2</items>
11     <items>3</items>
12     <items>4</items>
13     <items>5</items>
14     <items>6</items>
15     <items>7</items>
16     <items>8</items>
17     <items>9</items>
18     <items>0</items>
19     <items>.</items>
20     <items>
21      <key>fraction</key>
22      <text>fraction</text>
23      <value>\frac{\PH}{\PH}</value>
24     </items>
25    </rows>
26    <rows>
27     <title>Operations</title>
28     <type>row</type>
29     <items>+</items>
30     <items>-</items>
31     <items>times</items>
32     <items>div</items>
33    </rows>
34    <rows>
35     <title>Signs</title>
36     <type>row</type>
37     <items>lt</items>
38     <items>=</items>
39     <items>gt</items>
40    </rows>
41    <rows>
42     <title>Other</title>
43     <type>row</type>
44     <items>
45      <key>( )</key>
46      <value>(\PH)</value>
47     </items>
48    </rows>
49   </SBAC4>
50  </tabConfig>
```

☑	Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs. XML: 2014-08-11 Schema: 2014-08-11

SBAC5 Equation Editor Input Key Panel Configuration

The standard SBAC5 input key panel configuration is shown in Code Listing A.12. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-25) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and a fraction.
- Three rows of keys on the right:
  - A row (lines 26-33) with 4 basic arithmetic operators: + – ∗ /
  - A row (lines 34-40) with 3 basic comparison operators: < = >
  - A row (lines 41-52) with 2 other input keys: superscript and parenthesis.

Code Listing A.12: Standard SBAC5 Input Key Panel Configuration

```
01  <tabConfig>
02   <Order>SBAC5</Order>
03   <SBAC5>
04    <title>SBAC5</title>
05    <rows>
06     <title>Numbers</title>
07     <type>grid</type>
08     <cols>3</cols>
09     <items>1</items>
10     <items>2</items>
11     <items>3</items>
12     <items>4</items>
13     <items>5</items>
14     <items>6</items>
15     <items>7</items>
16     <items>8</items>
17     <items>9</items>
18     <items>0</items>
19     <items>.</items>
20     <items>
21      <key>fraction</key>
22      <text>fraction</text>
23      <value>\frac{\PH}{\PH}</value>
24     </items>
25    </rows>
26    <rows>
27     <title>Operations</title>
28     <type>row</type>
29     <items>+</items>
30     <items>-</items>
31     <items>times</items>
32     <items>div</items>
33    </rows>
34    <rows>
35     <title>Signs</title>
36     <type>row</type>
37     <items>lt</items>
38     <items>=</items>
39     <items>gt</items>
```

```
40      </rows>
41      <rows>
42        <title>Other</title>
43        <type>row</type>
44        <items>
45          <key>sup</key>
46          <value>\PH^\PH</value>
47        </items>
48        <items>
49          <key>( )</key>
50          <value>(\PH)</value>
51        </items>
52      </rows>
53    </SBAC5>
54  </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC6 Equation Editor Input Key Panel Configuration

The standard SBAC6 input key panel configuration is shown in Code Listing A.13. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Three rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + − * /
  - A row (lines 33-39) with 3 basic comparison operators: < = >
  - A row (lines 40-59) with 4 other input keys: fraction, superscript, parenthesis and vertical bars.

Code Listing A.13: Standard SBAC6 Input Key Panel Configuration

```
01  <tabConfig>
02    <Order>SBAC6</Order>
03    <SBAC6>
04      <title>SBAC6</title>
05      <rows>
06        <title>Numbers</title>
07        <type>grid</type>
08        <cols>3</cols>
09        <items>1</items>
10        <items>2</items>
11        <items>3</items>
12        <items>4</items>
13        <items>5</items>
14        <items>6</items>
15        <items>7</items>
16        <items>8</items>
17        <items>9</items>
18        <items>0</items>
19        <items>.</items>
20        <items>-</items>
21      </rows>
22      <rows>
```

```
23        <title>Operations_After_Grade_6</title>
24        <type>row</type>
25        <items>+</items>
26        <items>-</items>
27        <items>
28          <key>*</key>
29          <text>*</text>
30        </items>
31        <items>div</items>
32      </rows>
33      <rows>
34        <title>Signs</title>
35        <type>row</type>
36        <items>lt</items>
37        <items>=</items>
38        <items>gt</items>
39      </rows>
40      <rows>
41        <title>Other</title>
42        <type>row</type>
43        <items>
44          <key>fraction</key>
45          <value>\frac{\PH}{\PH}</value>
46        </items>
47        <items>
48          <key>sup</key>
49          <value>\PH^\PH</value>
50        </items>
51        <items>
52          <key>( )</key>
53          <value>(\PH)</value>
54        </items>
55        <items>
56          <key>| |</key>
57          <value>\lvert\PH\rvert</value>
58        </items>
59      </rows>
60    </SBAC6>
61  </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC7 Equation Editor Input Key Panel Configuration

The standard SBAC7 input key panel configuration is shown in Code Listing A.14. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Three rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + − ∗ /
  - A row (lines 33-39) with 3 basic comparison operators: < = >
  - A row (lines 40-60) with 5 other input keys: fraction, superscript, parenthesis, vertical bars and π.

Code Listing A.14: Standard SBAC7 Input Key Panel Configuration

```
<tabConfig>
 <Order>SBAC7</Order>
 <SBAC7>
  <title>SBAC7</title>
  <rows>
   <title>Numbers</title>
   <type>grid</type>
   <cols>3</cols>
   <items>1</items>
   <items>2</items>
   <items>3</items>
   <items>4</items>
   <items>5</items>
   <items>6</items>
   <items>7</items>
   <items>8</items>
   <items>9</items>
   <items>0</items>
   <items>.</items>
   <items>-</items>
  </rows>
  <rows>
   <title>Operations_After_Grade_6</title>
   <type>row</type>
   <items>+</items>
   <items>-</items>
   <items>
    <key>*</key>
    <text>*</text>
   </items>
   <items>div</items>
  </rows>
  <rows>
   <title>Signs</title>
   <type>row</type>
   <items>lt</items>
   <items>=</items>
   <items>gt</items>
  </rows>
  <rows>
   <title>Other</title>
   <type>row</type>
   <items>
    <key>fraction</key>
    <value>\frac{\PH}{\PH}</value>
   </items>
   <items>
    <key>sup</key>
    <value>\PH^\PH</value>
   </items>
   <items>
    <key>( )</key>
    <value>(\PH)</value>
   </items>
   <items>
    <key>| |</key>
    <value>\lvert\PH\rvert</value>
   </items>
   <items>pi</items>
  </rows>
 </SBAC7>
</tabConfig>
```

☑	Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs. XML: 2014-08-11 Schema: 2014-08-11

SBAC8 Equation Editor Input Key Panel Configuration

The standard SBAC8 input key panel configuration is shown in Code Listing A.15. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Three rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + – ∗ /
  - A row (lines 33-41) with 5 comparison operators: < ≤ = ≥ >
  - A row (lines 42-70) with 7 other input keys: fraction, superscript, parenthesis, vertical bars, square root (√), n^{th} root and π.

Code Listing A.15: Standard SBAC8 Input Key Panel Configuration

```
01  <tabConfig>
02   <Order>SBAC8</Order>
03   <SBAC8>
04    <title>SBAC8</title>
05    <rows>
06     <title>Numbers</title>
07     <type>grid</type>
08     <cols>3</cols>
09     <items>1</items>
10     <items>2</items>
11     <items>3</items>
12     <items>4</items>
13     <items>5</items>
14     <items>6</items>
15     <items>7</items>
16     <items>8</items>
17     <items>9</items>
18     <items>0</items>
19     <items>.</items>
20     <items>-</items>
21    </rows>
22    <rows>
23     <title>Operations_After_Grade_6</title>
24     <type>row</type>
25     <items>+</items>
26     <items>-</items>
27     <items>
28      <key>*</key>
29      <text>*</text>
30     </items>
31     <items>div</items>
32    </rows>
33    <rows>
34     <title>Signs</title>
35     <type>row</type>
36     <items>lt</items>
37     <items>le</items>
```

```
38        <items>=</items>
39        <items>ge</items>
40        <items>gt</items>
41       </rows>
42       <rows>
43        <title>Other</title>
44        <type>row</type>
45        <items>
46         <key>fraction</key>
47         <value>\frac{\PH}{\PH}</value>
48        </items>
49        <items>
50         <key>sup</key>
51         <value>\PH^\PH</value>
52        </items>
53        <items>
54         <key>( )</key>
55         <value>(\PH)</value>
56        </items>
57        <items>
58         <key>| |</key>
59         <value>\lvert\PH\rvert</value>
60        </items>
61        <items>
62         <key>sqrt</key>
63         <value>\sqrt{\PH}</value>
64        </items>
65        <items>
66         <key>nrt</key>
67         <value>\sqrt[\PH]{\PH}</value>
68        </items>
69        <items>pi</items>
70       </rows>
71     </SBAC8>
72   </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC9 Equation Editor Input Key Panel Configuration

The standard SBAC9 input key panel configuration is shown in Code Listing A.16. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Three rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + − * /
  - A row (lines 33-41) with 5 comparison operators: < ≤ = ≥ >
  - A row (lines 42-70) with 7 other input keys: fraction, superscript, parenthesis, vertical bars, square root (√), n^{th} root and π.

Code Listing A.16: Standard SBAC9 Input Key Panel Configuration

```
<tabConfig>
 <Order>SBAC9</Order>
 <SBAC9>
  <title>SBAC9</title>
  <rows>
   <title>Numbers</title>
   <type>grid</type>
   <cols>3</cols>
   <items>1</items>
   <items>2</items>
   <items>3</items>
   <items>4</items>
   <items>5</items>
   <items>6</items>
   <items>7</items>
   <items>8</items>
   <items>9</items>
   <items>0</items>
   <items>.</items>
   <items>-</items>
  </rows>
  <rows>
   <title>Operations_After_Grade_6</title>
   <type>row</type>
   <items>+</items>
   <items>-</items>
   <items>
    <key>*</key>
    <text>*</text>
   </items>
   <items>div</items>
  </rows>
  <rows>
   <title>Signs</title>
   <type>row</type>
   <items>lt</items>
   <items>le</items>
   <items>=</items>
   <items>ge</items>
   <items>gt</items>
  </rows>
  <rows>
   <title>Other</title>
   <type>row</type>
   <items>
    <key>fraction</key>
    <value>\frac{\PH}{\PH}</value>
   </items>
   <items>
    <key>sup</key>
    <value>\PH^\PH</value>
   </items>
   <items>
    <key>( )</key>
    <value>(\PH)</value>
   </items>
   <items>
    <key>| |</key>
    <value>\lvert\PH\rvert</value>
   </items>
   <items>
```

```
        <key>sqrt</key>
        <value>\sqrt{\PH}</value>
      </items>
      <items>
        <key>nrt</key>
        <value>\sqrt[\PH]{\PH}</value>
      </items>
      <items>pi</items>
    </rows>
  </SBAC9>
 </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC10 Equation Editor Input Key Panel Configuration

The standard SBAC10 input key panel configuration is shown in Code Listing A.17. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Three rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + − ∗ /
  - A row (lines 33-41) with 5 comparison operators: < ≤ = ≥ >
  - A row (lines 42-75) with 9 other input keys: fraction, superscript, subscript, parenthesis, vertical bars, square root (√), n^{th} root, π and i.

Code Listing A.17: Standard SBAC10 Input Key Panel Configuration

```
 <tabConfig>
  <Order>SBAC10</Order>
  <SBAC10>
   <title>SBAC10</title>
   <rows>
    <title>Numbers</title>
    <type>grid</type>
    <cols>3</cols>
    <items>1</items>
    <items>2</items>
    <items>3</items>
    <items>4</items>
    <items>5</items>
    <items>6</items>
    <items>7</items>
    <items>8</items>
    <items>9</items>
    <items>0</items>
    <items>.</items>
    <items>-</items>
   </rows>
   <rows>
    <title>Operations_After_Grade_6</title>
    <type>row</type>
    <items>+</items>
    <items>-</items>
```

```
      <items>
        <key>*</key>
        <text>*</text>
      </items>
      <items>div</items>
    </rows>
    <rows>
      <title>Signs</title>
      <type>row</type>
      <items>lt</items>
      <items>le</items>
      <items>=</items>
      <items>ge</items>
      <items>gt</items>
    </rows>
    <rows>
      <title>Other</title>
      <type>row</type>
      <items>
        <key>fraction</key>
        <value>\frac{\PH}{\PH}</value>
      </items>
      <items>
        <key>sup</key>
        <value>\PH^\PH</value>
      </items>
      <items>
        <key>sub</key>
        <value>\PH_\PH</value>
      </items>
      <items>
        <key>( )</key>
        <value>(\PH)</value>
      </items>
      <items>
        <key>| |</key>
        <value>\lvert\PH\rvert</value>
      </items>
      <items>
        <key>sqrt</key>
        <value>\sqrt{\PH}</value>
      </items>
      <items>
        <key>nrt</key>
        <value>\sqrt[\PH]{\PH}</value>
      </items>
      <items>pi</items>
      <items>i</items>
    </rows>
  </SBAC10>
</tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

SBAC11 Equation Editor Input Key Panel Configuration

The standard SBAC11 input key panel configuration is shown in Code Listing A.18. Only the tabConfig element of the equation editor configuration is shown.

The layout includes:

- A grid on the left (lines 05-21) for a three-column numeric keypad that includes the digits 0-9, a decimal point (.) and negative sign.
- Four rows of keys on the right:
  - A row (lines 22-32) with 4 basic arithmetic operators: + − ∗ /
  - A row (lines 33-41) with 5 comparison operators: < ≤ = ≥ >
  - A row (lines 42-75) with 9 other input keys: fraction, superscript, subscript, parenthesis, vertical bars, square root (√), n^{th} root, π and i.
  - A row (lines 76-85) with 6 trigonometric functions: sine, cosine, tangent, arc sine, arc cosine and arc tangent.

Code Listing A.18: Standard SBAC11 Input Key Panel Configuration

```
01  <tabConfig>
02   <Order>SBAC11</Order>
03   <SBAC11>
04    <title>SBAC11</title>
05    <rows>
06     <title>Numbers</title>
07     <type>grid</type>
08     <cols>3</cols>
09     <items>1</items>
10     <items>2</items>
11     <items>3</items>
12     <items>4</items>
13     <items>5</items>
14     <items>6</items>
15     <items>7</items>
16     <items>8</items>
17     <items>9</items>
18     <items>0</items>
19     <items>.</items>
20     <items>-</items>
21    </rows>
22    <rows>
23     <title>Operations_After_Grade_6</title>
24     <type>row</type>
25     <items>+</items>
26     <items>-</items>
27     <items>
28      <key>*</key>
29      <text>*</text>
30     </items>
31     <items>div</items>
32    </rows>
33    <rows>
34     <title>Signs</title>
35     <type>row</type>
36     <items>lt</items>
37     <items>le</items>
38     <items>=</items>
39     <items>ge</items>
40     <items>gt</items>
41    </rows>
```

```
    <rows>
     <title>Other</title>
     <type>row</type>
     <items>
      <key>fraction</key>
      <value>\frac{\PH}{\PH}</value>
     </items>
     <items>
      <key>sup</key>
      <value>\PH^\PH</value>
     </items>
     <items>
      <key>sub</key>
      <value>\PH_\PH</value>
     </items>
     <items>
      <key>( )</key>
      <value>(\PH)</value>
     </items>
     <items>
      <key>| |</key>
      <value>\lvert\PH\rvert</value>
     </items>
     <items>
      <key>sqrt</key>
      <value>\sqrt{\PH}</value>
     </items>
     <items>
      <key>nrt</key>
      <value>\sqrt[\PH]{\PH}</value>
     </items>
     <items>pi</items>
     <items>i</items>
    </rows>
    <rows>
     <title>Trigonometry</title>
     <type>row</type>
     <items>sin</items>
     <items>cos</items>
     <items>tan</items>
     <items>arcsin</items>
     <items>arccos</items>
     <items>arctan</items>
    </rows>
   </SBAC11>
  </tabConfig>
```

☑ Example Validated: XMLSPY 2014 2.1 using wrapper and local (not hosted) XSDs.
XML: 2014-08-11 Schema: 2014-08-11

Annex: XML Representation Design Decisions

Note: This section is informative.

General Document Design Decisions

Naming Conventions: The XML element and attribute names follow the original source AIR format. The styles of the names are thus inconsistent; some are lower case, some are lower camel case and some are upper camel case. Deprecating element names and replacing them with equivalent elements using lower camel case names would have the least impact to harmonize element name style. Changing names to consistent use of lower camel case is deferred until a future version of the Specification. Any extensions to the Specification SHOULD use lower camel case names.

Default Values: OPTIONAL elements and attributes do not have default values. To avoid the possibility of default values not being provided, they are not used throughout (Schema validation is NOT REQUIRED and thus other means of providing default values is needed).

Lengths/Sizes: Minimum string lengths and the minimum number of elements with multiplicity of [0..*] or [1..*] are selected to support most cases (~99%+).

- Strings used to hold major content blocks, such as item stems, have a minimum length of 64,000 characters. This should accommodate most item core content.
- Strings used to hold most content (e.g., with embedded markup) have a minimum length of 16,000 characters. This should accommodate most marked up content.
- Strings used to hold simple content (no markup) have a minimum length of 4,000 characters.
- Strings used to hold labels, names, identifiers, etc., have a minimum length of 4,000 characters. This will accommodate most URLs and URIs should these elements be treated as resources.
- Strings used to hold version numbers have a minimum length of 100 characters.
- Strings used to hold specific character strings defined by regular expressions are sized to the requirements of the pattern.
- Collections of subelements have a minimum collection size of 100 elements.
- Collections of alternatives (encoding or language variants) have a minimum collection size of 10 alternatives.
- Integers support values with a minimum range $[-2^{31}-1..2^{31}-1]$ (32 bit signed).

Item Numbers: Item numbers (item identifiers $[1..2^{31}-1]$.) are defined to be unique across all assessment items, i.e., the context is essentially global.

Identifier Uniqueness: Most identifiers are defined to be unique across all assessment items, i.e., the context is essentially global. This eliminates any potential conflict in creating and mixing items from different sources. Identifiers for parts of data that are strictly limited to the context of a single item need only be unique within the context of the item. A smaller context within an item is not used to eliminate any potential identifier conflicts within an item.

Extension Points: Well defined types and elements do not require extensibility. All other elements include extensibility points. In general, all simple elements and types do not include extensibility points. The default is to include extensibility.

Deprecated Elements: The original AIR element and attribute names are retained for backward compatibility. New elements with the same purpose, typically to address naming or data type/value

space issues, are added as needed. The related original AIR element is tagged IS DEPRECATED or TO BE DEPRECIATED.

Versioning: Versioning numbers have three parts: a major and a minor version for Specification and XML documents, and a sub version only for changes that do not impact XML documents.

General Schema Design Decisions

While specific schemata are NOT REQUIRED, the sample schemata are designed using common best practices in schema design.

Namespaces: Each top-level schema is defined in its own namespace. This permits documents using different namespaces to be used together (some of the document types use the same element name for different concepts). Schemata defining types that are used with other top-level schemata do not have a namespace.

Namespaces are URLs that follow a versioning naming convention. The root of the namespace name is the Smarter App domain. This is followed by a path denoting the location of the schemata and the individual versioned name for each namespace based on the document type.

Namespaces for the sample schemata are listed in Table A.1.

Table A.1: XSD Schema Namespaces

Schema	Version	Namespace
Assessment Item	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd
Passage Item	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/passageitem_v1p0.xsd
Tutorial	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/tutorial_v1p0.xsd
Wordlist	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/wordlist_v1p0.xsd
Assessment Item Accessibility	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/apip_v1p0.xsd
Grid Item Rendering Specification	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/griditemrenderingspec _v1p0.xsd
Equation Editor Configuration	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfig _v1p0.xsd
Assessment Item Usage Statistics	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/usagestatistics _v1p0.xsd
Assessment Item Machine Rubric	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/machinerubric _v1p0.xsd

Target Namespace: Each top-level schema is defined to have a target namespace. Schemata defining types that are used with other top-level schemata do not have a target namespace.

Default Namespace: Each top-level schema uses the target namespace as the default namespace. Not qualifying all elements reduces namespace clutter, but when different schemata are combined some elements will be namespace qualified while others will not. Schemata defining types that are used with other top-level schemata do not have a default namespace.

Name and Attribute Qualification: All element names in a top-level schema SHOULD be namespace qualified. All element attributes SHOULD NOT be qualified. Schemata defining types that are used with other top-level schemata are not namespace qualified. The types can be used in the

top-level schema without namespace qualification; they inherit the namespace of the top-level schema.

Types versus Elements: Types are created for all elements (either simple or complex types as needed). All document elements are defined using these defined types. Providing types maximizes flexibility and reuse. Elements that are depreciated used built in types.

Design Pattern: All types (simple and complex) are defined as global. The root elements of the document (typically only one) are defined as global. All other elements are defined as local elements within the global root. This pattern defines a set of reusable types. Limiting the number of global elements limits the number of valid documents forms to the number of root elements.

Default Values: Default values are not assigned in the schema for OPTIONAL elements and attributes. While schema validation is RECOMMENDED, it is NOT REQUIRED. Thus assuming the default value comes from the schema is unreliable and other means of providing default values is REQUIRED which leads to increased processing complexity and the decision not to include defaults in a schema.

Tokens: Token elements include restrictions on the value space, i.e., the simple xsd:string type is not used directly. Restrictions help limit the possibility of inserting malicious code in a string.

Extension Points: When allowed, element extensions are defined in a different namespace via including <xsd:any namespace="##other" processcontent="lax" /> in the element definition. Use of ##other requires extensions be at the end of an element. Elements from other namespaces need not be validated (validation is lax).

Schema Versioning: Schema namespaces and locations encode the schema major and minor version in the namespace or location. The use of the pattern v\d{1}p\d{1} to encode the version number in the schema name and locations follows the convention used by IMS. Different specification or schemata versions have different values of namespace and schema location.

Schemata include the xsd:schema version attribute. The values pattern combines the string SAAIF, a designator for the document type and the document type version number. Different specification or schemata versions have different values for the schema version attribute.

All schemata define an additional OPTIONAL schemaversion attribute on each root element of type xsd:token that contains the specification version number. The value of the schemaversion in an instance document SHOULD align with the schema version of the XSD.

Schema Structure: To promote reuse, types are defined in a separate file from the main schema file and included via xsd:include. There is one file of element and custom data type per schema. There is an additional file with common elements and data types that are shared across all of the schemata. Type files contain only global element types and no global elements. An enclosing main file contains only schema declarations and the global elements.

Documentation: Annotation elements are used to document the schemata.

Sample: A sample schema header for the assessment item document illustrating the schema design decisions is shown in Code Listing A.19. A complete schema header will include additional namespaces and additional documentation.

Code Listing A.19: Sample Schema Header

```
00 <?xml version="1.0" encoding="UTF-8"?>
01 <xsd:schema
02   targetNamespace="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd"
03   xmlns="http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd"
04   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
05   version="SAAIF AI 1.0"
06   elementFormDefault="qualified"
07   attributeFormDefault="unqualified">
08
09   <xsd:annotation>
10     <xsd:documentation xml:lang="en-US">
11 Copyright © 2014, The Regents of the University of California.  This schema may be used under the Creative
12 Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0)
13 [http://creativecommons.org/licenses/by-sa/4.0/].
14     </xsd:documentation>
15
16     <xsd:documentation xml:lang="en-US">
17 The complete license is included in the SAAIF specification, located at:
18 http://www.smarterapp.org/specifications.html
19     </xsd:documentation>
20   </xsd:annotation></xsd documentation>
21     </xsd annotation>
22 ...
   </xsd:schema>
```

☒ Example Validated: not validated
XML: ☒ Schema: ☒

Assessment Item Release XML Document and Schema Design Decisions

The *Assessment Item Release* XML document is not essential for item representation or exchange and use by non-AIR systems. It has been retained for compatibility. The *Assessment Item Release* XML document is a candidate TO BE DEPRECATED and removed in a future version of the Specification.

A producer MAY create an *Assessment Item* XML document or a *Passage Item* XML document without the *Assessment Item Release* XML container document and wrap the *Assessment Item* XML document or *Passage Item* XML document with the *Assessment Item Release* XML container document when exchanged with an AIR system that requires the *Assessment Item Release* XML container document.

A consumer that does not require the *Assessment Item Release* XML container document MAY extract the contained *Assessment Item* XML document or *Passage Item* XML document and discard the *Assessment Item Release* XML container document.

The version attribute is REQUIRED but not used. Thus, there are no constraints on the value of the attribute. The attribute is a candidate TO BE DEPRECATED and removed in a future version of the Specification.

Since the *Assessment Item Release* XML document is a candidate TO BE DEPRECATED, extensions are not permitted.

There is no schema for an *Assessment Item Release* XML document. The *Assessment Item* XML document, *Passage Item* XML document, *Tutorial* XML document and *Wordlist* XML document schemata each contain two root elements, one for the standalone item document and one for the item release document.

Assessment Item XML Document and Schema Design Decisions

The format and type attributes of the item element both serve the same purpose. Having two similar attributes is a legacy artifact of the AIR format. Both are retained for compatibility. The value space of the format attribute contains all values for both the format and type attributes to allow the type attribute TO BE DEPRECATED and replaced by the format attribute.

The format attribute value pass is reserved for future use to model passages directly as an *Assessment Item*.

The associatedpassage element contains an item number of the corresponding passage item. How to obtain the passage item or how to select from multiple versions of the passage item with the same item number is missing from the AIR design. A design change to convert the associatedpassage element to an empty element and add a filename attribute or other unique identifier attribute that can be used to obtain the appropriate version of the passage item is a potential change to the element in a future version of the Specification.

Some of the attriblist attributes are *Assessment Item* metadata. The attriblist element is retained for compatibility. Item metadata SHOULD NOT be maintained with the item but with the SBAC item metadata [SBAC Packaging 1.4]. Some attributes MAY be deprecated in a future version of the specification.

The tutorial element id attribute contains an item number of the corresponding *Tutorial*. How to obtain the *Tutorial* or how to select from multiple versions of the *Tutorial* with the same item number is missing from the AIR design. A design change to convert the tutorial element to add a filename attribute or other unique identifier attribute that can be used to obtain the appropriate version of the *Tutorial* is a potential change to the element in a future version of the Specification.

The resource element id attribute contains an item number of the corresponding resource. How to obtain the resource or how to select from multiple versions of the resource with the same item number is missing from the AIR design. A design change to convert the resource element to add a filename attribute or other unique identifier attribute that can be used to obtain the appropriate version of the resource is a potential change to the element in a future version of the Specification.

The resource element type attribute corresponds to the item format. The attribute name has not been changed from type to format.

A *Wordlist* is the only type of resource currently used. A *Tutorial* could be modeled as a resource. The vocabulary for the resource element type attribute includes the value of tutorial as a reserved entry to permit the tutorial element TO BE DEPRECATED and replaced by the resource element in a future version of the Specification.

The original AIR design permitted a MachineRubric to be held in an external file or stored inline within the MachineRubric element. The design has been simplified and a MachineRubric SHALL be stored in an external file.

The original AIR design permitted a RendererSpec to be held in an external file or stored inline within the RendererSpec element. The design has been simplified and a RendererSpec SHALL be stored in an external file.

Instances of AIR rendering specification XML documents used an AIR-specific file extension. All XML documents are stored in files with an .xml file extension. Existing items MAY use a file name extension that is specific to the type of rendering specification.

The gridanswerspace holds an inline rendering specification for a grid item. Eliminating the gridanswerspace element and using the RendererSpec element to reference an external file containing the rendering specification for a grid item is a potential change to the gridanswerspace element in a future version of the Specification.

The version attribute of the content element aligns with the version attribute of the itemrelease element. There are no constraints on the value of the attribute. The version attribute of the content element is a candidate TO BE DEPRECATED and removed in a future version of the specification when the itemrelease element is DEPRECATED.

The original AIR design permitted an illustration to be held in an external file or stored inline within the illustration element. The design has been simplified and an illustration SHALL be stored inline within an illustration element. Allowing an illustration to be stored inline or in an external file would require addition of an attribute describing which option is used.

The original AIR design permitted the stem to be held in an external file or stored inline within the stem element. The design has been simplified and the stem SHALL be stored inline within a stem element. Allowing a stem to be stored inline or in an external file would require addition of an attribute describing which option is used.

Within the rubriclist element, the rubric and samplelist subelements are paired one to one following the ordering of the subelement instances. In the PSVI, the ordering of the input MAY NOT be preserved. An additional index attribute has been added to both the rubric and samplelist subelements to specify order and pairing.

The original AIR design included mixed XML content within the content element. Mixed content is not used. The design has been simplified and mixed content is not permitted.

The pass attribute of the attachment element aligns duplicates other attribute/value combinations. The pass attribute of the attachment element is a candidate TO BE DEPRECATED and removed in a future version of the specification.

The original AIR design included an annotation element. The annotation element is not used. The element is not included in the model.

The original AIR design included concept, es and himi elements. The concept, es and himi elements are not used. These elements are not included in the model.

While the AIR design and the Specification uses only integers [1:2^32-1] for item numbers, the schema also permits strings to allow item numbers to be UUIDs.

xHTML content MAY use one of the xHTML XSDs for validation, e.g., http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd.

Passage Item XML Document and Schema Design Decisions

The *Passage Item* is modeled with a specific root element – passage. The subelements are a subset of the *Assessment Item* elements. The passage item could be converted to an *Assessment Item* with a format attribute of passage in a future version of the Specification.

The format attribute value pass has been added for future use to model passages directly as an *Assessment Item*.

Some of the attriblist attributes are *Passage Item* metadata. The attriblist element is retained for compatibility. Item metadata SHOULD NOT be maintained with the item but with the SBAC item metadata [SBAC Packaging 1.4]. Some attributes MAY be deprecated in a future version of the specification.

The resource element id attribute contains an item number of the corresponding resource. How to obtain the resource or how to select from multiple versions of the resource with the same item number is missing from the AIR design. A design change to convert the resource element to add a filename attribute or other unique identifier attribute that can be used to obtain the appropriate version of the resource is a potential change to the element in a future version of the Specification.

The resource element type attribute corresponds to the item format. The attribute name has not been changed from type to format.

The version attribute of the content element aligns with the version attribute of the itemrelease element. There are no constraints on the value of the attribute. The version attribute of the content element is a candidate TO BE DEPRECATED and removed in a future version of the Specification when the itemrelease element is DEPRECATED.

The Passage Item does not include usage statistics. If the item is converted into a special type of assessment item, the list of assessment items not used SHOULD include the statistics element.

While the AIR design and the Specification uses only integers [1:2^32-1] for item numbers, the schema also permits strings to allow item numbers to be UUIDs.

xHTML content MAY use one of the xHTML XSDs for validation, e.g., http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd.

Tutorial XML Document and Schema Design Decisions

Since a *Tutorial* XML document is an *Assessment Item* XML document, there are no specific design decisions.

Wordlist XML Document and Schema Design Decisions

The format and type attributes of the item element both serve the same purpose. Having two similar attributes is a legacy artifact of the AIR format. The format attribute has been added to an item. The value space of the format attribute contains all values for the type attributes to allow the type attribute TO BE DEPRECATED and removed in a future version of the Specification to align the modeling of formats and types across all assessment items.

Keywords in a wordlist do not include alternatives for accessibility. Adding wordlist accessibility alternatives is a potential change to the keyword element in a future version of the Specification.

While the AIR design and the Specification uses only integers [1:2^32-1] for item numbers, the schema also permits strings to allow item numbers to be UUIDs.

xHTML content MAY use one of the xHTML XSDs for validation, e.g., http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd.

Assessment Item Accessibility XML Document and Schema Design Decisions

The apipAccessibility element has no attributes and only includes the AccessibilityInfo element that also has no attributes and only includes the accessElement element. There are three levels of elements where it appears only two are needed. One of the levels of element nesting could be eliminated. The three levels of element nesting are retained for compatibility.

While there is both an .xsd file containing the types that define an assessment item accessibility XML document and an .xsd file containing the global declarations and global root element, there are no occurrences of a standalone assessment item accessibility XML document. The .xsd file containing the types is used by inclusion in the schemata for other types of documents.

Grid Item Rendering Specification XML Document and Schema Design Decisions

The original AIR design included the ScoreEngineVer attribute for the Question element. The attribute is not used. The attribute is not included in the model.

The Question element version attribute is not used and is candidate to be TO BE DEPRECATED and removed in a future version of the Specification. Until deprecated, the attribute is REQUIRED.

The text element is not used and is candidate to be TO BE DEPRECATED and removed in a future version of the Specification. Until deprecated, the element MAY be used. If used, it is ignored.

The images specified by IconSpec are to be placed in the order specified. In the PSVI, the ordering of the input MAY NOT be preserved. An additional index attribute has been added to the element to specify order.

The filename of an image is specified by the src attribute. Other filenames are specified using a filename attribute or filename element. The src attribute is a candidate TO BE DEPRECATED and removed in a future version of the Specification and replaced by a filename attribute. Until deprecated, the src attribute SHALL be used

The original AIR design included a recursive Question element within the PreSetAnswerPart element. The design was developed to store both a set of default rendering information created during item authoring and a set of student responses captured and presented to the student at a later time. Capture of student responses is not included in the Specification. To simplify the design, the recursive nesting has been eliminated and the Question and QuestionPart elements are not included in the PreSetAnswerPart element.

No simple schema exists that validate that fill and stroke colors conform to [SVG 1.1] and [CSS 2] (all combinations of upper and lower case letters anywhere in the color name is permitted, e.g., whItE). Color names are defined through a restriction whose base type is xsd:string. Color names MUST be validated external to schema validation. A schema MAY validate hex color codes.

While there is both an .xsd file containing the types that define a grid item rendering specification XML document and an .xsd file containing the global declarations and global root element, there are no occurrences of a standalone a grid item rendering specification XML document. The .xsd file containing the types is used by inclusion in the schemata for other types of documents.

Equation Editor Configuration XML Document and Schema Design Decisions

The original AIR design allowed all of the subelements of the editorconfig element to appear in any order. The design has been simplified and the element order is fixed.

The id, fontSizePt and fontSizeHt elements are not used or their values are ignored. These elements are candidates to be TO BE DEPRECATED and removed in a future version of the Specification. Until deprecated, the elements MAY be used.

The name of the magicDisabled element is has no inherit meaning. The element is a candidate to be TO BE DEPRECATED and replaced in a future version of the Specification by a more meaningful name. Until deprecated, the element MAY be used.

The tabs in the input keys panel specified by Order are to be placed in the order specified. In the PSVI, the ordering of the input MAY NOT be preserved. An additional index attribute has been added to the element to specify tab order.

The rows in an input keys panel tab specified by rows are to be placed in the order specified. In the PSVI, the ordering of the input MAY NOT be preserved. An additional index attribute has been added to the element to specify row order.

The items in a row or grid in the input keys panel tab specified by items are to be placed in the order specified. In the PSVI, the ordering of the input MAY NOT be preserved. An additional index attribute has been added to the element to specify item order.

The items in the input keys panel tab specified by items MAY come from a fixed vocabulary of terms without any subelements, or MAY be defined through a list of subelements. Both an item from the vocabulary and subelements in NON Conforming. The XML schema uses a mixed element type for items, but this does not constrain where the values can be place or if a value comes from the constrained vocabulary. The schema also permits both the vocabulary and the subelements.

The original AIR design provided a set of defaults input keys panel tabs and permitted each default to be overridden. Each default tab is defined by its own element (e.g., SBAC3, SBAC4, ...) and a the same set of subelements used to define the custom tab. A simplified design would have a single element with the enumerated list of predefined defaults and a single element and subelements that would provide the definition of the custom tab. Adding elements for such a simplified design and deprecating the specific elements for each tab type (e.g., SBAC3, SBAC4, ...) is a potential change in a future version of the Specification.

There is no XML schema for [TeX]. TeX elements are defined through a restriction whose base type is xsd:string. TeX content MUST be validated external to schema validation.

There is no XML schema for [CSS] content. The css element is defined through a restriction whose base type is xsd:string. The CSS content MUST be validated external to schema validation.

MathML content MAY use the MathML XSD for validation http://www.w3.org/Math/XMLSchema/mathml3/mathml3.xsd.

Assessment Item Usage Statistics XML Document and Schema Design Decisions

While there is both an .xsd file containing the types that define an assessment item usage statistics XML document and an .xsd file containing the global declarations and global root element, there are no occurrences of a standalone assessment item usage statistics XML document. The .xsd file containing the types is used by inclusion in the schemata for other types of documents.

Note: No additional design decisions are included. The *Assessment Item Usage Statistics* XML document specification is not included in the Specification.

Assessment Item Machine Rubric XML Document and Schema Design Decisions

Instances of AIR machine rubric XML documents used an AIR-specific file extension. Machine rubrics XML files have been assigned an .XML file name extension and corresponding MIME type. Existing items MAY use a file name extension that is specific to the type of the machine rubric.

Note: No additional design decisions are included. The *Assessment Item Machine Rubric* XML document specification is not included in the Specification.

Annex: XML Schemata

Note: This section is informative.

XML Schema Definitions (XSDs) [XSD 1] MAY be used to describe part of the XML document model for assessment items. An XSD is insufficient to represent the entire document model. The schemata are insufficient to determine if an XML document is fully conformant to the Specification. Different XSDs can be equivalent; there is no unique XSD.

A sample set of schemata has been developed that represent the Specification. An XML document that uses these schemata conforms to part of the Specification. A conformant XML document need not use these specific schemata. These schemata are informative.

The sample schemata are available on the web at the URIs listed in Table A.2.

Table A.2: XSD Schema Locations

Schema	Version	Schema Location
Assessment Item	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitem_v1p0.xsd
Passage Item	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/itempassage_v1p0.xsd
Tutorial	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/tutorial_v1p0.xsd
Wordlist	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/wordlist_v1p0.xsd
Assessment Item Accessibility	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/accessibility_v1p0.xsd
Grid Item Rendering Specification	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/griditemrenderingspec_v1p0.xsd
Equation Editor Configuration	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfig_v1p0.xsd
Assessment Item Usage Statistics	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/usagestatistic_v1p0.xsd
Assessment Item Machine Rubric	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/machinerubric_v1p0.xsd

Each sample schema contains only global declarations and the global root element definition. The rest of schema is defined in a separate file defining element types and attributes. There is a common schema for types that are shared across all schemata and a common schema for types that are just shared across the different types of items (assessment item, passage item, wordlist, tutorial). The files locations are listed in Table A.3.

Table A.3: XSD Element Type Schema Locations

Schema Types	Version	Element Type Schema Location
Assessment Item Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/assessmentitemtypes_v1p0.xsd
Passage Item Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/passageitemtypes_v1p0.xsd
Tutorial Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/tutorialtypes_v1p0.xsd
Wordlist Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/wordlisttypes_v1p0.xsd
Assessment Item Accessibility Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/accessibilitytypes_v1p0.xsd

Schema Types	Version	Element Type Schema Location
Grid Item Rendering Specification Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/griditemrenderingspectypes_v1p0.xsd
Equation Editor Configuration Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/equationeditorconfigtypes_v1p0.xsd
Assessment Item Usage Statistics Type	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/usagestatisticstypes_v1p0.xsd
Assessment Item Machine Rubric Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/machinerubrictypes_v1p0.xsd
SAAIF Shared Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/saaifcommontypes_v1p0.xsd
SAAIF Item Types	1.0.0	http://www.smarterapp.org/xsd/saaif/v1p0/saaifitemtypes_v1p0.xsd

The structural relationships between the schemata is illustrated in Figure A.7. The top row of the diagram shows the different document types. Each references a single schema shown in the second row. Each of these schemata link to a basic corresponding *type* schema, shown in the third row. The shared elements used in multiple schemata are shown in the fourth row. The SAAIF schemata are linked using xsd:include. And the bottom row shows external (non SAAIF) schemata used incorporated via xsd:import.

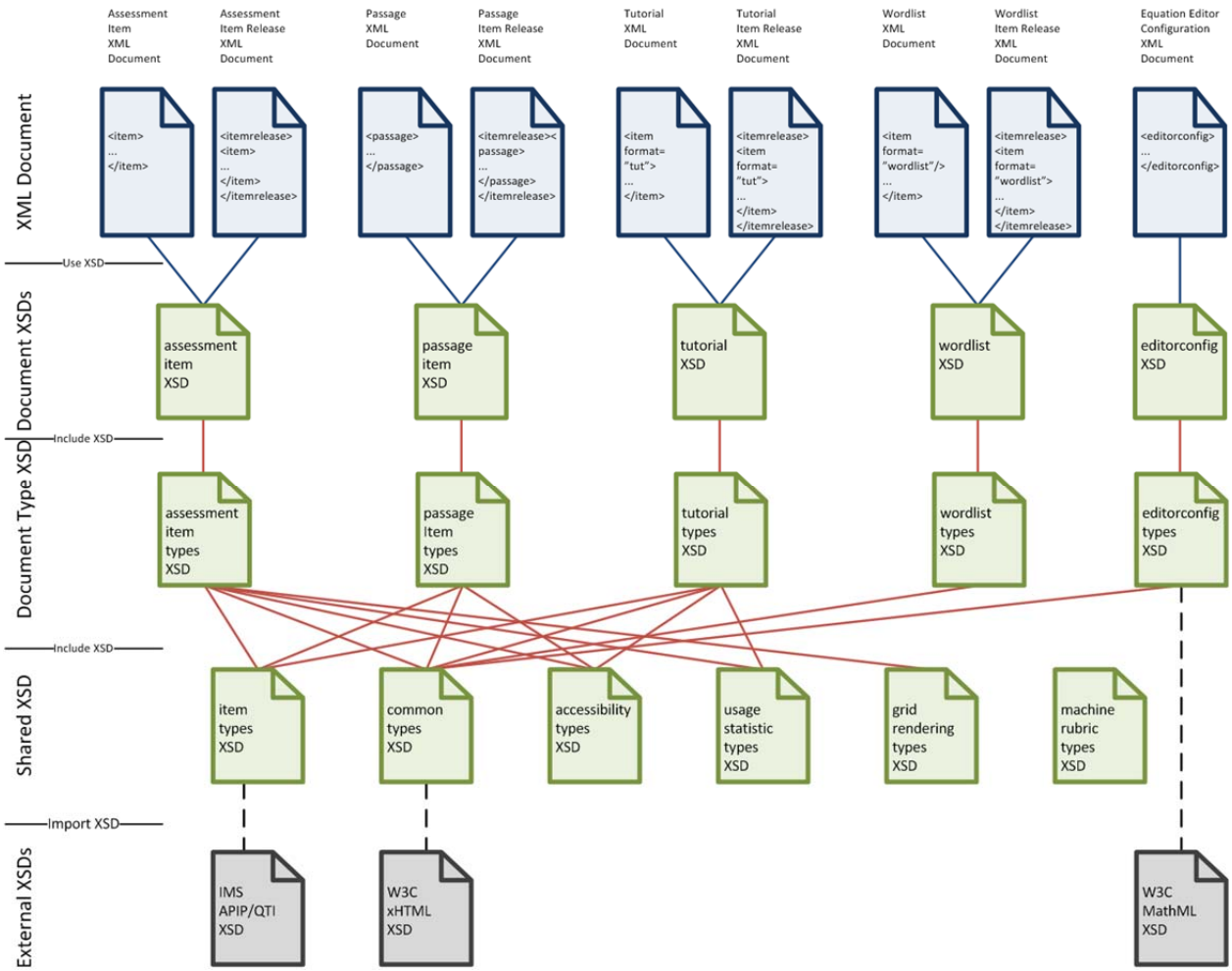


Figure A.7: Schemata Structure

The schemata are available for download or for direct references from an XML document using the schemaLocation attribute.

Multiple versions of each schema MAY exist, with the schema location following the schema versioning strategy. The schemata MAY be updated to reflect changes, errata or new versions. Users are advised to verify the version of any schema they use.

Annex: XML DTDs

Note: This section is informative.

XML Document Type Definitions (DTDs) MAY be used to describe part of the XML document model for assessment items. A DTD is insufficient to represent the entire document model. DTDs are insufficient to determine if an XML document is fully conformant to the Specification. Different DTDs can be equivalent.

An XML document that uses a DTD SHOULD include a DOCTYPE element referencing or including the appropriate DTD.

A future version of the Specification MAY include an informative set of DTDs. These DTDs SHALL conform to the Specification. An XML document that uses these DTDs conforms to part of the Specification. A conformant XML document need not use these specific DTDs. The files locations are listed in Table A.4.

Table A.4: DTD Locations

DTD	Version	Future DTD Location
Assessment Item Release	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0/assessmentitemrelease_v1p0.dtd
Assessment Item	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0assessmentitem_v1p0.dtd
Passage Item	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0itempassage_v1p0.dtd
Tutorial	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0tutorial_v1p0.dtd
Wordlist	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0wordlist_v1p0.dtd
Assessment Item Accessibility	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0/apip_v1p0.dtd
Grid Item Rendering Specification	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0griditemrenderingspec_v1p0.dtd
Equation Editor Configuration	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0equationeditorconfig_v1p0.dtd
Assessment Item Usage Statistics	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0usagestatistics_v1p0.dtd
Assessment Item Machine Rubric	1.0.0	http://www.smarterapp.org/dtd/saaif/v1p0machinerubric_v1p0.dtd

Once developed, the DTDs will be available for download or for direct references from an XML document using the DOCTYPE element.

Multiple versions of each DTD MAY exist, with the DTD location following the DTD versioning and naming strategy. The DTDs MAY be updated to reflect changes, errata or new versions. Users are advised to verify the version of any DTD they use.

DTD Versioning Strategy

DTD names and locations encode the DTD major and minor version in the location. The use of the pattern v\d{1}p\d{1} to encode the version number in the DTD name and locations follows the

convention used by IMS and parallels the convention used to identify versions of schemata. Different specification or DTD versions have different values of the DTD name and DTD location.

Index: XML Elements and Attributes

Note: Page numbers in bold indicate the location where the XML element is defined. Page numbers in italics indicate the location where the XML attribute is defined. If an index entry does not have a bold or italics page number, the item is both defined and referenced on the single page indicated.

Change Log

Date	Version	Author	Notes
20131107	0.35	DR	Baseline working document for AIR information gathering.
20131128	0.40	DR, JD	Incorporate initial AIR information on element descriptions. Internal release for SBAC stakeholders.
20140205	0.50	DR, JD	Incorporate additional AIR information on element descriptions. Add passage item document type. Editorial updates. Document organizational changes. Technical revisions and clarifications throughout. Unreleased.
20140224	0.60	DR, JD, DL	Incorporate additional AIR information on element descriptions. Add tutorial and wordlist item document types. Editorial updates. Document organizational changes. Technical revisions and clarifications throughout. Unreleased,
20140224	0.61	DR	Prerelease.
20140411	0.65	DR, JD, DL	Incorporate additional AIR information on element descriptions. Editorial updates. Technical revisions and clarifications throughout. Unreleased.
20140422	0.70	DR	Editorial updates. Technical revisions and clarifications throughout. Editorial review. Prerelease.
20140602	0.75	DR, JD, DL	Incorporate additional AIR information on element descriptions. Editorial updates. Technical revisions and clarifications throughout. Change from SBAIF to SAAIF Prerelease.
20140812	0.80	DR, JD, DL	Incorporate additional AIR information on element descriptions. Editorial updates. Technical revisions and clarifications throughout. Incorporate updates from XSD development. Project Draft.
2014xxxx	0.90	DR	Editorial review. Technical review. XSD review. Public Draft.
2014xxxx	1.0		Incorporate updates from feedback. Incorporate updates from XSD use. V1.0 Public Release.
Future	Future		Update all © Notices (Document, Examples, XSDs) to reflect final © owner. Update any references of SBAC Packaging to SAAIF Packaging. Replace example illustration placeholders with real illustrations. Add details of Machine Rubrics XML Document. Add details of Assessment Item Usage Statistics XML Document.