



IMS Question and Test Interoperability XML Binding

Version 2.1 Public Draft Specification

Copyright © 2006 IMS Global Learning Consortium, Inc. All Rights Reserved.
The IMS Logo is a registered trademark of IMS GLC.
Document Name: IMS Question and Test Interoperability XML Binding

Date Issued: 8 June 2006

Caution: this specification is incomplete in its current state. The IMS QTI project group is in the process of evolving this specification based on input from market participants. Suppliers of products and services are encouraged to participate by contacting Mark McKell at mmckell@imglobal.org. This specification will be superseded by an updated release based on the input of the project group participants.

Please note that supplier's claims as to implementation of QTI v2.1 and conformance to it HAVE NOT BEEN VALIDATED by IMS GLC. While such suppliers are likely well-intentioned, IMS GLC member organizations have not yet put in place the testing process to validate these claims. IMS GLC currently grants a conformance mark to the Common Cartridge profile of QTI v1.2.1. The authoritative source of products and services that meet this conformance is contained in the IMS online product directory
<http://www.imglobal.org/ProductDirectory/directory.cfm>

Thank you for your interest in and support of IMS QTI.

IPR and Distribution Notices

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

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

Copyright © 2006 IMS Global Learning Consortium. All Rights Reserved.

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

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

Use of this specification to develop products or services is governed by the license with IMS found on the IMS

website: <http://www.imsglobal.org/license.html>.

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

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

Table of Contents

1. [Introduction](#)
2. [References](#)
3. [Binding of Complex Types](#)
4. [Binding of Simple Types](#)
 - 4.1. [boolean](#)
 - 4.2. [coords](#)
 - 4.3. [date](#)
 - 4.4. [datetime](#)
 - 4.5. [duration](#)
 - 4.6. [float](#)
 - 4.7. [identifier](#)
 - 4.8. [integer](#)
 - 4.9. [language](#)
 - 4.10. [length](#)
 - 4.11. [mimeType](#)
 - 4.12. [string](#)
 - 4.13. [string256](#)
 - 4.14. [styleclass](#)
 - 4.15. [uri](#)
 - 4.16. [valueType](#)
5. [Binding Values of Item Variables](#)
 - 5.1. [boolean](#)
 - 5.2. [directedPair](#)
 - 5.3. [duration](#)
 - 5.4. [file](#)
 - 5.5. [float](#)
 - 5.6. [identifier](#)
 - 5.7. [integer](#)
 - 5.8. [pair](#)
 - 5.9. [point](#)
 - 5.10. [string](#)
 - 5.11. [uri](#)
6. [XML Binding Reference](#)

1. Introduction

The information model defines a set of classes and data types in an abstract way. When faced with the real task of communicating data between systems these abstract concepts need to be bound to some *physical* representation, such as a stream of bytes in a computer file.

This document describes the required method of binding the information model when the *physical* representation is an XML Document. The binding rules are given primarily in the form of an XML Schema with the following target namespace:

http://www.imsglobal.org/xsd/imsqti_v2p1. To assist developers who wish to check the validity of QTI XML documents using tools that support document type definitions (DTDs) the binding rules are also represented as a DTD, however, the DTD is not able to express all the constraints contained in the schema and document instances should validate against the schema. QTI document instances *must* validate against one of the schema or DTD.

2. References

ISO8601

ISO8601:2000 Data elements and interchange formats – Information interchange –
Representation of dates and times
Published: 2000

MathML

Mathematical Markup Language (MathML), Version Version 2.0 (Second Edition)
<http://www.w3.org/TR/2003/REC-MathML2-20031021/>
Published: 2003-10-21

RFC1766

RFC 1766 Tags for the Identification of Languages
H. Alvestrand
<http://www.ietf.org/rfc/rfc1766.txt>
Published: 1995-03

RFC2045

RFC 2045-2048 Multipurpose Internet Mail Extensions (MIME)

RFC3066

RFC 3066 Tags for the Identification of Languages
H. Alvestrand
<http://www.ietf.org/rfc/rfc3066.txt>
Published: 2001-01

XHTML

XHTML 1.1: The Extensible HyperText Markup Language

3. Binding of Complex Types

Complex types, those that contain other objects through composition and/or have attributes, are represented in the schema as elements with complex content in a straightforward way. Classes that are used as base classes (typically labeled *abstract* in the model) are used to define an element group consisting of a single choice listing all the non-abstract classes derived from them. Classes with attributes are used to define attribute groups and those that contain other objects (including through inheritance) are used to define element groups describing them.

Elements defined by [\[XHTML\]](#) are *not* imported directly from an external schema but are instead re-declared as part of the QTI schema. At the time of writing the use of schema for XHTML is at an experimental stage. However, the math element defined by [\[MathML\]](#) is imported from the MathML schema. Note that the use of the MathML schema excludes the use of the entity declarations for mathematical symbols defined in the original MathML DTD and therefore these should not be used for referring to characters within QTI document instances.

4. Binding of Simple Types

Simple datatypes are bound directly to the closest equivalent type in XML Schema, for example, [boolean](#) is bound directly to `xsd:boolean`. Enumerations are bound as enumerated restrictions of `xsd:NMTOKEN`.

The information model says nothing about how simple types are to be represented in text. For example, the [boolean](#) type is an abstract boolean which can be either logically true or logically false. By binding these simple types using XML Schema they automatically inherit the lexical bindings described there. A lexical binding is just a set of rules for representing the abstract type as a run of text characters. For example, XML Schema allows the following strings to represent the values of `xsd:boolean`: "true", "false", "1" and "0".

XML DTDs have a smaller vocabulary for defining simple types and is therefore not able to impose the same lexical constraints as the schema. Applications that generate QTI document instances that validate against the DTD must still use the same lexical bindings defined by schema even though these will not be checked automatically by the XML parser. For example, in the DTD the [boolean](#) type is bound only to NMTOKEN and therefore the XML parser will permit many more strings than the four listed above. However, strings such as "yes" and "no" are still forbidden.

The following sections spell out the bindings for the simple types defined in the information model. Simple types are defined as datatypes in the information model and are used as the types of attributes (which are bound to attributes in the XML representation) or as the types of simple aggregations (which are bound to elements without attributes in the XML representation). Where the information model indicates multiple occurrences of a simple type these are bound using space separation of individual values in attributes or multiple instances of the element in the case of aggregations.

4.1. boolean

Bound using the XML schema *boolean*. The legal literals are defined there to be:

```
true
false
1
0
```

4.2. coords

The [coords](#) datatype is bound as a single string value which is obtained by concatenating a list of *length* values using **comma** as the list separator. Note that the use of comma is for compatibility with [XHTML](#). Example values:

```
10,100,50,200
50,60,50%
60,50,110,100,60,150,10,150,60,50
50%,50%,20,25
```

4.3. date

Bound using the XML schema *date* type. The legal literals are defined as a subset of [\[ISO8601\]](#). Note that only *extended* calendar forms are supported (forms with hyphen field delimiters) and that all components of the date must be specified. The information model value space for the date type does not include time zone information or expanded forms (negative dates and dates beyond 9999) so these are excluded from the binding. Example values:

2006-03-22

4.4. datetime

Bound using the XML schema *dateTime* type. Expanded forms should not be used. Example values:

2006-03-22T12:33:25TZ
2006-03-022T12:33:25+01:00

4.5. duration

Bound using the XML schema *double* type. Example values:

3600.0
5400.0
7.2E+02

4.6. float

Bound using the XML schema *double* type. Note that the range of values accepted by float corresponds to XML schema's double type and not it's float type.

1
3.14
3E+08

4.7. identifier

Identifiers are defined in terms of allowable sequences of characters and are bound in the schema to *NCName*. Example values:

A
choice4
correct

4.8. integer

Bound using the XML schema *int* type. Example values:

1
+3
-4

4.9. language

Bound using the XML schema *language* type. Languages are represented according to the conventions of [RFC1766](#) which itself has been made obsolete by [RFC3066](#). Example values:

en-US
en-GB
en

4.10. length

Length values are represented as non-negative integers, optionally followed by the percent character. There is no corresponding type in XML schema, the binding is through a pattern restriction of the basic *string* type. Lengths must match the following pattern `[0-9]+%?`. Example values:

10
50%

4.11. mimeType

Represented as strings conforming to the definition of *type "/" subtype* in [RFC2045](#). Example values:

text/plain
image/gif
application/octet-stream

4.12. string

Bound using the XML schema *string* type.

4.13. string256

Bound using a simple restriction of the XML schema *string* type.

4.14. styleclass

Bound using the XML schema *string* type, though styleclass values cannot contain the space character as this is interpreted as a list separator when multiple values are given.

4.15. uri

Bound using the XML schema *anyURI* type. Example values:

`http://www.example.com/
images/icon.gif
mailto:user@example.com`

4.16. valueType

See [Binding Values of Item Variables](#).

5. Binding Values of Item Variables

This specification defines a model for maintaining the values of item variables during a test or item session. Item variables may have single values, or may be containers for multiple values. In each case, the single values are represented using instances of the class [value](#) in declarations and reports, the class [baseValue](#) when used in expressions and the datatype [valueType](#) when given as an attribute value. The following discussion applies to all three contexts.

Each value instance can contain a single value of any of the [baseTypes](#) supported by the specification. The content of a `<value>` *element* is therefore bound to the general *string* type in XML schema. However, the element content must satisfy the binding for the associated baseType as described below.

5.1. boolean

As for the simple type [boolean](#).

5.2. directedPair

A space separated list of two identifier values with the source identifier first, followed by the destination identifier. Example values:

```
A B
choice5 choice1
apple fruit
carrot vegetable
```

5.3. duration

The number of seconds, expressed as a floating point number and bound according to the rules for the simple type [float](#).

5.4. file

Files are represented using MIME message streams, including the specification of headers for providing information about content type and encoding. For more information see [\[RFC2045\]](#). Example value:

```
Content-Type: image/gif; name="works.gif" ; x-mac-type="47494666"
; x-mac-creator="4A565752"
Content-Disposition: attachment; filename="works.gif"
Content-Transfer-Encoding: base64
```

```
R0lGODlhGgAXAPcAAP////////zP//mf//Zv//M//AP/M//MzP/Mmf/MZv/MM//MAP+Z
//+ZzP+Zmf+ZZv+ZM/+ZAP9m//9mzP9mmf9mZv9mM/9mAP8z//8zzP8zmf8zZv8zM/8z
AP8A//8AzP8Amf8AZv8AM/8AAMz//8z/zMz/mcz/Zsz/M8z/AMzM/8zMzMzMmczMZszM
M8zMAMyZ/8yZzMyZmcyZZsyZM8yZAMxm/8xmzMxmMcmZsxmM8xmAMwz/8wzzMwzmcwz
ZswzM8wzAMwA/8wAzMwAmcwAZswAM8wAAJn//5n/zJn/mZn/Zpn/M5n/AJnM/5nMzJnM
mZnMZpnMM5nMAJmZ/5mZzJmZmZmZpmZM5mZAJlm/5lmzJlmmZlmZplmM5lmAJkz/5kz
```

zJkzmZkzZpkzM5kzAJkA/5kAzJkAmZkAZpkAM5kAAGb//2b/zGb/mWb/Zmb/M2b/AGbM
 /2bMzGbMmWbMZmbMM2bMAGaZ/2aZzGaZmWaZZmaZM2aZAGZm/2ZmzGZmmWZmZmZmM2Zm
 AGYz/2YzzGYzmWYzZmYzM2YzAGYA/2YAzGYAmWYAZmYAM2YAADP//zP/zDP/mTP/ZjP/
 MzP/ADPM/zPMzDPMmTPMZjPMMzPMADOZ/zOZzDOZmTOZjOZMzOZADNm/zNmzDNmmTNm
 ZjNmMzNmADMz/zMzZDMzmTMzZjMzMzMzADMA/zMAzDMAmTMAZjMAMzMAAAD//wD/zAD/
 mQD/ZgD/MwD/AADM/wDMzADMmQDMZgDMMwDMAACZ/wCZzACZmQCZGZCZMwCZAABm/wBm
 zABmmQBmZgBmMwBmAAAz/wAzZAAzmQAZgAZMwAZAAAA/wAAzAAAmQAAZgAAM+4AAN0A
 ALsAAKoAAIgaAHcAAFUAAEQAACIAABEAAADuAADdAAC7AACqAACIAAB3AABVAABEAAAi
 AAARAAA7gAA3QAAUwAAqgAAiAAAdwAAVQAARAAAiGAEE7u7t3d3bu7u6qqqoiIiHd3
 d1VVVURERCiiIHEREQAAACH5BAEAAAEALAAAAAAaABcABwigAAMIHEgwwL+D/woqXEjw
 HzZsCRlKHojwIcSJEys+jIhRYUUAADZ29PgQpMiRFEuCDHkR5ceVJlt2/OivZkyOGVXW
 9LfyZE5sIHf2lMnw5UqbN3/CBCCUJc6CRoPu5Om06EOqTKdS9dnwqlapWLka9Dr1aNOq
 KbFppWoTKdqxatdKnZsU7tq2bMNe1Gixr9+/Fg8CHkwYoeHdiBMfBAGgIAA7

5.5. float

As for the simple type [float](#).

5.6. identifier

As for the simple type [identifier](#).

5.7. integer

As for the simple type [integer](#).

5.8. pair

A space separated list of two identifier values.

```
A B
choice5 choice1
apple pear
carrot potato
```

5.9. point

A space separated list of two integer values, bound in the same way as the XML schema type *int*. The first value is the horizontal component, the second the vertical.

```
10 50
32 16
```

5.10. string

As for the simple type [string](#).

5.11. uri

As for the simple type [uri](#).

6. XML Binding Reference

<a>

Description: Defined by XHTML

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **href (required).**
Data type = uri.Type
- **type (optional).**
Data type = mimeType.Type

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<abbr>

Description: Defined by XHTML

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)

- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<acronym>

Description: Defined by XHTML

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<adaptive>

Description: Profile flag for the adaptive items feature.

Context:

- Occurs [1] within [contentProfile](#)

<address>

Description: Defined by XHTML

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<and>

Description: An expression that performs a boolean and.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<anyN>

Description: An expression for testing groups of boolean sub-expressions.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **min (required).**
Data type = integerOrTemplateRef.Type
- **max (required).**
Data type = integerOrTemplateRef.Type

Elements:

- Members of [expression.ElementGroup](#)

<areaMapEntry>

Description: Defines an area for mapping a set of points onto a single real number.

Context:

- Occurs [1..*] within [areaMapping](#)

Attributes:

- **shape (required).**
Data type = shape.Type
- **coords (required).**
Data type = coords.Type
- **mappedValue (required).**
Data type = float.Type

<areaMapping>

Description: Defines a mapping from the set of points to real numbers using areas.

Context:

- Occurs [0..1] within [responseDeclaration](#)

Attributes:

- **lowerBound (optional).**
Data type = float.Type
- **upperBound (optional).**
Data type = float.Type
- **defaultValue (required).**
Data type = float.Type

Elements:

- [<areaMapEntry>](#)

<assessmentItem>

Description: The root element for QTI items

Context:

Can be the root element of an XML document.

Attributes:

- **identifier (required).**
Data type = string.Type
- **title (required).**
Data type = string.Type
- **label (optional).**
Data type = string256.Type
- **xml:lang (optional).**
- **adaptive (required).**
Data type = boolean.Type
- **timeDependent (required).**
Data type = boolean.Type
- **toolName (optional).**
Data type = string256.Type
- **toolVersion (optional).**
Data type = string256.Type

Elements:

- [<responseDeclaration>](#)
- [<outcomeDeclaration>](#)
- [<templateDeclaration>](#)
- [<templateProcessing>](#)
- [<stylesheet>](#)
- [<itemBody>](#)
- [<responseProcessing>](#)
- [<modalFeedback>](#)

<assessmentItemRef>

Description: The element used to incorporate an item into a test.

Context:

See:

- [sectionPart.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **required (optional).**
Data type = boolean.Type
- **fixed (optional).**
Data type = boolean.Type
- **href (required).**
Data type = uri.Type
- **category (optional).**
Data type = list of identifier.Type

Elements:

- [<preCondition>](#)
- [<branchRule>](#)
- [<itemSessionControl>](#)
- [<timeLimits>](#)
- [<variableMapping>](#)
- [<weight>](#)
- [<templateDefault>](#)

<assessmentResult>**Context:**

Can be the root element of an XML document.

Elements:

- [<context>](#)
- [<testResult>](#)
- [<itemResult>](#)

<assessmentSection>

Description: A section of a test.

Context:

- Occurs [1..*] within [testPart](#)

See:

- [sectionPart.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **required (optional).**
Data type = boolean.Type
- **fixed (optional).**
Data type = boolean.Type
- **title (required).**
Data type = string.Type
- **visible (required).**
Data type = boolean.Type
- **keepTogether (optional).**
Data type = boolean.Type

Elements:

- [<preCondition>](#)
- [<branchRule>](#)
- [<itemSessionControl>](#)
- [<timeLimits>](#)
- [<selection>](#)
- [<ordering>](#)
- [<rubricBlock>](#)
- Members of [sectionPart.ElementGroup](#)

<assessmentTest>

Description: The root element for QTI tests.

Context:

Can be the root element of an XML document.

Attributes:

- **identifier (required).**
Data type = string.Type
- **title (required).**
Data type = string.Type
- **toolName (optional).**
Data type = string256.Type
- **toolVersion (optional).**
Data type = string256.Type

Elements:

- [<outcomeDeclaration>](#)
- [<timeLimits>](#)
- [<testPart>](#)
- [<outcomeProcessing>](#)
- [<testFeedback>](#)

associableChoice.ElementGroup

Description: The group of elements that are used in association-type interactions.

Group members: [<associableHotspot>](#), [<gap>](#), [<gapImg>](#), [<gapText>](#), [<simpleAssociableChoice>](#)

Context:

Not applicable.

<associableHotspot>

Description: An element that defines a hot area of a graphic for association tasks.

Context:

- Occurs [1..*] within [graphicAssociateInteraction](#)
- Occurs [1..*] within [graphicGapMatchInteraction](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **matchGroup (optional).**
Data type = list of identifier.Type
- **shape (required).**
Data type = shape.Type
- **coords (required).**
Data type = coords.Type
- **hotspotLabel (optional).**
Data type = string256.Type
- **matchMax (required).**
Data type = integer.Type
- **matchMin (optional).**
Data type = integer.Type

<associateInteraction>

Description: Interaction in which the candidate must identify pairs of choices from a single set.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type

- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **shuffle (required).**
Data type = boolean.Type
- **maxAssociations (required).**
Data type = integer.Type
- **minAssociations (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<simpleAssociableChoice>](#)

atomicBlock.ElementGroup

Description: A group of elements that provide structure but that can only contain elements behaving like spans of text.

Group members: [<address>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<p>](#), [<pre>](#)

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

atomicInline.ElementGroup

Description: A group of empty elements that behave like spans of text.

Group members: [
](#), [](#)

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)

- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<bankProfile>

Description: An element for describing the capabilities or requirements of an item-bank system.

Context:

Can be the root element of an XML document.

Elements:

- [<usageDataVocabulary>](#)
- [<metadataProfile>](#)
- Elements from other namespaces

baseType.Type

One of the following symbolic values:

- boolean
- directedPair
- duration
- file
- float
- identifier
- integer
- pair
- point
- string
- uri

<baseValue>

Description: A simple expression returning a constant value of the given base-type.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **baseType (required).**
Data type = baseType.Type

<big>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

block.ElementGroup

Description: A group of elements that provide structure to the item body.

Group members: [<address>](#), [<associateInteraction>](#), [<blockquote>](#), [<choiceInteraction>](#), [<customInteraction>](#), [<div>](#), [<dl>](#), [<drawingInteraction>](#), [<extendedTextInteraction>](#), [<feedbackBlock>](#), [<gapMatchInteraction>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hotspotInteraction>](#), [<hottextInteraction>](#), [<hr>](#), [<m:math>](#), [<matchInteraction>](#), [<mediaInteraction>](#), [](#), [<orderInteraction>](#), [<p>](#), [<positionObjectStage>](#), [<pre>](#), [<rubricBlock>](#), [<selectPointInteraction>](#), [<sliderInteraction>](#), [<table>](#), [<templateBlock>](#), [](#), [<uploadInteraction>](#)

Context:

- Occurs [*] within [itemBody](#)
- Occurs [*] within members of [simpleBlock.ElementGroup](#)

blockInteraction.ElementGroup

Description: The group of interactions which have inherent structure of their own.

Group members: [<associateInteraction>](#), [<choiceInteraction>](#), [<drawingInteraction>](#), [<extendedTextInteraction>](#), [<gapMatchInteraction>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<hotspotInteraction>](#), [<hottextInteraction>](#), [<matchInteraction>](#), [<mediaInteraction>](#), [<orderInteraction>](#), [<selectPointInteraction>](#), [<sliderInteraction>](#), [<uploadInteraction>](#)

Context:

See:

- [block.ElementGroup](#)

- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

blockStatic.ElementGroup

Description: A group of elements that provides structure excluding interactions.

Group members: [<address>](#), [<blockquote>](#), [<div>](#), [<dl>](#), [<feedbackBlock>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hr>](#), [<m:math>](#), [](#), [<p>](#), [<pre>](#), [<rubricBlock>](#), [<table>](#), [<templateBlock>](#), [](#)

Context:

- Occurs [1..*] within [gapMatchInteraction](#)
- Occurs [1..*] within [hottextInteraction](#)
- Occurs [*] within [templateBlock](#)

See:

- [block.ElementGroup](#)

<blockquote>

Description: Defined by XHTML

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **cite (optional).**
Data type = uri.Type

Elements:

- Members of [block.ElementGroup](#)

bodyElement.ElementGroup

Description: The group of elements used to describe the structure, material, and interactions within the item body.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [<address>](#), [<associableHotspot>](#), [<associateInteraction>](#), [](#), [<big>](#), [<blockquote>](#), [
](#), [<caption>](#), [<choiceInteraction>](#), [<cite>](#), [<code>](#), [<col>](#), [<colgroup>](#), [<customInteraction>](#), [<dd>](#), [<dfn>](#), [<div>](#), [<dl>](#), [<drawingInteraction>](#), [<dt>](#), [](#), [<endAttemptInteraction>](#), [<extendedTextInteraction>](#), [<feedbackBlock>](#), [<feedbackInline>](#), [<gap>](#), [<gapImg>](#), [<gapMatchInteraction>](#), [<gapText>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hotspotChoice>](#), [<hotspotInteraction>](#), [<hottext>](#), [<hottextInteraction>](#), [<hr>](#), [<i>](#), [](#), [<inlineChoice>](#), [<inlineChoiceInteraction>](#), [<itemBody>](#), [<kbd>](#), [](#), [<matchInteraction>](#), [<mediaInteraction>](#), [<object>](#), [](#), [<orderInteraction>](#), [<p>](#), [<positionObjectInteraction>](#), [<pre>](#), [<printedVariable>](#), [<prompt>](#), [<q>](#), [<rubricBlock>](#), [<samp>](#), [<selectPointInteraction>](#), [<simpleAssociableChoice>](#), [<simpleChoice>](#), [<sliderInteraction>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<table>](#), [<tbody>](#), [<td>](#), [<templateBlock>](#), [<templateInline>](#), [<textEntryInteraction>](#), [<tfoot>](#), [<th>](#), [<thead>](#), [<tr>](#), [<tt>](#), [](#), [<uploadInteraction>](#), [<var>](#)

Context:

Not applicable.

boolean.Type

A trivial restriction of xsd:boolean.

**
**

Description: Defined by XHTML

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

<branchRule>**Context:**

- Occurs [*] within members of [sectionPart.ElementGroup](#)
- Occurs [*] within [testPart](#)

Attributes:

- **target (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

<candidateResponse>**Context:**

- Occurs [1] within [responseVariable](#)

Elements:

- [<value>](#)

<caption>

Description: Defined by XHTML.

Context:

- Occurs [0..1] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

cardinality.Type

One of the following symbolic values:

- multiple
- ordered
- record
- single

<categorizedStatistic>

Description: A categorized statistic, such as a score-conversion table, expressed as a mapping.

Context:

See:

- [itemStatistic.ElementGroup](#)

Attributes:

- **name (required).**
Data type = identifier.Type
- **glossary (optional).**
Data type = uri.Type
- **context (required).**
Data type = uri.Type
- **caseCount (optional).**
Data type = integer.Type
- **stdError (optional).**
Data type = float.Type
- **stdDeviation (optional).**
Data type = float.Type

- **lastUpdated (optional).**
Data type = date.Type

Elements:

- [<targetObject>](#)
- [<mapping>](#)

choice.ElementGroup

Description: The group of elements that provides choices from which the candidate may select a response.

Group members: [<associableHotspot>](#), [<gap>](#), [<gapImg>](#), [<gapText>](#), [<hotspotChoice>](#), [<hottext>](#), [<inlineChoice>](#), [<simpleAssociableChoice>](#), [<simpleChoice>](#)

Context:

Not applicable.

<choiceInteraction>

Description: The element that represents simple multiple choice and multiple-response interactions.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **shuffle (required).**
Data type = boolean.Type

- **maxChoices (required).**
Data type = integer.Type
- **minChoices (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<simpleChoice>](#)

<cite>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<code>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<col>

Description: Defined by XHTML.

Context:

- Occurs [*] within [colgroup](#)
- Occurs [*] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **span (optional).**
Data type = integer.Type

<colgroup>

Description: Defined by XHTML.

Context:

- Occurs [*] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **span (optional).**
Data type = integer.Type

Elements:

- [<col>](#)

<composite>

Description: Profile flag for the composite items feature.

Context:

- Occurs [1] within [contentProfile](#)
- Occurs [0..1] within [qtiMetadata](#)

<containerSize>

Description: An expression that returns the size of a container.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<contains>

Description: An expression that tests a container to determine if it contains another.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<contentProfile>

Description: An element for describing the capabilities or requirements of a system that handles QTI content.

Context:

Can be the root element of an XML document.

Elements:

- [<composite>](#)
- [<adaptive>](#)
- [<timeDependent>](#)
- [<templates>](#)
- [<textElements>](#)
- [<listElements>](#)
- [<objectElements>](#)
- [<objectType>](#)
- [<presentationElements>](#)
- [<tableElements>](#)
- [<imageElement>](#)
- [<imageType>](#)
- [<hypertextElement>](#)
- [<mathElement>](#)
- [<mathVariable>](#)
- [<feedbackIntegrated>](#)
- [<feedbackModal>](#)
- [<rubric>](#)
- [<printedVariables>](#)
- [<interactionType>](#)
- [<responseRules>](#)
- [<rpTemplate>](#)
- [<rounding>](#)
- [<regexp>](#)
- [<metadataProfile>](#)
- Elements from other namespaces

<context>

Context:

- Occurs [1] within [assessmentResult](#)

Elements:

- [<sessionIdentifier>](#)
- [<identification>](#)

coords.Type

A trivial restriction of xsd:string.

<correct>

Description: A simple expression returning the correct value associated with a response variable.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

<correctResponse>

Description: Declares the (or a) correct response for a response variable.

Context:

- Occurs [0..1] within [responseDeclaration](#)
- Occurs [0..1] within [responseVariable](#)

Attributes:

- **interpretation (optional).**
Data type = string.Type

Elements:

- [<value>](#)

<customInteraction>

Description: An element that allows extensions to QTI to be developed for interaction types not covered by the information model.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- Attributes from other namespaces

Elements:

- Elements from any namespace

<customOperator>

Description: An element allowing externally defined expressions to be used.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **class (optional).**
Data type = identifier.Type

- **definition (optional).**
Data type = uri.Type
- Attributes from other namespaces

Elements:

- Members of [expression.ElementGroup](#)

date.Type

A trivial restriction of xsd:date.

datetime.Type

A trivial restriction of xsd:dateTime.

<dd>

Description: Defined by XHTML.

Context:

See:

- [dlElement.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [flow.ElementGroup](#)
- PCDATA

<default>

Description: A simple expression returning the default value of an item variable.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

<defaultValue>

Description: Defines the default value for an item variable.

Context:

- Occurs [0..1] within members of [variableDeclaration.ElementGroup](#)

Attributes:

- **interpretation (optional).**
Data type = string.Type

Elements:

- [<value>](#)

<delete>

Description: An expression that derives a new container by removing a given value from an existing one.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<dfn>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<div>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

- **xml:base (optional).**

Elements:

- Members of [flow.ElementGroup](#)
- PCDATA

<divide>

Description: An expression for dividing one number by another.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<dl>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [dlElement.ElementGroup](#)

dlElement.ElementGroup

Description: Elements that can appear within <dl>.

Group members: [<dd>](#), [<dt>](#)

Context:

- Occurs [*] within [dl](#)

<drawingInteraction>

Description: Interaction in which the candidate draws freely on a given backdrop image.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type

Elements:

- [<prompt>](#)
- [<object>](#)

<dt>

Description: Defined by XHTML.

Context:

See:

- [dlElement.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

duration.Type

A trivial restriction of xsd:double.

<durationGTE>

Description: An expression for testing the greater-than-or-equal relation between two durations.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<durationLT>

Description: An expression for testing the less-than relation between two durations.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

Description: Defined by XHTML

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<endAttemptInteraction>

Description: A special interaction for providing alternative ways to end an attempt.

Context:

See:

- [flow.ElementGroup](#)
- [inline.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **title (required).**
Data type = string.Type

<equal>

Description: An expression for testing two numeric values for equality.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **toleranceMode (required).**
Data type = toleranceMode.Type
- **tolerance (optional).** When two values are given, the lower bound (t0) is given first
Data type = list of floatOrTemplateRef.Type
- **includeLowerBound (optional).**
Data type = boolean.Type
- **includeUpperBound (optional).**
Data type = boolean.Type

Elements:

- Members of [expression.ElementGroup](#)

<equalRounded>

Description: An expression for testing two numeric values for equality after rounding.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **roundingMode (required).**
Data type = roundingMode.Type
- **figures (required).**
Data type = integerOrTemplateRef.Type

Elements:

- Members of [expression.ElementGroup](#)

<exitResponse>

Description: A response rule that terminates response processing immediately.

Context:

See:

- [responseRule.ElementGroup](#)

<exitTemplate>

Description: A template rule that terminates template processing immediately.

Context:

See:

- [templateRule.ElementGroup](#)

<exitTest>

Context:

See:

- [outcomeRule.ElementGroup](#)

expression.ElementGroup

Description: The group of elements that are used to create expressions for setting variable values and controlling conditional groups of rules.

Group members: [<and>](#), [<anyN>](#), [<baseValue>](#), [<containerSize>](#), [<contains>](#), [<correct>](#), [<customOperator>](#), [<default>](#), [<delete>](#), [<divide>](#), [<durationGTE>](#), [<durationLT>](#), [<equal>](#), [<equalRounded>](#), [<fieldValue>](#), [<gt>](#), [<gte>](#), [<index>](#), [<inside>](#), [<integerDivide>](#), [<integerModulus>](#), [<integerToFloat>](#), [<isNull>](#), [<lt>](#), [<lte>](#), [<mapResponse>](#), [<mapResponsePoint>](#), [<match>](#), [<member>](#), [<multiple>](#), [<not>](#), [<null>](#), [<numberCorrect>](#), [<numberIncorrect>](#), [<numberPresented>](#), [<numberResponded>](#), [<numberSelected>](#), [<or>](#), [<ordered>](#), [<outcomeMaximum>](#), [<outcomeMinimum>](#), [<patternMatch>](#), [<power>](#), [<product>](#), [<random>](#), [<randomFloat>](#), [<randomInteger>](#), [<round>](#), [<stringMatch>](#), [<substring>](#), [<subtract>](#), [<sum>](#), [<testVariables>](#), [<truncate>](#), [<variable>](#)

Context:

- Occurs [1..*] within [and](#)
- Occurs [1..*] within [anyN](#)
- Occurs [1] within [branchRule](#)
- Occurs [1] within [containerSize](#)
- Occurs [2] within [contains](#)
- Occurs [*] within [customOperator](#)
- Occurs [2] within [delete](#)
- Occurs [2] within [divide](#)
- Occurs [2] within [durationGTE](#)
- Occurs [2] within [durationLT](#)
- Occurs [2] within [equal](#)
- Occurs [2] within [equalRounded](#)
- Occurs [1] within [fieldValue](#)
- Occurs [2] within [gt](#)
- Occurs [2] within [gte](#)
- Occurs [1] within [index](#)
- Occurs [1] within [inside](#)
- Occurs [2] within [integerDivide](#)
- Occurs [2] within [integerModulus](#)
- Occurs [1] within [integerToFloat](#)
- Occurs [1] within [isNull](#)
- Occurs [1] within [lookupOutcomeValue](#)
- Occurs [2] within [lt](#)
- Occurs [2] within [lte](#)
- Occurs [2] within [match](#)
- Occurs [2] within [member](#)
- Occurs [*] within [multiple](#)
- Occurs [1] within [not](#)
- Occurs [1..*] within [or](#)
- Occurs [*] within [ordered](#)
- Occurs [1] within [outcomeElseIf](#)
- Occurs [1] within [outcomeIf](#)
- Occurs [1] within [patternMatch](#)
- Occurs [2] within [power](#)
- Occurs [1] within [preCondition](#)
- Occurs [1..*] within [product](#)

- Occurs [1] within [random](#)
- Occurs [1] within [responseElseIf](#)
- Occurs [1] within [responseIf](#)
- Occurs [1] within [round](#)
- Occurs [1] within [setCorrectResponse](#)
- Occurs [1] within [setDefaultValue](#)
- Occurs [1] within [setOutcomeValue](#)
- Occurs [1] within [setTemplateValue](#)
- Occurs [2] within [stringMatch](#)
- Occurs [2] within [substring](#)
- Occurs [2] within [subtract](#)
- Occurs [1..*] within [sum](#)
- Occurs [1] within [templateDefault](#)
- Occurs [1] within [templateElseIf](#)
- Occurs [1] within [templateIf](#)
- Occurs [1] within [truncate](#)

<extendedTextInteraction>

Description: Interaction that allows the candidate to supply an extended free-text response.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **base (optional).**
Data type = integer.Type
- **stringIdentifier (optional).**
Data type = identifier.Type
- **expectedLength (optional).**
Data type = integer.Type

- **patternMask (optional).**
Data type = string.Type
- **placeholderText (optional).**
Data type = string.Type
- **maxStrings (optional).**
Data type = integer.Type
- **minStrings (optional).**
Data type = integer.Type
- **expectedLines (optional).**
Data type = integer.Type
- **format (optional).**
Data type = textFormat.Type

Elements:

- [<prompt>](#)

<feedbackBlock>

Description: A structural part of the item body whose visibility can be conditionally controlled by the current outcomes.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **outcomeIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

- **xml:base (optional).**

Elements:

- Members of [block.ElementGroup](#)

feedbackElement.ElementGroup

Description: Elements whose visibility can be conditionally controlled depending on the current outcomes.

Group members: [<feedbackBlock>](#), [<feedbackInline>](#)

Context:

Not applicable.

<feedbackInline>

Description: A span of text (or equivalent) whose visibility can be conditionally controlled by the current outcomes.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **outcomeIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<feedbackIntegrated>

Description: Profile flag for the integrated feedback feature.

Context:

- Occurs [1] within [contentProfile](#)

<feedbackModal>

Description: Profile flag for the modal feedback feature.

Context:

- Occurs [1] within [contentProfile](#)

<feedbackType>**Context:**

- Occurs [0..1] within [qtiMetadata](#)

Content: a value of type feedbackType.Type.

feedbackType.Type

One of the following symbolic values:

- adaptive
- nonadaptive
- none

<fieldValue>

Description: An expression that extracts a single field value from a record container.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **fieldIdentifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

float.Type

A trivial restriction of xsd:double.

floatOrTemplateRef.Type

A trivial restriction of xsd:string.

flow.ElementGroup

Description: A group of elements that can appear in contexts where either further structure or spans of text are allowed.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [<address>](#), [<associateInteraction>](#), [](#), [<big>](#), [<blockquote>](#), [
](#), [<choiceInteraction>](#), [<cite>](#), [<code>](#), [<customInteraction>](#), [<dfn>](#), [<div>](#), [<dl>](#), [<drawingInteraction>](#), [](#), [<endAttemptInteraction>](#), [<extendedTextInteraction>](#), [<feedbackBlock>](#), [<feedbackInline>](#), [<gapMatchInteraction>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hotspotInteraction>](#), [<hottext>](#), [<hottextInteraction>](#), [<hr>](#), [<i>](#), [](#), [<inlineChoiceInteraction>](#), [<kbd>](#), [<m:math>](#), [<matchInteraction>](#), [<mediaInteraction>](#), [<object>](#), [](#), [<orderInteraction>](#), [<p>](#), [<pre>](#), [<printedVariable>](#), [<q>](#), [<rubricBlock>](#), [<samp>](#), [<selectPointInteraction>](#), [<sliderInteraction>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<table>](#), [<templateBlock>](#), [<templateInline>](#), [<textEntryInteraction>](#), [<tt>](#), [](#), [<uploadInteraction>](#), [<var>](#), [<xi:include>](#)

Context:

- Occurs [*] within [dd](#)
- Occurs [*] within [div](#)
- Occurs [*] within [li](#)
- Occurs [*] within members of [tableCell.ElementGroup](#)

See:

- [objectFlow.ElementGroup](#)

flowStatic.ElementGroup

Description: A group of elements that can appear in contexts where either further structure or spans of text are allowed excluding interactions.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [<address>](#), [](#), [<big>](#), [<blockquote>](#), [
](#), [<cite>](#), [<code>](#), [<dfn>](#), [<div>](#), [<dl>](#), [](#), [<feedbackBlock>](#), [<feedbackInline>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hottext>](#), [<hr>](#), [<i>](#), [](#), [<kbd>](#), [<m:math>](#), [<object>](#), [](#), [<p>](#), [<pre>](#), [<printedVariable>](#), [<q>](#), [<rubricBlock>](#), [<samp>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<table>](#), [<templateBlock>](#), [<templateInline>](#), [<tt>](#), [](#), [<var>](#)

Context:

- Occurs [*] within [modalFeedback](#)
- Occurs [*] within [simpleAssociableChoice](#)
- Occurs [*] within [simpleChoice](#)
- Occurs [*] within [testFeedback](#)

See:

- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

<gap>

Description: Defines the position of a gap in the text for gap-filling tasks.

Context:

See:

- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type

- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **matchGroup (optional).**
Data type = list of identifier.Type
- **required (optional).**
Data type = boolean.Type

gapChoice.ElementGroup

Description: A group of elements that can be used to define choices for gap-filling tasks.

Group members: [<gapImg>](#), [<gapText>](#)

Context:

- Occurs [1..*] within [gapMatchInteraction](#)

<gapImg>

Description: An image that can be used to fill a gap.

Context:

- Occurs [1..*] within [graphicGapMatchInteraction](#)

See:

- [gapChoice.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type

- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **matchGroup (optional).**
Data type = list of identifier.Type
- **matchMax (required).**
Data type = integer.Type
- **matchMin (optional).**
Data type = integer.Type
- **objectLabel (optional).**
Data type = string.Type

Elements:

- [<object>](#)

<gapMatchInteraction>

Description: An interaction in which the candidate fills gaps in a passage of text from a set of predefined choices.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **shuffle (required).**
Data type = boolean.Type

Elements:

- [<prompt>](#)
- Members of [gapChoice.ElementGroup](#)
- Members of [blockStatic.ElementGroup](#)

<gapText>

Description: A simple run of text that can be used to fill a gap.

Context:

See:

- [gapChoice.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **matchGroup (optional).**
Data type = list of identifier.Type
- **matchMax (required).**
Data type = integer.Type
- **matchMin (optional).**
Data type = integer.Type

Elements:

- [<textOrVariable>](#)
- PCDATA

<graphicAssociateInteraction>

Description: Interaction in which the candidate associates pairs of pre-defined hotspots on an image.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **maxAssociations (required).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<object>](#)
- [<associableHotspot>](#)

<graphicGapMatchInteraction>

Description: Interaction in which the candidate fills gaps within a given image, e.g., through drag-and-drop.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type

- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type

Elements:

- [<prompt>](#)
- [<object>](#)
- [<gapImg>](#)
- [<associableHotspot>](#)

graphicInteraction.ElementGroup

Description: A group of elements for describing image-based tasks.

Group members: [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<hotspotInteraction>](#), [<selectPointInteraction>](#)

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

<graphicOrderInteraction>

Description: Interaction in which the candidate orders a set of predefined hotspots on an image.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type

- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **minChoices (optional).**
Data type = integer.Type
- **maxChoices (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<object>](#)
- [<hotspotChoice>](#)

<gt>

Description: An expression for testing the greater-than relation between two numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<gte>

Description: An expression for testing the greater-than-or-equal relation between two numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<h1>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<h2>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<h3>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<h4>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<h5>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**

- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<h6>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

hotspot.ElementGroup

Description: A group of elements that define hot areas of a graphic.

Group members: [<associableHotspot>](#), [<hotspotChoice>](#)

Context:

Not applicable.

<hotspotChoice>

Description: An element that defines a hot area of a graphic for simple area selection tasks.

Context:

- Occurs [1..*] within [graphicOrderInteraction](#)
- Occurs [1..*] within [hotspotInteraction](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **shape (required).**
Data type = shape.Type
- **coords (required).**
Data type = coords.Type
- **hotspotLabel (optional).**
Data type = string256.Type

<hotspotInteraction>

Description: Interaction in which the candidate selects pre-defined hotspots on an image.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **maxChoices (required).**
Data type = integer.Type
- **minChoices (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<object>](#)
- [<hotspotChoice>](#)

<hottext>

Description: Defines an area of hottext for hottext selection tasks.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **xml:base (optional).**

Elements:

- Members of [inlineStatic.ElementGroup](#)
- PCDATA

<hottextInteraction>

Description: Interaction in which the candidate selects pre-defined hot areas within the text.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **maxChoices (required).**
Data type = integer.Type
- **minChoices (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)

- Members of [blockStatic.ElementGroup](#)

<hr>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

<hypertextElement>

Description: Profile flag for the XHTML hypertext element.

Context:

- Occurs [1] within [contentProfile](#)

<i>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)

- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<lip:identification>

Externally defined.

identifier.Type

A trivial restriction of xsd:NCName.

<imageElement>

Description: Profile flag for the XHTML image element.

Context:

- Occurs [1] within [contentProfile](#)

<imageType>

Description: Profile element for identifying a specific image mime-type.

Context:

- Occurs [*] within [contentProfile](#)

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **src (required).**
Data type = uri.Type
- **alt (required).**
Data type = string256.Type
- **longdesc (optional).**
Data type = uri.Type
- **height (optional).**
Data type = length.Type
- **width (optional).**
Data type = length.Type

<imsmd>

Description: Profile flag for IMS meta-data.

Context:

- Occurs [1] within [metadataProfile](#)

<imsqtimd>

Description: Profile flag for IMS QTI meta-data.

Context:

- Occurs [1] within [metadataProfile](#)

<xi:include>

Externally defined.

<index>

Description: An expression that extracts a single value from an ordered container.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **n (required).**
Data type = integer.Type

Elements:

- Members of [expression.ElementGroup](#)

inline.ElementGroup

Description: The group of elements that behave like spans of text.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [](#), [<big>](#), [
](#), [<cite>](#), [<code>](#), [<dfn>](#), [](#), [<endAttemptInteraction>](#), [<feedbackInline>](#), [<gap>](#), [<hottext>](#), [<i>](#), [](#), [<inlineChoiceInteraction>](#), [<kbd>](#), [<m:math>](#), [<object>](#), [<printedVariable>](#), [<q>](#), [<samp>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<templateInline>](#), [<textEntryInteraction>](#), [<tt>](#), [<var>](#)

Context:

- Occurs [*] within members of [atomicBlock.ElementGroup](#)
- Occurs [*] within [caption](#)
- Occurs [*] within [dt](#)
- Occurs [*] within members of [simpleInline.ElementGroup](#)

<inlineChoice>

Description: A string of text that can be used to fill a gap.

Context:

- Occurs [1..*] within [inlineChoiceInteraction](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type

<inlineChoiceInteraction>

Description: An interaction that allows a single gap in the text to be filled from a set of predefined strings.

Context:

See:

- [flow.ElementGroup](#)
- [inline.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type

- **shuffle (required).**
Data type = boolean.Type
- **required (optional).**
Data type = boolean.Type

Elements:

- [<inlineChoice>](#)

inlineInteraction.ElementGroup

Description: The group of interactions that behave like spans of text.

Group members: [<endAttemptInteraction>](#), [<inlineChoiceInteraction>](#), [<textEntryInteraction>](#)

Context:

See:

- [flow.ElementGroup](#)
- [inline.ElementGroup](#)
- [objectFlow.ElementGroup](#)

inlineStatic.ElementGroup

Description: A group of elements that behave like spans of text excluding interactions.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [](#), [<big>](#), [
](#), [<cite>](#), [<code>](#), [<dfn>](#), [](#), [<feedbackInline>](#), [<gap>](#), [<hottext>](#), [<i>](#), [](#), [<kbd>](#), [<m:math>](#), [<object>](#), [<printedVariable>](#), [<q>](#), [<samp>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<templateInline>](#), [<tt>](#), [<var>](#)

Context:

- Occurs [*] within [hottext](#)
- Occurs [*] within [prompt](#)
- Occurs [*] within [templateInline](#)

See:

- [inline.ElementGroup](#)

<inside>

Description: An expression for testing if a point is inside a given area.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **shape (required).**
Data type = shape.Type
- **coords (required).**
Data type = coords.Type

Elements:

- Members of [expression.ElementGroup](#)

integer.Type

A trivial restriction of xsd:int.

<integerDivide>

Description: An expression for performing integer-division.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<integerModulus>

Description: An expression for calculating the remainder following integer-division.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

integerOrTemplateRef.Type

A trivial restriction of xsd:string.

<integerToFloat>

Description: An expression for converting an integer to the corresponding value of float base-type.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

interaction.ElementGroup

Description: The group of elements that allow the candidate to interact with the item, setting or updating the values of the response variables.

Group members: [<associateInteraction>](#), [<choiceInteraction>](#), [<customInteraction>](#), [<drawingInteraction>](#), [<endAttemptInteraction>](#), [<extendedTextInteraction>](#), [<gapMatchInteraction>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<hotspotInteraction>](#), [<hottextInteraction>](#), [<inlineChoiceInteraction>](#), [<matchInteraction>](#), [<mediaInteraction>](#), [<orderInteraction>](#), [<positionObjectInteraction>](#), [<selectPointInteraction>](#), [<sliderInteraction>](#), [<textEntryInteraction>](#), [<uploadInteraction>](#)

Context:

Not applicable.

<interactionType>

Context:

- Occurs [*] within [contentProfile](#)
- Occurs [*] within [qtiMetadata](#)

Content: a value of type interactionType.Type.

interactionType.Type

One of the following symbolic values:

- associateInteraction
- choiceInteraction
- customInteraction
- drawingInteraction
- endAttemptInteraction
- extendedTextInteraction
- gapMatchInteraction
- graphicAssociateInteraction
- graphicGapMatchInteraction
- graphicOrderInteraction
- hotspotInteraction
- hottextInteraction
- inlineChoiceInteraction
- matchInteraction
- orderInteraction
- positionObjectInteraction
- selectPointInteraction
- sliderInteraction
- textEntryInteraction
- uploadInteraction

<interpolationTable>

Context:

See:

- [lookupTable.ElementGroup](#)

Attributes:

- **defaultValue (optional).**
Data type = valueType.Type

Elements:

- [<interpolationTableEntry>](#)

<interpolationTableEntry>

Context:

- Occurs [1..*] within [interpolationTable](#)

Attributes:

- **sourceValue (required).**
Data type = float.Type
- **includeBoundary (optional).**
Data type = boolean.Type
- **targetValue (required).**
Data type = valueType.Type

<isNull>

Description: An expression that tests for the NULL value.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<itemBody>

Description: Describes the body of an item, the text, graphics and interactions that are presented to the candidate.

Context:

- Occurs [0..1] within [assessmentItem](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [block.ElementGroup](#)

<itemResult>

Context:

- Occurs [*] within [assessmentResult](#)

Attributes:

- **identifier (required).**
Data type = string.Type
- **sequenceIndex (optional).**
Data type = integer.Type
- **datestamp (required).**
Data type = datetime.Type
- **sessionStatus (required).**
Data type = sessionStatus.Type

Elements:

- Members of [itemVariable.ElementGroup](#)
- PCDATA

<itemSessionControl>

Context:

- Occurs [0..1] within members of [sectionPart.ElementGroup](#)
- Occurs [0..1] within [testPart](#)

Attributes:

- **maxAttempts (optional).**
Data type = integer.Type
- **showFeedback (optional).**
Data type = boolean.Type
- **allowReview (optional).**
Data type = boolean.Type
- **showSolution (optional).**
Data type = boolean.Type
- **allowComment (optional).**
Data type = boolean.Type
- **allowSkipping (optional).**
Data type = boolean.Type
- **validateResponses (optional).**
Data type = boolean.Type

itemStatistic.ElementGroup

Description: A group of elements used to define item statistics.

Group members: [<categorizedStatistic>](#), [<ordinaryStatistic>](#)

Context:

- Occurs [*] within [usageData](#)

itemSubset.ElementGroup

Group members: [<numberCorrect>](#), [<numberIncorrect>](#), [<numberPresented>](#), [<numberResponded>](#), [<numberSelected>](#), [<outcomeMaximum>](#), [<outcomeMinimum>](#), [<testVariables>](#)

Context:

Not applicable.

<itemTemplate>

Description: A meta-data flag indicating that the item is a template.

Context:

- Occurs [0..1] within [qtiMetadata](#)

itemVariable.ElementGroup

Group members: [<outcomeVariable>](#), [<responseVariable>](#), [<templateVariable>](#)

Context:

- Occurs [*] within [itemResult](#)
- Occurs [*] within [testResult](#)

<kbd>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

language.Type

A trivial restriction of xsd:language.

length.Type

A trivial restriction of xsd:string.

Description: Defined by XHTML

Context:

- Occurs [*] within [ol](#)
- Occurs [*] within [ul](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**

- **label (optional).**
Data type = string256.Type

Elements:

- Members of [flow.ElementGroup](#)
- PCDATA

<listElements>

Description: Profile flag for the XHTML list elements.

Context:

- Occurs [1] within [contentProfile](#)

<lomMetadata>

Description: Profile flag for LOM meta-data.

Context:

- Occurs [1] within [metadataProfile](#)

<lookupOutcomeValue>

Description: A response/outcome rule that looks up the value of an outcome variable in an associated lookuptable

Context:

See:

- [outcomeRule.ElementGroup](#)
- [responseRule.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

lookupTable.ElementGroup

Description: Defines a lookup table for setting an outcome value based on a numeric source value.

Group members: [<interpolationTable>](#), [<matchTable>](#)

Context:

- Occurs [0..1] within [outcomeDeclaration](#)

<lt>

Description: An expression for testing the less-than relation between two numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<lte>

Description: An expression for testing the less-than-or-equal relation between two numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<mapEntry>

Description: Defines a mapping from a single value, the key, onto a real number.

Context:

- Occurs [1..*] within [mapping](#)

Attributes:

- **mapKey (required).**
Data type = valueType.Type
- **mappedValue (required).**
Data type = float.Type

<mapResponse>

Description: An expression for mapping the value(s) of a response variable onto a real number using the declared mapping.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

<mapResponsePoint>

Description: An expression for mapping the value(s) of a point-type response variable onto a real number using the declared area-mapping.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

<mapping>

Description: Defines a mapping from a source set of values onto real numbers.

Context:

- Occurs [1] within [categorizedStatistic](#)
- Occurs [0..1] within [responseDeclaration](#)

Attributes:

- **lowerBound (optional).**
Data type = float.Type
- **upperBound (optional).**
Data type = float.Type
- **defaultValue (required).**
Data type = float.Type

Elements:

- [<mapEntry>](#)

<match>

Description: An expression for testing if two values are identical.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<matchInteraction>

Description: Interaction in which the candidate must identify pairs of choices between two distinct sets.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type

- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **shuffle (required).**
Data type = boolean.Type
- **maxAssociations (required).**
Data type = integer.Type
- **minAssociations (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<simpleMatchSet>](#)

<matchTable>**Context:**

See:

- [lookupTable.ElementGroup](#)

Attributes:

- **defaultValue (optional).**
Data type = valueType.Type

Elements:

- [<matchTableEntry>](#)

<matchTableEntry>**Context:**

- Occurs [1..*] within [matchTable](#)

Attributes:

- **sourceValue (required).**
Data type = integer.Type
- **targetValue (required).**
Data type = valueType.Type

<m:math>

Externally defined.

<mathElement>

Description: Profile flag for the MathML math element.

Context:

- Occurs [1] within [contentProfile](#)

<mathVariable>

Description: Profile flag for the mathVariable feature.

Context:

- Occurs [1] within [contentProfile](#)

<mediaInteraction>

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **autostart (required).**
Data type = boolean.Type
- **minPlays (optional).**
Data type = integer.Type

- **maxPlays (optional).**
Data type = integer.Type
- **loop (optional).**
Data type = boolean.Type

Elements:

- [<prompt>](#)
- [<object>](#)

<member>

Description: An expression that tests a container to determine if it contains a given value.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<metadataProfile>

Description: An element for describing the capabilities or requirements of a system with respect to meta-data.

Context:

- Occurs [1] within [bankProfile](#)
- Occurs [1] within [contentProfile](#)

Elements:

- [<imsmd>](#)
- [<lomMetadata>](#)
- [<imsqtimd>](#)

mimeType.Type

A trivial restriction of xsd:string.

<modalFeedback>

Description: Feedback conditionally shown to the candidate directly following response processing depending on the value of the given outcome variable.

Context:

- Occurs [*] within [assessmentItem](#)

Attributes:

- **outcomeIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type
- **title (optional).**
Data type = string.Type

Elements:

- Members of [flowStatic.ElementGroup](#)
- PCDATA

<multiple>

Description: An expression that aggregates sub-expressions into a single unordered container.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

navigationMode.Type

One of the following symbolic values:

- linear
- nonlinear

<not>

Description: An expression that performs boolean negation.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<null>

Description: A simple expression that returns the NULL value.

Context:

See:

- [expression.ElementGroup](#)

<numberCorrect>

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

<numberIncorrect>

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

<numberPresented>**Context:**

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

<numberResponded>**Context:**

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

<numberSelected>**Context:**

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

<object>

Description: Defined by XHTML.

Context:

- Occurs [1] within [drawingInteraction](#)
- Occurs [1] within [gapImg](#)
- Occurs [1] within members of [graphicInteraction.ElementGroup](#)
- Occurs [1] within [mediaInteraction](#)
- Occurs [1] within [positionObjectInteraction](#)
- Occurs [1] within [positionObjectStage](#)

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **data (required).**
Data type = string.Type

- **type (required).**
Data type = mimeType.Type
- **width (optional).**
Data type = length.Type
- **height (optional).**
Data type = length.Type

Elements:

- Members of [objectFlow.ElementGroup](#)
- PCDATA

<objectElements>

Description: Profile flag for the XHTML object element.

Context:

- Occurs [1] within [contentProfile](#)

objectFlow.ElementGroup

Description: The group of elements that can appear within <object>.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [<address>](#), [<associateInteraction>](#), [](#), [<big>](#), [<blockquote>](#), [
](#), [<choiceInteraction>](#), [<cite>](#), [<code>](#), [<customInteraction>](#), [<dfn>](#), [<div>](#), [<dl>](#), [<drawingInteraction>](#), [](#), [<endAttemptInteraction>](#), [<extendedTextInteraction>](#), [<feedbackBlock>](#), [<feedbackInline>](#), [<gapMatchInteraction>](#), [<graphicAssociateInteraction>](#), [<graphicGapMatchInteraction>](#), [<graphicOrderInteraction>](#), [<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#), [<hotspotInteraction>](#), [<hottext>](#), [<hottextInteraction>](#), [<hr>](#), [<i>](#), [](#), [<inlineChoiceInteraction>](#), [<kbd>](#), [<m:math>](#), [<matchInteraction>](#), [<mediaInteraction>](#), [<object>](#), [](#), [<orderInteraction>](#), [<p>](#), [<param>](#), [<pre>](#), [<printedVariable>](#), [<q>](#), [<rubricBlock>](#), [<samp>](#), [<selectPointInteraction>](#), [<sliderInteraction>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<table>](#), [<templateBlock>](#), [<templateInline>](#), [<textEntryInteraction>](#), [<tt>](#), [](#), [<uploadInteraction>](#), [<var>](#), [<xi:include>](#)

Context:

- Occurs [*] within [object](#)

<objectType>

Description: Profile element for identifying a specific object mime-type.

Context:

- Occurs [*] within [contentProfile](#)

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- [](#)

<or>

Description: An expression that performs a boolean or.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<orderInteraction>

Description: Interaction in which candidates must (re-)order a set of choices.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **shuffle (required).**
Data type = boolean.Type
- **minChoices (optional).**
Data type = integer.Type
- **maxChoices (optional).**
Data type = integer.Type
- **orientation (optional).**
Data type = orientation.Type

Elements:

- [<prompt>](#)
- [<simpleChoice>](#)

<ordered>

Description: An expression that aggregates sub-expressions into a single ordered container.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<ordering>

Context:

- Occurs [0..1] within [assessmentSection](#)

Attributes:

- **shuffle (required).**
Data type = boolean.Type
- Attributes from other namespaces

Elements:

- Elements from any namespace

<ordinaryStatistic>

Description: An ordinary single-valued item statistic.

Context:

See:

- [itemStatistic.ElementGroup](#)

Attributes:

- **name (required).**
Data type = identifier.Type
- **glossary (optional).**
Data type = uri.Type
- **context (required).**
Data type = uri.Type
- **caseCount (optional).**
Data type = integer.Type
- **stdError (optional).**
Data type = float.Type
- **stdDeviation (optional).**
Data type = float.Type
- **lastUpdated (optional).**
Data type = date.Type

Elements:

- [<targetObject>](#)
- [<value>](#)

orientation.Type

One of the following symbolic values:

- horizontal
- vertical

<outcomeCondition>

Description: An outcome rule that contains groups of sub-rules which are followed conditionally based on the value of one or more expressions.

Context:

See:

- [outcomeRule.ElementGroup](#)

Elements:

- [<outcomeIf>](#)
- [<outcomeElseIf>](#)
- [<outcomeElse>](#)

<outcomeDeclaration>

Description: Declares an outcome variable.

Context:

- Occurs [*] within [assessmentItem](#)
- Occurs [*] within [assessmentTest](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type
- **view (optional).**
Data type = list of view.Type

- **interpretation (optional).**
Data type = string.Type
- **longInterpretation (optional).**
Data type = uri.Type
- **normalMaximum (optional).**
Data type = float.Type
- **normalMinimum (optional).**
Data type = float.Type
- **masteryValue (optional).**
Data type = float.Type

Elements:

- [<defaultValue>](#)
- Members of [lookupTable.ElementGroup](#)

<outcomeElse>

Description: A final group of rules to be followed unconditionally if all previous groups have been skipped for the containing outcome condition.

Context:

- Occurs [0..1] within [outcomeCondition](#)

Elements:

- Members of [outcomeRule.ElementGroup](#)

<outcomeElseIf>

Description: A conditional group of rules to be considered if all previous groups have been skipped for the containing outcome condition.

Context:

- Occurs [*] within [outcomeCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [outcomeRule.ElementGroup](#)

<outcomeIf>

Description: The first part of an outcome condition.

Context:

- Occurs [1] within [outcomeCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [outcomeRule.ElementGroup](#)

<outcomeMaximum>

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type
- **outcomeIdentifier (required).**
Data type = identifier.Type
- **weightIdentifier (optional).**
Data type = identifier.Type

<outcomeMinimum>

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type

- **outcomeIdentifier (required).**
Data type = identifier.Type
- **weightIdentifier (optional).**
Data type = identifier.Type

<outcomeProcessing>

Description: Defines the set of rules for calculating test-level outcomes.

Context:

- Occurs [0..1] within [assessmentTest](#)

Elements:

- Members of [outcomeRule.ElementGroup](#)

<outcomeProcessingFragment>

Context:

See:

- [outcomeRule.ElementGroup](#)

Can be the root element of an XML document.

Elements:

- Members of [outcomeRule.ElementGroup](#)

outcomeRule.ElementGroup

Description: A group of elements that define rules for outcome processing.

Group members: [<exitTest>](#), [<lookupOutcomeValue>](#), [<outcomeCondition>](#), [<outcomeProcessingFragment>](#), [<setOutcomeValue>](#), [<xi:include>](#)

Context:

- Occurs [*] within [outcomeElse](#)
- Occurs [*] within [outcomeElseIf](#)
- Occurs [*] within [outcomeIf](#)
- Occurs [*] within [outcomeProcessing](#)
- Occurs [*] within [outcomeProcessingFragment](#)

<outcomeVariable>

Context:

See:

- [itemVariable.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type
- **view (optional).**
Data type = list of view.Type
- **interpretation (optional).**
Data type = string.Type
- **longInterpretation (optional).**
Data type = uri.Type
- **normalMaximum (optional).**
Data type = float.Type
- **normalMinimum (optional).**
Data type = float.Type
- **masteryValue (optional).**
Data type = float.Type

Elements:

- [<value>](#)

<p>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<param>

Description: Defined by XHTML.

Context:

See:

- [objectFlow.ElementGroup](#)

Attributes:

- **name (required).**
Data type = string.Type
- **value (required).**
Data type = string.Type
- **valuetype (required).**
Data type = paramType.Type
- **type (optional).**
Data type = mimeType.Type

paramType.Type

One of the following symbolic values:

- DATA
- REF

<patternMatch>

Description: An expression for testing two string values using regular expressions.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **pattern (required).**
Data type = stringOrTemplateRef.Type

Elements:

- Members of [expression.ElementGroup](#)

<positionObjectInteraction>

Description: Interaction in which the candidate must position objects over an existing image.

Context:

- Occurs [1..*] within [positionObjectStage](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **centerPoint (optional).**
Data type = list of integer.Type
- **maxChoices (required).**
Data type = integer.Type
- **minChoices (optional).**
Data type = integer.Type

Elements:

- [<object>](#)

<positionObjectStage>

Description: Defines the background image to be used for positioning object tasks.

Context:

See:

- [block.ElementGroup](#)

Elements:

- [<object>](#)
- [<positionObjectInteraction>](#)

<power>

Description: An expression for raising one number to the power of another.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<pre>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<preCondition>

Context:

- Occurs [*] within members of [sectionPart.ElementGroup](#)
- Occurs [*] within [testPart](#)

Elements:

- Members of [expression.ElementGroup](#)

<presentationElements>

Description: Profile flag for the XHTML presentation elements.

Context:

- Occurs [1] within [contentProfile](#)

<printedVariable>

Description: An element used to insert the value of an item variable into the item body.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

- [textOrVariable.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **identifier (required).**
Data type = identifier.Type
- **format (optional).**
Data type = string256.Type
- **base (optional).**
Data type = integer.Type

<printedVariables>

Description: Profile flag for the printedVariable feature.

Context:

- Occurs [1] within [contentProfile](#)

<product>

Description: An expression for calculating the product of a group of numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<prompt>

Description: An optional prompt for an interaction.

Context:

- Occurs [0..1] within members of [blockInteraction.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [inlineStatic.ElementGroup](#)
- PCDATA

<q>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **cite (optional).**
Data type = uri.Type

Elements:

- Members of [inline.ElementGroup](#)

- PCDATA

<qtiMetadata>

Description: The root element for QTI-specific meta-data. Typically used within a manifest file in an IMS content package.

Context:

Can be the root element of an XML document.

Elements:

- [<itemTemplate>](#)
- [<timeDependent>](#)
- [<composite>](#)
- [<interactionType>](#)
- [<feedbackType>](#)
- [<solutionAvailable>](#)
- [<toolName>](#)
- [<toolVersion>](#)
- [<toolVendor>](#)

<random>

Description: An expression that extracts a random value from a container.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<randomFloat>

Description: An expression that returns a random float from the specified range.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **min (required).**
Data type = floatOrTemplateRef.Type
- **max (required).**
Data type = floatOrTemplateRef.Type

<randomInteger>

Description: An expression that returns a random integer from the specified range.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **min (required).**
Data type = integerOrTemplateRef.Type
- **max (required).**
Data type = integerOrTemplateRef.Type
- **step (optional).**
Data type = integerOrTemplateRef.Type

<regexp>

Description: Profile flag for the regular-expression feature.

Context:

- Occurs [1] within [contentProfile](#)

<responseCondition>

Description: A response rule that contains groups of sub-rules which are followed conditionally based on the value of one or more expressions.

Context:

See:

- [responseRule.ElementGroup](#)

Elements:

- [<responseIf>](#)
- [<responseElseIf>](#)
- [<responseElse>](#)

<responseDeclaration>

Description: Declares a response variable.

Context:

- Occurs [*] within [assessmentItem](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type

Elements:

- [<defaultValue>](#)
- [<correctResponse>](#)
- [<mapping>](#)
- [<areaMapping>](#)

<responseElse>

Description: A final group of rules to be followed unconditionally if all previous groups have been skipped for the containing response condition.

Context:

- Occurs [0..1] within [responseCondition](#)

Elements:

- Members of [responseRule.ElementGroup](#)

<responseElseIf>

Description: A conditional group of rules to be considered if all previous groups have been skipped for the containing response condition.

Context:

- Occurs [*] within [responseCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [responseRule.ElementGroup](#)

<responseIf>

Description: The first part of a response condition.

Context:

- Occurs [1] within [responseCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [responseRule.ElementGroup](#)

<responseProcessing>

Description: Defines the rules used to process the candidate's responses.

Context:

- Occurs [0..1] within [assessmentItem](#)

Can be the root element of an XML document.

Attributes:

- **template (optional).**
Data type = uri.Type
- **templateLocation (optional).**
Data type = uri.Type

Elements:

- Members of [responseRule.ElementGroup](#)

<responseProcessingFragment>**Context:**

See:

- [responseRule.ElementGroup](#)

Can be the root element of an XML document.

Elements:

- Members of [responseRule.ElementGroup](#)

responseRule.ElementGroup

Description: A group of elements that define rules to be followed during response processing.

Group members: [<exitResponse>](#), [<lookupOutcomeValue>](#), [<responseCondition>](#), [<responseProcessingFragment>](#), [<setOutcomeValue>](#), [<xi:include>](#)

Context:

- Occurs [*] within [responseElse](#)
- Occurs [*] within [responseElseIf](#)
- Occurs [*] within [responseSelf](#)
- Occurs [*] within [responseProcessing](#)
- Occurs [*] within [responseProcessingFragment](#)

<responseRules>

Description: Profile flag for the general response processing feature.

Context:

- Occurs [1] within [contentProfile](#)

<responseVariable>

Context:

See:

- [itemVariable.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type
- **choiceSequence (optional).**
Data type = list of identifier.Type

Elements:

- [<correctResponse>](#)
- [<candidateResponse>](#)

<round>

Description: An expression for rounding a real number to an integer.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<rounding>**Context:**

- Occurs [1] within [contentProfile](#)

roundingMode.Type

One of the following symbolic values:

- decimalPlaces
- significantFigures

<rpTemplate>

Description: Profile element for identifying a specific response processing template.

Context:

- Occurs [*] within [contentProfile](#)

<rubric>

Description: Profile flag for the rubric feature.

Context:

- Occurs [1] within [contentProfile](#)

<rubricBlock>

Description: A structural part of the item body whose visibility depends on the role of the person viewing the item, typically containing instructions to a specific actor.

Context:

- Occurs [*] within [assessmentSection](#)

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **view (required).**
Data type = list of view.Type

Elements:

- Members of [block.ElementGroup](#)

<samp>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

sectionPart.ElementGroup

Description: The group of elements used to define the structure of sections.

Group members: [<assessmentItemRef>](#), [<assessmentSection>](#), [<xi:include>](#)

Context:

- Occurs [*] within [assessmentSection](#)

<selectPointInteraction>

Description: Interaction in which the candidate selects one or more points on an image.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **maxChoices (required).**
Data type = integer.Type
- **minChoices (optional).**
Data type = integer.Type

Elements:

- [<prompt>](#)
- [<object>](#)

<selection>**Context:**

- Occurs [0..1] within [assessmentSection](#)

Attributes:

- **select (required).**
Data type = integer.Type
- **withReplacement (optional).**
Data type = boolean.Type
- Attributes from other namespaces

Elements:

- Elements from any namespace

<sessionIdentifier>**Context:**

- Occurs [*] within [context](#)

Attributes:

- **sourceID (required).**
Data type = uri.Type
- **identifier (required).**
Data type = string.Type

sessionStatus.Type

One of the following symbolic values:

- final
- initial
- pendingResponseProcessing
- pendingSubmission

<setCorrectResponse>

Description: A template rule that sets the correct value of a response variable.

Context:

See:

- [templateRule.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

<setDefaultValue>

Description: A template rule that sets the default value of an item variable.

Context:

See:

- [templateRule.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

<setOutcomeValue>

Description: A response/outcome rule that sets the value of an outcome variable.

Context:

See:

- [outcomeRule.ElementGroup](#)
- [responseRule.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

<setTemplateValue>

Description: A template rule that sets the value of a template variable.

Context:

See:

- [templateRule.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

shape.Type

One of the following symbolic values:

- circle
- default
- ellipse
- poly
- rect

showHide.Type

One of the following symbolic values:

- hide
- show

<simpleAssociableChoice>

Description: The element used to describe the choices in simple association or matching tasks.

Context:

- Occurs [1..*] within [associateInteraction](#)
- Occurs [*] within [simpleMatchSet](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type
- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type
- **matchGroup (optional).**
Data type = list of identifier.Type

- **matchMax (required).**
Data type = integer.Type
- **matchMin (optional).**
Data type = integer.Type

Elements:

- Members of [flowStatic.ElementGroup](#)
- PCDATA

simpleBlock.ElementGroup

Description: A group of elements that provide simple aggregating structure to the item body.

Group members: [<blockquote>](#), [<feedbackBlock>](#), [<rubricBlock>](#)

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

<simpleChoice>

Description: The element used to describe the choices in simple selection or ordering tasks.

Context:

- Occurs [1..*] within [choiceInteraction](#)
- Occurs [1..*] within [orderInteraction](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **identifier (required).**
Data type = identifier.Type

- **fixed (optional).**
Data type = boolean.Type
- **templateIdentifier (optional).**
Data type = identifier.Type
- **showHide (optional).**
Data type = showHide.Type

Elements:

- Members of [flowStatic.ElementGroup](#)
- PCDATA

simpleInline.ElementGroup

Description: A group of elements that both behave as and contain things that behave like spans of text.

Group members: [<a>](#), [<abbr>](#), [<acronym>](#), [](#), [<big>](#), [<cite>](#), [<code>](#), [<dfn>](#), [](#), [<feedbackInline>](#), [<i>](#), [<kbd>](#), [<q>](#), [<samp>](#), [<small>](#), [](#), [](#), [<sub>](#), [<sup>](#), [<tt>](#), [<var>](#)

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

<simpleMatchSet>

Description: Describes a set of choices for use in a matching task.

Context:

- Occurs [2] within [matchInteraction](#)

Elements:

- [<simpleAssociableChoice>](#)

<sliderInteraction>

Description: Interaction in which the candidate uses a slider to enter a constrained numeric response.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **lowerBound (required).**
Data type = float.Type
- **upperBound (required).**
Data type = float.Type
- **step (optional).**
Data type = integer.Type
- **stepLabel (optional).**
Data type = boolean.Type
- **orientation (optional).**
Data type = orientation.Type
- **reverse (optional).**
Data type = boolean.Type

Elements:

- [<prompt>](#)

<small>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGrou](#)
- PCDATA

<solutionAvailable>

Description: A meta-data flag indicating that the item contains a solution.

Context:

- Occurs [0..1] within [qtiMetadata](#)

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

string.Type

A trivial restriction of xsd:string.

string256.Type

A trivial restriction of xsd:string.

stringInteraction.ElementGroup

Description: A group of interactions that allow text-based responses to be entered by the user.

Group members: [<extendedTextInteraction>](#), [<textEntryInteraction>](#)

Context:

Not applicable.

<stringMatch>

Description: An expression for testing two string values.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **caseSensitive (required).**
Data type = boolean.Type

- **substring (optional).**
Data type = boolean.Type

Elements:

- Members of [expression.ElementGroup](#)

stringOrTemplateRef.Type

A trivial restriction of xsd:string.

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

styleclass.Type

A trivial restriction of xsd:string.

<stylesheet>

Description: Associates a stylesheet with an item.

Context:

- Occurs [0..*] within [assessmentItem](#)

Attributes:

- **href (required).**
Data type = uri.Type
- **type (required).**
Data type = mimeType.Type
- **media (optional).**
Data type = string.Type
- **title (optional).**
Data type = string.Type

<sub>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

submissionMode.Type

One of the following symbolic values:

- individual
- simultaneous

<substring>

Description: An expression that tests a string to determine if it contains another.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **caseSensitive (required).**
Data type = boolean.Type

Elements:

- Members of [expression.ElementGroup](#)

<subtract>

Description: An expression for subtracting one number from another.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<sum>

Description: An expression for calculating the sum of a group of numbers.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<sup>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<table>

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)

- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**
- **summary (optional).**
Data type = string.Type

Elements:

- [<caption>](#)
- [<col>](#)
- [<colgroup>](#)
- [<thead>](#)
- [<tfoot>](#)
- [<tbody>](#)

tableCell.ElementGroup

Description: Elements that can appear within a row of a table.

Group members: [<td>](#), [<th>](#)

Context:

- Occurs [1..*] within [tr](#)

tableCellScope.Type

One of the following symbolic values:

- col
- colgroup
- row
- rowgroup

<tableElements>

Description: Profile flag for the XHTML table element.

Context:

- Occurs [1] within [contentProfile](#)

<targetObject>

Description: A reference to the object, such as an assessment item, or a sub-part of the object, to which an item statistic relates.

Context:

- Occurs [1..*] within members of [itemStatistic.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = string.Type
- **partIdentifier (optional).**
Data type = identifier.Type

<tbody>

Description: Defined by XHTML.

Context:

- Occurs [1..*] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- [<tr>](#)

<td>

Description: Defined by XHTML.

Context:

See:

- [tableCell.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **headers (optional).**
Data type = list of identifier.Type
- **scope (optional).**
Data type = tableCellScope.Type
- **abbr (optional).**
Data type = string.Type
- **axis (optional).**
Data type = string.Type
- **rowspan (optional).**
Data type = integer.Type
- **colspan (optional).**
Data type = integer.Type

Elements:

- Members of [flow.ElementGroup](#)
- PCDATA

<templateBlock>

Description: A structural part of the item body whose visibility is conditionally controlled by a template variable.

Context:

See:

- [block.ElementGroup](#)

- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **templateIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [blockStatic.ElementGroup](#)

<templateCondition>

Description: A template rule that contains groups of sub-rules which are followed conditionally based on the value of one or more expressions.

Context:

See:

- [templateRule.ElementGroup](#)

Elements:

- [<templateIf>](#)
- [<templateElseIf>](#)
- [<templateElse>](#)

<templateDeclaration>

Description: Declares a template variable.

Context:

- Occurs [*] within [assessmentItem](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type
- **paramVariable (optional).**
Data type = boolean.Type
- **mathVariable (optional).**
Data type = boolean.Type

Elements:

- [<defaultValue>](#)

<templateDefault>

Context:

- Occurs [*] within [assessmentItemRef](#)

Attributes:

- **templateIdentifier (required).**
Data type = identifier.Type

Elements:

- Members of [expression.ElementGroup](#)

templateElement.ElementGroup

Description: Elements whose visibility is conditionally controlled by a template variable.

Group members: [<templateBlock>](#), [<templateInline>](#)

Context:

Not applicable.

<templateElse>

Description: A final group of rules to be followed unconditionally if all previous groups have been skipped for the containing template condition.

Context:

- Occurs [0..1] within [templateCondition](#)

Elements:

- Members of [templateRule.ElementGroup](#)

<templateElseIf>

Description: A conditional group of rules to be considered if all previous groups have been skipped for the containing template condition.

Context:

- Occurs [*] within [templateCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [templateRule.ElementGroup](#)

<templateIf>

Description: The first part of a template condition.

Context:

- Occurs [1] within [templateCondition](#)

Elements:

- Members of [expression.ElementGroup](#)
- Members of [templateRule.ElementGroup](#)

<templateInline>

Description: A span of text (or equivalent) whose visibility is conditionally controlled by a template variable.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **templateIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type

Elements:

- Members of [inlineStatic.ElementGroup](#)
- PCDATA

<templateProcessing>

Description: Defines the rules used for calculating the values of the template variables.

Context:

- Occurs [0..1] within [assessmentItem](#)

Elements:

- Members of [templateRule.ElementGroup](#)

templateRule.ElementGroup

Description: A group of elements that define rules to be followed during template processing.

Group members: [<exitTemplate>](#), [<setCorrectResponse>](#), [<setDefaultValue>](#),
[<setTemplateValue>](#), [<templateCondition>](#)

Context:

- Occurs [*] within [templateElse](#)
- Occurs [*] within [templateElseIf](#)
- Occurs [*] within [templateIf](#)
- Occurs [1..*] within [templateProcessing](#)

<templateVariable>

Context:

See:

- [itemVariable.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **cardinality (required).**
Data type = cardinality.Type
- **baseType (optional).**
Data type = baseType.Type

Elements:

- [<value>](#)

<templates>

Description: Profile flag for the item template feature.

Context:

- Occurs [1] within [contentProfile](#)

<testFeedback>

Description: Test or test-part feedback.

Context:

- Occurs [*] within [assessmentTest](#)

- Occurs [*] within [testPart](#)

Attributes:

- **access (required).**
Data type = testFeedbackAccess.Type
- **outcomeIdentifier (required).**
Data type = identifier.Type
- **showHide (required).**
Data type = showHide.Type
- **identifier (required).**
Data type = identifier.Type
- **title (optional).**
Data type = string.Type

Elements:

- Members of [flowStatic.ElementGroup](#)
- PCDATA

testFeedbackAccess.Type

One of the following symbolic values:

- atEnd
- during

<testPart>

Description: Each test is divided into one or more parts

Context:

- Occurs [1..*] within [assessmentTest](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type
- **navigationMode (required).**
Data type = navigationMode.Type
- **submissionMode (required).**
Data type = submissionMode.Type

Elements:

- [<preCondition>](#)

- [<branchRule>](#)
- [<itemSessionControl>](#)
- [<timeLimits>](#)
- [<assessmentSection>](#)
- [<testFeedback>](#)

<testResult>

Context:

- Occurs [0..1] within [assessmentResult](#)

Attributes:

- **identifier (required).**
Data type = string.Type
- **datestamp (required).**
Data type = datetime.Type

Elements:

- Members of [itemVariable.ElementGroup](#)

<testVariables>

Description: A simple expression returning a container of values looked up in the test namespace.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **sectionIdentifier (optional).**
Data type = identifier.Type
- **includeCategory (optional).**
Data type = list of identifier.Type
- **excludeCategory (optional).**
Data type = list of identifier.Type
- **variableIdentifier (required).**
Data type = identifier.Type
- **baseType (optional).**
Data type = baseType.Type

- **weightIdentifier (optional).**
Data type = identifier.Type

<textElements>

Description: Profile flag for the XHTML text elements.

Context:

- Occurs [1] within [contentProfile](#)

<textEntryInteraction>

Description: Interaction that allows the candidate to supply a span of text to fill a gap.

Context:

See:

- [flow.ElementGroup](#)
- [inline.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **base (optional).**
Data type = integer.Type
- **stringIdentifier (optional).**
Data type = identifier.Type
- **expectedLength (optional).**
Data type = integer.Type
- **patternMask (optional).**
Data type = string.Type
- **placeholderText (optional).**
Data type = string.Type

textFormat.Type

One of the following symbolic values:

- plain
- preFormatted
- xhtml

<tfoot>

Context:

- Occurs [0..1] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- [<tr>](#)

<th>

Description: Defined by XHTML.

Context:

See:

- [tableCell.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**

- **label (optional).**
Data type = string256.Type
- **headers (optional).**
Data type = list of identifier.Type
- **scope (optional).**
Data type = tableCellScope.Type
- **abbr (optional).**
Data type = string.Type
- **axis (optional).**
Data type = string.Type
- **rowspan (optional).**
Data type = integer.Type
- **colspan (optional).**
Data type = integer.Type

Elements:

- Members of [flow.ElementGroup](#)
- PCDATA

<thead>**Context:**

- Occurs [0..1] within [table](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- [<tr>](#)

<timeDependent>

Description: Profile flag for the time-dependence feature.

Context:

- Occurs [1] within [contentProfile](#)

- Occurs [0..1] within [qtiMetadata](#)

<timeLimits>

Context:

- Occurs [0..1] within [assessmentTest](#)
- Occurs [0..1] within members of [sectionPart.ElementGroup](#)
- Occurs [0..1] within [testPart](#)

Attributes:

- **minTime (optional).**
Data type = duration.Type
- **maxTime (optional).**
Data type = duration.Type

toleranceMode.Type

One of the following symbolic values:

- absolute
- exact
- relative

<toolName>

Description: A meta-data element containing the name of the tool used to create the object.

Context:

- Occurs [0..1] within [qtiMetadata](#)

<toolVendor>

Description: A meta-data element containing the name of the company that produced the tool used to create the object.

Context:

- Occurs [0..1] within [qtiMetadata](#)

<toolVersion>

Description: A meta-data element containing the version of the tool used to create the object.

Context:

- Occurs [0..1] within [qtiMetadata](#)

<tr>

Description: Defined by XHTML.

Context:

- Occurs [1..*] within [tbody](#)
- Occurs [1..*] within [tfoot](#)
- Occurs [1..*] within [thead](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type

Elements:

- Members of [tableCell.ElementGroup](#)

<truncate>

Description: An expression for truncating a real number to an integer.

Context:

See:

- [expression.ElementGroup](#)

Elements:

- Members of [expression.ElementGroup](#)

<tt>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

Description: Defined by XHTML.

Context:

See:

- [block.ElementGroup](#)
- [blockStatic.ElementGroup](#)
- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**

- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- [](#)

<uploadInteraction>

Description: Interaction in which the candidate uploads a file representing their response.

Context:

See:

- [block.ElementGroup](#)
- [flow.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **xml:base (optional).**
- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **responseIdentifier (required).**
Data type = identifier.Type
- **type (optional).**
Data type = mimeType.Type

Elements:

- [<prompt>](#)

uri.Type

A trivial restriction of xsd:anyURI.

<usageData>

Description: The root element of a usage data XML document instance. Usage data are also referred to as item statistics.

Context:

Can be the root element of an XML document.

Attributes:

- **glossary (optional).** The default glossary in which the names of the itemStatistics are defined.
Data type = uri.Type

Elements:

- Members of [itemStatistic.ElementGroup](#)

<usageDataVocabulary>

Description: Profile flag for usage-data files.

Context:

- Occurs [*] within [bankProfile](#)

<value>

Description: A single data value from one of the base-types.

Context:

- Occurs [*] within [candidateResponse](#)
- Occurs [1..*] within [correctResponse](#)
- Occurs [1..*] within [defaultValue](#)
- Occurs [1] within [ordinaryStatistic](#)
- Occurs [*] within [outcomeVariable](#)
- Occurs [*] within [templateVariable](#)

Attributes:

- **fieldIdentifier (optional).**
Data type = identifier.Type
- **baseType (optional).**
Data type = baseType.Type

valueType.Type

A trivial restriction of xsd:string.

<var>

Description: Defined by XHTML.

Context:

See:

- [flow.ElementGroup](#)
- [flowStatic.ElementGroup](#)
- [inline.ElementGroup](#)
- [inlineStatic.ElementGroup](#)
- [objectFlow.ElementGroup](#)

Attributes:

- **id (optional).**
Data type = identifier.Type
- **class (optional).**
Data type = list of styleclass.Type
- **xml:lang (optional).**
- **label (optional).**
Data type = string256.Type
- **xml:base (optional).**

Elements:

- Members of [inline.ElementGroup](#)
- PCDATA

<variable>

Description: A simple expression returning the value of an item variable.

Context:

See:

- [expression.ElementGroup](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

- **weightIdentifier (optional).**
Data type = identifier.Type

variableDeclaration.ElementGroup

Group members: [<outcomeDeclaration>](#), [<responseDeclaration>](#), [<templateDeclaration>](#)

Context:

Not applicable.

<variableMapping>

Context:

- Occurs [*] within [assessmentItemRef](#)

Attributes:

- **sourceIdentifier (required).**
Data type = identifier.Type
- **targetIdentifier (required).**
Data type = identifier.Type

view.Type

One of the following symbolic values:

- author
- candidate
- proctor
- scorer
- testConstructor
- tutor

<weight>

Context:

- Occurs [*] within [assessmentItemRef](#)

Attributes:

- **identifier (required).**
Data type = identifier.Type

- **value (required).**
Data type = float.Type

About This Document

Title	IMS Question and Test Interoperability XML Binding
Editors	Steve Lay (Cambridge Assessment), Pierre Gorissen (SURF)
Version	Public Draft v2.1
Version Date	8 June 2006
Status	Public Draft Specification
Summary	This document describes the QTI XML Binding specification.
Revision Information	8 June 2006
Purpose	This document has been approved by the IMS Technical Board and is made available for public review and comment.
Document Location	

List of Contributors

The following individuals contributed to the development of this document:

Name	Organization
Dick Bacon	University of Surrey
Niall Barr	Question Mark
Lance Blackstone	Pearson VUE
Jeanne Ferrante	ETS
Pierre Gorissen	SURF
Regina Hoag	ETS
Gopal Krishnan	Pearson VUE
Steve Lay	Cambridge Assessment
Rowin Young	CETIS

Revision History

Version No.	Release Date	Comments
Base Document 2.1	14 October 2005	The first version of the QTI v2.1 specification.
Public Draft 2.1	9 January 2006	The Public Draft v2.1 of the QTI specification.
Public Draft 2.1 (revision 2)	8 June 2006	The Public Draft v2.1 (revision 2) of the QTI specification.

IMS Global Learning Consortium, Inc. ("IMS/GLC") is publishing the information contained in this IMS Question and Test Interoperability XML Binding ("Specification") for purposes of scientific, experimental, and scholarly collaboration only.

IMS/GLC makes no warranty or representation regarding the accuracy or completeness of the Specification.

This material is provided on an "As Is" and "As Available" basis.

The Specification is at all times subject to change and revision without notice.

It is your sole responsibility to evaluate the usefulness, accuracy, and completeness of the Specification as it relates to you.

IMS/GLC would appreciate receiving your comments and suggestions.

Please contact IMS/GLC through our website at <http://www.imsglobal.org>

*Please refer to Document Name: IMS Question and Test Interoperability XML Binding
Revision: 8 June 2006*