



IMS Question and Test Interoperability: QTILite Specification

Version 1.2 Final Specification

IPR and Distribution Notices

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

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

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

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

This document may be copied and furnished to others by Licensee organizations 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 work 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/question/qtiv1p2/qtiv1p2speclicense.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.

**Copyright © 2002 by 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: QTILite Specification

Date: 11 February 2002

Table of Contents

TABLE OF CONTENTS.....	2
1. INTRODUCTION.....	4
1.1 QUESTION & TEST INTEROPERABILITY OVERVIEW	4
1.2 SCOPE & CONTEXT.....	4
1.3 STRUCTURE OF THIS DOCUMENT	5
1.4 NOMENCLATURE.....	5
1.5 REFERENCES	5
2. OVERALL DATA MODEL.....	7
2.1 INFORMATION MODEL.....	7
2.2 QTILITE XML SCHEMA TREE	8
3. XML BINDING	10
3.1 ITEM BINDING	10
3.1.1 <questestinterop> Elements	10
3.1.2 <item> Elements.....	10
3.1.3 <objectives> Elements.....	11
3.1.4 <rubric> Elements.....	11
3.1.5 <presentation> Elements	12
3.1.6 <response_lid> Elements	12
3.1.7 <render_choice> Elements.....	13
3.1.8 <response_label> Elements	13
3.1.9 <resprocessing> Elements.....	14
3.1.10 <outcomes> Elements.....	14
3.1.11 <rescondition> Elements.....	15
3.1.12 <conditionvar> Elements	16
3.1.13 <itemfeedback> Elements.....	18
3.1.14 <material> Elements	18
3.1.15 <altmaterial> Elements	21
4. EXAMPLE XML INSTANCES.....	22
4.1 BASIC ASI EXAMPLES	22
4.1.1 Standard True/False (Text).....	22
4.1.2 Standard Multiple Choice (Text).....	23
4.1.3 Enhanced Standard Multiple Choice (Text).....	25
4.1.4 Standard Multiple Choice (Image).....	30
5. XML INSTANCE EXAMPLE LISTS.....	33
5.1 THE ASI XML INSTANCE EXAMPLE FILES	33
6. IMPLEMENTATION GUIDANCE	34
6.1 ITEMS	34
6.1.1 Elements and their Attributes	34
6.1.2 Groups of Elements	34
6.2 AGGREGATED SCORING AND RESPONSE PROCESSING.....	34
6.3 NAMING CONVENTIONS	36
6.3.1 Identities and Labels	36
6.4 SCOPING RULES	36
6.4.1 Identities and Labels	36
7. COMPATIBILITY WITH THE FULL IMS QTI.....	38
7.1 QTI ASI COMPATIBILITY	38

7.2	RESULTS REPORTING COMPATIBILITY	38
8.	CONFORMANCE.....	39
8.1	VALID DATA ISSUES	39
8.2	CONFORMANCE SUMMARY	39
8.3	INTEROPERABILITY STATEMENT	40
8.4	A QTILITE CONFORMANCE EXAMPLE	41
APPENDIX A – GLOSSARY OF TERMS		43
ABOUT THIS DOCUMENT		47
	LIST OF CONTRIBUTORS	47
REVISION HISTORY		48
INDEX.....		49

1. Introduction

1.1 Question & Test Interoperability Overview

The Question & Test Interoperability (QTI) specification describes a basic structure for the representation of question (item) and test (assessment) data and their corresponding results reports [QTI, 02i]. Therefore, the specification enables the exchange of this test, assessment and results data between Learning Management Systems, as well as content authors and, content libraries and collections. The QTI specification is defined in XML to promote the widest possible adoption. XML is a powerful, flexible, industry standard markup language used to encode data models for Internet-enabled and distributed applications. The QTI specification is extensible and customizable to permit immediate adoption, even in specialized or proprietary systems. Leading suppliers and consumers of learning products, services and content contributed time and expertise to produce this final specification.

This document describes the components that are required to construct the simplest form of a QTI-compliant system. QTILite supports multiple-choice questions (this includes the true/false questions) only and limits the rendering form to the classical one response from a set of choices. Multiple Items can be exchanged in a single QTI-XML instance but Assessments and Sections are not supported. The QTILite specification is a standalone document in that none of the others are required to understand and construct QTILite-compliant systems. All QTILite compliant Items are compliant with the full IMS QTI V1.1 and V1.2 specifications but they are not backwards compatible with V1.0 or 1.01 of the specification.

1.2 Scope & Context

This document is the ‘IMS Question & Test Interoperability QTILite Specification’. This specification is based upon the ‘IMS QTI: ASI Information Model’ [QTI, 02a] and is the realization of a subset of that model. QTILite is presented as the entry-level specification to the full QTI specification. QTILite does not support all of the features of the full QTI specification however an instance that conforms to QTILite will also conform to the full QTI specification. The key differences between QTILite and the full specification are:

- The only question-types to be supported within QTILite are:
 - Yes/No
 - True/false
 - Likert scale examples could be: strongly agree, agree, neutral, disagree, strongly disagree, strongly agree, agree, disagree, strongly disagree, agree, neutral and disagree
 - Other forms of multiple choice (i.e. one choice from many);
- Simple response processing to provide for a single right answer and using the default mechanisms;
- No support for:
 - Hints and solutions
 - Meta-data
 - Comments
 - Extensions
 - Options that are “fuzzy”
 - Limited media types and limited text types
 - All time-based mechanisms.

The development of the full IMS QTI specification is accompanied by a release of the associated updated QTILite specification. The QTILite will always be a subset of the full QTI specification and the updates will normally focus on a subset of the additions made to the full specification. The IMS QTI Results Reporting specifications [QTI, 02f], [QTI, 02g], [QTI, 02h] are fully compatible with QTILite i.e. QTILite results can be exchanged using the IMS QTI Results Reporting XML binding.

1.3 Structure of this Document

The structure of this document is:

2. Overall Data Model	A brief summary of the Question & Test Interoperability: ASI Information Model;
3. XML Binding	The realization of the QTILite specification in XML in terms of XSD and DTD;
4. Example XML Instances	Examples of the basic data structures that are supported by this specification;
5. XML Instance Example Lists	Some complete examples of the XML instance required to realize QTILite;
6. Implementation Guidance	Tips on how the distributed learning engines can make best usage of the QTILite specification;
7. Compatibility with the Full IMS QTI	Compatibility of the QTILite specification with the full IMS QTI: ASI and IMS QTI: Results Reporting specifications;
8. Conformance	The expectations on systems that claim conformance to the QTILite specification;
Appendix A – Glossary of Terms	A glossary of the key terms and elements used within the specification.

1.4 Nomenclature

API	Application Programming Interface
ASI	Assessment, Section, Item
CBT	Computer Based Training
DTD	Document Type Definition
QTI	Question & Test Interoperability
VLE	Virtual Learning Environment
W3C	World Wide Web Consortium
XML	Extensible Mark-up Language
XSD	XML Schema Data

1.5 References

[IMS, 01]	<i>IMS Persistent, Location-Independent Resource Identifier Implementation Handbook</i> , M.McKell, Version 1.0, IMS , April 2001.
[QTI, 02a]	<i>IMS Question & Test Interoperability: ASI Information Model Specification</i> , C.Smythe, E.Shepherd, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02b]	<i>IMS Question & Test Interoperability: ASI XML Binding Specification</i> , C.Smythe, E.Shepherd, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02c]	<i>IMS Question & Test Interoperability: ASI Best Practice & Implementation Guide</i> , C.Smythe, E.Shepherd, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02d]	<i>IMS Question & Test Interoperability: ASI Selection & Ordering</i> , C.Smythe, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02e]	<i>IMS Question & Test Interoperability: ASI Outcomes Processing Specification</i> , C.Smythe, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.

[QTI, 02f]	<i>IMS Question & Test Interoperability: Results Reporting Information Model</i> , C.Smythe, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02g]	<i>IMS Question & Test Interoperability: Results Reporting XML Binding</i> , C.Smythe, L.Brewer and S.Lay, Public Draft Specification, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02h]	<i>IMS Question & Test Interoperability: Results Reporting Best Practice & Implementation Guide</i> , C.Smythe, L.Brewer and S.Lay, Final Specification, Version 1.2, IMS , February 2002.
[QTI, 02i]	<i>IMS Question & Test Interoperability: An Overview</i> , C.Smythe, E.Shepherd, L.Brewer and S.Lay, Public Draft Specification, Final Specification, Version 1.2, IMS , February 2002.

2. Overall Data Model

2.1 Information Model

The system model for QTI is shown in Figure 2.1 (this is the same as that described in the QTI: ASI Information Model, [QTI, 02a]).

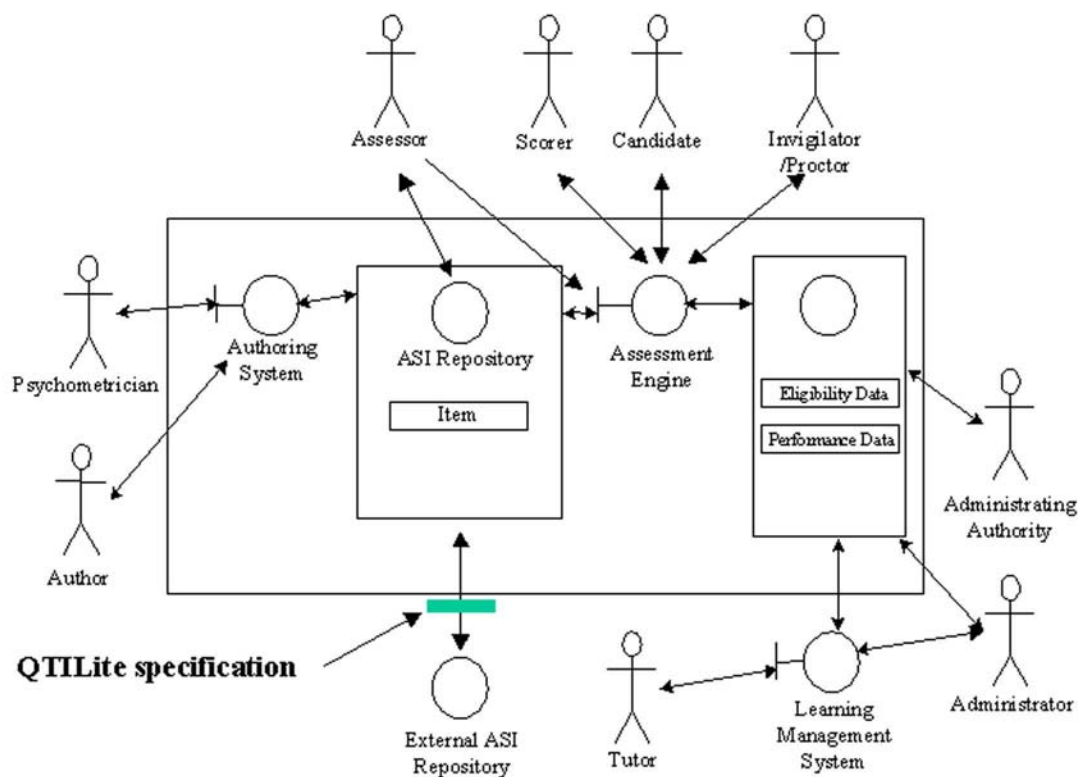


Figure 2.1 The IMS QTILite object data model.

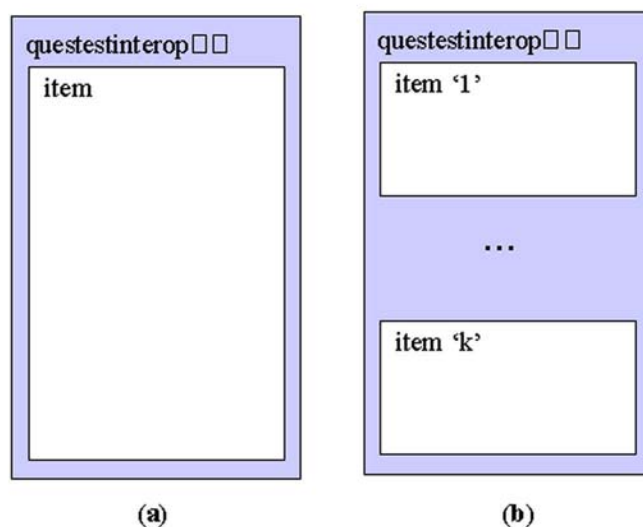


Figure 2.2 The principal QTILite interchange data objects.

The QTILite specification is concerned with the exchange of Items between Assessment systems. The internal representation may conform to the QTILite but the adoption in this way is beyond the scope of the specification. Nine distinct 'views' have been identified for each of the core participants i.e. Administering, Administrator, Assessor, Author, Candidate, Invigilator/Proctor, Psychometrician, Scorer and Tutor. Different types of information may be made available to each of these actors.

The core data structures that can be exchanged using the QTILite Specification are shown in Figure 2.2. QTILite supports the exchange of Items only cf. the full specification that also supports the exchange of Assessments and Sections.

2.2 QTILite XML Schema Tree¹

The generic XML schema tree is shown in Figure 2.3. This representation reflects the overall structure of an Item.

1. The XML schema trees shown in this document were generated by the XML Authority V2.2.1 product from Extensibility Inc.

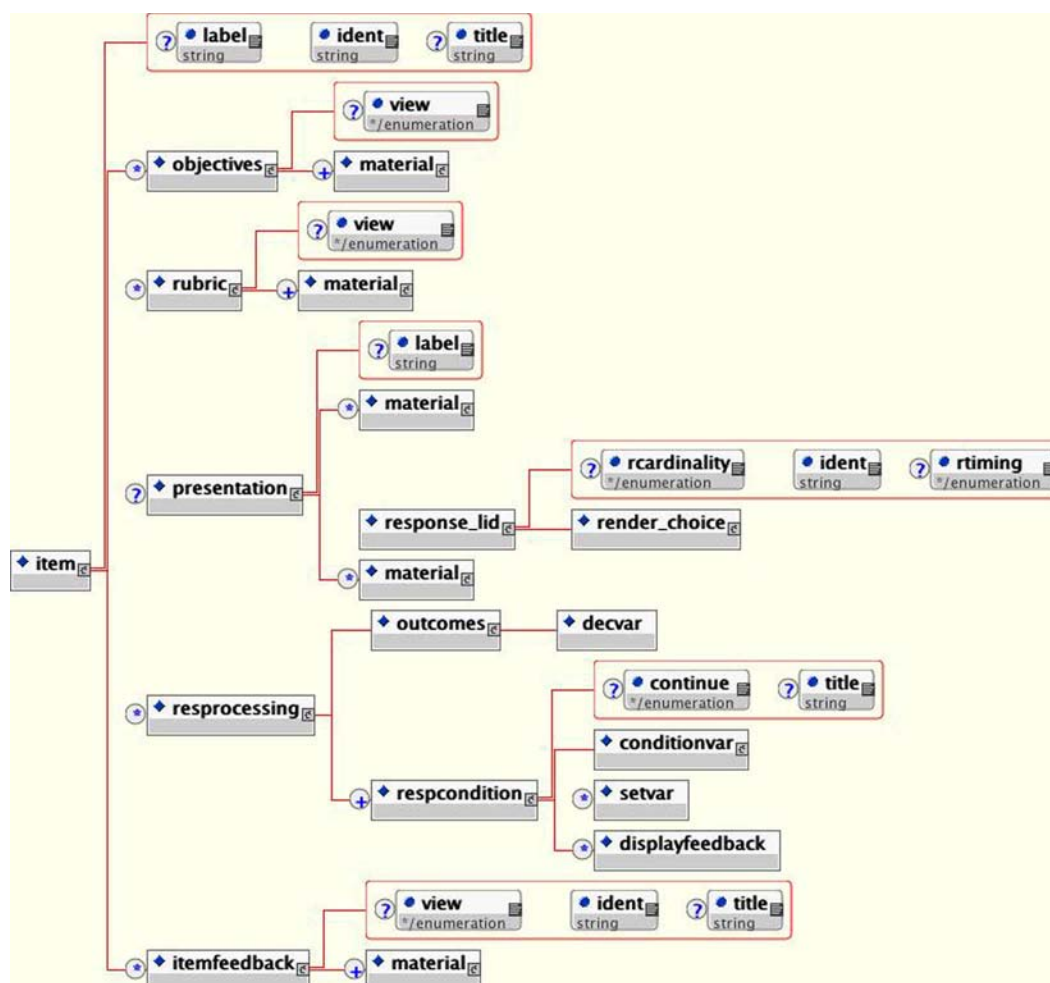


Figure 2.3 The generic structure of the QTILite XML schema tree.

3. XML Binding

3.1 Item Binding

3.1.1 <questestinterop> Elements

Description: The <questestinterop> is the holder for the core QTILite objects. This may contain one or more Items.



Figure 3.1 <questestinterop> elements.

Multiplicity: This is the core element and must occur only once in the XML instance file.

Attributes: None.

Elements:

- item

3.1.2 <item> Elements

Description: The Item is the only data object that can be exchanged using the QTILite specification (cf. Assessments, Sections and Items in the full specification). Each Item consists of five distinct parts, namely: objectives – the materials used to describe the objectives with respect to each view; rubric – the materials used to define the context of the Item and available for each view; presentation – the instructions describing the nature of the question to be asked; resprocessing – the instructions to be followed when analyzing the responses to create a corresponding score and feedback; itemfeedback – the materials to be presented as feedback to the entered response.

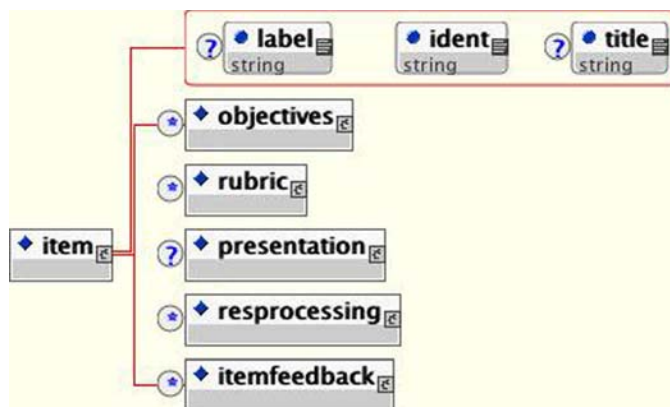


Figure 3.2 <item> elements.

Multiplicity: Occurs one or more times within the <questestinterop> element.

Attributes:

- **title (optional).** The title of the Item.
Data-type = String (max of 256 chars).

- **label (optional).** A label that can be used by authoring tools to identify key features.
Data-type = String (max of 256 chars).
- **ident (required).** The unique identifier for the Item. This identifier should be globally unique (a possible naming convention is included later in this specification).
Data-type = String (max of 256 chars).

Elements:

- objectives
- rubric
- presentation
- resprocessing
- itemfeedback

3.1.3 <objectives> Elements

Description: The objectives element is used to store the information that describes the educational aims of the Item. These objectives can be defined for each of the different ‘view’ perspectives. This element should not be used to contain information specific to an Item because the question-engine may not make this information available to the Item during the actual test.

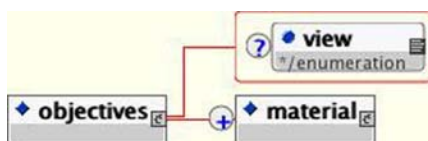


Figure 3.3 <objectives> elements.

Multiplicity: Occurs zero or more times within the <item> element.

Attributes:

- **view (optional with selection from the enumerated list of: All, Administrator, AdminAuthority, Assessor, Author, Candidate, InvigilatorProctor, Psychometrician, Scorer, Tutor. Default=All).** The view defines the scope for the display of the associated information i.e. to whom the material can be presented.
Data-type = Enumerated list.

Elements:

- material

3.1.4 <rubric> Elements

Description: The rubric element is used to contain contextual information that is important to the Item e.g. it could contain standard data values that might or might not be useful for answering the question. Different sets of rubric can be defined for each of the possible ‘views’.

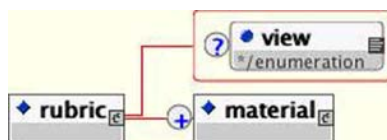


Figure 3.4 <rubric> elements.

Multiplicity: Occurs zero or more times within the <item> element.

Attributes:

- **view** (optional with selection from the enumerated list of: **All, Administrator, AdminAuthority, Assessor, Author, Candidate, InvigilatorProctor, Psychometrician, Scorer, Tutor. Default=All**). The view defines the scope for the display of the associated information i.e. to whom the material can be presented.
Data-type = Enumerated list.

Elements:

- material

3.1.5 <presentation> Elements

Description: This element contains all of the instructions for the presentation of the question during a test. This information includes the actual material to be presented. The labels for the possible responses are also identified and these are used by the response processing element defined elsewhere in the Item.

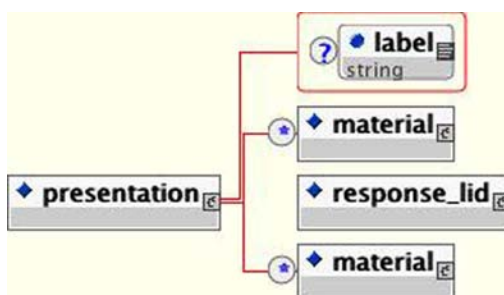


Figure 3.5 <presentation> elements.

Multiplicity: Occurs zero or once within the <item> element.

Attributes:

- **label** (optional). A label that can be used by authoring tools to identify key features.
Data-type = String (max of 256 chars).

Elements:

- material
- response-lid

3.1.6 <response_lid> Elements

Description: The <response_lid> element contains the instructions for the presentation of questions whose response will be the logical label of the selected answer. QTILite supports the <response_lid> form of response only (cf. the full specification). The QTILite specification supports the render_choice option only (cf. the full specification).



Figure 3.6 <response_lid> elements.

Multiplicity: Occurs zero or more times within the <presentation> element.

Attributes:

- **ident (required).** The unique identifier for the response presentation block. This identifier will be used within the response processing structure to ensure the right set of response labels are processed.
Data-type = String (max of 256 chars).
- **rcardinality (optional – enumerated list: Single).** Indicates the number of responses expected from the user. All QTILite questions are defined as requiring a single response i.e. 'rcardinality=Single'.
Data-type = Enumerated list.
- **rtiming (optional – enumerated list: No).** Indicates whether or not the responses are time dependent. All QTILite questions are defined as time independent i.e. 'rtiming=No'.
Data-type = Enumerated list.

Elements:

- material
- render_choice

3.1.7 <render_choice> Elements

Description: The <render_choice> element instructs the question-engine to render the question using a classical multiple-choice format. The number of possible responses is determined by the <response_label> elements contained.



Figure 3.7 <render_choice> elements.

Multiplicity: Occurs zero or once within the <response_lid> element.

Attributes:

- **shuffle (optional – enumerated list of: Yes, No. Default = No).** Shows whether or not the list of possible responses can be shuffled between consecutive displays to the user.
Data-type = Enumerated list.
- **minnumber (fixed = 1).** The minimum number of responses that must be supplied by the participant.
Data-type = Integer (1).
- **maxnumber (fixed = 1).** The maximum number of responses that must be supplied by the participant.
Data-type = Integer (1).

Elements:

- response_label

3.1.8 <response_label> Elements

Description: The <response_label> is used to define the possible response choices that are presented to the user. This information includes the material to be shown to the user and the logical label that is associated with that response. The label is used in the response processing.



Figure 3.8 <response_label> elements.

Multiplicity: This occurs zero or more times within the <render_choice> element.

Attributes:

- **labelrefid (optional).** A label that can be used by authoring tools to identify key features.
Data-type = String (max of 256 chars).
- **ident (required).** The unique identifier for the response_label section. This identifier is used by the response processing mechanism to identify the selected response.
Data-type = String (max of 256 chars).
- **rshuffle (optional – enumerated list of: Yes, No. Default = Yes).** Defines whether the associated response_label can be shuffled between consecutive displays to the user.
Data-type = Enumerated list.

Elements:

- material

3.1.9 <resprocessing> Elements

Description: This is the element within which all of the instructions for the response processing are contained. This includes the scoring variables to contain the associated scores and the set of response condition tests that are to be applied to the received user response. Multiple <resprocessing> elements should be used to provide alternative response processing algorithms which may or may not be used by the response processing engine.

Multiplicity: Occurs zero or more times within the <item> element.

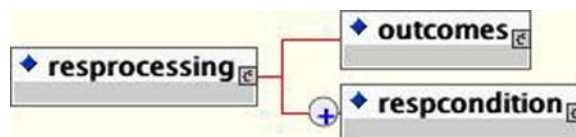


Figure 3.9 <resprocessing> elements.

Attributes: None.

Elements:

- outcomes
- respcondition

3.1.10 <outcomes> Elements

Description: The <outcomes> element contains all of the variable declarations that are to be made available to the scoring algorithm. Each variable is declared using the <decvar> element apart from the default variable called 'SCORE' that is an integer and has a default value of zero (0). In QTILite only one other variable declaration is supported.



Figure 3.10 <outcomes> elements.

Multiplicity: This occurs once within the <resprocessing> element.

Attributes: None.

Elements:

- decvar

3.1.10.1 <decvar> Element

Description: The <decvar> element declares a single scoring variable. In QTILite only integer variables are supported.

Multiplicity: This occurs once within the <outcomes> element.

Attributes:

- **varname (optional. Default = 'SCORE').** The name of the variable that is to be declared. The default name is 'SCORE'.
Data-type = String (max of 256 chars).
- **vartype (Enumerated list: Integer).** The type of the variable declared – for QTILite this is fixed.
Data-type = Enumerated list.
- **defaultval (optional).** The default value to which the variable is to be initialized.
Data-type = String (max of 16 chars).

Elements: None.

3.1.11 <rescondition> Elements

Description: This element contains the actual test to be applied to the user responses to determine their correctness or otherwise. Each <rescondition> contains an actual test, the assignment of a value to the associate scoring variables and the identification of the feedback to be associated with the test.

Multiplicity: This occurs one or more times within the <resprocessing> element.

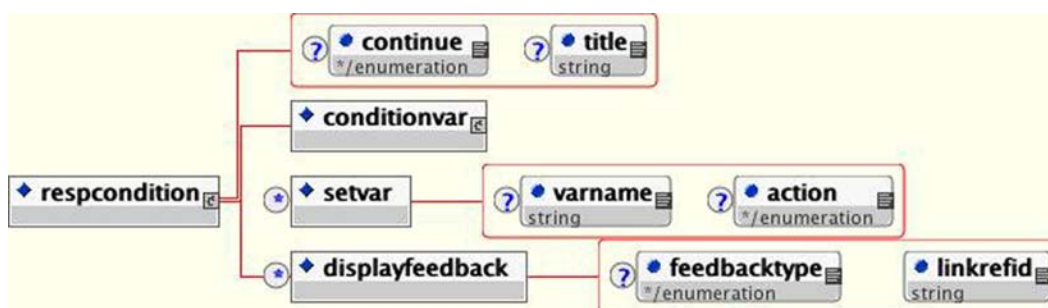


Figure 3.11 <rescondition> elements.

Attributes:

- **title (optional).** The title of the response condition test e.g. 'correct response test', etc.
Data-type = String (max of 256 chars).
- **continue (optional – enumerated list: Yes, No. Default=No).** This is a switch to indicate if further response condition tests are to be applied. The state of this switch is only valid if the application of the previous condition resulted in 'True'.
Data-type = Enumerated list.

Elements:

- conditionvar
- setvar
- displayfeedback

3.1.11.1 <setvar> Element

Description: The <setvar> element is responsible for changing the value of the scoring variable as a result of the associated response processing test. Within QTILite the only supported action is to set the value of the integer variable to some defined number.

Multiplicity: This occurs once within the <responcondition> element.

Elements: None.

Attributes:

- **varname (optional. Default = 'SCORE').** The name of the variable that is to be processed. The default name is 'SCORE'.
Data-type = String (max of 256 chars).
- **action (optional – enumerated list: Set. Default=Set).** The action that is to be applied to the named variable – in QTILite this is fixed.
Data-type = Enumerated list.

3.1.11.2 <displayfeedback> Element

Description: The <displayfeedback> element is responsible for assigning an associated feedback to the response processing if the 'True' state results. Within QTILite the only supported feedback is generic content for the user.

Multiplicity: This occurs zero or more times within the <responcondition> element.

Elements: None.

Attributes:

- **feedbacktype (optional – enumerated list: Response. Default = Response).** The type of feedback that has been triggered by the associated response condition – In QTILite this is fixed.
Data-type = Enumerated list.
- **linkrefid (required).** The identifier of the associated feedback. An <itemfeedback> element must exist with this identifier as defined by the 'ident' attribute.
Data-type = String (max of 256 chars).

3.1.12 <conditionvar> Elements

Description: The conditional test that is to be applied to the user's response. In QTILite the only test supported is the equivalence test on the label. The negative test is also available along with the condition for detecting no attempt at the question.

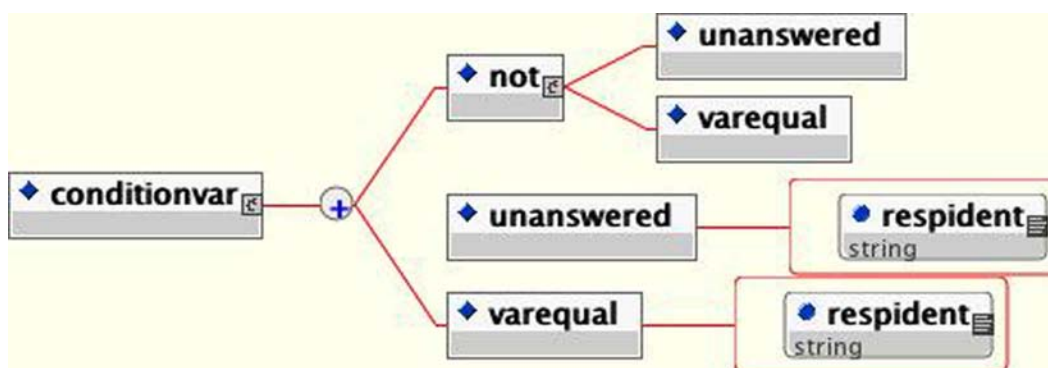


Figure 3.12 <conditionvar> elements.

Multiplicity: This occurs once within the <rescondition> element.

Attributes: None.

Elements:

- not
- unanswered
- varequal

3.1.12.1 <varequal> Element

Description: The <varequal> element is the test of equivalence. The data for the test is contained within the element's PCDATA string and must be the same as one of the <response_label> values (this were assigned using the *ident* attribute).

Multiplicity: This occurs zero or once within the <conditionvar> element, and zero or once within the <not> element.

Elements: None.

Attributes:

- **respidant (required).** The identifier of the <response_lid> element (this was assigned using its *ident* attribute). Data-type = String (max of 256 chars).

3.1.12.2 <not> Element

Description: The <not> element inverts the logical test outcome that is required. In the case of the <varequal> element produces a 'not equals' test. In the case of <unanswered> this becomes the equivalent of answered i.e. the correctness or otherwise is unimportant.

Multiplicity: This occurs zero or more times within the <conditionvar> element.

Elements:

- unanswered
- varequal

Attributes: None.

3.1.12.3 <unanswered> Element

Description: The <unanswered> element is the condition to be applied if a response is not received for the Item i.e. it is unanswered.

Multiplicity: This occurs zero or more times within the <conditionvar> element, and zero or once within the <not> element.

Elements: None.

Attributes:

- **respidnt (required).** The identifier of the <response_lid> element (this was assigned using its *ident* attribute).
Data-type = String (max of 256 chars).

3.1.13 <itemfeedback> Elements

Description: The container for the feedback that is to be presented as a result of the user's responses. In QTILite no hints or solutions are supported.



Figure 3.13 <itemfeedback> elements.

Multiplicity: Occurs zero or more times within the <item> element.

Attributes:

- **title (optional).** The title of the feedback section.
Data-type = String (max of 256 chars).
- **ident (required).** The unique identifier for the feedback. This identifier is used within the <resprocessing> element to identify the feedback to be presented as a consequence of the processing of the user's response.
Data-type = String (max of 256 chars).
- **view (optional with selection from the enumerated list of: All, Administrator, AdminAuthority, Assessor, Author, Candidate, InvigilatorProctor, Psychometrician, Scorer, Tutor. Default=All).** The view defines the scope for the display of the associated information i.e. to whom the material can be presented.
Data-type = Enumerated list.

Elements:

- material

3.1.14 <material> Elements

Description: This is the container for any content that is to be displayed by the question-engine. Within QTILite the only supported content types are text (emphasized or not) and images. The content can be internally referenced to avoid the need for duplicate copies. Alternative information can be defined – this is used if the primary content cannot be displayed.

Multiplicity: This occurs once or more times in: <objectives>, <rubric>, and <itemfeedback>. Occurs zero or more times in <presentation> and <response_label>.

Attributes:

- **label (optional).** A label that can be used by authoring tools to identify key features.
Data-type = String (max of 256 chars).

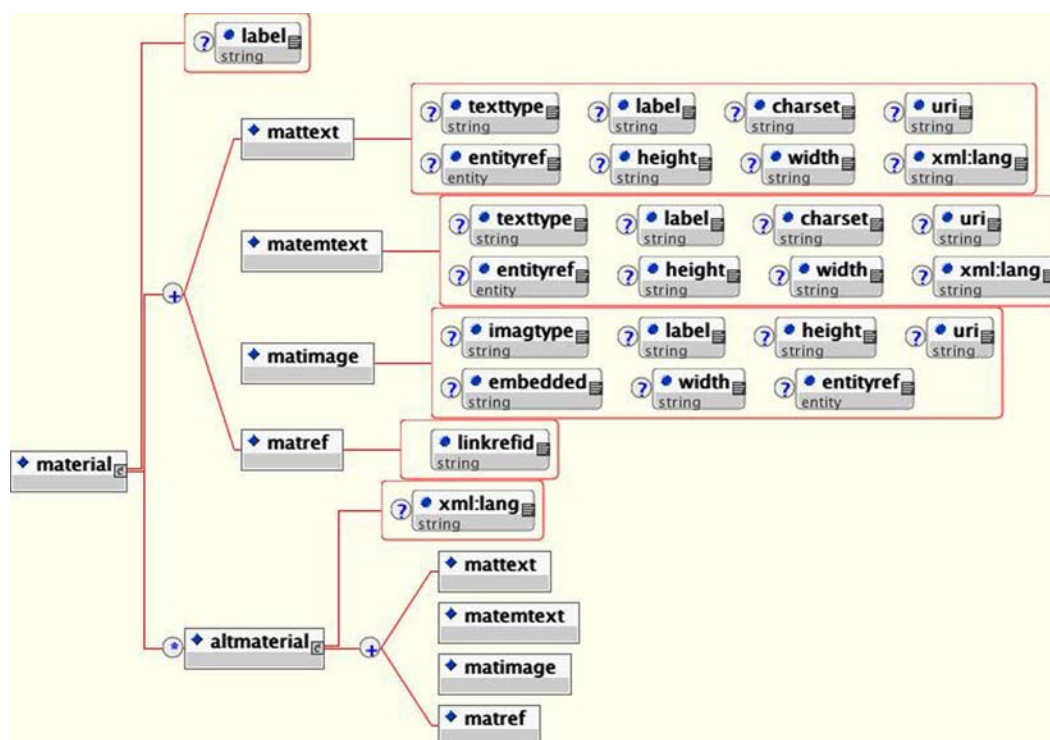


Figure 3.14 <material> elements.

Elements:

- mattext
- matemtext
- matimage
- matref
- altmaterial

3.1.14.1 <mattext> Element

Description: The <mattext> element contains any text that is to be displayed to the users.

Multiplicity: This occurs zero or more times within the <material> element.

Elements: None.

Attributes:

- **texttype (optional. Default='text/plain').** This identifies the type of text string. The format of the string is as defined by RFC1521.
Data-type = String (max of 32 chars).
- **charset (optional. Default='ascii-us').** The character set that is to be used to represent the text string.
Data-type = String (max of 32 chars).
- **label (optional).** A label that is used to uniquely identify the text content.
Data-type = String (max of 256 chars).
- **uri (optional).** The 'uri' identifying the external reference that contains the text to be presented.
Data-type = String (max of 256 chars).

- **entityref (optional).** An alternative mechanism for identifying the external reference that contains the text to be presented. This allows the file reference to be bound to the XML instance itself.
Data-type = String (max of 256 chars).
- **width (optional).** This identifies the width of the text box i.e. its x-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
- **height (optional).** This identifies the height of the text box i.e. its y-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
- **xml:lang (optional).** The language of the text content. As per the ISO639 standard vocabulary.
Data-type = String (max of 32 chars).

3.1.14.2 <matemtext> Element

Description: The <matemtext> element contains any emphasized text that is to be displayed to the users. The type of emphasis is dependent on the question-engine rendering the text.

Multiplicity: This occurs zero or more times within the <material> element.

Elements: None.

Attributes:

- **texttype (optional. Default='text/plain').** This identifies the type of text string. The format of the string is as defined by RFC1521.
Data-type = String (max of 32 chars).
- **charset (optional. Default='ascii-us').** The character set that is to be used to represent the text string.
Data-type = String (max of 32 chars).
- **label (optional).** A label that is used to uniquely identify the text content.
Data-type = String (max of 256 chars).
- **uri (optional).** The 'uri' identifying the external reference that contains the text to be presented.
Data-type = String (max of 256 chars).
- **entityref (optional).** An alternative mechanism for identifying the external reference that contains the text to be presented. This allows the file reference to be bound to the XML instance itself.
- **width (optional).** This identifies the width of the text box i.e. its x-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
- **height (optional).** This identifies the height of the text box i.e. its y-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
Data-type = String (max of 256 chars).
- **xml:lang (optional).** The language of the text content. As per the ISO639 standard vocabulary.
Data-type = String (max of 32 chars).

3.1.14.3 <matimage> Element

Description: The <matimage> element is used to contain image content that is to be displayed to the users.

Multiplicity: This occurs zero or more times within the <material> element.

Elements: None.

Attributes:

- **imagetype (optional. Default='image/jpeg').** This identifies the type of image. The format of the string is as defined by RFC1521.
Data-type = String (max of 32 chars).
- **label (optional).** A label that is used to uniquely identify the image content.
Data-type = String (max of 256 chars).

- **uri (optional).** The 'uri' identifying the external reference that contains the image to be presented.
Data-type = String (max of 256 chars).
- **entityref (optional).** An alternative mechanism for identifying the external reference that contains the image to be presented. This allows the file reference to be bound to the XML instance itself.
Data-type = String (max of 256 chars).
- **width (optional).** This identifies the width of the image i.e. its x-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
- **height (optional).** This identifies the height of the image i.e. its y-axis length.
Data-type = String (max of 32 chars representing an integer in the range $0-2^{31}-1$).
- **embedded (optional. Default='Base64').** This defines the nature of the encoding of the image if it is embedded within the XML instance itself.
Data-type = String (max of 32 chars).

3.1.14.4 <matref> Element

Description: The <matref> element is used to content by reference to the individual material components e.g. <mattext>. This material will have had an identifier assigned to enable such a reference to be reconciled when the instance is parsed into the system.

Multiplicity: This occurs zero or more times within the <material> element.

Elements: None.

Attributes:

- **linkrefid (required).** The identifier of the content that is being referenced. This material will have been assigned by the *label* attribute of the associated element e.g. <matimage>.
Data-type = String (max of 256 chars).

3.1.15 <altmaterial> Elements

Description: This is the container for alternative content. This content is to be displayed if, for whatever reason, the primary content cannot be rendered.

Multiplicity: This occurs zero or once within the <material> element.

Attributes: None.

Elements:

- mattext
- matemtext
- matimage
- matref

4. Example XML Instances

4.1 Basic ASI Examples

The examples of the basic Item types are lists under:

- Standard True/False (text-based options) – two choice-based rendering with response processing;
- Standard Multiple Choice (text-based options) – four choice-based rendering without response processing;
- Standard Multiple Choice (text-based options) – five choice-based rendering without response processing;
- Standard Multiple Choice e (text-based options) – five choice-based rendering with correct answer response processing;
- Standard Multiple Choice (text-based options) – five choice-based rendering with correct and incorrect answer response processing;
- Standard Multiple Choice (text-based options) – five choice-based rendering with response processing and rubric and objectives content;
- Standard Multiple Choice (image-based options) – four choice-based rendering with correct answer response processing.

Note: *These examples also comply with the full IMS QTI Specification V1.1 and V1.2. They DO NOT necessarily comply with the IMS QTI V1.0 or V1.01 specifications.*

4.1.1 Standard True/False (Text)

Figure 4.1 shows a typical True/False multiple-choice question where the possible answers are formatted in to different ways. The corresponding XML is listed after the figure. The user is expected to select either the ‘Agree’ or ‘Disagree’ radio buttons.

<p>Paris is the Capital of France</p> <p><input checked="" type="radio"/> Agree <input type="radio"/> Disagree</p>	<p>Paris is the Capital of France</p> <p><input checked="" type="radio"/> Agree</p> <p><input type="radio"/> Disagree</p>
---	---

Figure 4.1a Standard true/false item.

Figure 4.1b Standard true/false item.

The equivalent XML using the QTILite V1.2 specification is:

1	<questestinterop>
2	<item ident="IMS_V01_I_QTILiteExample001">
3	<presentation label="QTILiteExample001">
4	<material>
5	<mattext>Paris is the Capital of France</mattext>
6	</material>
7	<response_lid ident="TF01" rcardinality="Single" rtiming="No">
8	<render_choice>
9	<response_label ident="T">
10	<material><mattext>Agree</mattext></material>
11	</response_label>
12	<response_label ident="F">
13	<material><mattext>Disagree</mattext></material>
14	</response_label>
15	</render_choice>
16	</response_lid>
17	</presentation>
18	<resprocessing>
19	<outcomes><decvar/></outcomes>
20	<respcondition title="Correct">
21	<conditionvar>
22	<varequal respident="TF01">T</varequal>
23	</conditionvar>
24	<setvar action="Set">1</setvar>
25	<displayfeedback feedbacktype="Response" linkrefid="Correct"/>
26	</respcondition>
27	</resprocessing>
28	<itemfeedback ident="Correct" view="Candidate">
29	<material><mattext>Yes, you are right.</mattext></material>
30	</itemfeedback>
31	</item>
32	</questestinterop>

This XML code is available in the file: 'ims_qtiasiv1p2/qt-lite/trfl_ir_001/trfl_ir_001.xml'. The key points of this example are:

- The actual material to be presented to pose the question is shown in lines 4-6;
- The possible responses and the content to be displayed with each option are shown in lines 8-15. The form of rendering is denoted by the <render_choice> element (line 8);
- The type and number of responses is determined by the <response_lid> element which has the cardinality set as "Single" i.e. only one response is permitted and that there is no time dependence for the answer;
- The processing to be undertaken once a user's response has been obtained is described in lines 18-27. The test for the correct response is shown in lines 21-23 i.e. the label of the correct response is 'T' (the two possible labels were assigned in lines 9 and 12). Each distinct test is enclosed in its own <respcondition> element (line 20) and so a separate test is required to detect an incorrect answer;
- The result of getting the correct answer is to set the default scoring variable to 1 (line 24) and to trigger some feedback (line 25);
- The feedback that is displayed as a result of getting the right answer is shown in lines 28-30.

It should be noted that the actual rendering of the question as shown in Figures 4.1a and 4.1b is dependent upon the actual rendering-engine. V1.2 of the full QTI specification provides mechanisms to influence the ways in which the rendering-engine will render a question but these capabilities are **not** included within QTILite.

4.1.2 Standard Multiple Choice (Text)

Figure 4.2 shows a typical text-based multiple-choice question. The corresponding XML is listed after the figure. The user is required to choose one of the available options by clicking the appropriate radio button.

Which one of the listed standards committees is responsible for developing the token ring specification ?

- ☐ IEEE 802.3
- ☒ IEEE 802.5
- ☐ IEEE 802.6
- ☐ IEEE 802.11

Figure 4.2 Standard multiple choice (text) item.

The XML instance for Figure 4.2 is:

1	<questestinterop>
2	<item title="Standard Multiple Choice Item" id="IMS_V01_I_QTILiteExample004">
3	<presentation label="QTILiteExample004">
4	<material>
5	<mattext>
6	Which one of the listed standards committees is responsible
7	for developing the token ring specification ?
8	</mattext>
9	</material>
10	<response_lid id="MCb_01" rcardinality="Single" rtiming="No">
11	<render_choice>
12	<response_label id="A">
13	<material><mattext>IEEE 802.3</mattext></material>
14	</response_label>
15	<response_label id="B">
16	<material><mattext>IEEE 802.5</mattext></material>
17	</response_label>
18	<response_label id="C">
19	<material><mattext>IEEE 802.6</mattext></material>
20	</response_label>
21	<response_label id="D">
22	<material><mattext>IEEE 802.11</mattext></material>
23	</response_label>
24	</render_choice>
25	</response_lid>
26	</presentation>
27	</item>
28	</questestinterop>

This XML code is available in the file: 'ims_qtiasiv1p2/qt-lite/mchc_i_001/mchc_i_001.xml'. The key points of this example are:

- The actual material to be presented to pose the question is shown in lines 4-9;

- The possible responses and the content to be displayed with each option are shown in lines 12-23. The form of rendering is denoted by the <render_choice> element (line 11). Four possible choices are shown and these each have a unique label identifier;
- The type and number of responses is determined by the <response_lid> element which has the cardinality set as “Single” i.e. only one response is permitted and that there is no time dependence for the answer.

4.1.3 Enhanced Standard Multiple Choice (Text)

Figure 4.3 shows a typical text-based multiple-choice question. The corresponding XML is listed after the figure. The user is required to choose one of the available options by clicking the appropriate radio button. Note that this example is very similar to that shown in Figure 4.2.

Which ***one*** of the listed standards committees is responsible for developing the token ring specification ?

☐ IEEE 802.3

☒ IEEE 802.5

☐ IEEE 802.6

☐ IEEE 802.11

☐ None of the above.

Figure 4.3 Enhanced standard multiple choice (text) item.

The XML instance for Figure 4.3 is:

```

1      <questestinterop>
2          <item title="Standard Multiple Choice Item" ident="IMS_V01_I_QTILiteExample005">
3              <presentation label="QTILiteExample005">
4                  <material>
5                      <mattext>Which </mattext>
6                      <matemtext>one </matemtext>
7                      <mattext>of the listed standards committees is responsible
8                          for developing the token ring specification ?
9                      </mattext>
10                 </material>
11                 <response_lid ident="MCb_01" rcardinality="Single" rtiming="No">
12                     <render_choice shuffle="Yes">
13                         <response_label ident="A">
14                             <material><mattext>IEEE 802.3</mattext></material>
15                         </response_label>
16                         <response_label ident="B">
17                             <material><mattext>IEEE 802.5</mattext></material>
18                         </response_label>
19                         <response_label ident="C">
20                             <material><mattext>IEEE 802.6</mattext></material>
21                         </response_label>
22                         <response_label ident="D">
23                             <material><mattext>IEEE 802.11</mattext></material>
24                         </response_label>
25                         <response_label ident="E" rshuffle="No">
26                             <material><mattext>None of the above.</mattext></material>
27                         </response_label>
28                     </render_choice>
29                 </response_lid>
30             </presentation>
31         </item>
32     </questestinterop>

```

This XML code is available in the file: 'ims_qtiasiv1p2/qtilite/mchc_i_002/mchc_i_002.xml'. The key points of this example are:

- The actual material to be presented to pose the question is shown in lines 4-10. The difference between this example and that discussed in Figure 4.2 is the inclusion of the emphasized word 'one'. This emphasis is created using line 5;
- The possible responses and the content to be displayed with each option are shown in lines 13-27. The form of rendering is denoted by the <render_choice> element (line 11). Five possible choices are now shown (four were given in Figure 4.2). The relative order of these choices can be varied as denoted by the use of the shuffle="Yes" attribute (line 12). The exception to the shuffle is the last choice that is fixed to occur always as the last choice by the usage of the rshuffle="No" attribute line 25).

The next stage is to use the same example shown in Figure 4.3 but to add response processing and feedback to the XML instance. This gives rise to the XML instance (lines 1-30 are identical to the previous XML example):

1	<questestinterop>
2	<item title="Standard Multiple Choice Item" ident="IMS_V01_I_QTILiteExample006">
3	<presentation label="QTILiteExample006">
4	<material>
5	<mattext>Which </mattext>
6	<matemtext>one </matemtext>
7	<mattext>of the listed standards committees is responsible
8	for developing the token ring specification ?
9	</mattext>
10	</material>
11	<response_lid ident="MCb_01" rcardinality="Single" rtiming="No">
12	<render_choice shuffle="Yes">
13	<response_label ident="A">
14	<material><mattext>IEEE 802.3</mattext></material>
15	</response_label>
16	<response_label ident="B">
17	<material><mattext>IEEE 802.5</mattext></material>
18	</response_label>
19	<response_label ident="C">
20	<material><mattext>IEEE 802.6</mattext></material>
21	</response_label>
22	<response_label ident="D">
23	<material><mattext>IEEE 802.11</mattext></material>
24	</response_label>
25	<response_label ident="E" rshuffle="No">
26	<material><mattext>None of the above.</mattext></material>
27	</response_label>
28	</render_choice>
29	</response_lid>
30	</presentation>
31	<resprocessing>
32	<outcomes>
33	<decvar vartype="Integer" defaultval="0"/>
34	</outcomes>
35	<respcondition title="Correct">
36	<conditionvar>
37	<varequal respident="MCb_01">B</varequal>
38	</conditionvar>
39	<setvar action="Set">1</setvar>
40	<displayfeedback feedbacktype="Response" linkrefid="Correct"/>
41	</respcondition>
42	</resprocessing>
43	<itemfeedback ident="Correct" view="Candidate">
44	<material><mattext>Yes, you are right.</mattext></material>
45	</itemfeedback>
46	</item>
47	</questestinterop>

This XML code is available in the file: 'ims_qtiasiv1p2/qt-lite/mchc_ir_002/mchc_ir_002a.xml'. The key points of this example are:

- The response processing information is contained in lines 31-42 and the associated feedback is given in lines 43-45;
- The response processing is designed to identify the correct answer. The test for the correct answer is shown in lines 36-38 (the presence of label 'B' is tested). If the correct answer is detected then the default scoring variable is set to 1 (line 39) and the feedback is triggered (line 40). The scoring variable is initialized as '0' (line 33) and so the score for an incorrect answer is 0 by implication;
- The feedback for the correct answer is denoted by the content shown in line 44.

The next stage is to use the same example shown in Figure 4.3 but to add response processing and feedback associate with the user selecting an incorrect answer. This gives rise to the XML instance (lines 1-41 are identical to the previous XML example):

1	<questestinterop>
2	<item title="Standard Multiple Choice Item" ident="IMS_V01_I_QTILiteExample007">
3	<presentation label="QTILiteExample007">
4	<material>
5	<mattext>Which </mattext>
6	<matemtext>one </matemtext>
7	<mattext>of the listed standards committees is responsible
8	for developing the token ring specification ?
9	</mattext>
10	</material>
11	<response_lid ident="MCb_01" rcardinality="Single" rtiming="No">
12	<render_choice shuffle="Yes">
13	<response_label ident="A">
14	<material><mattext>IEEE 802.3</mattext></material>
15	</response_label>
16	<response_label ident="B">
17	<material><mattext>IEEE 802.5</mattext></material>
18	</response_label>
19	<response_label ident="C">
20	<material><mattext>IEEE 802.6</mattext></material>
21	</response_label>
22	<response_label ident="D">
23	<material><mattext>IEEE 802.11</mattext></material>
24	</response_label>
25	<response_label ident="E" rshuffle="No">
26	<material><mattext>None of the above.</mattext></material>
27	</response_label>
28	</render_choice>
29	</response_lid>
30	</presentation>
31	<resprocessing>
32	<outcomes>
33	<decvar vartype="Integer" defaultval="0"/>
34	</outcomes>
35	<respcondition title="Correct">
36	<conditionvar>
37	<varequal respident="MCb_01">B</varequal>
38	</conditionvar>
39	<setvar action="Set">1</setvar>
40	<displayfeedback feedbacktype="Response" linkrefid="Correct"/>
41	</respcondition>
42	<respcondition title="Incorrect">
43	<conditionvar>
44	<not><varequal respident="MCb_01">B</varequal></not>
45	</conditionvar>
46	<setvar action="Set">-1</setvar>
47	<displayfeedback feedbacktype="Response" linkrefid="Incorrect"/>
48	</respcondition>
49	</resprocessing>
50	<itemfeedback ident="Correct" view="Candidate">
51	<material><mattext>Yes, you are right.</mattext></material>
52	</itemfeedback>
53	<itemfeedback ident="Incorrect" view="Candidate">
54	<material>

55	<code><matemtext>No.</matemtext></code>
56	<code><mattext> The right answer is B.</mattext></code>
57	<code></material></code>
58	<code></itemfeedback></code>
59	<code></item></code>
60	<code></questestinterop></code>

This XML code is available in the file: 'ims_qtiasiv1p2/qtillite/mchc_ir_002/mchc_ir_002b.xml'. The key points of this example are:

- The response processing information to detect an incorrect answer is contained in lines 42-48 and the associated feedback is given in lines 53-58;
- The response processing is designed to identify the incorrect answer. The test for the incorrect answer is shown in lines 43-45 i.e. not the correct answer. If an incorrect answer is detected then the default scoring variable is set to -1 (line 46) and the feedback is triggered (line 47). Note that the score now has three possible values of 1 (correct), -1 (incorrect) and 0 (unanswered);
- The feedback for the incorrect answer is denoted by the content shown in lines 55-56 (note that the word 'No' is to be emphasized).

The next stage is to use the same example shown in Figure 4.3 but objectives and rubric information are added. This gives rise to the XML instance:

1	<code><questestinterop></code>
2	<code><item title="Standard Multiple Choice Item" ident="IMS_V01_I_QTILiteExample008"></code>
3	<code><objectives view="Candidate"></code>
4	<code><material></code>
5	<code><mattext>To test your understanding of LAN standards.</mattext></code>
6	<code></material></code>
7	<code></objectives></code>
8	<code><rubric view="Candidate"></code>
9	<code><material></code>
10	<code><mattext>Attempt all questions.</mattext></code>
11	<code></material></code>
12	<code></rubric></code>
13	<code><rubric view="Scorer"></code>
14	<code><material></code>
15	<code><mattext>Negative marking is employed.</mattext></code>
16	<code></material></code>
17	<code></rubric></code>
18	<code><presentation label="QTILiteExample008"></code>
19	<code><material></code>
20	<code><mattext>Which </mattext></code>
21	<code><matemtext>one </matemtext></code>
22	<code><mattext>of the listed standards committees is responsible</code>
23	<code>for developing the token ring specification ?</code>
24	<code></mattext></code>
25	<code></material></code>
26	<code><response_lid ident="MCb_01" rcardinality="Single" rtiming="No"></code>
27	<code><render_choice shuffle="Yes"></code>
28	<code><response_label ident="A"></code>
29	<code><material><mattext>IEEE 802.3</mattext></material></code>
30	<code></response_label></code>
31	<code><response_label ident="B"></code>
32	<code><material><mattext>IEEE 802.5</mattext></material></code>
33	<code></response_label></code>
34	<code><response_label ident="C"></code>

35	<code><material><mattext>IEEE 802.6</mattext></material></code>
36	<code></response_label></code>
37	<code><response_label ident="D"></code>
38	<code><material><mattext>IEEE 802.11</mattext></material></code>
39	<code></response_label></code>
40	<code><response_label ident="E" rshuffle="No"></code>
41	<code><material><mattext>None of the above.</mattext></material></code>
42	<code></response_label></code>
43	<code></render_choice></code>
44	<code></response_lid></code>
45	<code></presentation></code>
46	<code><resprocessing></code>
47	<code><outcomes></code>
48	<code><decvar vartype="Integer" defaultval="0"/></code>
49	<code></outcomes></code>
50	<code><respcondition title="Correct"></code>
51	<code><conditionvar></code>
52	<code><varequal respident="MCb_01">B</varequal></code>
53	<code></conditionvar></code>
54	<code><setvar action="Set">1</setvar></code>
55	<code><displayfeedback feedbacktype="Response" linkrefid="Correct"/></code>
56	<code></respcondition></code>
57	<code><respcondition title="Incorrect"></code>
58	<code><conditionvar></code>
59	<code><not><varequal respident="MCb_01">B</varequal></not></code>
60	<code></conditionvar></code>
61	<code><setvar action="Set">-1</setvar></code>
62	<code><displayfeedback feedbacktype="Response" linkrefid="Incorrect"/></code>
63	<code></respcondition></code>
64	<code></resprocessing></code>
65	<code><itemfeedback ident="Correct" view="Candidate"></code>
66	<code><material><mattext>Yes, you are right.</mattext></material></code>
67	<code></itemfeedback></code>
68	<code><itemfeedback ident="Incorrect" view="Candidate"></code>
69	<code><material></code>
70	<code><matemtext>No.</matemtext></code>
71	<code><mattext> The right answer is B.</mattext></code>
72	<code></material></code>
73	<code></itemfeedback></code>
74	<code></item></code>
75	<code></questestinterop></code>

This XML code is available in the file: 'ims_qtiasiv1p2/qtilite/mchc_ir_003/mchc_ir_003.xml'. The key points of this example are:

- The objectives are given in lines 3-7. These objectives are available to everyone as denoted by the view="All" attribute;
- Two sets of rubric information are supplied in lines 8-17. The first set is for the candidate (line 8) and the second set for the scorer (line 13).

4.1.4 Standard Multiple Choice (Image)

Figure 4.4 shows a typical image-based multiple-choice question. The corresponding XML is listed after the figure. The user is required to choose one of the available options by clicking the appropriate radio button.

Which symbol is the 'Stop' sign ?



Figure 4.4 Standard multiple choice (image) item.

The XML instance for Figure 4.4 is:

```

1  <!DOCTYPE questestinterop SYSTEM "IMS_QTIV1p1.dtd"[
2      <!NOTATION gif PUBLIC "">
3      <!ENTITY image01 SYSTEM "image1.gif" NDATA gif>
4      <!ENTITY image02 SYSTEM "image2.gif" NDATA gif>
5      <!ENTITY image03 SYSTEM "image3.gif" NDATA gif>
6      <!ENTITY image04 SYSTEM "image4.gif" NDATA gif>
7  ]>
8  <questestinterop>
9      <item title="Standard Multiple Choice with Images Item"
10         ident="IMS_V01_I_QTILiteExample010">
11          <presentation label="QTILiteExample0010">
12              <material>
13                  <mattext>Which symbol is the 'Stop' sign ?</mattext>
14              </material>
15              <response_lid ident="MC02" rcardinality="Single" rtiming="No">
16                  <render_choice shuffle="Yes">
17                      <response_label ident="A">
18                          <material>
19                              <matimage imagtype="image/gif" entityref="image01">
20                                  </matimage>
21                              </material>
22                          </response_label>
23                      <response_label ident="B">
24                          <material>
25                              <matimage imagtype="image/gif" entityref="image02">
26                                  </matimage>
27                              </material>
28                          </response_label>
29                      <response_label ident="C">
30                          <material>
31                              <matimage imagtype="image/gif" entityref="image03">
32                                  </matimage>
33                              </material>
34                          </response_label>
35                      <response_label ident="D">
36                          <material>
37                              <matimage imagtype="image/gif" entityref="image04">
38                                  </matimage>
39                              </material>
40                          </response_label>
41                      </render_choice>
42                  </response_lid>
43          </presentation>
44          <resprocessing>
45              <outcomes>
46                  <decvar varname="SCORE1" vartype="Integer" defaultval="1"/>

```

47	</outcomes>
48	<respcondition>
49	<conditionvar>
50	<varequal respident="MC02">B</varequal>
51	</conditionvar>
52	<setvar action="Set" varname="SCORE1">10</setvar>
53	<displayfeedback feedbacktype="Response" linkrefid="Correct"/>
54	</respcondition>
55	</resprocessing>
56	<itemfeedback ident="Correct" view="Candidate">
57	<material><mattext>Yes, you are right.</mattext></material>
58	</itemfeedback>
59	</item>
60	</questestinterop>

This XML code is available in the file: 'ims_qtiasiv1p2/qtillite/mchc_ir_004/mchc_ir_004b.xml'. The key points of this example are:

- The actual material to be presented to pose the question is shown in lines 12-14 and the associated graphics are identified in lines 19-20, 25-26, 31-32 and 37-38. The image files are externally referenced using the *entityref* attribute. These references are bound to the XML instance using the XML <!ENTITY...> statements in lines 2-6;
- The possible responses and the content to be displayed with each option are shown in lines 17-40. The form of rendering is denoted by the <render_choice> element (line 16) and shuffling of the options is to be used;
- The type and number of responses is determined by the <response_lid> element which has the cardinality set as "Single" i.e. only one response is permitted and that there is no time dependence for the answer;
- The processing to be undertaken once a user's response has been obtained is described in lines 44-55. The test for the correct response is shown in lines 49-51 i.e. the label of the correct response is 'B'. Each distinct test is enclosed in its own <respcondition> element (lines 48-54) and so a separate test is required to detect an incorrect answer;
- The result of getting the correct answer is to set the default scoring variable, 'SCORE1' to 10 (line 52) and to trigger some feedback (line 53). The scoring variable is declared in line 46;
- The feedback that is displayed as a result of getting the right answer is shown in lines 56-58.

5. XML Instance Example Lists

5.1 The ASI XML Instance Example Files

The full set of example files, as referred to in Section 4, are available as part of the Q&TILite Resource Kit. These files are listed in Table 5.1. The XML files are denote by an '.xml' extension.

Table 5.1 The Q&TILite XML example files.

File Name	Nature	Description
trfl_ir_001.xml	True/false Item	A true/false question. Response processing is supplied with the corresponding feedback.
trfl_ir_002.xml	True/false Item	A true/false question with objectives, rubric, response processing and feedback.
mchc_i_001.xml	Multiple-choice Item	A multiple-choice question with four possible responses. No response processing is supplied.
mchc_i_002.xml	Multiple-choice Item	The previous multiple-choice question has a further option added and has emphasised text. No response processing is supplied.
mchc_ir_002a.xml	Multiple-choice Item	The previous multiple-choice question has response processing added to identify the correct response and to give the corresponding feedback.
mchc_ir_002b.xml	Multiple-choice Item	The previous example is extended by adding feedback to respond to incorrect responses from the user.
mchc_ir_003.xml	Multiple-choice Item	The previous example is extended by adding 'objectives' and 'rubric' materials.
mchc_ir_004a.xml	Multiple-choice Item	A multiple-choice question using images. The images are referenced using the 'uri' attribute. Response processing is supplied with the corresponding feedback.
mchc_ir_004b.xml	Multiple-choice Item	The previous example is used but the images are referenced using the 'entityref' attribute.

The directory for these files is: 'ims_qtiasiv1p2/qt-lite/...'

6. Implementation Guidance

6.1 Items

6.1.1 Elements and their Attributes

Objectives

The *Objectives* element should be used to define the objectives of the Item for each of the available actors. The objectives can include any type of content and so they can be presented in a wide range of forms.

Rubric

The *rubric* element should be used to present material that is to be applied to a set of the contained Items. These descriptions can be supplied for each view that is to be supported. The element `<itemrubric>` is **not** a supported alternative in QTILite (this is deprecated in favour of `<rubric>` and is a legacy issue with V1.0 and V1.01)

6.1.2 Groups of Elements

Response Types

In the full QTI are five basic response types, as listed in Section 3.2 of the ‘IMS Q&TI: ASI Information Model’. It is important to realize that the response-type is determined by the manner in which the response is to be internally processed. In QTILite, only the single response classical multiple-choice question types are supported using the *response_lid* element.

Rendering Types

At present there are four rendering types: *render_choice*, *render_hotspot*, *render_slider* and *render_fib*. It is important to note that the rendering type is only loosely imposed by the response-type but it is closely linked to the educational objective of the question. In QTILite, only the classical multiple-choice question types can be supported using *render_choice*.

Itemfeedback

The *itemfeedback* element contains the *hint* and *solution* elements and its contents are triggered using the *displayfeedback* element. In QTILite, only the standard feedback is supported i.e. hints and solutions are not available.

Variable Manipulation

The manipulation of the scoring variables declared in the *outcomes/decvar* combination is contained within the *conditionvar* element. In QTILite the only variable comparisons are made individually using the *varequals* element. The state of these comparisons can be inverted using the logical ‘NOT’ element.

The response processing variables are declared using the *decvar* element. Each implementation of the QTILite must generate a default integer variable called ‘SCORE’ whose default value is zero. This variable is used whenever a condition test is applied and the corresponding *setvar* does not include a particular variable name. When supporting the response processing there is one special condition that can be trapped:

- When the response has not been answered - this can be supported using the *unanswered* element placed within *conditionvar* and is activated whenever that response has not been attempted.

6.2 Aggregated Scoring and Response Processing

A key question is how the test engine is to decide the type of responses and their subsequent response processing i.e. how does an implementation ascertain the sequence in which the response tests are to be applied to an Item that expects multiple responses. Consider the most simple case of a single response multiple choice question, “Which is the first working day of the week?”. The XML QTI code is shown below:

1	<questestinterop>
2	<item title="Single response" ident="A">
3	<presentation label="BasicExample002a">
4	<material>
5	<mattext>Which is the first working day of the week ?</mattext>
6	</material>
7	<response_lid ident="MCb_01" rcardinality="Single" rtiming="No">
8	<render_choice>
9	<response_label ident="A">
10	<material><mattext>Saturday</mattext></material>
11	</response_label>
12	<response_label ident="B">
13	<material><mattext>Monday</mattext></material>
14	</response_label>
15	<response_label ident="C">
16	<material><mattext>Wednesday</mattext></material>
17	</response_label>
18	<response_label ident="D">
19	<material><mattext>Tuesday</mattext></material>
20	</response_label>
21	<response_label ident="E">
22	<material><mattext>Sunday</mattext></material>
23	</response_label>
24	<response_label ident="F">
25	<material><mattext>Friday</mattext></material>
26	</response_label>
27	<response_label ident="G">
28	<material><mattext>Thursday</mattext></material>
29	</response_label>
30	</render_choice>
31	</response_lid>
32	</presentation>
33	<resprocessing>
34	<outcomes><decvar/></outcomes>
35	<respcondition title="Correct">
36	<conditionvar>
37	<varequal respident="MCb_01">B</varequal>
38	</conditionvar>
39	<setvar action="Set">1</setvar>
40	<displayfeedback feedbacktype="Response" linkrefid="Correct"/>
41	</respcondition>
42	</resprocessing>
43	<itemfeedback ident="Correct" view="Candidate">
44	<material><mattext>Yes, you are right.</mattext></material>
45	</itemfeedback>
46	</item>
47	</questestinterop>

The response processing test on line 37 does the check to see if the response has been 'Monday', the correct answer. The system is aware that a single response is required because of the value in line 7 of the *rcardinality* attribute i.e. 'Single'.

6.3 Naming Conventions

6.3.1 Identities and Labels

Many elements within the Q&TI specifications require unique identities and labels. Uniqueness is particularly important when importing from multiple sources. The earlier version of the QTILite specification (V1.1) recommended a 32-character naming convention. A new approach is now recommended that makes use of the IMS Persistent, Location-Independent Resource Identifier [IMS, 01]. This recommendation is based upon the following structure:

URN:<nid>:<nss>

Where:

<nid> = Namespace identifier
 = IMS-PLIRID-V1
 <nss> = Namespace specific string
 = <sourcelabel>:<sourceidentifier>:<schemelabel>:<schemeidentifier>
 = ABCDEFGH:abcdefghijklmnp:qtiv1p2:?_***...***

in which:

- ‘ABCDEFGH’ denotes the 1-8 character code for the organization responsible for creating the names/labels;
- ‘abcdefghijklmnp’ is the 1-16 character identifier assigned to the organization responsible for creating the names/labels;
- ‘qtiv1p2’ is the IMS specification and release version identifier for IMS QTILite;
- ‘?’ is either ‘A’ for *Assessment*, ‘S’ for *Section*, ‘I’ for *Item* or ‘B’ for *Object-bank*;
- ‘***...***’ is the 1-16 character code for the name/label of the object itself.

An example of this for an assessment identifier created by ‘ETS’ under the Q&TI v1.2 specification is:

URN:IMS-PLIRID-V1:ETS:23459:qtiv1p2:L_TESTITEMv001^a

Therefore, we request that organizations wishing to create proprietary extensions register their 1-8 letter organization identifier with IMS so that IMS can ensure a unique set of identifier names.

a. Note that the usage of the ‘ETS’ identifier is a fictional allocation used for the purposes of the example.

6.4 Scoping Rules

6.4.1 Identities and Labels

Scoping within XML is very limited. It is possible to create globally unique identifiers within a file by using the ‘ID’ attribute and reference to these elements is possible through the usage of ‘IDREF’ and ‘IDREFs’. During the development of the Q&TILite specification this global uniqueness was considered too constraining and so the scoping rules listed in Table 6.1 should be followed whenever possible:

Table 6.1 Scoping rules for identifiers.

Identifier Name	Role of the Identifier	Scoping Rule
item	Identifier for each Item.	Unique across the instance. Should be unique to the organization creating it. Should be based upon the naming convention described in Subsection 6.3.
response_lid	Identifier for the response_lid response-type.	This identifier is used to link the response supplied by the user and the corresponding response processing. This identifier <i>must</i> be unique across all of the <i>response</i> elements with an Item.

response_label	Identifier for each of the possible responses to be made by the user.	This identifier is used as part of the <i>respcondition</i> element processing within each <i>resprocessing</i> element. This identifier must be unique within the <i>render_choice</i> element it is used.
mattext	Identifier for text-based material. Note that the attribute <i>label</i> is used as this is optional and so not all <i>mattext</i> entries require a unique identifier.	This allows the text to be defined once and then referenced from any other location using the <i>matref</i> element. This identifier must be unique to all of the <i>mat***</i> elements used internally or from the Section and Item pool.
matemtext	Identifier for text-based material. Note that the attribute <i>label</i> is used as this is optional and so not all <i>matemtext</i> entries require a unique identifier.	This allows the emphasized text to be defined once and then referenced from any other location using the <i>matref</i> element. This identifier must be unique to all of the <i>mat***</i> elements used internally or from the Item pool.
matimage	Identifier for image-based material. Note that the attribute <i>label</i> is used as this is optional and so not all <i>matimage</i> entries require a unique identifier.	This allows the image to be defined once and then referenced from any other location using the <i>matref</i> element. This identifier must be unique to all of the <i>mat***</i> elements used internally or from the Item pool.

7. Compatibility with the Full IMS QTI

7.1 QTI ASI Compatibility

The compatibility between the different versions of the specification and the full QTI and QTILite is shown schematically in Figure 7.1.

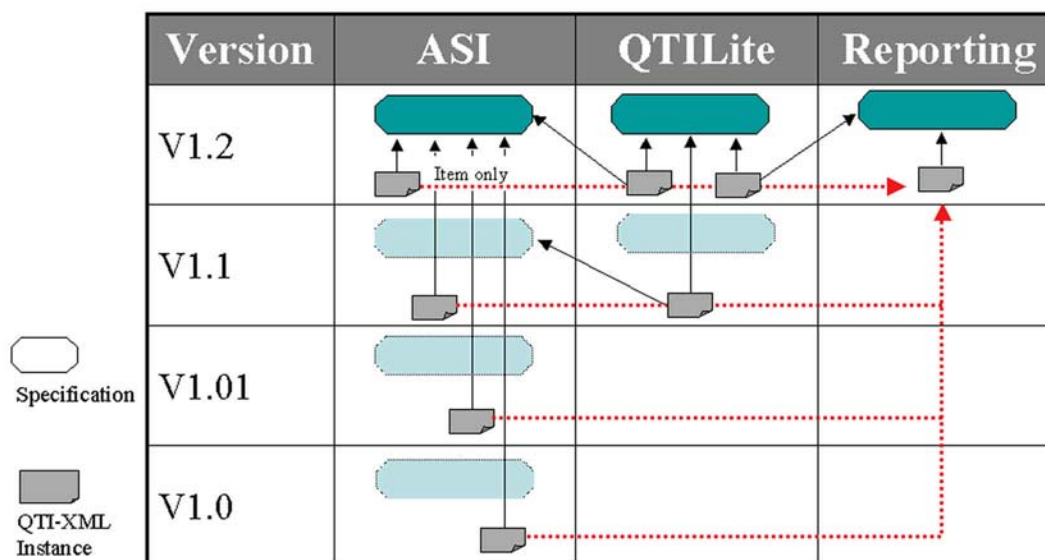


Figure 7.1 Compatibility representation of the set of released QTI specifications.

At the current time four versions of the full QTI specification have been released and two versions of the QTILite. Figure 7.1 shows that any instance file created under versions 1.0 and 1.01 are fully compatible with version 1.1. From the QTILite perspective, any instances created under V1.1 are not backwards compatible. This is because of the following issues:

- The *itemrubric* element is not supported in QTILite – this is deprecated in favour of *rubric*;
- The *entityref* and *xml:lang* attributes are available in QTILite but not in any release of the full QTI earlier than V1.1;
- The *matemtext* element is supported as a valid entry within content material.

All QTILite instances are compatible with the full QTI V1.1 specification. QTILite V1.1 and V1.2 instances are fully compatible as no functionality was added or altered to the Item data object. QTILite V1.2 and QTI V1.2 are compatible with the exception of:

- The *material* element was changed in V1.2 to support multiple *altmaterial* elements. This change was made to support accessibility issues and to allow the same content to be rendered in alternative languages.

7.2 Results Reporting Compatibility

The QTILite V1.1 and V1.2 specifications are fully compatible with the IMS QTI Results reporting specifications [QTI, 02f], [QTI, 02g], [QTI, 02h]. This means that the results from a QTILite-based evaluation can be reported using the IMS QTI results reporting XML binding (this is shown by the dotted lines leading into the Results Reporting structure).

8. Conformance

The purpose of this statement is to provide a mechanism for customers to fairly compare vendors of assessment systems, tools and content.² It is **not** mandatory for a vendor to support every feature of the QTILite specification, but a vendor must detail their level of support with a “Conformance Statement”. For example vendors may choose to accept or publish QTI data, but not choose to repackage QTI data. Compliance is represented by:

- Conformance summary – this is a summary that shows, in colloquial terms, the capabilities of a particular implementation with respect to the IMS QTILite specification;
- Interoperability statement – this is a detailed technical checklist that identifies all of the feature capabilities of the implementation in terms of the QTILite specification functions.

8.1 Valid Data Issues

Vendors claiming conformance shall publish, accept, and/or repackage valid QTILite data as defined by the DTD including proprietary extensions where applicable. Vendors claiming their tools publish QTILite shall export valid QTI data. Vendors claiming their system tools accept QTILite data shall be able to parse and recognize valid QTI data. Vendors claiming their system tools repackage QTILite data shall be able “pass through” valid QTILite data whether the tool recognizes the optional elements or not. Vendors claiming their assessment content conforms to this specification shall provide valid QTILite data. Publishers claiming their content conforms to QTILite shall provide valid QTILite data.

8.2 Conformance Summary

Vendors claiming conformance must provide a “Conformance Summary”, detailing their level of conformance, substantially similar to the information shown below, upon a reasonable request from a member of the IMS, or a prospective customer(s). It is expected that this table, a template of which is shown in Table 8.1, is a summary of the information given in the ‘Interoperability statement’. The intention is for the ‘Conformance Summary’ to be informative in nature.

Table 8.1 QTILite conformance summary.

	QTILite Conformance Summary (Version 1.2)		
	Publish (export, data)	Accept (import, display)	Repackage Feature
Items supported	Y or N	Y or N	Y or N
Question types	Y or N	Y or N	Y or N
Multiple choice	Y or N	Y or N	Y or N
Objectives & Rubric	Y or N	Y or N	Y or N
Response processing	Y or N	Y or N	Y or N
Feedback	Y or N	Y or N	Y or N
Material Content	Y or N	Y or N	Y or N
Text	Y or N	Y or N	Y or N
Emphasized text	Y or N	Y or N	Y or N
Image	Y or N	Y or N	Y or N

2. This conformance statement should be used for guidance only. IMS is working on a formal Certification and Conformance process but this will only be available in late 2002.


























































Completion of the three columns is intended to reflect:

- Publish – this implies that the XML instance contains the identified elements. If such an element is not ticked then it will not occur within the exported QTI-XML instance(s);
- Accept – it is assumed that the ability to accept the contents of an element is accompanied by the ability to use, and if appropriate, display that content. If this is not the case but the content of the material can be exported then the ‘Repackage’ column can still be ticked;
- Repackage – this is the ability to import QTI_XML instances from one or more sources and to create a new instance that combines the imported information. It is not necessary for the repackaging system to be able to operate on the information supplied.

8.3 Interoperability Statement

An example of the detailed ‘Interoperability Statement’ is shown in Tables 8.2. Compliance to QTI means that at least one of the columns must be completed.

Table 8.2 QTILite interoperability statement.

QTILite Item		Version 1.2	
Optional Fields: <i>Identify all of that that are supported.</i>			
	Publish (export, data)	Accept (import, display)	Repackage Feature
objectives			
rubric			
presentation			
response_label			
response_lid			
render_choice			
resprocessing			
outcomes			
decvar			
rescondition			
setvar			
conditionvar			
itemfeedback			
material			
matemtext			
mattext			
matimage			
matref			
altmaterial			

It is recommended that the IMS Content and Packaging specification is used to package QTILite content to enable the system to determine that it has the resources to display the items. The conformance statement provide by a supplier must describe any optional attributes.

Note that the ‘Interoperability Statement’ addresses support for the various elements within the binding. The set of attributes are not considered. Inclusion of conformance with respect to attributes will be considered in later versions of the specification.

It is important that the ‘Interoperability Statement’ is clear in showing what is and, perhaps more importantly, **what is not** supported. The usage of descriptive conformance approach has been adopted to encourage vendors to be as clear as possible when describing the capabilities of their QTI-compliant systems.





































8.4 A QTILite Conformance Example



















An example of QTI conformance is shown in Tables 8.3 and 8.4. In this example the system being defined is an import only tool i.e. a tool that imports QTILite Items. This means that only the middle column in each table has content.

Table 8.3 A QTILite conformance summary example.

	QTILite Conformance Summary (Version 1.2)		
	Publish (export, data)	Accept (import, display)	Repackage Feature
Items supported	N/A	Y	N/A
Question types	N/A	Y	N/A
Multiple choice	N/A	Y	N/A
Objectives & Rubric	N/A	Y	N/A
Response processing	N/A	Y	N/A
Feedback	N/A	Y	N/A
Material Content	N/A	Y	N/A
Text	N/A	Y	N/A
Emphasized text	N/A	Y	N/A
Image	N/A	Y	N/A

Table 8.4 QTILite interoperability statement example.

Item	Version 1.2		
Optional Fields: <i>Identify all of that that are supported.</i>			
	Publish (export, data)	Accept (import, display)	Repackage Feature
objectives			
rubric			
presentation			
response_label			
response_lid			
render_choice			
resprocessing			
outcomes			
decvar			
respcondition			
setvar			
conditionvar			
itemfeedback			

material			
matemtext			
mattext			
matimage			
matref			
altmaterial			

Note that the 'Interoperability Statement' addresses support for the various elements within the binding. The set of attributes are not considered. Inclusion of conformance with respect to attributes will be considered in later versions of the specification.

Appendix A – Glossary of Terms

action	The <i>action</i> attribute is used by the <i>setvar</i> element to define the manner in which the identified variable is to be modified. The only action supported is 'set'.
altmaterial	The <i>altmaterial</i> element is used to define alternative material that is to be presented should the original material be incompatible with the capabilities of the display system. While the <i>altmaterial</i> can support the same types as the <i>material</i> element it would normally be conservative in manner i.e. typically consist of text only as an alternative to video, etc.
charset	The <i>charset</i> attribute is used to define the character set that is to be used for the text presentation. The character set is typically described in the string form such as 'ascii/us' that defines the repertoire.
conditionvar	This element contains the set of conditions that are to be applied to the user responses. Two mechanisms are possible for logical operations i.e. implicit and explicit. The explicit mechanism requires the usage of the <i>and</i> and <i>or</i> elements whereas the implicit approach assumes the elements contained with the <i>conditionvar</i> element are combined using the logical AND.
continue	The <i>continue</i> attribute is used to control the order of the response processing for each of the data structures. It is applied to the <i>respcondition</i> and <i>scorecondition</i> elements. When the attribute is set as 'No' then the condition processing terminates if that condition has been evaluated as 'True'.
decvar	The <i>decvar</i> element is used to declare the variables that are to be required to support the ASI scoring models. The integer variable 'SCORE' is the default declaration with a value of zero. The name, type, minimum, maximum and default value of the variables is determined by the <i>varname</i> , <i>vartype</i> , <i>minvalue</i> , <i>maxvalue</i> and <i>defaultval</i> attributes.
defaultval	The <i>defaultval</i> attribute is used by the <i>decvar</i> element to define the start value for the variable. This value is bounded by the <i>minvalue</i> and <i>maxvalue</i> attributes, when used. The type of value is determined by the type of variable declared i.e. it should conform to the <i>vartype</i> attribute as used by the <i>decvar</i> element.
displayfeedback	The <i>displayfeedback</i> element is the trigger for the presentation of feedback to the users. The type of feedback to be displayed is determined by the <i>feedbacktype</i> attribute. The <i>linkrefid</i> attribute is used to identify the element containing the feedback.
embedded	The <i>embedded</i> attribute is used by the <i>material</i> and <i>mat***</i> elements to indicate if the material is encoded within the element itself as opposed to referenced through a URL. The string value with the attribute is used to denote the type of encoding of the material - the default type is 'base64' encoding.
entityref	The <i>entityref</i> attribute is used as an alternative to the <i>uri</i> attribute. The key difference is that the <i>entityref</i> refers to an XML entity whose external linkage is bound to the XML instance itself. This binding also allows the XML-capable processes to intelligently handle the material. It is recommended that this attribute be used in preference to the <i>uri</i> attribute.
feedbacktype	The <i>feedbacktype</i> attribute is used by the <i>displayfeedback</i> element to denote the type of feedback that is to be invoked. The possible values for the <i>feedbacktype</i> attribute are Response, Solution and Hint. This attribute is required because the element containing the feedback may have more than one type of feedback within its body. For QTILite 'Response' is the only feedback type available.
height	The <i>height</i> attribute is used to denote the total y-axis size, in pixels, of the material being presented. If this attribute is not used then the system should determine the y-axis size from the defined x-axis size (given by the <i>width</i> attribute) and the subsequent scaling of the image itself to maintain its aspect ratio. If neither attribute is used then the size is determined directly by the image itself.

ident	The <i>ident</i> attribute is used to define the unique identifier for the element. The uniqueness of the identifier is not maintained by the XML and so a naming convention should be adopted. This identifier will be referenced by other elements and so it is essential that no logical conflicts occur within the XML file. The identifier string is typically 1-256 characters in length and whenever possible it should conform to the 'IMS Persistent, Location-Independent Resource Identifier'. Any character can be used within the 'ident'.
imagtype	This attribute is used in the <i>matimage</i> element to define the type of image file. The file type is assumed to take the form of 'image/****' where '****' is indicative of the file source e.g. 'jpeg' etc (the MIME format will be used). The mapping of this to an actual file type is beyond the scope of this specification and will be vendor specific.
item	The <i>item</i> element is one of the three primary interoperable ASI data structures. An Item is a combination of the question, the rendering form, the block formatting, the processing of the subsequent response and the corresponding feedback (including hints and solutions). Items can be imported/exported. Items are the only core ASI data structure that can be exchanged using the QTILite specification.
itemfeedback	The <i>itemfeedback</i> element is used to contain the full range of materials that are to be displayed as feedback to the user. The type of feedback can be determined by the sub-elements contained e.g. <i>hint</i> for hints, <i>solution</i> for solutions and <i>material</i> for responses. The <i>view</i> attribute is used to define the actors to whom the material can be displayed. Hints and solutions are not supported within QTILite.
label	The <i>label</i> attribute is intended for optionally labelling elements so that authoring tools can easily identify them. This means that elements that are to be used in similar manners should be commonly labelled e.g. all QuickTime video material could be labelled as "quicktimevideo". An editing/authoring tool could then easily search for all such instances with the file.
labelrefid	The <i>labelrefid</i> attribute is used by the <i>response_label</i> element to allow the different response labels to be linked to more complex scoring models. This attribute is for further study.
linkrefid	The <i>linkrefid</i> attribute is used to associate Items and Material that are defined elsewhere with a specific location within the defining data structures. This means that only a single copy of the data structure is required with multiple references to it.
matemtext	This element is used to define text that should be emphasized. It can be used wherever the <i>mattext</i> element is available and has the same attributes. The nature of the emphasis is specific to the rendering engine however all text should be handled in the same way.
material	This is the container for all types of content that is to be presented to the users. Each <i>material</i> element can have multiple types of content as defined by <i>matref</i> , <i>mattext</i> , <i>matemtext</i> , and <i>matimage</i> . Alternative material is available using the <i>altmaterial</i> element.
matimage	This element is responsible for the presentation of graphics/image content. The type of image must be described using the <i>imagtype</i> attribute. It is always a sub-element of <i>material</i> .
matref	The <i>matref</i> element enables material that has not been included in the associated data structures to be 'pulled into scope'. This means that material defined elsewhere can be readily utilized. The <i>linkrefid</i> attribute is used to identify the associated material.
mattext	This element is responsible for the presentation of text content. The type of text must be described using the <i>texttype</i> , <i>charset</i> , <i>xml:lang</i> and <i>encoding</i> attributes. It is always a sub-element of <i>material</i> .
maxnumber	The <i>maxnumber</i> attribute is used to limit the maximum number of responses that a user can enter for a particular response. Within QTILite this value is fixed as one.
minnumber	The <i>minnumber</i> attribute is used to limit the minimum number of responses that a user can enter for a particular response. Within QTILite this value is fixed as one.
not	The <i>not</i> element provides the mechanism by which the result of a comparison elements (e.g. <i>varequal</i> , <i>varlt</i> , etc.) can be inverted. If the comparison element returns a 'True' state then the <i>NOT</i> returned state is 'False' and vice-versa.

objectives	This <i>objectives</i> element is the container for the description of the objectives of the Assessments, Sections and Items - the context is defined by the usage. These objectives are defined with respect to the actor as defined by the <i>view</i> attribute.
outcomes	The <i>outcomes</i> element is used to contain the declaration of the variables for the Item-level response processing. The <i>outcomes</i> element contains the <i>decvar</i> and <i>interpretvar</i> sub-elements to define the scoring variables.
presentation	The <i>presentation</i> element contains all of the sub-elements used to present a question to a user. This includes the response-type, render-type and the material itself.
questestinterop	This is the root element. The top-most elements are either the <i>assessment</i> , <i>section</i> or <i>item</i> elements (excluding the comment element <i>qticomment</i>). Within QTILite only single or multiple Item exchanges are supported.
rcardinality	The <i>rcardinality</i> attribute is used by the response-type elements to indicate the number and nature of the response actions required from the user. The enumerated choices are Single, Multiple and Ordered as defined in Table 3.1 of the IMS Q&TI Information Model. Within QTILite this attribute is fixed as 'Single'.
render_choice	The <i>render_choice</i> element is used to indicate that the response-type is to be presented to the user in a text-based multiple-choice/multiple response format. The rendering system is responsible for the actual selection mechanism e.g. radio buttons, check buttons, etc.
rescondition	The <i>rescondition</i> element is used as the container for the application of the conditions to the Item-level responses. It also contains the associated score elements (<i>setvar</i>) and the feedback trigger (<i>displayfeedback</i>). The <i>continue</i> attribute is used to denote if further <i>rescondition</i> elements should be processed.
respident	The <i>respident</i> attribute is used by all of the condition test elements (<i>varequal</i> , <i>varlt</i> , <i>varlte</i> , etc.) to identify the response-type as defined by its <i>ident</i> attribute. This identification acts as the scope for the <i>response_label</i> to be considered by the condition tests.
response_label	This element is used to present the response options to the user. The input options presented to the user are influenced by the attributes: <i>rshuffle</i> (used for the <i>render_choice</i> element), <i>rarea</i> (used for the <i>render_hotspot</i> element), <i>rrange</i> (used for the <i>render_num</i> element), and <i>labelrefid</i> . The <i>response_label</i> element is available to all of the response-types and render-types.
response_lid	The <i>response_lid</i> element is the response-type associated with the selection of responses from a set of defined answers presented to the user. The actual content of the answer is not important. Instead the label associated with the defined answer is what will be used in the response processing.
resprocessing	The <i>resprocessing</i> element is used to contain the information on how Item responses are to be processed. This processing requires the declaration of the scoring variables, the condition processing of the responses and the associated score allocation.
rshuffle	The <i>rshuffle</i> attribute is used with the <i>response_label</i> element in the context of the <i>response_choice</i> element. The <i>rshuffle</i> attribute is enumerated as either Yes/No and it is used to denote that the associated response must not be shuffled i.e. its location must remain fixed with respect to the number of pre and post options. This attribute only has meaning if the shuffle attribute has been defined as 'Yes' in the <i>response_choice</i> element.
rtiming	This attribute is used with the response-type elements (<i>response_lid</i> , <i>response_xy</i> , etc.) to indicate that the time taken by a user to generate a response must be recorded. The manner and accuracy of this recording is specific to the question engine. Within QTILite this attribute is fixed as 'No'.
rubric	The <i>rubric</i> element is a specialized form of the <i>material</i> element available as a sub-element to <i>assessment</i> , <i>section</i> and <i>item</i> . This provides a mechanism to supply content that can be used to describe a range of information about the ASI as a whole. The content is made available as defined through the <i>View</i> attribute.

setvar	The <i>setvarelement</i> is used to control the allocation of the Item level scoring as a consequence of the response condition processing within the <i>conditionvar</i> element. The scoring is controlled by the <i>action</i> attribute and the corresponding variable is identified by the <i>varname</i> attribute. Within QTILite, 'SCORE' is the only permitted variable name.
shuffle	The <i>shuffle</i> attribute is used with the <i>response_choice</i> element. The <i>shuffle</i> attribute is enumerated as either Yes/No and it is used to denote whether the following <i>response_label</i> elements i.e. the allowed options to be chosen by the user, can be shuffled before the question is actually rendered and displayed. The default value is 'No'.
texttype	This attribute is used in the <i>mattext</i> and <i>matemtext</i> elements to define the type of text. The type of text is assumed to take the form of 'text/****' where '****' is indicative of the text source e.g. 'basic' etc (the MIME format is to be used. The inclusion of rtf and html documents also adopt this approach using 'text/rtf', 'text/html' and 'text/xhtml' respectively. It is assumed that any html is well-formed but not necessarily complete i.e. well formed fragments are supported.
title	The <i>title</i> attribute is an optional naming of the associated element. The title is used to help readability of the XML files. The title is usually 1-256 characters in length.
unanswered	The <i>unanswered</i> element is a sub-element of <i>conditionvar</i> . It is used to indicate that the condition test should take into account the situation when the question is not attempted.
uri	The <i>uri</i> attribute is used to identify an external resource i.e. typically a file name or URL. An alternative, and preferred method is the usage of the <i>entityref</i> attribute.
varequal	The <i>varequal</i> element is the equivalence test for the response identified by the accompanying <i>respident</i> attribute. If equivalent the value 'True' is returned.
varname	The <i>varname</i> attribute is used to define the name of the variable. This attribute is used by the <i>decvar</i> and <i>interpretvar</i> elements. The name can consist of any lower- and upper-case characters i.e. a-z, A-Z, 0-9 and underscore (0-9 should not be used for the first character).
vartype	The <i>vartype</i> attribute is used by the <i>decvar</i> element to define the type of variable being declared. The type of variable is: String, Boolean, Integer, Decimal, Scientific and Enumerated. The 'Enumerated' option enables the declaration of typed entries and the range of types is defined within the <i>members</i> attribute that must accompany the declaration. Within QTILite the only supported type of variable is 'Integer'.
view	The <i>view</i> attribute is used to define the 'actors' permitted to see the associated information e.g. feedback, objectives, etc. The supported actors are All (used to indicate access to all), Administrating Authority, Administrator, Assessor, Author, Candidate, Invigilator/Proctor, Psychometrician, Scorer and Tutor
width	The <i>width</i> attribute is used to denote the total x-axis size, in pixels, of the material being presented. If this attribute is not used then the system should determine the x-axis size from the defined y-axis size (given by the <i>height</i> attribute) and the subsequent scaling of the image itself to maintain its aspect ratio. If neither attribute is used then the size is determined directly by the image itself.
xml:lang	The <i>xml:lang</i> attribute is used wherever the language of the entry text can be varied. This attribute is used to define the language of the associated text. The format of the attribute shows that it is one of the core attributes provided by XML itself.

About This Document

Title	IMS Question & Test Interoperability QTILite Specification
Authors	Colin Smythe, Eric Shepherd, Lane Brewer and Steve Lay
Version	1.2
Version Date	11 February 2002
Status	Final Specification
Summary	This document presents the IMS QTILite Specification. This specification is one of the set of the IMS Question & Test Interoperability specifications. The QTILite specification conforms to the IMS QTI Information Model but realizes only a subset of that model.
Revision Information	22 January 2002
Purpose	Defines the entry-level version of the QTI specification. The QTILite functionality is a subset of the full QTI specification features and functionality.
Document Location	http://www.imsglobal.org/question/v1p2/imsqti_litev1p2.html

List of Contributors

The following individuals contributed to the development of this document:

Russell Almond	ETS, USA
Lane Brewer	Galton Technologies Inc.
Todd Brewer	Galton Technologies Inc.
Russell Grocott	Can Studios Ltd.
Andy Heath	CETIS/JISC, UK
Paul Hilton	Can Studios Ltd.
Richard Johnson	Goal Design Inc.
John Kleeman	Question Mark Computing Ltd.
Steven Lay	University of Cambridge Local Examinations Syndicate, UK
Jez Lord	Can Studios Ltd.
Paul Roberts	Question Mark Computing Ltd.
Nial Sclater	CETIS/University of Strathclyde, UK
Eric Shepherd	Question Mark Corporation
Colin Smythe	Dunelm Services Ltd.

Revision History

Version No.	Release Date	Comments
Final Version 1.1	9 March 2001	The released version 1.1 of the IMS QTI: QTILite Specification.
Final Version 1.2	11 February 2002	<p>Updated to be consistent with the changes made to the core V1.2 ASI documentation. No core changes have been made to the specification itself. The only small change is:</p> <ul style="list-style-type: none">• The inclusion of the <i>minnumber</i> and <i>maxnumber</i> attributes to the <i>render_choice</i> element (they have a fixed value of 1);• The inclusion of the usage of the IMS Persistent Location-independent Resource Identifier for the Item <i>ident</i> attribute naming convention;• The usage of the Turbo XML V2.2.1 tool to provide a more detailed visualization of the XML DTD structures;• Removal of the Appendix containing the uncommented listing of the QTILite DTD source;• Editorial corrections throughout the document. <p>All changes are denoted by shaded text.</p>

Index

A

Administrating Authority 46
 Administrator 8, 11, 12, 18, 46
 ASI 4, 5, 7, 22, 33, 34, 38, 43, 45, 48
 Assessment 5, 8, 36
 Assessment Elements
 assessment 4, 36, 39, 45
 Assessor 8, 11, 12, 18, 46
 Attributes
 action 16, 23, 27, 28, 30, 32, 35, 43, 46
 charset 19, 20, 43, 44
 columns 40
 continue 16, 43, 45
 defaultval 15, 27, 28, 30, 31, 43
 embedded 21, 43
 encoding 21, 43, 44
 entityref 20, 21, 31, 32, 33, 38, 43, 46
 height 20, 21, 43, 46
 ident 11, 13, 14, 16, 17, 18, 23, 24, 26, 27, 28, 29, 30, 31, 32, 35, 44, 45
 imagetype 20, 31, 44
 label 11, 12, 13, 14, 16, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 31, 32, 35, 36, 37, 44, 45
 linkrefid 16, 21, 23, 27, 28, 30, 32, 35, 43, 44
 maxvalue 43
 members 46
 minvalue 43
 rarea 45
 rcardinality 13, 23, 24, 26, 27, 28, 29, 31, 35, 45
 respident 17, 18, 23, 27, 28, 30, 32, 35, 45, 46
 rrange 45
 rshuffle 14, 26, 27, 28, 30, 45
 rtiming 13, 23, 24, 26, 27, 28, 29, 31, 35, 45
 shuffle 13, 26, 27, 28, 29, 31, 45, 46
 texttype 19, 20, 44, 46
 title 10, 16, 18, 23, 24, 26, 27, 28, 29, 30, 31, 35, 46
 type 15, 16, 19, 20, 23, 25, 32, 34, 36, 43, 44, 46
 uri 19, 20, 21, 33, 43, 46
 varname 15, 16, 31, 32, 43, 46
 vartype 15, 27, 28, 30, 31, 43, 46

view 10, 11, 12, 18, 23, 27, 28, 29, 30, 32, 34, 35, 44, 45, 46
 width 20, 21, 43, 46
 xmllang 20, 38, 44, 46
 Author 8, 11, 12, 18, 46

B

Basic 22

C

Candidate 8, 11, 12, 18, 23, 27, 28, 29, 30, 32, 35, 46
 Common Elements
 altmaterial 19, 21, 38, 40, 42, 43, 44
 conditionvar 16, 17, 18, 23, 27, 28, 30, 32, 34, 35, 40, 41, 43, 46
 decvar 14, 15, 23, 27, 28, 30, 31, 34, 35, 40, 41, 43, 45, 46
 displayfeedback 16, 23, 27, 28, 30, 32, 34, 35, 43, 45
 interpretvar 45, 46
 matemtext 19, 20, 21, 26, 27, 28, 29, 30, 37, 38, 40, 42, 44, 46
 material 11, 12, 13, 14, 18, 19, 20, 21, 23, 24, 26, 27, 28, 29, 30, 31, 32, 34, 35, 37, 38, 40, 42, 43, 44, 45, 46
 matimage 19, 20, 21, 31, 37, 40, 42, 44
 matref 19, 21, 37, 40, 42, 44
 mattext 19, 21, 23, 24, 26, 27, 28, 29, 30, 31, 32, 35, 37, 40, 42, 44, 46
 matvideo 44
 objectives 10, 11, 18, 22, 29, 30, 33, 34, 40, 41, 45, 46
 other 14, 37, 44
 outcomes 14, 15, 23, 27, 28, 30, 31, 32, 34, 35, 40, 41, 45
 qticomment 45
 reference 19, 20, 21, 36
 rubric 10, 11, 18, 22, 29, 30, 33, 34, 38, 40, 41, 45
 scorecondition 43
 setvar 16, 23, 27, 28, 30, 32, 34, 35, 43, 45, 46
 unanswered 17, 18, 29, 34, 46
 varequal 17, 23, 27, 28, 30, 32,

35, 44, 45, 46
 varlt 44, 45
 varlte 45
 Conformance 5, 39, 41

E

Elements

altmaterial 19, 21, 38, 40, 42, 43, 44
 assessment 4, 36, 39, 45
 conditionvar 16, 17, 18, 23, 27, 28, 30, 32, 34, 35, 40, 41, 43, 46
 decvar 14, 15, 23, 27, 28, 30, 31, 34, 35, 40, 41, 43, 45, 46
 displayfeedback 16, 23, 27, 28, 30, 32, 34, 35, 43, 45
 hint 34, 44
 interpretvar 45, 46
 itemfeedback 10, 11, 16, 18, 23, 27, 28, 29, 30, 32, 34, 35, 40, 41, 44
 itemrubric 34, 38
 matemtext 19, 20, 21, 26, 27, 28, 29, 30, 37, 38, 40, 42, 44, 46
 material 11, 12, 13, 14, 18, 19, 20, 21, 23, 24, 26, 27, 28, 29, 30, 31, 32, 34, 35, 37, 38, 40, 42, 43, 44, 45, 46
 matimage 19, 20, 21, 31, 37, 40, 42, 44
 matref 19, 21, 37, 40, 42, 44
 mattext 19, 21, 23, 24, 26, 27, 28, 29, 30, 31, 32, 35, 37, 40, 42, 44, 46
 matvideo 44
 objectives 10, 11, 18, 22, 29, 30, 33, 34, 40, 41, 45, 46
 order 26, 43
 other 14, 37, 44
 outcomes 14, 15, 23, 27, 28, 30, 31, 32, 34, 35, 40, 41, 45
 presentation 10, 11, 12, 13, 18, 23, 24, 26, 27, 28, 29, 30, 31, 35, 40, 41, 43, 44, 45
 qticomment 45
 reference 19, 20, 21, 36
 render_choice 12, 13, 14, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 34, 35, 37, 40, 41, 45, 48

render_fib 34	25, 26, 27, 28, 29, 30,	46
render_hotspot 34, 45	31, 32, 34, 35, 37, 40,	Scorer 8, 11, 12, 18, 29, 46
render_slider 34	41, 45, 48	Tutor 8, 11, 12, 18, 46
rescondition 14, 15, 16, 17, 23,	render_fib 34	Proctor 8, 46
27, 28, 30, 32, 35, 37,	render_hotspot 34, 45	Psychometrician 8, 11, 12, 18, 46
40, 41, 43, 45	render_slider 34	
response_label 13, 14, 17, 18,	rescondition 14, 15, 16, 17, 23,	Q
23, 24, 26, 27, 28, 29,	27, 28, 30, 32, 35, 37,	QTILite 1, 4, 5, 7, 8, 9, 10, 12, 13,
30, 31, 35, 37, 40, 41,	40, 41, 43, 45	14, 15, 16, 18, 22, 23, 34, 36,
44, 45, 46	response_label 13, 14, 17, 18,	38, 39, 40, 41, 47, 48
response_lid 12, 17, 18, 23, 24,	23, 24, 26, 27, 28, 29,	Question 1, 4, 5, 6, 39, 41, 47
25, 26, 27, 28, 29, 30,	30, 31, 35, 37, 40, 41,	
31, 32, 34, 35, 36, 40,	44, 45, 46	R
41, 45	response_lid 12, 17, 18, 23, 24,	Resource Identifier 5, 36, 44
response_xy 45	25, 26, 27, 28, 29, 30,	Response 16, 23, 27, 28, 30, 32, 33,
resprocessing 10, 11, 14, 15, 18,	31, 32, 34, 35, 36, 40,	34, 35, 39, 41, 43
23, 27, 28, 30, 31, 32,	41, 45	Response processing 33, 39, 41
35, 37, 40, 41, 45	response_xy 45	
scorecondition 43	resprocessing 10, 11, 14, 15, 18,	S
selection 11, 12, 18, 45	23, 27, 28, 30, 31, 32,	Section 5, 33, 34, 36, 37
setvar 16, 23, 27, 28, 30, 32, 34,	35, 37, 40, 41, 45	Selection & ordering 5
35, 43, 45, 46	solution 34, 44	Elements
solution 34, 44		order 26, 43
unanswered 17, 18, 29, 34, 46	M	selection 11, 12, 18, 45
varequal 17, 23, 27, 28, 30, 32,	Meta-data	Solution 43
35, 44, 45, 46	Description 11, 12, 33	
varlt 44, 45	Elements	T
varlte 45	qmd_renderingtype 34	True/false 4, 33
	qmd_responsetype 34	Tutor 8, 11, 12, 18, 46
H	Language 5	
Hints 4	Objectives 34, 39, 41	V
	Resource Identifier 5, 36, 44	Version 1.01 Additions
I	Version 1, 5, 6, 39, 40, 41, 48	Elements
Interoperability structures	Multiple choice 39, 41	rubric 10, 11, 18, 22, 29,
Assessment 5, 8, 36	O	30, 33, 34, 38, 40,
Item 5, 8, 10, 11, 12, 18, 22, 24,	Outcomes 5	41, 45
26, 27, 28, 29, 31, 33,	Outcomes processing	Version 1.1 Additions
34, 36, 37, 38, 40, 41,	Elements	Attributes
44, 45, 46	outcomes 14, 15, 23, 27, 28,	entityref 20, 21, 31, 32, 33,
Section 5, 33, 34, 36, 37	30, 31, 32, 34, 35,	38, 43, 46
Invigilator 8, 46	40, 41, 45	xmlang 20, 38, 44, 46
Item 5, 8, 10, 11, 12, 18, 22, 24, 26,		Elements
27, 28, 29, 31, 33, 34, 36, 37,	P	matemtext 19, 20, 21, 26,
38, 40, 41, 44, 45, 46	Participant	27, 28, 29, 30, 37,
Item Elements	Administering Authority 11, 12,	38, 40, 42, 44, 46
hint 34, 44	18	Version 1.2 Additions
itemfeedback 10, 11, 16, 18, 23,	Administrator 8, 11, 12, 18, 46	Elements
27, 28, 29, 30, 32, 34,	Assessor 8, 11, 12, 18, 46	order 26, 43
35, 40, 41, 44	Author 8, 11, 12, 18, 46	outcomes 14, 15, 23, 27, 28,
itemrubric 34, 38	Candidate 8, 11, 12, 18, 23, 27,	30, 31, 32, 34, 35,
presentation 10, 11, 12, 13, 18,	28, 29, 30, 32, 35, 46	40, 41, 45
23, 24, 26, 27, 28, 29,	Invigilator 8, 46	reference 19, 20, 21, 36
30, 31, 35, 40, 41, 43,	Proctor 8, 46	selection 11, 12, 18, 45
44, 45	Psychometrician 8, 11, 12, 18,	
render_choice 12, 13, 14, 23, 24,		

X

XML 4, 5, 6, 8, 9, 10, 20, 21, 22, 23,
24, 25, 26, 27, 28, 29, 30, 31,
32, 33, 34, 36, 38, 40, 43, 44, 46

XML Schema

DTD 5, 39

XSD 5

XSD 5

IMS Global Learning Consortium, Inc. (“IMS”) is publishing the information contained in this IMS Question & Test Interoperability QTILite Specification (“Specification”) for purposes of scientific, experimental, and scholarly collaboration only.

IMS 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 would appreciate receiving your comments and suggestions.

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

Please refer to Document Name: IMS Question & Test Interoperability QTILite Specification

Date: 11 February 2002