IMS Question and Test Interoperability Conformance Guide

Version 2.0 Final Specification

Date Issued:	24 January 2005
Latest version:	http://www.imsglobal.org/question/qti_v2p0/imsqti_confv2p0.html
Register comments or implementations:	http://www.imsglobal.org/developers/ims/imsforum/categories.cfm?catid=23

IPR and Distribution Notices

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

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

If you wish to 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/qti_v2p0/qtiv2p0speclicense.html.

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

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

Document Name: IMS Question and Test Interoperability Conformance Guide
Date: 24 January 2005

Table of Contents

1. Introduction

In order to make meaningful statements about interoperability it is necessary to consider the issue of QTI-conformant data and the associated issue of what a system developer needs to do to ensure that their system conforms.

A system vendor or data publisher makes a conformance statement that can be used by the community to compare the capabilities of their product with others. To facilitate creation of conformance statements contentProfile and bankProfile classes are defined that enable a rigorous approach to describing the extent to which the item information and packaging models are supported. The same classes can of course be used to describe a set of requirements. Used in this way they enable smaller communities to express profiles of this specification. For information and advice about setting up and running such communities, readers are referred to the IMS Application Profile Guidelines Whitepaper [IMS_AP].

This specification defines two profiles that can be used as the basis for determining interoperability needs in the absence of any more specific profiling requirement. These profiles are called *QTI-Lite Version 2* (which applies only to content) and *QTI-All Version 2* and can be used to interpret statements such as “conforms to all of QTI Version 2”.

Communities that define their own profiles are strongly encouraged to ensure that all objects conforming to their profile also conform to the QTI-All Version 2 profile described in this document except with respect to additional media types (see objectType and imageType). Profiles that allow (or even require) objects that do not conform to QTI-All Version 2 should describe themselves as *extensions* of QTI.

1.1 References

IMS_AP	IMS Application Profile Guidelines Whitepaper, Version 1.0
IMS_MD_Binding	IMS Learning Resource Meta-Data XML Binding, Version 1.2.1
LOM	IEEE 1484.12.1-2002 Standard for Learning Object Meta-data (LOM)

2. Conforming Data

This specification defines several types of data objects that may be exchanged between systems and hence require defined levels of interoperability. For example, a set of item statistics may be described as *QTI Version 2 Conformant*. This section explains what such conformance statements mean.

2.1 Assessment Items

Assessment Items must be XML documents that conform to the XML schema for assessmentItem defined by this specification and to the additional content constraints described in the information model.

2.2 Item Packages

Item packages must conform to the IMS Content Packaging specification and contain assessment items packaged in accordance with the requirements described in the Integration Guide.

2.3 Item Statistics

Item statistics must be XML documents that conform to the XML schema for usageData defined by this specification.

2.4 Response Processors

Response Processors must be XML documents that conform to the XML schema for responseProcessing defined by this specification and to the additional content constraints described in the information model.

3. Conforming Systems

In addition to defining conformance criteria for the data objects that are exchanged between interoperable systems this specification also describes requirements on the way those systems interpret the information described by those data objects. Systems that describe themselves as conforming to "QTI Version 2" must make reference to an appropriate profile. The requirements on each type of system are described below.

3.1 Publishing System

A conformant publishing system is any system that can export conforming *assessment items* packaged as *item packages* without *requiring* the use of the extension elements customInteraction and customOperator.

A publishing system may also publish content in a variety of other formats, including some QTI-based formats that make use of the extension elements, but it must be possible to separate this output or the modes of operation that generate it. For example, a publishing system may contain a flag to turn off the use of QTI extensions when publishing content and skip items from the selected data set that would have required them.

A publishing system should create a contentProfile that describes the range of content it can export. The main purpose of such a profile is to describe the requirements for a system that needs to import the data and does not imply that the publishing system exploits the full range of functionality it describes. For example, a publishing system that exports only single response multi-choice questions as conformant QTI assessment items would still add *choiceInteraction* as an interactionType to its contentProfile even though this describes multiple-response multi-choice questions too (these two question types being inseparable in the contentProfile).

3.2 Authoring Systems

A conformant authoring system allows item authors to create new items, to edit existing items imported from conforming item packages and to export items into new or updated item packages.

Authoring systems must set or adjust the toolName and toolVersion appropriately when exporting items (unless no changes have been made). When exporting items, all use of extensions must be consistent with the conventions of the tool referred to by these attributes. The extension mechanisms are:

- The label attribute on bodyElement
- The customInteraction class.
- The customOperator class.

Authoring systems should ignore information represented by the extension mechanisms when importing an item that was created by an incompatible tool.

Authoring systems should also ensure that data that *can* be represented by the information model defined by this specification *is* represented in that way. In other words, authoring systems should not make use of the extension mechanisms to represent information that could have been represented without them.

This requirement is made to ensure that authoring systems meet the reasonable expectations of authors when exporting assessment items. For example, an author who creates a question containing a simple choice represented by hotspots on a background image can reasonably expect the exported data to contain a hotspotChoice and not a customInteraction containing a proprietary applet that implements the same functionality on a limited set of delivery engines.

A system that uses an extension mechanism to represent data that can be represented directly in the information model must not claim conformance for that part of the information model in its conformance profile.

Note that an tool may combine the functions of authoring system and delivery engine, to allow authors to try out their items, but it is not required to do so. Where a tool contains a conformant authoring system and a delivery engine it should ensure that the delivery engine is also conformant to prevent authors being misled.

An authoring system should create a contentProfile to describe the range of QTI content that it supports.

3.3 Item Bank Systems

An item bank system is a tool for managing collections of items, their meta-data and any associated usage data.

A conformant item bank system allows item bank managers to import and export collections of items from item packages. Item bank systems must not alter the items' assessmentItem data. Though a given tool may combine the features of an item bank system with an authoring system, to be a *conformant* item bank system it must still be capable of importing, managing and exporting collections of items without modification of the associated assessmentItem data.

An item bank system should create a bankProfile to describe the range of features that it supports. Version 1 of this specification described an information model for *objectbank*s, *assessments* and *results* which have not been updated by this version but **may** be updated by future versions. Therefore, the conformance of item bank systems with respect to the interoperability of item banks, assessments and results and the associated bankProfile class is subject to change.

3.4 Delivery Engines

A delivery engine is the component of a system that allows the user or *candidate* to interact with an item, to assign values to response variables and to invoke response processing and provide feedback as appropriate. A delivery engine may be part of a full-blown assessment system or it may simply be a component of an authoring or editing system.

A conformant delivery engine conforms to the requirements described in the information model with respect to its behavior in delivering the items. For example, it must provide suitable controls that operate in accordance with the requirements of each supported interaction and maintain the data described by the itemSession.

4. Conformance Profiles

4.1 Authoring and Delivery Systems

Class : `contentProfile`

This class provides a framework for describing the capabilities or requirements of an authoring system or delivery engine. Most of the elements of the profile are booleans that indicate whether or not a specific feature is supported (*true*) or not supported (*false*). When being used in the context of expressing requirements the values correspond to required or optional respectively. This profile class does not support exclusion of features.

Contains : `composite boolean [1]` Whether or not the system supports composite items.

Contains : `adaptive boolean [1]` Whether or not the system supports adaptive items.

Contains : `timeDependent boolean [1]` Whether or not the system supports time dependent items.

Contains : `templates boolean [1]` Whether or not the system supports item templates.

Contains : `textElements boolean [1]` Whether or not the system supports the XHTML text elements. A profile that supports any of the other XHTML element groups should support this one too.

Contains : `listElements boolean [1]` Whether or not the system supports the XHTML list elements.

Contains : `objectElements boolean [1]` Whether or not the system supports the XHTML object elements.

Contains : `objectType mimeType [*]` For systems that support the object element, a list of the types of object supported. For example: image/jpeg, audio/aiff, etc.

Contains : `presentationElements boolean [1]` Whether or not the system supports the XHTML presentation elements.

Contains : `tableElements boolean [1]` Whether or not the system supports the XHTML table elements.

Contains : `imageElement boolean [1]` Whether or not the system supports the XHTML image element.

Contains : `imageType mimeType [*]` For systems that support the image element, a list of the types of images supported. For example: image/png, image/jpeg, etc.

Contains : `hypertextElement boolean [1]` Whether or not the system supports the XHTML hypertext element.

Contains : `mathElement boolean [1]` Whether or not the system supports the MathML <math> element.

Contains : `mathVariable boolean [1]` Whether or not the system support the expansion of template variable names in MathML expressions.

Contains : `feedbackIntegrated boolean [1]` Whether or not the system supports integrated feedback, i.e., the feedbackBlock class.

Contains : `feedbackModal boolean [1]` Whether or not the system supports modal feedback, i.e., the modalFeedback class.

Contains : `rubric boolean [1]` Whether or not the system supports rubric blocks, i.e., the rubricBlock class.

Contains : `printedVariables boolean [1]` Whether or not the system has core support for the printedVariable element. Note that support for the r conversion type specifier is controlled separately rounding.

Contains : `interactionType [*]` The supported interaction type(s). The vocabulary is comprised of the names, as defined in the information model, of the leaf classes derived from interaction with the exception of customInteraction. See below for interaction-specific conformance notes.

Contains : `responseRules boolean [1]` Whether or not the system supports response rules in response processing. Systems that set this to true are assumed to be able to process arbitrary templates so need not list these individually. Note that support for the equalRounded and patternMatch operators is optional, see rounding and regexp respectively.

Contains : `rpTemplate uri [*]` For systems that only support response processing templates, a list of the templates supported.

Contains : `rounding boolean [1]` Whether or not the system supports advanced rounding: if printedVariables is supported then the *r* conversion type specifier is also supported.

Contains : `regexp boolean [1]` Whether or not the system supports regular expression matching: if the textEntryInteraction or extendedTextInteraction then the patternMask attribute is also supported; if responseRules is supported then the patternMatch operator is also supported.

Contains : `metadataProfile [1]` The parameters concerning the range of meta-data supported are described by a separate class.

Class : `metadataProfile`

Associated classes:

bankProfile, contentProfile

Contains : `imsmd boolean [1]` The system supports meta-data described by and bound according to the IMS meta-data specification [IMS_MD_Binding].

Contains : `lomMetadata boolean [1]` The system supports meta-data described by [LOM] and bound according to the associated XML binding.

Contains : `imsqtimd boolean [1]` The system supports meta-data described by and bound according to the qtiMetadata class defined in the associated Meta-data and Usage Data.

4.1.1 Interaction-Specific Conformance Notes

Most of the simple interactions can be supported in isolation. For example, it is possible to define a meaningful profile with the a single value of *choiceInteraction* for interactionType and no other conforming features.

Some interaction types require the use of XHTML-based elements that are subject to their own flag in the profile. A profile that contains an interactionType indicating support for one of these types must also set the flags for any required XHTML-based element to be valid. These requirements are listed below.

gapMatchInteraction	Requires textElements. If a system supports gapMatchInteraction and objectElements then it must support use of gapImg with any image objectTypes in the profile. A system that supports gapMatchInteraction but no image objectTypes does not support gapImg.
inlineChoiceInteraction, textEntryInteraction, hotTextInteraction, endAttemptInteraction	Require textElements.
hotspotInteraction, selectPointInteraction, graphicOrderInteraction, graphicAssociateInteraction, graphicGapMatchInteraction, positionObjectInteraction, drawingInteraction	Require objectElements and at least one suitable objectType.

4.2 Item Bank Systems

Class : `bankProfile`

This class provides a framework for describing the capabilities or requirements of an item bank system. It has a similar dual use for specifying capabilities and requirements as the contentProfile class.

Note that item bank systems must be able to import *and* export items from content packages and must be able to operate in a mode whereby all imported usage data and meta-data from a vocabulary or scheme to which conformance is claimed can be exported again with the same set of items.

Contains : `usageDataVocabulary uri [*]` The URI of a vocabulary file (or files) describing the vocabulary of supported usage data. Reference to a vocabulary indicates that a system supports usage-data files packaged according to the method described in Integration Guide.

Contains : `metadataProfile [1]` The flags describing the range of meta-data supported are the same as those used in the contentProfile.

4.3 QTI-Lite

QTI-Lite is presented as the entry-level profile to the full QTI specification and only concerns content, its creation, modification and delivery. In other words, it does not concern item bank systems. QTI-Lite does not support all of the features of the full specification but it is a proper profile, in other words an assessment item that conforms to the QTI-Lite profile also conforms to the default “QTI-All” profile defined below.

QTI-Lite Profile Definition
`conformance/imsqti_lite_profile.xml`

The key differences between the QTI-Lite and the QTI-All profile are:

- Only one interaction per item.
- The only interaction type to be supported by QTI-Lite is the choiceInteraction, suitable for use with simple multi-choice questions like one choice from many (e.g., “Yes/No”, “True/false” and “Likert scale”) and also with multiple response questions like one or more choice from many (e.g., select all that apply).
- Simple response processing using the Match Correct template enabling only a single right answer (or an exact matching group for multiple response).
- No support for integrated feedback
- Limited image types and structural formatting.
- No support for advanced features like adaptive items, templates or time based scoring.

Note that the inclusion of multiple-response questions represents an expansion of the scope of QTI-Lite since version 1 of this specification but that the restrictions on response processing, in particular the *lack* of support for the Map Response template, should not present a significant burden to implementors.

4.4 QTI All

The content profile that describes conformance to the full QTI Version 2 specification includes a complete list of features and a minimal set of media types.

QTI-All Content Profile Definition
`conformance/imsqticontent_all_profile.xml`

QTI-All Bank Profile Definition
`conformance/imsqtibank_all_profile.xml`

About This Document

Title	IMS Question and Test Interoperability Conformance Guide
Editor	Steve Lay (University of Cambridge)
Version	2.0
Version Date	24 January 2005
Status	**Final Specification**
Summary	This document describes the QTI Conformance Guide specification.
Revision Information	24 January 2005
Purpose	This document has been approved by the IMS Technical Board and is made available for adoption.
Document Location	http://www.imsglobal.org/question/qti_v2p0/imsqti_confv2p0.html

To register any comments or questions about this specification please visit:
http://www.imsglobal.org/developers/ims/imsforum/categories.cfm?catid=23

List of Contributors

The following individuals contributed to the development of this document:

Name	Organization	Name	Organization
Niall Barr	CETIS	Joshua Marks	McGraw-Hill
Sam Easterby-Smith	Canvas Learning	David Poor	McGraw-Hill
Jeanne Ferrante	ETS	Greg Quirus	ETS
Pierre Gorissen	SURF	Niall Sclater	CETIS
Regina Hoag	ETS	Colin Smythe	IMS
Christian Kaefer	McGraw-Hill	GT Springer	Texas Instruments
John Kleeman	Question Mark	Colin Tattersall	OUNL
Steve Lay	UCLES	Rowin Young	CETIS
Jez Lord	Canvas Learning		

Revision History

Version No.	Release Date	Comments
Base Document 2.0	09 March 2004	The first version of the QTI Item v2.0 specification.
Public Draft 2.0	07 June 2004	The Public Draft version 2.0 of the QTI Item specification.
Final 2.0	24 January 2005	The Final version 2.0 of the QTI specification.

Index