IMS Question & Test Interoperability
Best Practice & Implementation Guide

Final Specification
Version 1.0

About This Document

Title	IMS Question & Test Interoperability Best Practice & Implementation Guide
Authors	Colin Smythe and Eric Shepherd
Version	1.0
Version Date	5th June, 2000
Status	**Final Specification.**
Summary	This document provides additional information regarding IMS Question & Test Interoperability Best Practices and Implementation guidelines. It is meant to complement the IMS Question & Test Interoperability XML Binding and IMS Question & Test Interoperability Information Model documents.
Revision Information	5th June, 2000
Purpose	Defines the best practice usage of the XML binding for the Question & Test Interoperability Information Model specification.
Document Location	http://www.imsproject.org/question/qtbest01.html

List of Contributors

The following individuals contributed to the development of this document:

Russell Almond	ETS	Eric Shepherd	Question Mark Corporation
Richard Johnson	Goal Design Inc.	Colin Smythe	Dunelm Services Ltd
John Kleeman	Question Mark Ltd	Linda Steinberg	ETS
Wayne Martin	Miami Dade Community College		
Mike Pettit	Blackboard Inc		
Paul Roberts	Question Mark Ltd		

Revision History

Version No.	Release Date	Comments
Base 1.0	22nd December, 1999	The first released version of the IMS Question & Test Interoperability Best Practice & Implementation Guide Base Document.
Public Draft 1.0	18th February, 2000	The first public draft release of the IMS Question & Test Interoperability Best Practice & Implementation Guide.
Final 1.0	5th June, 2000	The version 1.0 of the IMS Question & Test Interoperability XML Binding Final Specification. The amendments from the Public Draft release are: • The meta-data elements have been added to each of the *assessmentmetada*, *sectionmetadata* and *itemmetadata* elements; • The *showdraw* attribute has been added to the render_hotspot element; • The objectives element has been added to replace the *assessobjectives*, *sectionobjectives* and *itemobjectives* elements; • The *varcontains* element has been renamed *varsubset*; • The *setmatch* attribute has been added to the *varsubset* element to control set testing; • The addition of new examples in Section 4 and the inclusion of Item response processing for several of those examp les; • All of the examples files have a new naming convention. This convention is described in Appendix C. The full list of files is given in Section 6.4. The list of new files includes examples with intentional errors to be used to aid parser validation; • A significant reworking of Section 7 (creating two new Sections) to present a clearer explanation of the specifications.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Overview

The IMS Question & Test Interoperability (Q&TI) Best Practice and Implementation Guide supplies information to help vendors implement tools to support *Assessment*, *Section* and *Item* content interoperability. This information is based upon:

- Examples of the range of *Item* types, and their corresponding XML code, supported by the specifications;
- Descriptions of the elements and recommendations on how they could be used;
- Descriptions of how combinations of the elements can be used to support certain functional requirements.

It is important to note that the Q&TI is concerned with interoperability and not processing capability.

1.2 Scope & Context

This document is the IMS Question & Test Interoperability (Q&TI) Best Practice & Implementation Guide. As such it should be used in conjunction with the:

- IMS Question & Test Interoperability Information Model Specification v1.0 [QTI, 00a];
- IMS Question & Test Interoperability XML Binding Specification v1.0 [QTI, 00b].

1.3 Structure of this Document

The structure of this document is:

2. RELATIONSHIP TO OTHER SPECIFICATIONS	The relationship of this specification activity to other IMS and external specification activities;
3. OVERALL DATA MODEL	A brief summary of the Question & Test Interoperability information model;
4. EXAMPLE BASIC ITEM TYPES	Examples of the basic Item types that are supported by this specification;
5. EXAMPLE COMPOSITE ITEM TYPES	Examples of the composite (combinations of the basic types) Item types supported by this specification;
6. EXAMPLE XML SCHEMA	Some complete examples of the XML required to present an Assessment, Section or Item;
7. IMPLEMENTATION GUIDANCE	Tips on how the distributed learning engines can make best usage of the Q&TI specification;
8. EXTENSIBILITY	Usage of the extensions facilities to support proprietary requirements;
9. V2.0 ISSUES & COMPATIBILITY	The initial scoping of the version 2 specifications and IMS's commitment to backwards compatibility;
10. CONFORMANCE	The expectations on systems that claim conformance to the Q&TI specifications;
APPENDIX A – DTDS & XDRS	The range of available Q&TI DTDs and XDRs;
APPENDIX B – GLOSSARY OF TERMS	A glossary of the key terms and elements used within the specification.
APPENDIX C – EXAMPLES INFORMATION	Specific descriptive information about the QTI example files.

1.4 Nomenclature

ADL	Advanced Distributed Learning
AICC	Aviation Industry CBT Committee
ANSI	American National Standards Institute
ASI	Assessment, Section, Item
CBT	Computer Based Training
DTD	Document Type Definition
FIB	Fill-in-Blank
IEEE	Institute of Electronic & Electrical Engineering
IHS	Image Hotspot
ISO	International Standards Organisation
JTC	Joint Technical Committee
LTSC	Learning Technology Standards Committee
NATO	North Atlantic Treaty Organisation
NUM	Numeric
Q&TI	Question & Test Interoperability
SCORM	Shareable Courseware Object Reference Model
STR	String
W3C	World Wide Web Consortium
XML	Extensible Mark-up Language

1.5 References

[QTI, 00a] *IMS Question & Test Interoperability Information Model Specification*, C.Smythe and E.Shepherd, Version 1.0, IMS, May 2000.

[QTI, 00b] *IMS Question & Test Interoperability XML Binding Specification*, C.Smythe and E.Shepherd, Version 1.0, IMS, May 2000.

2. Relationship to Other Specifications

2.1 IMS Specifications

Version 1.0 of the IMS Question & Test Interoperability specification is made up of three documents:

1. "IMS Question & Test Interoperability Information Model Specification – Version 1.0.
   This document describes the data structures that are used to provide interoperability of on-line distributed learning question and test systems;
2. "IMS Question & Test Interoperability XML Binding Specification – Version 1.0.
   This document describes how to encode the Question & Test Interoperability objects in XML and provides the corresponding XML DTD;
3. "IMS Question & Test Interoperability Best Practice & Implementation Guide – Version 1.0.
   This document (the one you are reading now) provides an overview and describes how the IMS Question & Test Interoperability Information Model and XML Binding specifications can be applied to specific types of interoperability scenarios.

The IMS Question & Test Interoperability specification is related to several other IMS specifications, both complete and in progress. This specification is intended to be consistent with these other initiatives wherever possible, in order to reduce redundancy and confusion between specifications. The related specifications are:

- IMS Meta-data Specification – the IMS Q&TI specification shares a number of common data object elements with the IMS Meta-data specification. A set of unique meta-data extensions are also used and these are defined within the Q&TI specifications themselves;
- IMS Content & Management Specification – the IMS Q&TI data model is a subset of the Content & Management data model i.e. Q&TI Assessments, Sections and Items are defined as content;
- IMS Profiles & Enterprise Interchange Specification – this specification adopts some of the Assessment information to construct results records (the Q&TI will address results issues as part of Version 2.0).

2.2 Related Specifications

2.2.1 IEEE P1484

The IEEE Learning Technology Standardisation Committee P1484 is the only body engaged in the educational domain, which has a recognised formal standing. Given the diversity of the fora represented by the participants in the IEEE, there exist a large number of working groups focused on specific activities, as well as more horizontal activities (such as the Architecture and Reference Model and the Glossary working groups) that attempt to tie the other work together. None of the current IEEE working groups and study groups (note a study group is formed to do preliminary work to scope any subsequent working group in the particular area) are focussed on Question & Test Interoperability.

2.2.2 Advanced Distributed Learning (ADL) Initiative

ADL is a US military programme started by the White House in 1997 which aims to advance the use of state-of-the-art online training amongst the countries defence forces. There is some collaboration with experts in military training applications from other NATO countries. ADL is very focused on content for particular areas of training. It also has the Shareable Courseware Object Reference Model (SCORM v0.9) as a working document to encourage discussion and input on the emerging standards. Again no separate Question & Test Interoperability specification development is underway.

2.2.3 Aviation Industry CBT Committee (AICC)

The Aviation Industry CBT Committee is a membership-based international forum that develops recommendations on interoperable learning technology, principally for the commercial aviation and related industries. As such its members include both plane and equipment manufacturers, carriers, software and multimedia vendors and a growing number of interested parties not directly engaged in the sector, but nevertheless interested in the work being

undertaken. A subgroup of the AICC are working with the ADL and other organisations from the IEEE LTSC. The IMS Q&TI specifications are to be presented to the AICC.

2.2.4 ISO/IEC JTC1/SC36 Learning Technology

As of 10th November 1999, the ISO/IEC Joint Technical Committee 1 meeting in Seoul agreed resolution 6, which brought into existence Sub-Committee 36 - Learning Technology. The international secretariat for SC36 will be provided by the US National Body: the American National Standards Institute (ANSI). ISO/IEC JTC1/SC36 is intended to address standardisation in the area of information technologies that support automation for learners, learning institutions, and learning resources. It is the intention that SC36 shall not create standards or technical reports that define educational standards, cultural conventions, learning objectives, or specific learning content. Their activity in the field of question and test has yet to be defined.

2.3 Specification Development Process

The development life-cycle for an IMS specification has been established as:

- Month 1 – set-up of the team including identification of the team lead (for QTI this is Eric Shepherd, Question Mark Corp), editor (for QTI this is Colin Smythe, Dunelm Services Ltd) and key collaborating groups and organisations;
- Month 2 – Team and scope/requirements development;
- Month 3 – Initial internal team documents developed;
- Month 4 – Base document development and vote. Approval of the Base Documents by the Technical Board;
- Month 5 – Document improvement. Open issues are identified and solutions developed. Companies are encouraged to develop code against the base documents;
- Month 6 – Further document improvement and feedback from organisations involved in developing implementations;
- Month 7 – Completion of the Public Draft Specification and approval by the IMS Technical Board;
- Months 8 & 9 – Accept feedback from organisations working to the Public Draft Specification. Resolve any issues raised;
- Month 10 – Completion of the Final Specification and approval by the IMS Technical Board. This is the combination of the Public Draft Specification plus resolutions due to experience gained in working with it.

A draft specification is developed within the IMS developer and user community, which currently includes more than 200 organisations from around the world. In a number of cases, one of these organisations represents many other organisations, such as the Australian Government's DETYA organisation, which provides access to the IMS community for all institutions of learning in Australia.

The term 'Base Document' is used for draft specifications that have reached a relatively high level of stability based on input from the team and the Technical Board. Base documents represent the stage in the specification process of final development and refinement. It is base documents that are presented in their final forms to the IMS Technical Board for vote. If approved, the document becomes a 'Public Draft Specification' and is listed as such on the IMS public web site. If not approved, the team works through whatever adjustments and recommendations the Technical Board provides, and then resubmits the document. After three months the Public Draft Specification should be adopted as a 'Final Specification'.

After a final specification is released, the team develops the next scope document for the subsequent work. New requirements and features dropped from the previous specification constitute the scope of the next effort.

3. Overall Data Model

3.1 Information Model

The data model for the Q&TI is shown in Figure 3.1 (this is the same as that described in the QTI information Model, [QTI, 00a]).

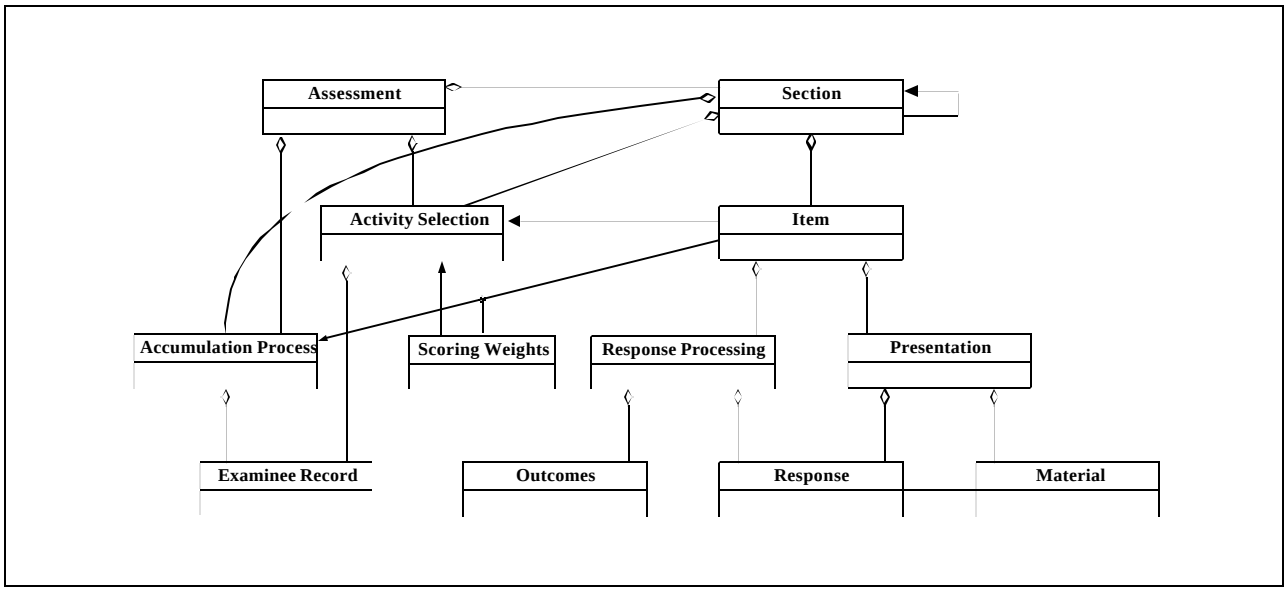

Figure 3.1 The IMS Question & Test Interoperability object data model.

The objects in this model and their key behaviours are:

- Assessment – the object that represents the *Assessment* data structure;
- Section – the object that represents the *Section* data structure;
- Item – the object that represents the *Item* data structure;
- Activity Selection – selection of the next activity determined by the progress and results obtained up to the moment of activity selection;
- Accumulation Process – the reconciliation of all the evaluation outputs to produce an overall Assessment/Section evaluation;
- Scoring Weights – the scoring weights that are to be assigned to the results output from the response processing;
- Response Processing – the processing and evaluation of the user responses;
- Presentation – the rendering of the content and the possible responses;
- Examinee Record – the set of collated results that are output from the complete process. This is a 'life-long' record in that it contains the historical progress of the individual;
- Outcomes – the set of outcomes that are to be evaluated by the response processing object. These determine the scoring metrics to be applied to the response evaluations;
- Response – the responses that are supplied by the user of the Items i.e. the input user selections;
- Material – the content that is to be displayed.

This structure shows the relationship between the three core data objects, namely Items, Sections and Assessments. The type of objects that can be exchanged are shown in Figure 3.2 (Figure 4.1 in the Q&TI Information Model Specification).

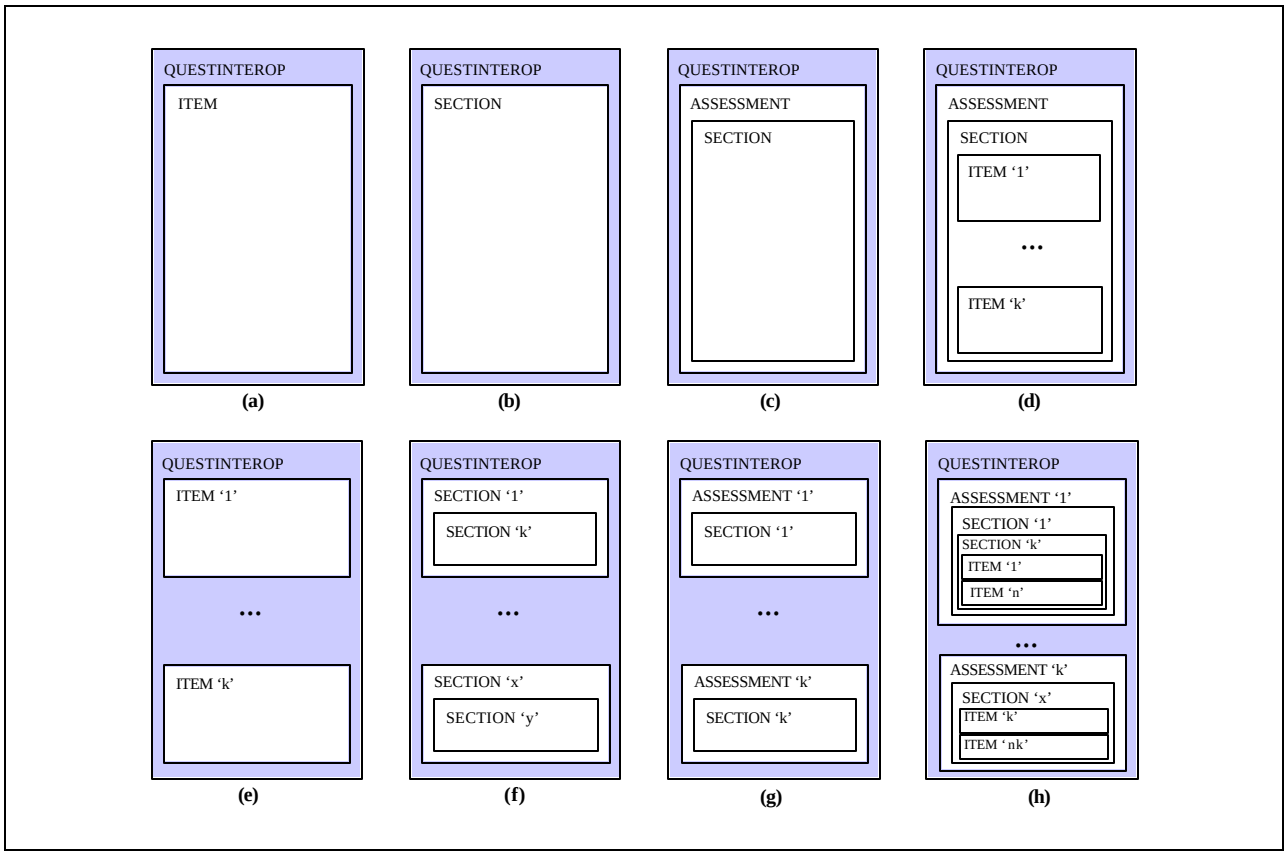

Figure 3.2 The principle Q&TI interchange data objects.

3.2 XML Schema Tree[1]

The generic XML schema tree is shown in Figure 3.3. This representation reflects the structure of an Item, Section and Assessment.

[1] The XML schema trees shown in this document were generated by the XML Authority V1.2 product from Extensibility Inc.

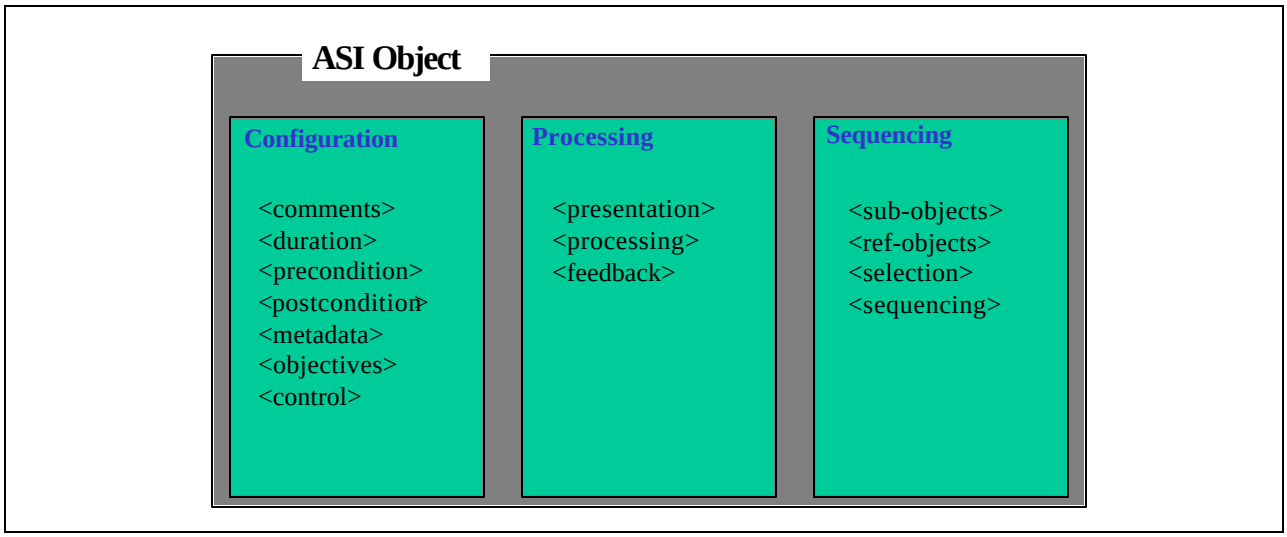

Figure 3.3 The generic structure of the XML schema tree.

This structure has three core components:

- Configuration - generation of the appropriate environment for the correct interpretation of the information contained within the object;
- Processing - the actual processing represented by the object e.g. the presentation of a question and the corresponding response processing and feedback;
- Sequencing - linkage to referenced objects and the selection and sequencing of the next object to be processed.

3.2.1 Item

The XML schema tree for the Item data structure is shown in Figure 3.4.

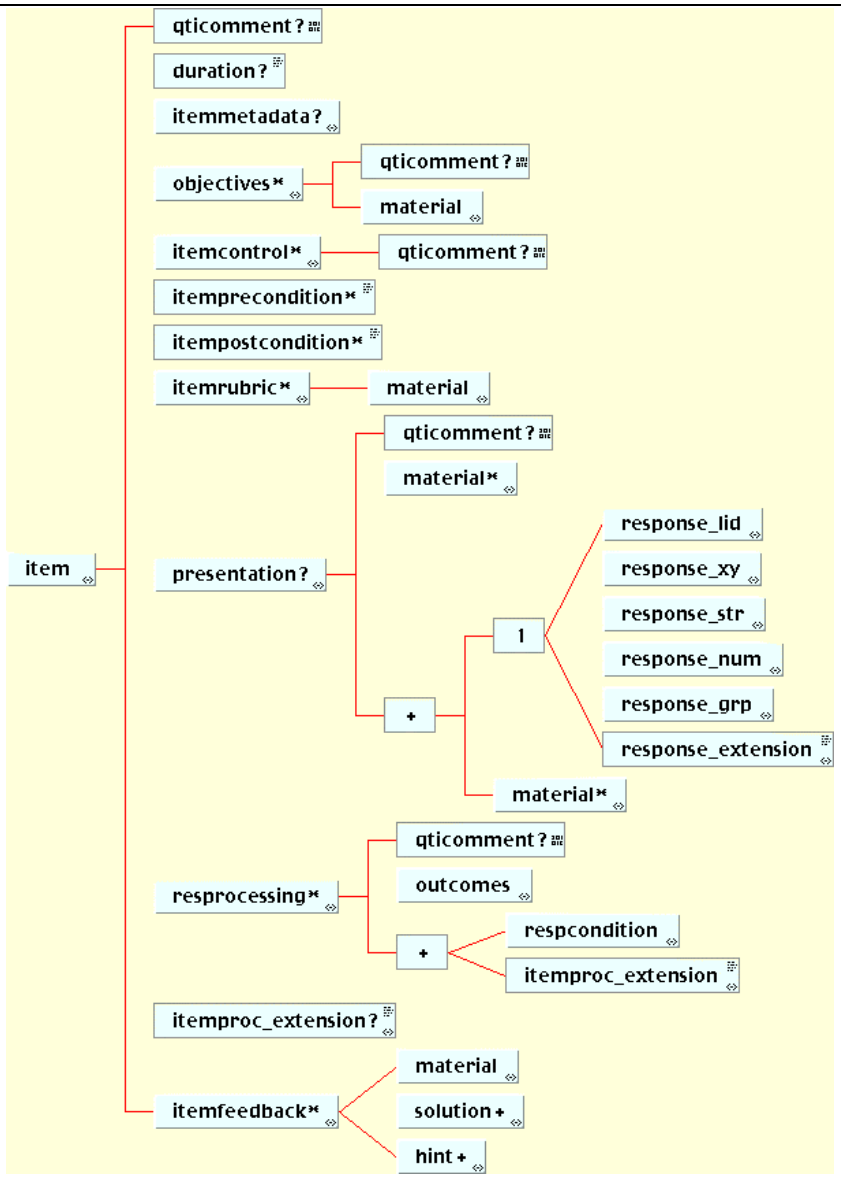

Figure 3.4 The Item element XML schema tree.

The corresponding XML schema trees for the <presentation> and <resprocessing> elements are shown in Figures 3.5 and 3.6 respectively.

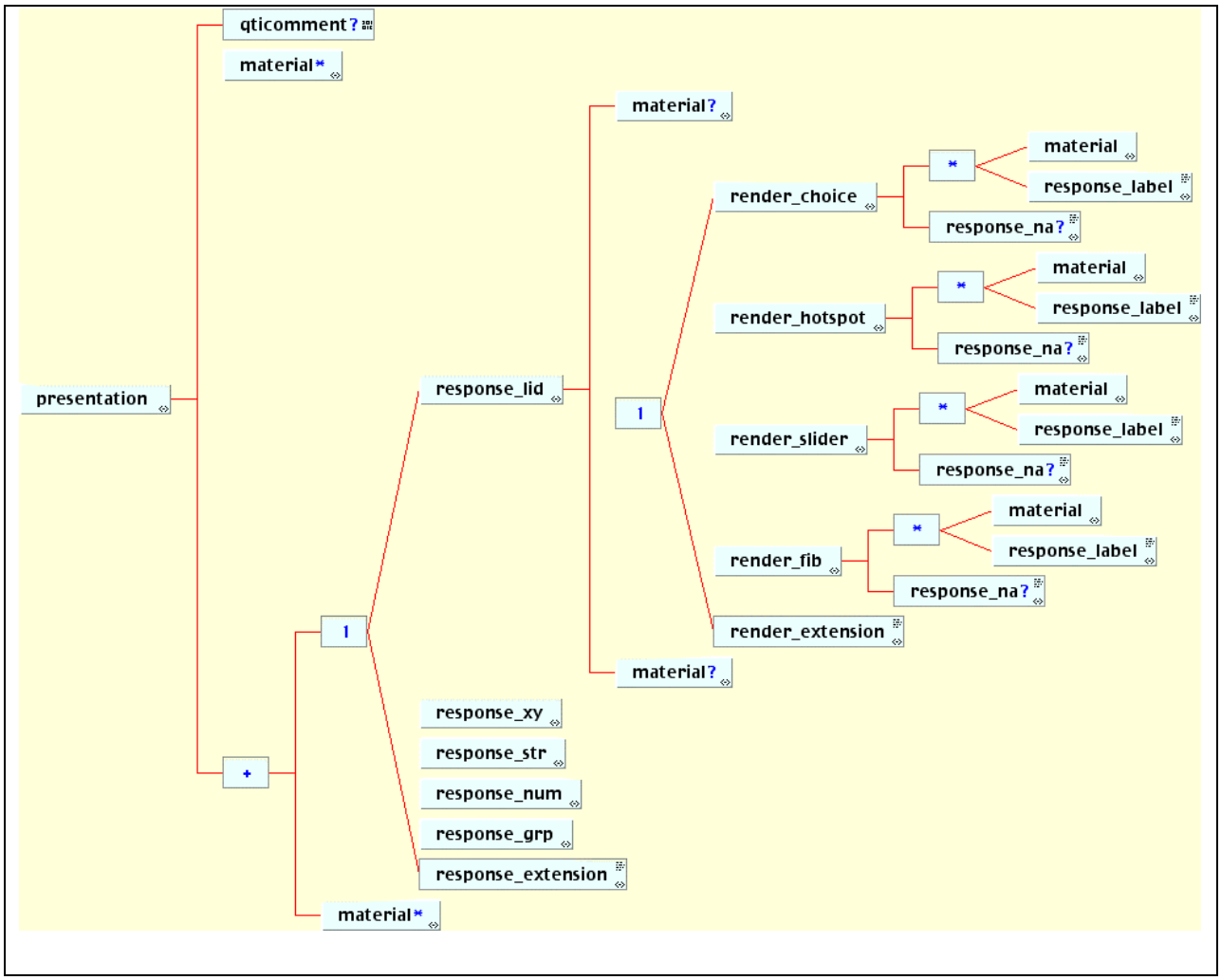

Figure 3.5 The Presentation element XML schema tree.

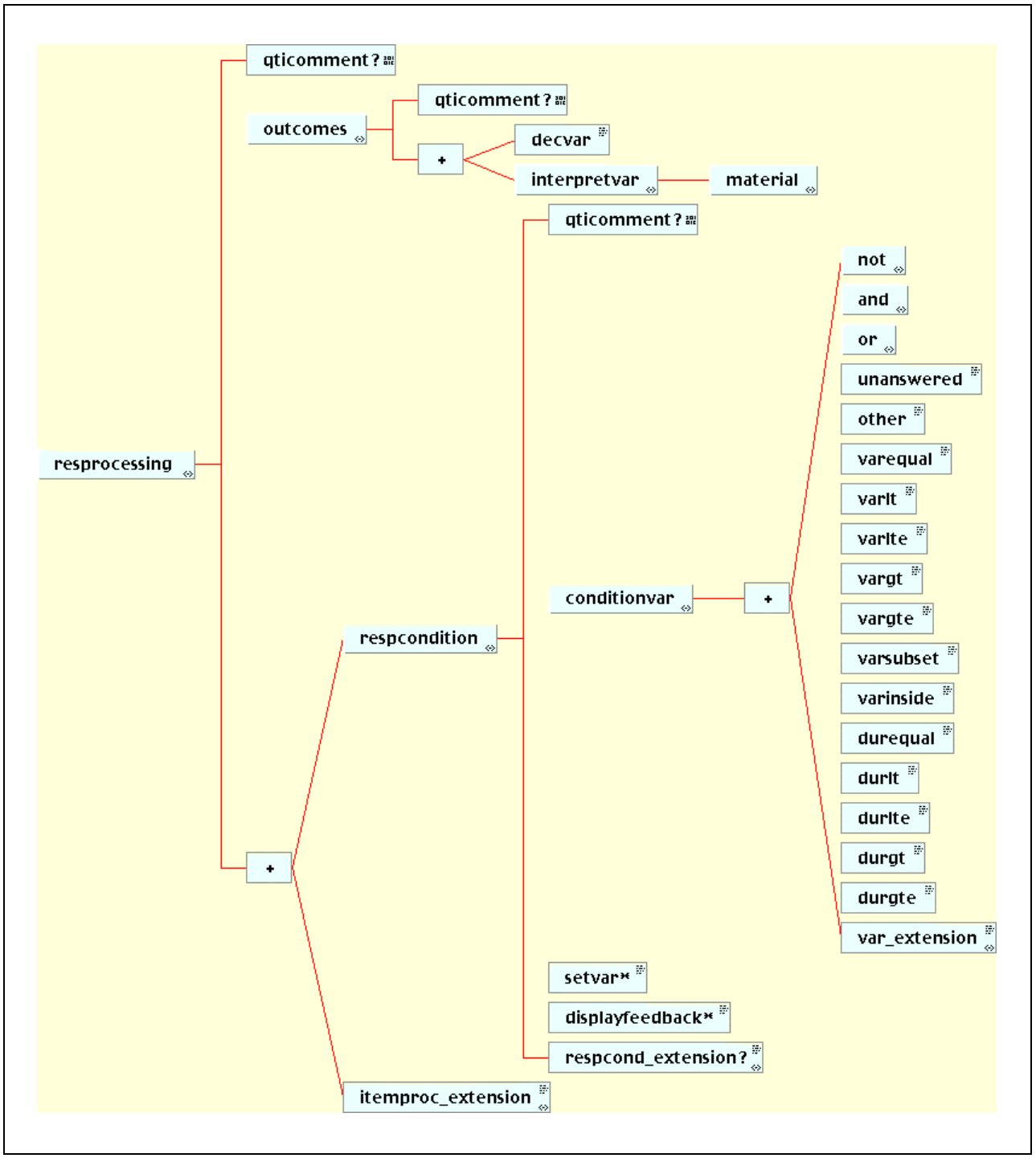

Figure 3.6 The Resprocessing element XML schema tree.

3.2.2 Section

The Section data structure XML schema tree is shown in Figure 3.7.

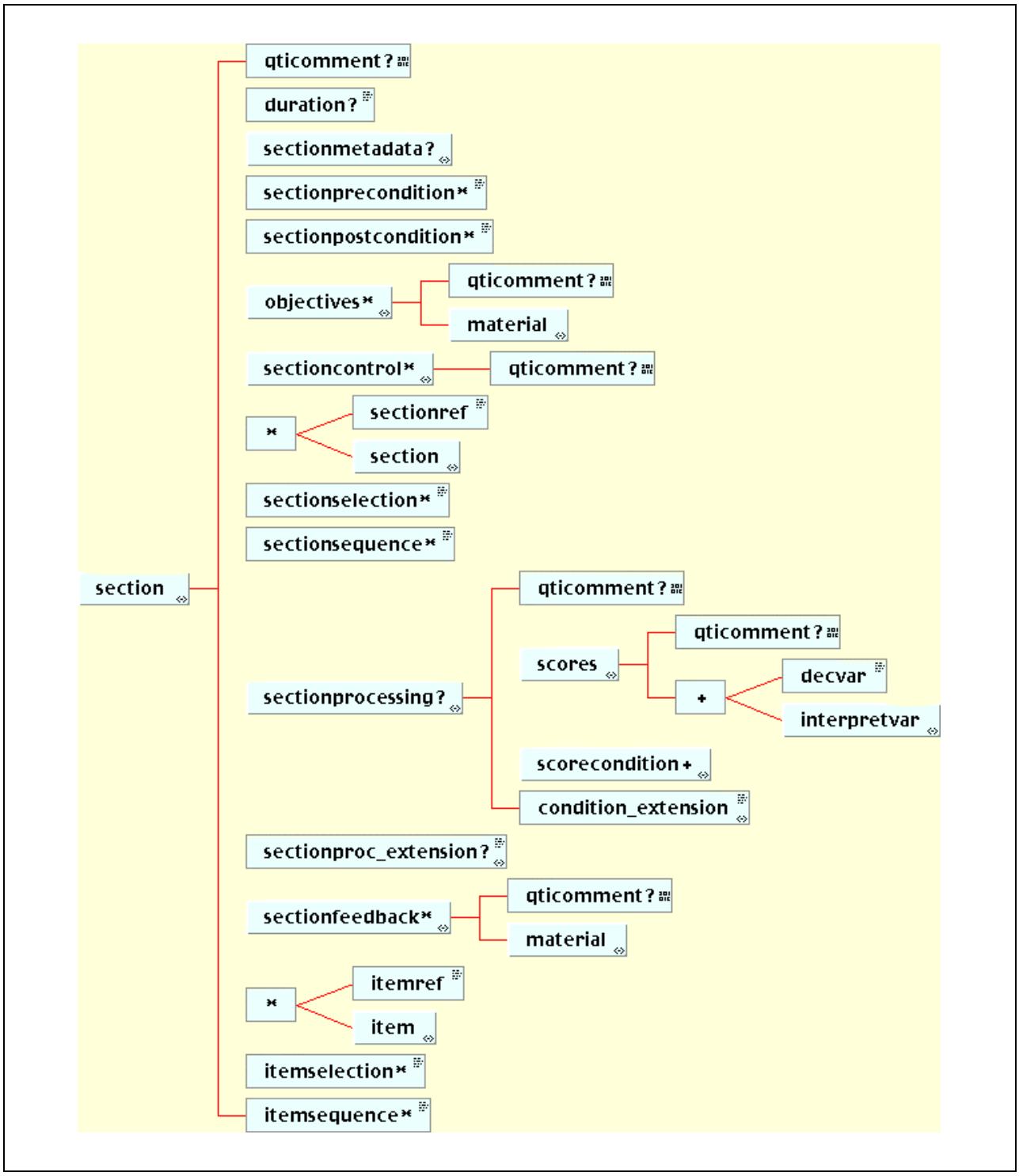

Figure 3.7 The Section XML schema tree.

3.2.3 Assessment

The Assessment data structure XML schema tree is shown in Figure 3.8.

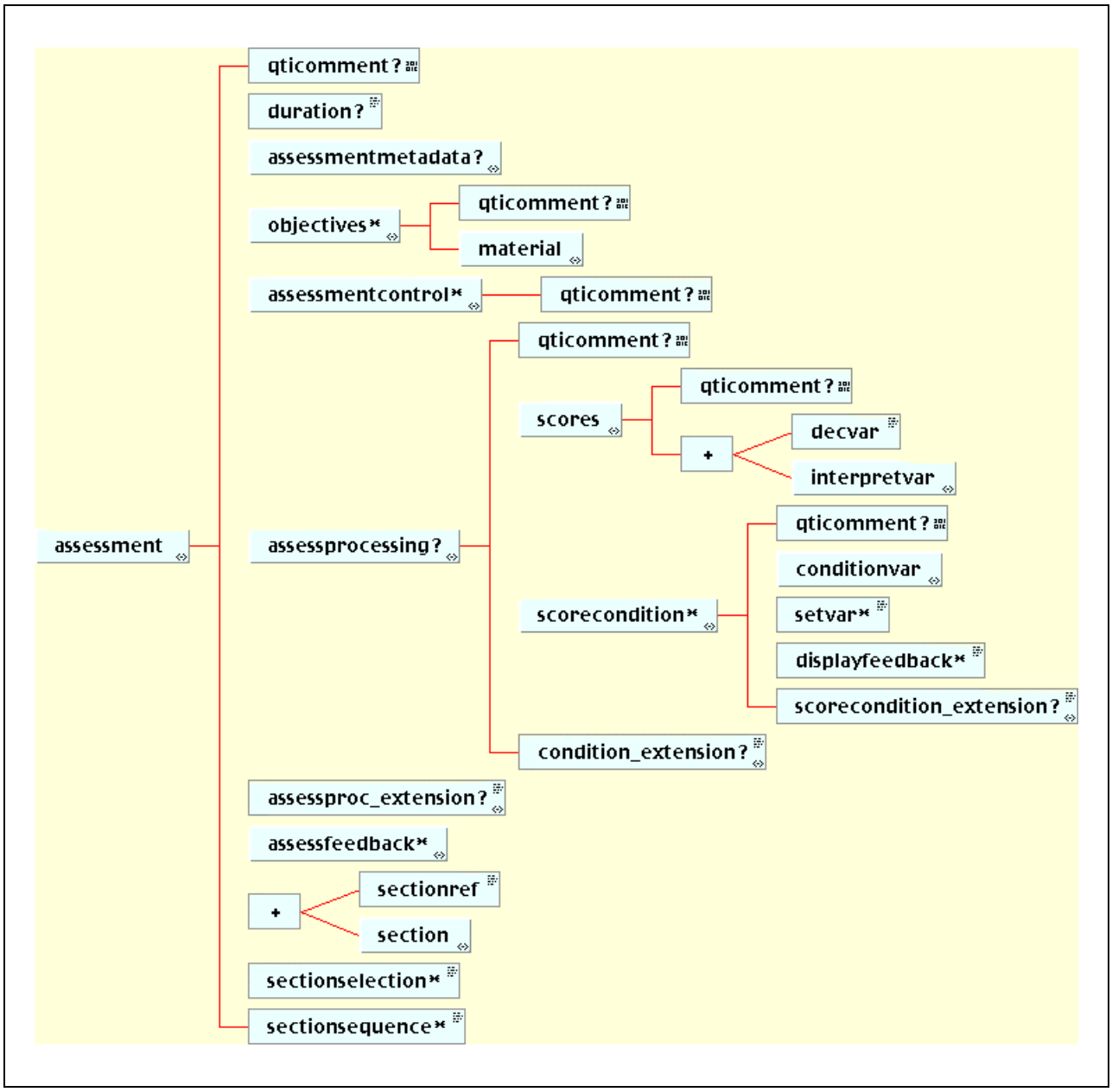

Figure 3.8 The Assessment XML schema tree.

3.2.4 Material Element

The Material element XML schema tree is shown in Figure 3.9.

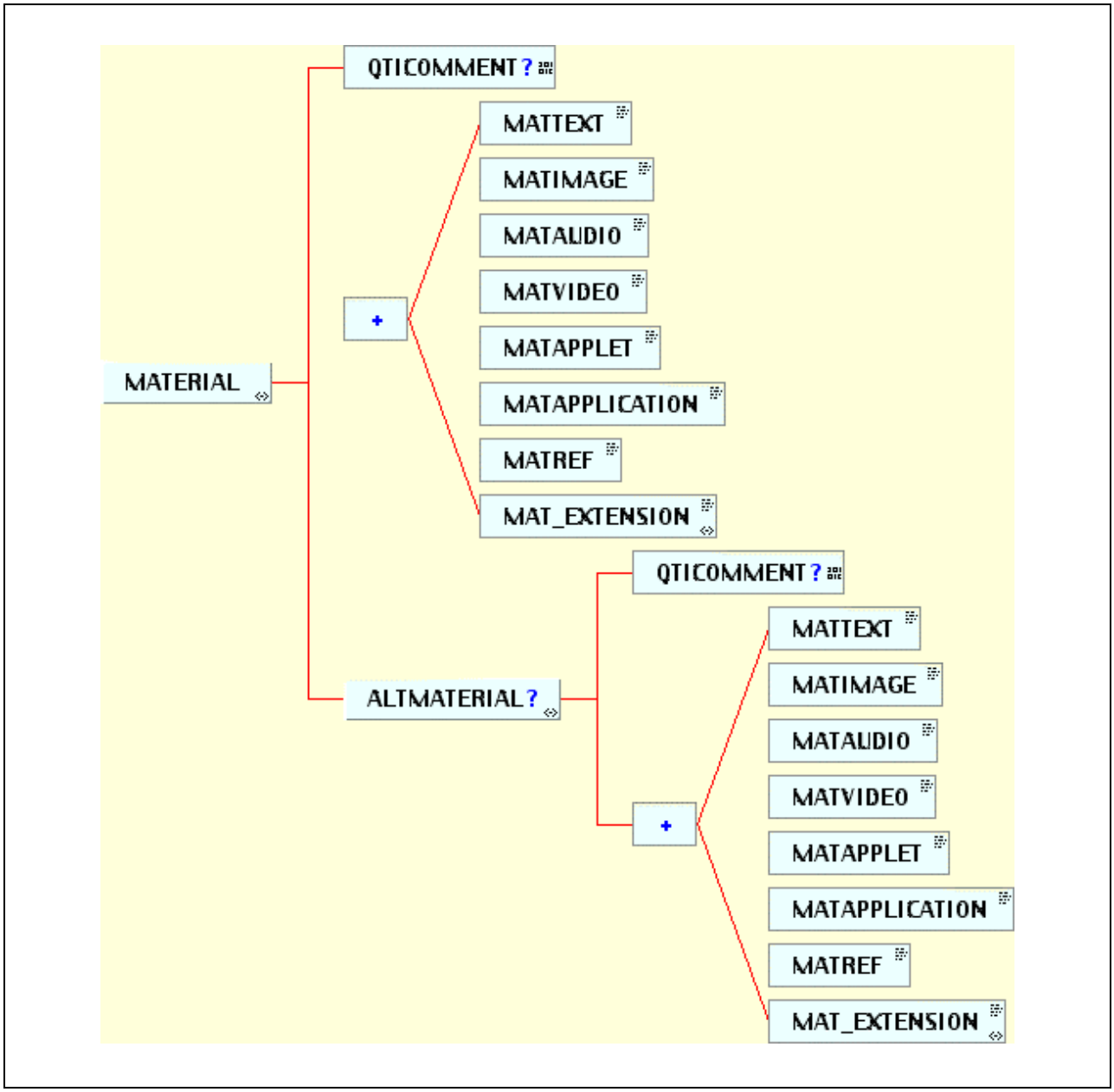

Figure 3.9 The Material element XML schema tree.

3.2.5 Meta-data

The meta-data for the three ASI structures are shown in Figures 10a, 10b and 10c.

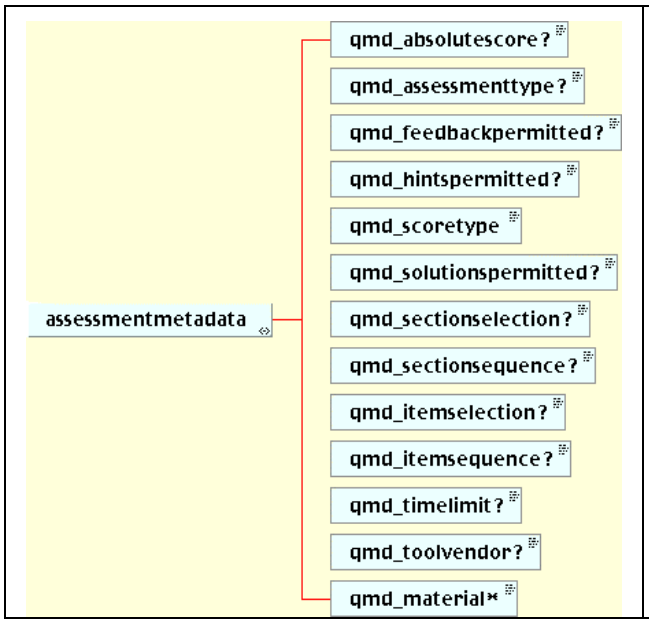

Figure 3.10a The Assessmentmetadata XML tree.

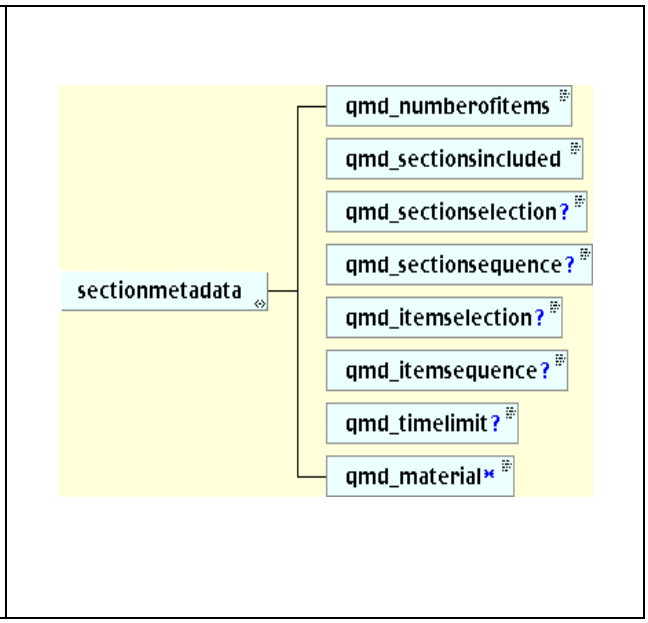

Figure 3.10b The Sectionmetadata XML tree.

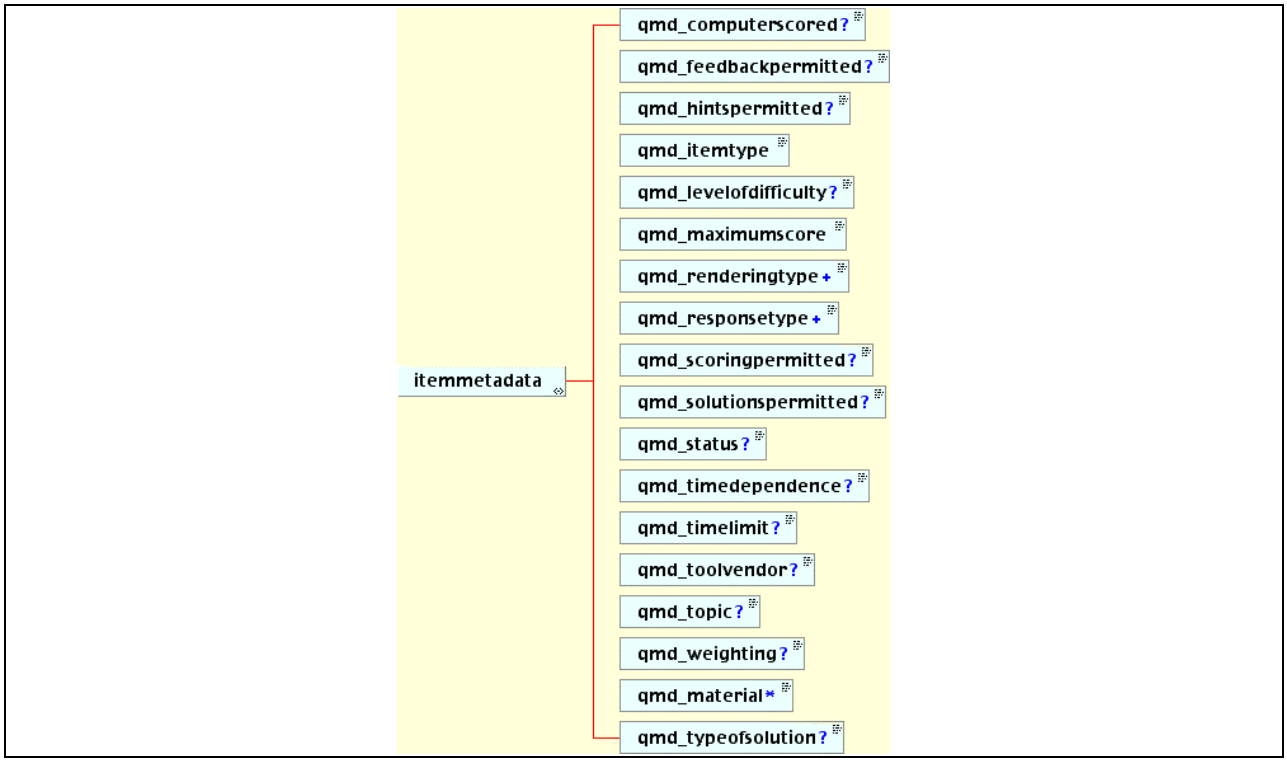

Figure 3.10c The Itemmetadata XML tree.

4. Example Basic Item Types

The examples of the basic Item types are lists under:

- Logical identifier;
- X-Y co-ordinate;
- String;
- Numeric;
- Group.

The LID examples are:

- Standard True/False (text-based options) – choice-based rendering;
- Standard Multiple Choice (text-based options) – choice-based rendering;
- Standard Multiple Choice (image-based options) – choice-based rendering;
- Standard Multiple Choice (audio-based options) – choice-based rendering;
- Standard Multiple Response (text-based options) – choice-based rendering;
- Multiple Choice with Single Image (image-based options) – IHS-based rendering;
- Multiple Response with Multiple Images (image-based options) – IHS-based rendering;
- Multiple Choice (slider-based options) – slider-based rendering;
- Standard Order Objects (text-based objects) – object-based rendering;
- Standard Order Objects (image-based objects) – object-based rendering;
- Connect-the-points (image-based) – IHS-based rendering.

The XY co-ordinate examples are:

- Standard Image Hot Spot (single image) – IHS-based rendering;
- Standard Drag-and-drop (multiple images) – IHS-based rendering;
- Connect-the-points (image-based) – IHS-based rendering.

The STR examples are:

- Standard Single Fill-in-Blank – FIB-based rendering;
- Standard Multiple Fill-in-Blank – FIB-based rendering;
- Standard Short Answer (text required) – FIB-based rendering.

The NUM examples are:

- Standard Integer Fill-in-Blank – FIB-based rendering;
- Standard Real number Fill-in-Blank – FIB-based rendering;
- Numerical entry with Slider – slider-based rendering.

4.1 Logical Identifier

4.1.1 Standard True/False (Text)

Figure 4.1 shows a typical True/False multiple choice question. The corresponding XML is listed after the figure. The user is expected to select either the 'True' or 'False' radio buttons.

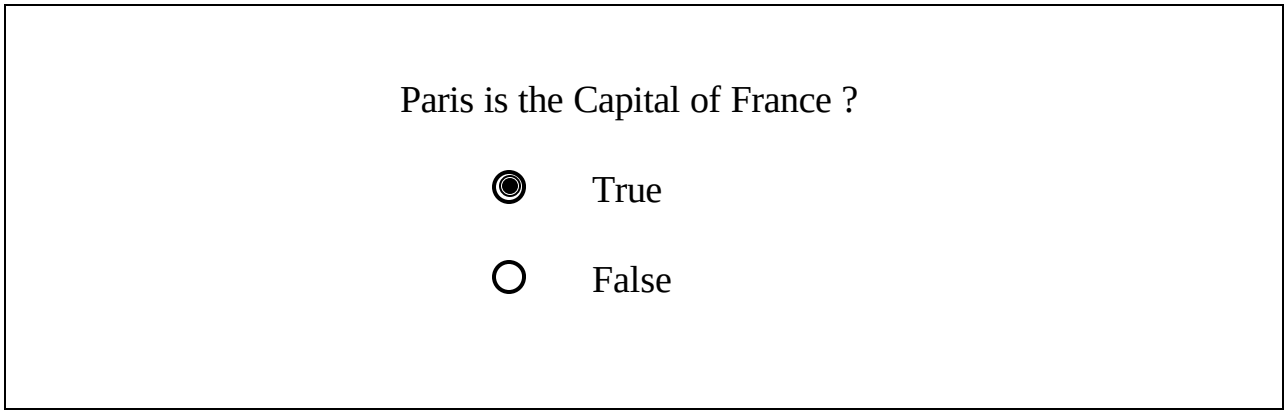

Figure 4.1 Standard true/false item.

Equivalent XML (without response processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a simple True/False multiple-choice example.
4               The rendering is a standard radio button style.
5               No response processing is incorporated.
6           </qticomment>
7           <item ident="IMS_V01_I_BasicExample001">
8           <presentation label="BasicExample001">
9               <material>
10                      <mattext>Paris is the Capital of France ?</mattext>
11              </material>
12              <response_lid ident="TF01" rcardinality=”Single” rtiming=”No”>
13                  <render_choice>
14                      <response_label ident="T">
15                          <material><mattext> True </mattext></material>
16                      </response_label>
17                      <response_label ident="F">
18                          <material><mattext> False </mattext></material>
19                      </response_label>
20                  </render_choice>
21              </response_lid>
22              </presentation>
23          </item>
24      </questestinterop>
```

Equivalent XML (with response processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a simple True/False multiple choice example.
4               The rendering is a standard radio button style.
5               Response processing is incorporated.
6           </qticomment>
7           <item ident="IMS_V01_I_BasicExample001">
8               <presentation label="BasicExample001">
9                   <material>
10                      <mattext>Paris is the Capital of France ?</mattext>
11                  </material>
```

```
12                <response_lid ident="TF01" rcardinality="Single" rtiming="No">
13                    <render_choice>
14                        <response_label ident="T">
15                            <material><mattext>True</mattext></material>
16                        </response_label>
17                        <response_label ident="F">
18                            <material><mattext>False</mattext></material>
19                        </response_label>
20                    </render_choice>
21                </response_lid>
22            </presentation>
23            <resprocessing>
24                <outcomes><decvar/></outcomes>
25                <respcondition title="Correct">
26                    <conditionvar>
27                        <varequal respident="TF01">T</varequal>
28                    </conditionvar>
29                    <setvar action="Set" >1</setvar>
30                    <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
31                </respcondition>
32            </resprocessing>
33            <itemfeedback ident="Correct" view="Candidate">
34                <material><mattext>Yes, you are right.</mattext></material>
35            </itemfeedback>
36        </item>
37    </questestinterop>
```

This XML code is available in the file: ‘IMS_QTIv1/Valid/Basic/trfl_i_001/trfl_i_001.xml’. The equivalent example with the response processing incorporated is available in the file: ‘IMS_QTIv1/Valid/Basic/trfl_ir_001/trfl_ir_001.xml’.

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.

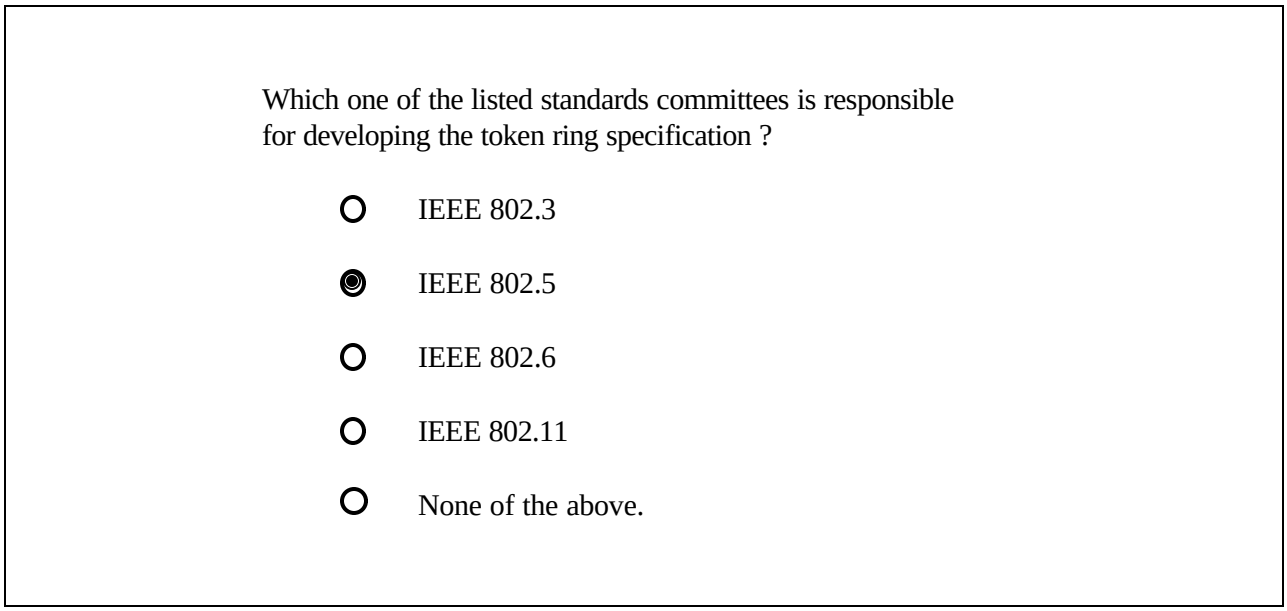

Figure 4.2 Standard multiple choice (text) item.

Equivalent XML (without response processing):

```
<questestinterop>
    <qticomment>
        This is a simple multiple-choice example.
        The rendering is a standard radio button style.
        No response processing is incorporated.
    </qticomment>
    <item title="Standard MC Item" Ident="IMS_V01_I_BasicExample002">
        <presentation label="BasicExample002">
        <material>
            <mattext>Which one of the listed standards committees is responsible for
                    developing the token ring specification ?
            <mattext>
        </material>
            <response_lid ident="MC01" rcardinality="Single" rtiming="No">
                <render_choice shuffle="Yes">
                    <response_label ident="A">
                        <material><mattext>IEEE 802.3</mattext></material>
                    </response_label>
                    <response_label ident="B">
                        <material><mattext>IEEE 802.5</mattext></material>
                    </response_label>
                    <response_label ident="C">
                        <material><mattext>IEEE 802.6</mattext></material>
                    </response_label>
                    <response_label ident="D">
                        <material><mattext>IEEE 802.11</mattext></material>
                    </response_label>
                    <response_label ident="E" rshuffle="No">
                        <material><mattext>None of the above.</mattext></material>
                    </response_label>
                </render_choice>
            </response_lid>
        </presentation>
    </item>
</questestinterop>
```

Equivalent XML (with response processing):

```
<questestinterop>
    <qticomment>
        This is a simple multiple choice example.
        The rendering is a standard radio button style.
        Response processing is incorporated.
    </qticomment>
    <item title="Standard Multiple Choice Item" ident="IMS_V01_I_BasicExample002b">
        <presentation label="BasicExample002a">
            <material>
                <mattext>
                    Which one of the listed standards committees is responsible
                    for developing the token ring specification ?
                </mattext>
            </material>
            <response_lid ident="MCb_01" rcardinality="Single" rtiming="No">
                <render_choice shuffle="Yes">
                    <response_label ident="A">
                        <material><mattext>IEEE 802.3</mattext></material>
                    </response_label>
                    <response_label ident="B">
                        <material><mattext>IEEE 802.5</mattext></material>
                    </response_label>
                    <response_label ident="C">
                        <material><mattext>IEEE 802.6</mattext></material>
                    </response_label>
                    <response_label ident="D">
                        <material><mattext>IEEE 802.11</mattext></material>
                    </response_label>
                    <response_label ident="E" rshuffle="No">
                        <material><mattext>None of the above.</mattext></material>
                    </response_label>
                </render_choice>
            </response_lid>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar  vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition title="Correct">
                <conditionvar>
                    <varequal respident="MCb_01">A</varequal>
                </conditionvar>
                <setvar action="Set" >1</setvar>
                <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
            </respcondition>
        </resprocessing>
        <itemfeedback ident="Correct" view="Candidate">
            <material><mattext>Yes, you are right.</mattext></material>
        </itemfeedback>
    </item>
</questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_001/mchc_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_001/mchc_ir_001.xml'.

4.1.3 Standard Multiple Choice (Images)

Figure 4.3 shows a typical image-based multiple-choice question. The corresponding XML is listed after the figure. The user is required to select one of the options by clicking on the appropriate radio button.

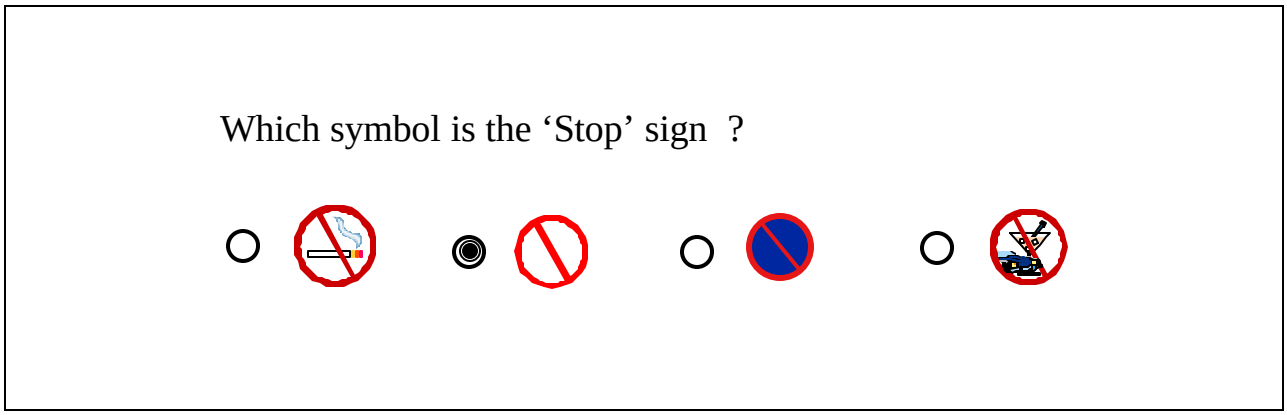

Figure 4.3 Standard multiple choice (images) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a multiple-choice example with image content.
4               The rendering is a standard radio button style.
5               No response processing is incorporated.
6           </qticomment>
7           <item title="Standard MC with Images Item" ident="IMS_V01_I_BasicExample003">
8               <presentation label="BasicExample003">
9                   <material>
10                      <mattext>Which symbol is the 'Stop' sign ?<mattext>
11                  </material>
12                  <response_lid ident="MC02" rcardinality="Single" rtiming="No">
13                      <render_choice shuffle="Yes">
14                          <response_label ident="A">
15                              <material>
16                                  <matimage imagtype="image/gif" uri="image1.gif"></matimage>
17                              </material>
18                          </response_label>
19                          <response_label ident="B">
20                              <material>
21                                  <matimage imagtype="image/gif" uri="image2.gif"></matimage>
22                              </material>
23                          </response_label>
24                          <response_label ident="C">
25                              <material>
26                                  <matimage imagtype="image/gif" uri="image3.gif"></matimage>
27                              </material>
28                          </response_label>
29                          <response_label ident="D">
30                              <material>
31                                  <matimage imagtype="image/gif" uri="image4.gif"></matimage>
32                              </material>
33                          </response_label>
34                      </render_choice>
35                  </response_lid>
36              </presentation>
37          </item>
38      </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_002/mchc_i_002.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_002/mchc_ir_002.xml'.

4.1.4 Standard Multiple Choice (Audio)

Figure 4.4 shows a typical audio-based multiple-choice question. The corresponding XML is listed after the figure. The user is required to make their choice by selecting the appropriate radio button but the audio can only be activated by clicking on each sound source symbol.

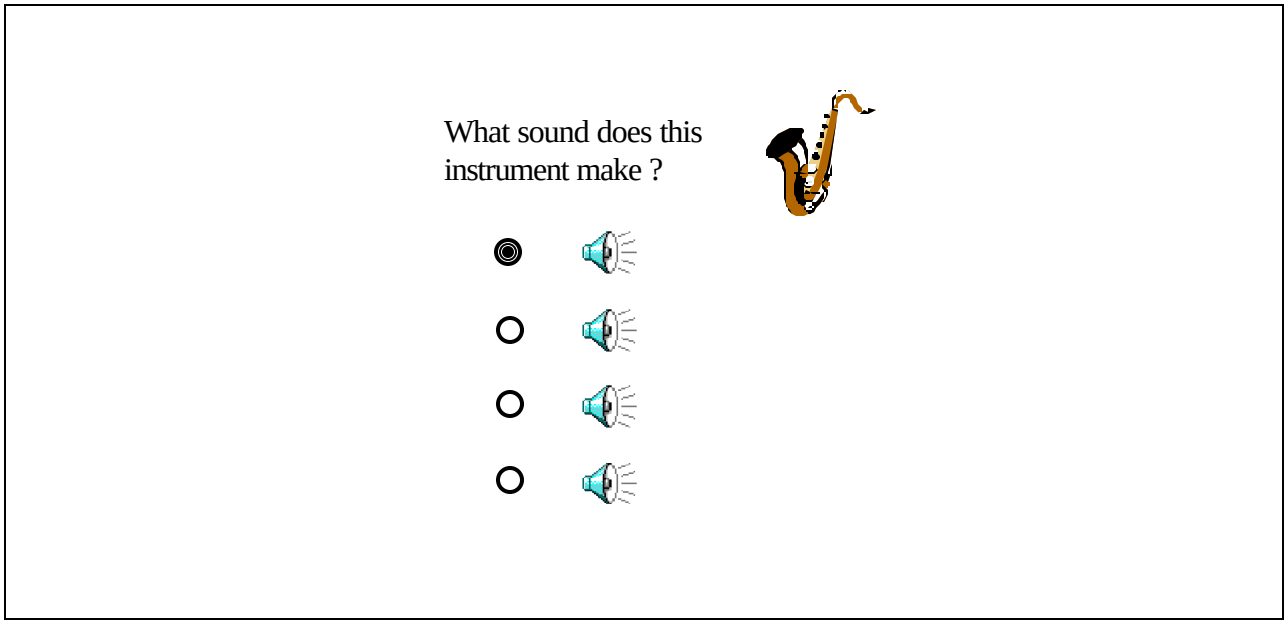

Figure 4.4 Standard multiple choice (audio) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a multiple-choice example with audio content.
4               The rendering is a standard radio button style.
5               No response processing is incorporated.
6           </qticomment>
7           <item title="Standard MC with Audio Item" ident="IMS_V01_I_BasicExample004">
8               <presentation label="BasicExample004">
9                   <material>
10                      <mattext>What sound does this instrument make ?</mattext>
11                      <matimage imagtype="image/gif" uri="imageinstrument.gif"></matimage>
12                  </material>
13                  <response_lid ident="MC03" rcardinality="Single" rtiming="No">
14                      <render_choice>
15                          <response_label ident="A">
16                              <material>
17                                  <mataudio audiotype="audio/wav" uri="sound1.wav"></mataudio>
18                              </material>
19                          </response_label>
20                          <response_label ident ="B">
21                              <material>
22                                  <mataudio audiotype="audio/wav" uri="sound2.wav"></mataudio>
23                              </material>
24                          </response_label>
25                          <response_label ident="C">
26                              <material>
27                                  <mataudio audiotype="audio/wav" uri="sound3.wav"></mataudio>
28                              </material>
29                          </response_label>
30                          <response_label ident="D">
31                              <material>
32                                  <mataudio audiotype="audio/wav" uri="sound4.wav"></mataudio>
33                              </material>
34                          </response_label>
```

```
35                    </render_choice>
36                </response_lid>
37            </presentation>
38        </item>
39    </questestinterop>
```

Note: The icon used to denote the sound file is dependent on the rendering system.

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_003/mchc_i_003.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_003/mchc_ir_003.xml'.

4.1.5 Standard Multiple Response (Text)

Figure 4.5 shows a typical multiple response question. The corresponding XML is listed after the figure. The user is expected to click on each of the correct solutions using the appropriate check buttons.

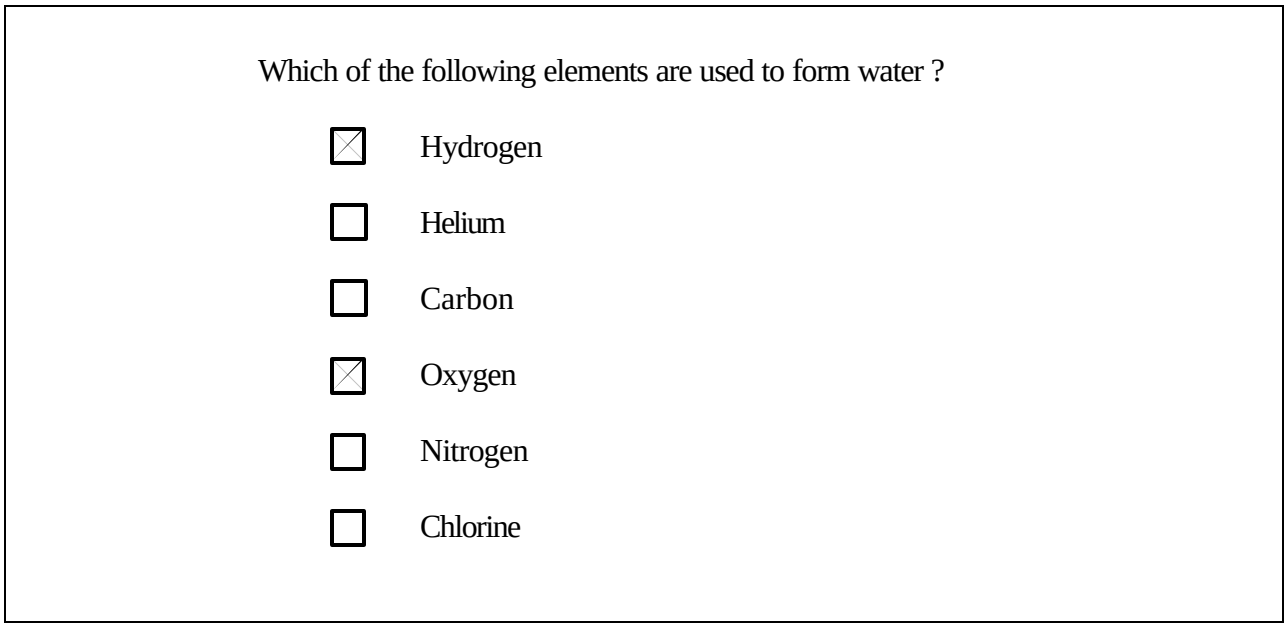

Figure 4.5 Standard multiple response item.

```
<questestinterop>
    <qticomment>
        This is a multiple-response example.
        The rendering is a standard check button style.
        No response processing is incorporated.
    </qticomment>
    <item title="Standard Multiple Response Item" ident="IMS_V01_I_BasicExample005">
        <presentation label="RS05">
            <material>
                <mattext>Which of the following elements are used to form water ?
                </mattext>
            </material>
            <response_lid ident="MR01" rcardinality="Multiple" rtiming="No">
                <render_choice shuffle="Yes" minnumber="1" maxnumber="4">
                    <response_label ident="A">
                        <material>
                            <mattext>Hydrogen</mattext>
                        </material>
                    </response_label>
                    <response_label ident="B">
                        <material>
                            <mattext>Helium</mattext>
                        </material>
                    </response_label>
                    <response_label ident="C">
                        <material>
                            <mattext>Carbon</mattext>
                        </material>
                    </response_label>
                    <response_label ident="D">
                        <material>
                            <mattext>Oxygen</mattext>
                        </material>
                    </response_label>
                    <response_label ident="E">
                        <material>
```

```
37                              <mattext>Nitrogen</mattext>
38                          </material>
39                      </response_label>
40                      <response_label ident="F">
41                          <material>
42                              <mattext>Chlorine</mattext>
43                          </material>
44                      </response_label>
45                  </render_choice>
46              </response_lid>
47          </presentation>
48      </item>
49  </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mrsp_i_001/mrsp_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mrsp_ir_001/mrsp_ir_001.xml'.

4.1.6 Multiple Choice with Image Hot Spot Rendering

Figure 4.6 shows a typical multiple response question using hotspot rendering. The corresponding XML is listed after the figure. The user is expect to click on the appropriate radio button.

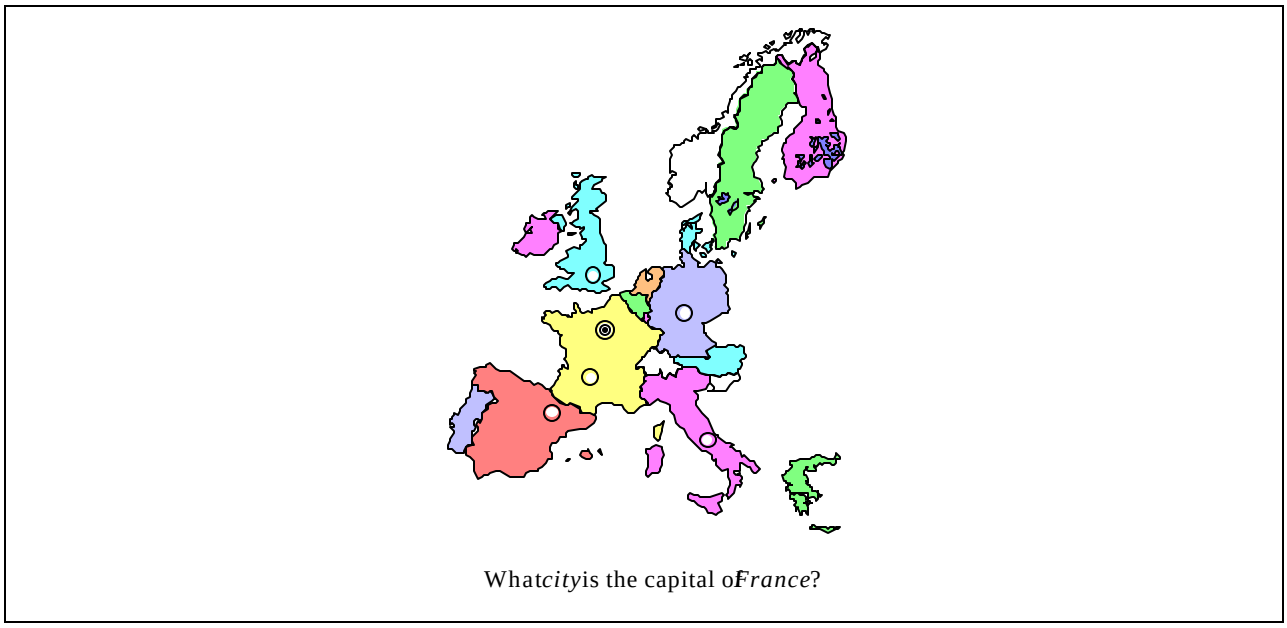

Figure 4.6 Multiple choice with hot spot rendering item.

```
<questestinterop>
    <qticomment>
        This is a multiple-choice with image hot spot. The rendering uses image-based
        hotspots. No response processing is incorporated.
    </qticomment>
    <item title="Multiple Choice with Image Hotspot Rendering Item"
        ident="IMS_V01_I_BasicExample006">
        <presentation label="BasicExample006">
            <response_lid ident="MC04" rcardinality="Single" rtiming="No">
                <render_hotspot>
                    <material>
                        <matimage imagtype="image/gif" uri="mchotspot1.gif" x0="0"
                            width="300" y0="512" height="400"></matimage>
                        <mattext>What city is the capital of France ?</mattext>
                    </material>
                    <response_label ident="A" rarea="Ellipse"> 100, 100, 2, 2
                    </response_label>
                    <response_label ident="B" rarea="Ellipse"> 150, 150, 2, 2
                    </response_label>
                    <response_label ident="C" rarea="Ellipse"> 180, 200, 2, 2
                    </response_label>
                    <response_label ident="D" rarea="Ellipse"> 280, 230, 2, 2
                    </response_label>
                    <response_label ident="E" rarea="Ellipse"> 30, 80, 2, 2
                    </response_label>
                </render_hotspot>
            </response_lid>
        </presentation>
    </item>
</questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_004/mchc_i_004.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_004/mchc_ir_004.xml'.

4.1.7 Multiple Response with Multiple Image Hot Spot Rendering

Figure 4.7 shows a typical multiple response question using multiple image hotspot rendering. The corresponding XML is listed after the figure. The user is expected to use the mouse to click on the four wheels.

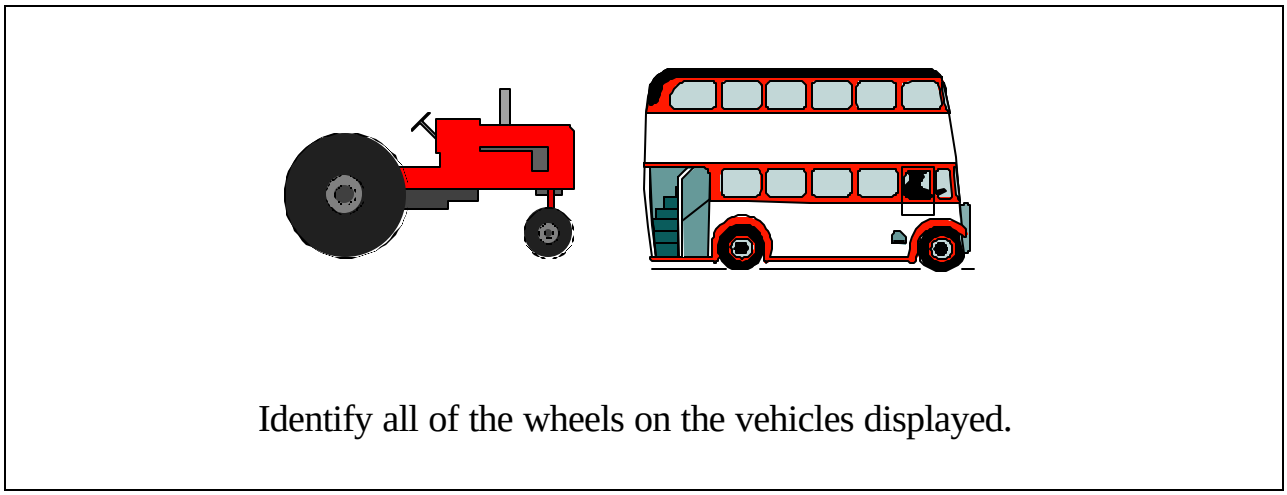

Figure 4.7 Multiple response with multiple image hot spot rendering item.

```
1       <questestinterop>
2           <qticomment>
3               This is a multiple-response with image hot spot rendering.
4               The rendering is based on image hotspots.
5               No response processing is incorporated.
6           </qticomment>
7           <item title="Multiple Response with Image Hotspot Rendering Item"
8                   ident="BasicExample007">
9               <presentation label="BasicExample007">
10                  <response_lid ident="MR02" rcardinality="Multiple" rtiming="No">
11                      <render_hotspot minnumber="2" maxnumber="4">
12                          <material>
13                              <matimage imagtype="image/gif" uri="tractor.gif" x0="100"
14                                        width="100" y0="100" height="100"></matimage>
15                              <matimage imagtype="image/gif" uri="bus.gif"
16                                        x0="300" width="100" y0="0" height="100"></matimage>
17                              <mattext>Identify all of the wheels on the vehicles displayed.
18                              </mattext>
19                          </material>
20                          <response_label ident="A" rarea=Ellipse> 110, 10, 20, 20
21                          </response_label>
22                          <response_label ident="B" rarea=Ellipse> 190, 110, 5, 5
23                          </response_label>
24                          <response_label ident="C" rarea=Ellipse> 220, 110, 5, 5
25                          </response_label>
26                          <response_label ident="D" rarea=Ellipse> 380, 110, 5, 5
27                          </response_label>
28                      </render_hotspot>
29                  </response_lid>
30              </presentation>
31          </item>
32      </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_005/mchc_i_005.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_005/mchc_ir_005.xml'.

4.1.8 Multiple Choice with Slider Rendering

Figure 4.8 shows a typical multiple response question using hotspot rendering. The corresponding XML is listed after the figure. The user is expected to move the slider to point towards the correct integer value.

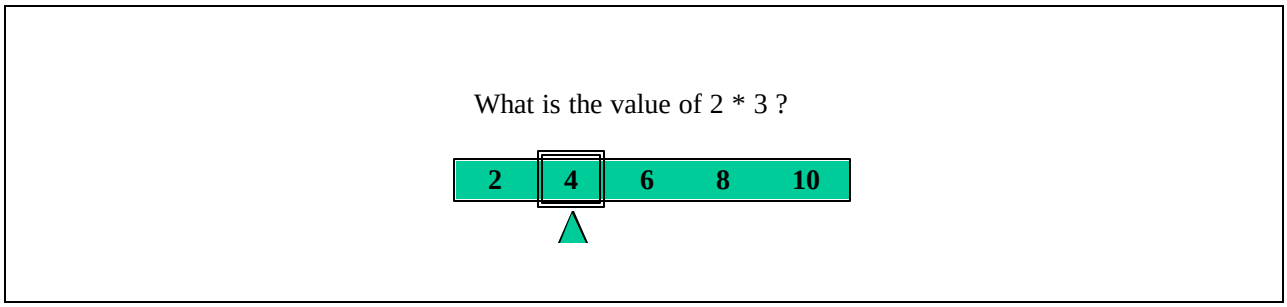

Figure 4.8 Multiple choice with slider rendering item.

Equivalent XML (withoutresponse processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a multiple-choice with slider rendering.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Multiple-choice and Slider rendering Item"
7               ident="IMS_V01_I_BasicExample008">
8               <presentation label=" BasicExample008">
9                   <material>
10                      <mattext>What is the value of 2 * 3 ?</mattext>
11                  </material>
12                  <response_lid ident="MC05" rcardinality="Single" rtiming="No">
13                      <render_slider lowerbound="2" upperbound="10" step="2" startval="4"
14                          steplabel="Yes">
15                          <response_label ident="a" rrange="absolute"> 2
16                          </response_label>
17                          <response_label ident="b" rrange="Absolute"> 4
18                          </response_label>
19                          <response_label ident="c" rrange="Absolute"> 6
20                          </response_label>
21                          <response_label ident="d" rrange="Absolute"> 8
22                          </response_label>
23                          <response_label ident="e" rrange="Absolute"> 10
24                          </response_label>
25                      </render_slider>
26                  </response_lid>
27              </presentation>
28          </item>
29      </questestinterop>
```

Equivalent XML (with response processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a multiple-choice with slider rendering.
4               Response processing is included.
5           </qticomment>
6           <item title="Multiple Choice with Slider rendering Item"
7                   ident="IMS_V01_I_BasicExample008b">
8               <presentation label="BasicExample008b">
9                   <material>
10                      <mattext>
11                          What is the value of 2 * 3 ?
```

```
12                  </mattext>
13              </material>
14              <response_lid ident="MC05" rcardinality="Single" rtiming="No">
15                  <render_slider lowerbound="2" upperbound="10" step="2" startval="4"
16                         steplabel="Yes">
17                      <response_label ident="A" rrange="Exact">2</response_label>
18                      <response_label ident="B" rrange="Exact">4</response_label>
19                      <response_label ident="C" rrange="Exact">6</response_label>
20                      <response_label ident="D" rrange="Exact">8</response_label>
21                      <response_label ident="E" rrange="Exact">10</response_label>
22                  </render_slider>
23              </response_lid>
24          </presentation>
25          <resprocessing>
26              <outcomes>
27                  <decvar varname="SLIDECHOICE" vartype="Integer" defaultval="0"/>
28              </outcomes>
29              <respcondition>
30                  <qticomment>Scoring for the correct answer. </qticomment>
31                  <conditionvar>
32                      <varequal respident="MC05">C</varequal>
33                  </conditionvar>
34                  <setvar action="Add" varname="SLIDECHOICE"> 5 </setvar>
35                  <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
36              </respcondition>
37              <respcondition>
38                  <qticomment>Detecting the wrong answer.</qticomment>
39                  <conditionvar>
40                      <or>
41                          <varequal respident="MC05"> A </varequal>
42                          <varequal respident="MC05"> B </varequal>
43                          <varequal respident="MC05"> D </varequal>
44                          <varequal respident="MC05"> E </varequal>
45                      </or>
46                  </conditionvar>
47                  <displayfeedback feedbacktype="Response" linkrefid="Incorrect"/>
48              </respcondition>
49          </resprocessing>
50          <itemfeedback ident="Correct" view="Candidate">
51              <material><mattext>Correct.</mattext></material>
52          </itemfeedback>
53          <itemfeedback ident="Incorrect" view="Candidate">
54              <material><mattext>The correct answer is 6.</mattext></material>
55          </itemfeedback>
56          <itemfeedback ident="Incorrect" view="Tutor">
57              <material>
58                  <mattext>The student chose the wrong answer.</mattext>
59              </material>
60          </itemfeedback>
61      </item>
62  </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_i_006/mchc_i_006.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/mchc_ir_006/mchc_ir_006.xml'.

4.1.9 Standard Order Objects (Text)

Figure 4.9 shows a typical order text -based objects. The corresponding XML is listed after the figure. The user is expected to click on each text object and to place them in the correct order.

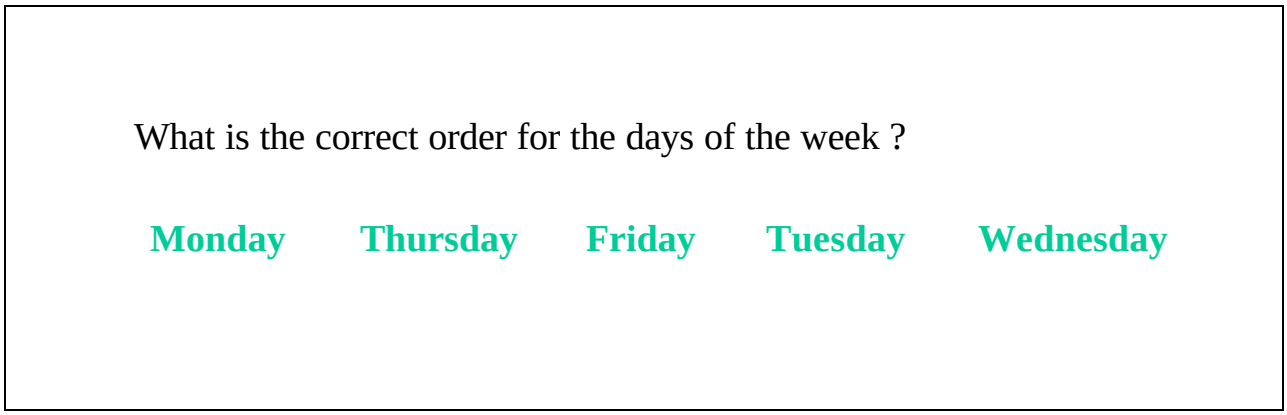

Figure 4.9 Standard order objects (text) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard ordering of a list of words.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard Object Ordering of text Item" ident="IMS_V01_I_BasicExample009">
7               <presentation label="BasicExample009">
8                   <material>
9                       <mattext>What is the correct order for the days of the week ?</mattext>
10                  </material>
11                  <response_lid ident="OB01" rcardinality="Ordered" rtiming="No">
12                      <render_extension>
13                          <ims_render_object shuffle="Yes" orientation="Row">
14                              <response_label ident="A">
15                                  <material><mattext>Monday</mattext></material>
16                              </response_label>
17                              <response_label ident="B">
18                                  <material><mattext>Thursday</mattext></material>
19                              </response_label>
20                              <response_label ident="C">
21                                  <material><mattext>Friday</mattext></material>
22                              </response_label>
23                              <response_label ident="D">
24                                  <material><mattext>Tuesday</mattext></material>
25                              </response_label>
26                              <response_label ident="E">
27                                  <material><mattext>Wednesday</mattext></material>
28                              </response_label>
29                          </ims_render_object>
30                      </render_extension>
31                  </response_lid>
32              </presentation>
33          </item>
34      </questestinterop>
```

Note: This example makes use of an extension i.e. *ims_render_object*. In a later version of the specification this extension may be adopted as part of the core specification.

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/oobj_i_001/oobj_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/oobj_ir_001/oobj_ir_001.xml'.

4.1.10 Standard Order Objects (Image)

Figure 4.10 shows a typical order image-based objects. The corresponding XML is listed after the figure. The user is expected to move each of the objects by using the mouse and mo ving them around the screen.

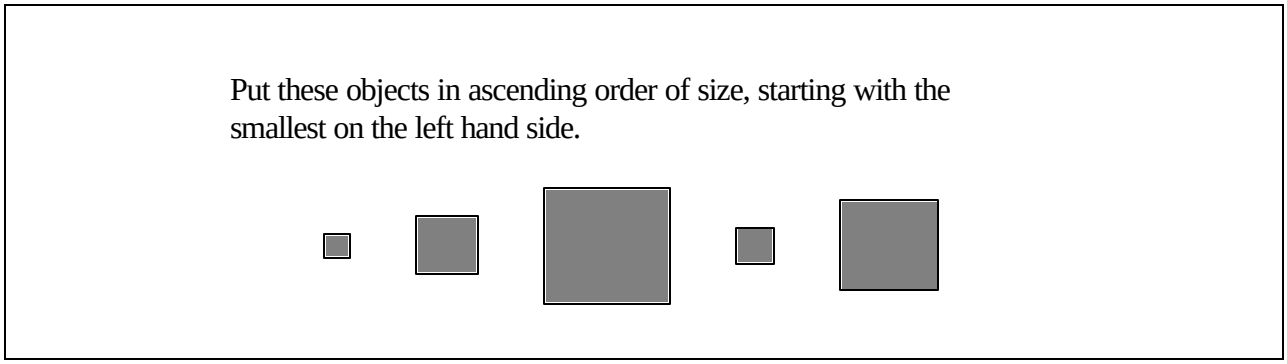

Figure 4.10 Standard order objects (image) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard ordering of a group of images.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard Object Ordering of images Item"
7                   ident="IMS_V01_I_BasicExample010">
8               <presentation label="BasicExample010">
9                   <response_lid ident="OB02" rcardinality="Ordered" rtiming="No">
10                      <material>
11                          <mattext>Put these objects in ascending order of size,
12                              starting with the smallest on the left hand side.</mattext>
13                      </material>
14                      <render_extension>
15                          <ims_render_object shuffle="Yes" orientation="Row">
16                              <response_label ident="A">
17                                  <material>
18                                      <matimage imagtype="image/gif" uri="object1.gif"></matimage>
19                                  </material>
20                              </response_label>
21                              <response_label ident="B">
22                                  <material>
23                                      <matimage imagtype="image/gif" uri="object2.gif"></matimage>
24                                  </material>
25                              </response_label>
26                              <response_label ident="C">
27                                  <material>
28                                      <matimage imagtype="image/gif" uri="object3.gif"></matimage>
29                                  </material>
30                              </response_label>
31                              <response_label ident="D">
32                                  <material>
33                                      <matimage imagtype="image/gif" uri="object4.gif"></matimage>
34                              </material>
35                              </response_label>
36                              <response_label ident="E">
37                                  <material>
38                                      <matimage imagtype="image/gif" uri="object5.gif"></matimage>
39                                  </material>
40                              </response_label>
41                          </ims_render_object>
42                      </render_extension>
43                  </response_lid>
44              </presentation>
45          </item>
46      </questestinterop>
```

Note: This example makes use of an extension i.e. *ims_render_object*. In a later version of the specification this extension may be adopted as part of the core specification.

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/oobj_i_002/oobj_i_002.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/oobj_ir_002/oobj_ir_002.xml'.

4.1.11 Connect-the-Points

Figure 4.11 shows a typical connect-the -points question. The corresponding XML is listed after the figure. The user is expected to click on the appropriate area in the image and to draw the corresponding figure.

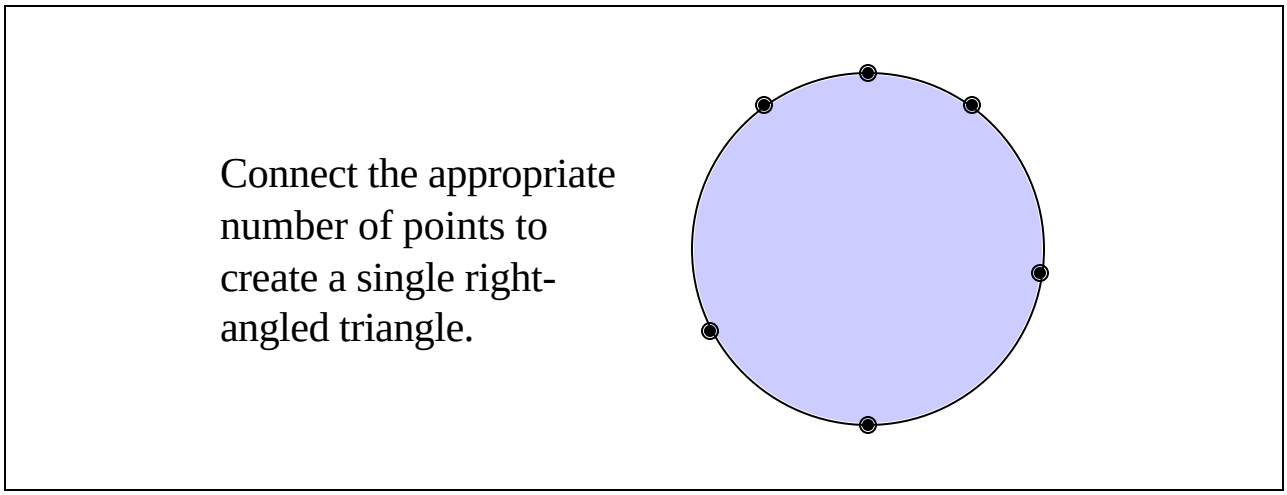

Figure 4.11 Connect-the-points.

The equivalent XML (without response processing):

```
<questestinterop>
    <qticomment>
        This is a standard connect-the-points example.
        The logical identifier response-type is used.
        No response processing is incorporated.
    </qticomment>
    <item title="standard connect-the-points item" ident="IMS_V01_I_BasicExample018">
        <presentation label="BasicExample018">
            <response_lid ident="CTP01" rcardinality="Multiple" rtiming="No">
                <material><mattext>Connect the appropriate number of points to create a
                    single right-angled triangle.</mattext></material>
                <render_hotspot showdraw="Yes">
                    <material>
                        <matimage imagtype=”image/gif” uri=”ctpoint.gif” x0="0" width="400"
                            y0="0" height="200">
                        </matimage>
                    <material>
                    <response_label ident="A" rarea="Ellipse"> 300,20,1,1
                    </response_label>
                    <response_label ident="B" rarea="Ellipse"> 320,40,1,1
                    </response_label>
                    <response_label ident="C" rarea="Ellipse"> 380,100,1,1
                    </response_label>
                    <response_label ident="D" rarea="Ellipse"> 300,180,1,1
                    </response_label>
                    <response_label ident="E" rarea="Ellipse"> 240,120,1,1
                    </response_label>
                    <response_label ident="F" rarea="Ellipse"> 280,40,1,1
                    </response_label>
                </render_hotspot>
            </response_lid>
        </presentation>
    </item>
<questestinterop>
```

The equivalent XML (with response processing):

```
<questestinterop>
    <qticomment>
        This is a standard connect-the-points example.
        The logical identifier response-type is used.
        No response processing is incorporated.
    </qticomment>
    <item title="standard connect-the-points item" ident="IMS_V01_I_BasicExample018">
        <presentation label="BasicExample018">
            <response_lid ident="CTP01" rcardinality="Multiple" rtiming="No">
                <material><mattext>Connect the appropriate number of points to create a
                    single right-angled triangle.</mattext></material>
                <render_hotspot showdraw="Yes">
                    <material>
                        <matimage imagtype=”image/gif” uri=”ctpoint.gif” x0="0" width="400"
                            y0="0" height="200">
                        </matimage>
                    <material>
                    <response_label ident="A" rarea="Ellipse"> 300,20,1,1
                    </response_label>
                    <response_label ident="B" rarea="Ellipse"> 320,40,1,1
                    </response_label>
                    <response_label ident="C" rarea="Ellipse"> 380,100,1,1
                    </response_label>
                    <response_label ident="D" rarea="Ellipse"> 300,180,1,1
                    </response_label>
                    <response_label ident="E" rarea="Ellipse"> 240,120,1,1
                    </response_label>
                    <response_label ident="F" rarea="Ellipse"> 280,40,1,1
                    </response_label>
                </render_hotspot>
            </response_lid>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar varname="CTPCHOICE" vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition>
                <qticomment>Scoring for the correct answer.</qticomment>
                <conditionvar>
                    <and>
                        <varequal respident="CTP01">A</varequal>
                        <varequal respident="CTP01">D</varequal>
                        <or>
                            <varequal respident="CTP01">B</varequal>
                            <varequal respident="CTP01">C</varequal>
                            <varequal respident="CTP01">E</varequal>
                            <varequal respident="CTP01">F</varequal>
                        </or>
                    </and>
                </conditionvar>
                <setvar action="Add" varname="CTPCHOICE">3</setvar>
                <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
            </respcondition>
        </resprocessing>
        <itemfeedback ident="Correct" view="Candidate">
            <material><mattext>Correct.</mattext></material>
        </itemfeedback>
    </item>
<questestinterop>
```

This XML code is available in the file: ‘IMS_QTIv1/Valid/Basic/ctpt_i_001/ctpt_i_001.xml’. The equivalent example with the response processing incorporated is available in the file: ‘IMS_QTIv1/Valid/Basic/ctpt_ir_001/ctpt_ir_001.xml’.

4.2 XY Co-ordinate

4.2.1 Standard Image Hot Spot

Figure 4.12 shows a typical image hot spot question. The corresponding XML is listed after the figure. The user is expected to click on the appropriate area in the image.

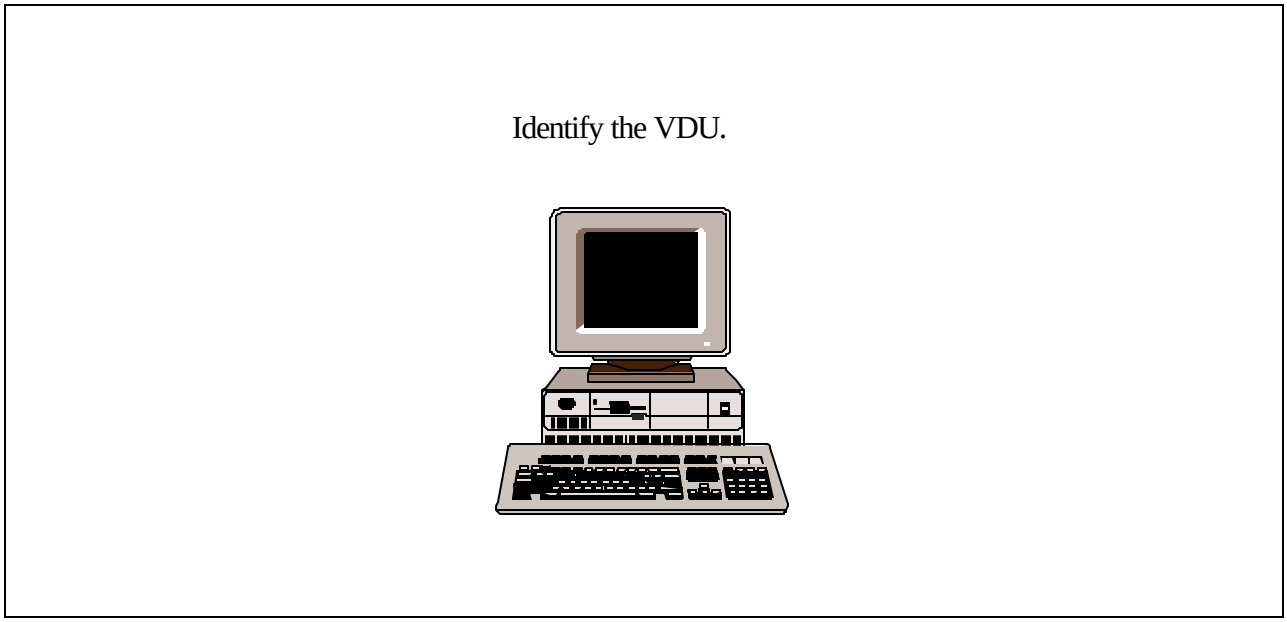

Figure 4.12 Standard image hot spot item.

The equivalent XML (without response processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a standard image hotspot example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="standard image hotspot item" ident="IMS_V01_I_BasicExample011">
7               <presentation label="BasicExample011">
8                   <response_xy ident="IHS01" rcardinality="Single" rtiming="No">
9                       <material><mattext>Identify the vdu.</mattext></material>
10                      <render_hotspot>
11                          <material>
12                              <matimage imagtype=”image/gif” uri=”ihsvdu.gif” x0="0" width="300"
13                                  y0="0" height="400">
14                              </matimage>
15                          <material>
16                          <response_label ident="A" rarea="Rectangle">50,200,250,350
17                          </response_label>
18                      </render_hotspot>
19                  </response_xy>
20              </presentation>
21          </item>
22      <questestinterop>
```

The equivalent XML (with response processing):

```
<questestinterop>
    <qticomment>
        This is a standard image hotspot example.
        No response processing is incorporated.
    </qticomment>
    <item title="standard image hotspot item" ident="IMS_V01_I_BasicExample011b">
        <presentation label="BasicExample011b">
            <response_xy ident="IHS01" rcardinality="Single" rtiming="No">
                <material>
                    <mattext> Identify the vdu.
                    </mattext>
                </material>
                <render_hotspot>
                    <material>
                        <matimage imagtype="image/gif" uri="ihsvdu.gif"
                            x0="0" width="300" y0="0" height="400">
                        </matimage>
                    </material>
                    <response_label ident="A" rarea="Rectangle"> 50,200,250,350
                    </response_label>
                </render_hotspot>
            </response_xy>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar varname="IHSSCORE" vartype="Integer" defaultval="1"/>
            </outcomes>
            <respcondition>
                <qticomment>Scoring for the correct answer.</qticomment>
                <conditionvar>
                    <varinside respident="IHS01" areatype="Rectangle"> 50,200,250,350
                    </varinside>
                </conditionvar>
                <setvar action="Add" varname="IHSSCORE"> 1 </setvar>
                <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
            </respcondition>
            <respcondition>
                <qticomment>Scoring for the incorrect answer.</qticomment>
                <conditionvar>
                    <not>
                        <varinside respident="IHS01" areatype="Rectangle"> 50,200,250,350
                        </varinside>
                    </not>
                </conditionvar>
                <setvar action="Subtract" varname="IHSSCORE"> 1 </setvar>
                <displayfeedback feedbacktype="Response" linkrefid="Incorrect"/>
            </respcondition>
        </resprocessing>
        <itemfeedback ident="Correct" view="Candidate">
            <material><mattext>Correct. </mattext></material>
        </itemfeedback>
        <itemfeedback ident="Incorrect" view="Candidate">
            <material><mattext>No, that is not the VDU.</mattext></material>
        </itemfeedback>
    </item>
</questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/ihsp_i_001/ihsp_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/ihsp_ir_001/ihsp_ir_001.xml'.

4.2.2 Drag-and-Drop (images)

Figure 4.13 shows a typical drag-and-drop question. The corresponding XML is listed after the figure. The user is expected to click on the appropriate answer images and to place them in the appropriate response holders.

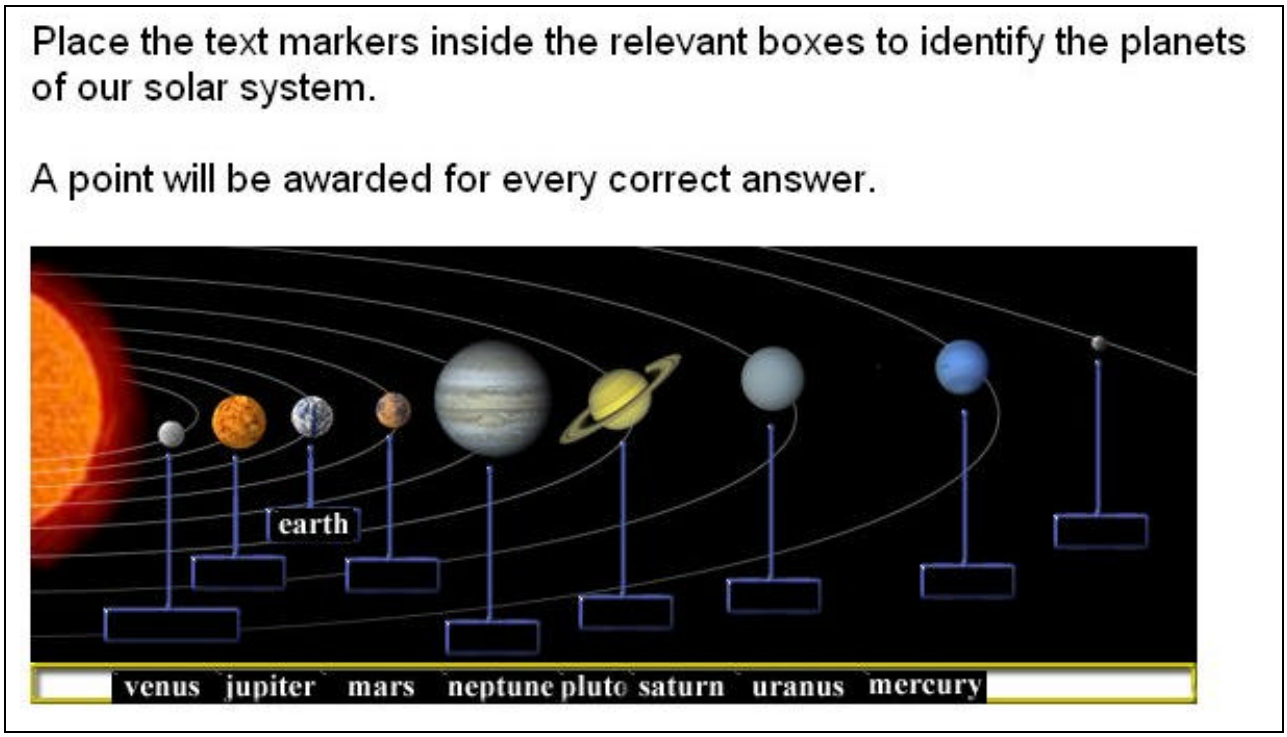

Figure 4.13 Drag-and-drop.

```
1       <questestinterop>
2           <qticomment>
3               This illustrates an example 'drag - drop' question.  Candidates are required to
4               place a label showing the name of a planet in the appropriate position to identify
5               the planets in our solar system.  Each correctly placed label gives the candidate
6               one mark.  If one or more of the labels is incorrectly placed, feedback is given
7               naming the planets in correct order.
8           </qticomment>
9           <item title="The Planets" ident="qm_1052138399372757">
10              <presentation>
11                  <material>
12                      <mattext texttype="text/plain">
13                          Place the text markers inside the relevant boxes to identify the
14                          planets of our solar system.
15                          A point will be awarded for every correct answer.
16                      </mattext>
17                  </material>
18                  <response_xy ident="planets" rcardinality="Multiple">
19                      <render_hotspot>
20                          <material>
21                              <matimage imagtype="image/jpg" uri="solar_system.jpg" height="220"
22                                  width="560"/>
23                          </material>
24                          <response_label ident="earth">
25                              <material>
26                                  <matimage imagtype="image/jpg" uri="earth.jpg" height="16"
27                                      width="39"/>
28                              </material>
29                          </response_label>
30                          <response_label ident="venus">
31                              <material>
```

```
                                <matimage imagtype="image/jpg" uri="venus.jpg" height="16"
                                width="50"/>
                            </material>
                        </response_label>
                        <response_label ident="jupiter">
                            <material>
                                <matimage imagtype="image/jpg" uri="jupiter.jpg" height="16"
                                    width="50"/>
                            </material>
                        </response_label>
                        <response_label ident="mars">
                            <material>
                                <matimage imagtype="image/jpg" uri="mars.jpg" height="16"
                                    width="58"/>
                            </material>
                        </response_label>
                        <response_label ident="neptune">
                            <material>
                                <matimage imagtype="image/jpg" uri="neptune.jpg" height="16"
                                    width="56"/>
                            </material>
                        </response_label>
                        <response_label ident="pluto">
                            <material>
                                <matimage imagtype="image/jpg" uri="pluto.jpg" height="16"
                                width="34"/>
                            </material>
                        </response_label>
                        <response_label ident="saturn">
                            <material>
                                <matimage imagtype="image/jpg" uri="saturn.jpg" height="16"
                                    width="51"/>
                            </material>
                        </response_label>
                        <response_label ident="uranus">
                            <material>
                                <matimage imagtype="image/jpg" uri="uranus.jpg" height="16"
                                    width="62"/>
                            </material>
                        </response_label>
                        <response_label ident="mercury">
                            <material>
                                <matimage imagtype="image/jpg" uri="mercury.jpg" height="16"
                                    width="59"/>
                            </material>
                        </response_label>
                    </render_hotspot>
                </response_xy>
            </presentation>
        </item>
    <questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basicdobj_i_001/dobj_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/dobj_ir_001/dobj_ir_001.xml'.

4.2.3 Connect-the-Points

Figure 4.14 shows a typical connect-the -points question. The corresponding XML is listed after the figure. The user is expected to click on the appropriate area in the image and to draw the corresponding figure.

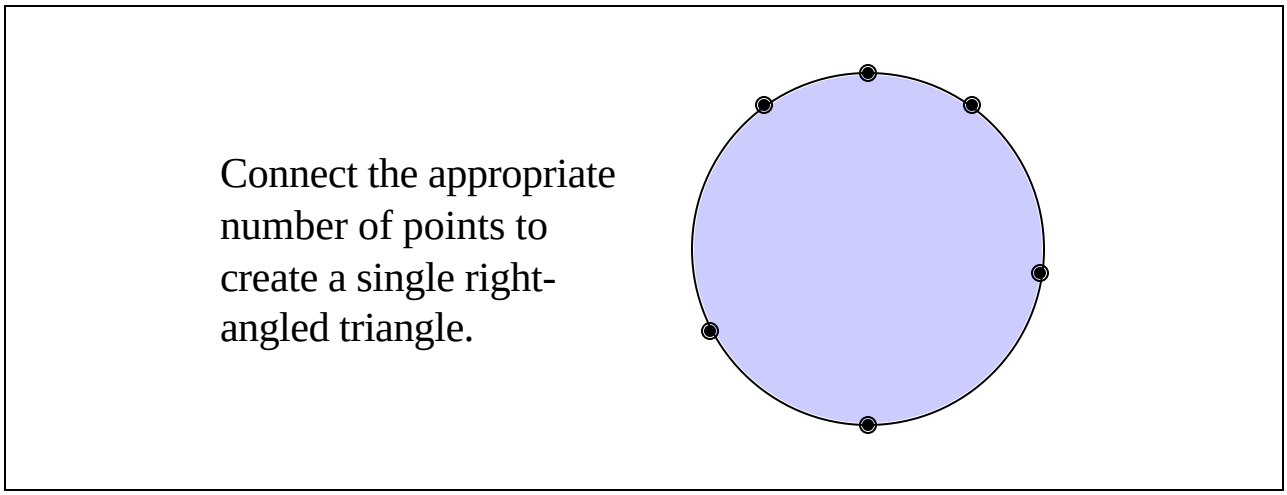

Figure 4.14 Connect-the-points.

The equivalent XML (without response processing):

```
1       <questestinterop>
2           <qticomment>
3               This is a standard connect-the-points example.
4               The XY response-type is used.
5           </qticomment>
6           <item title="standard connect-the-points item" ident="IMS_V01_I_BasicExample019">
7               <presentation label="BasicExample019">
8                   <response_xy ident="CTP02" rcardinality="Multiple" rtiming="No">
9                       <material>
10                          <mattext>Connect the appropriate number of points to create a single
11                                  right-angled triangle.
12                          </mattext>
13                      </material>
14                      <render_hotspot showdraw="Yes">
15                          <material>
16                              <matimage imagtype=”image/gif” uri=”ctpoint.gif” x0="0" width="400"
17                                  y0="0" height="200">
18                              </matimage>
19                          <material>
20                          <response_label ident="A" rarea="Ellipse"> 300,20,1,1
21                          </response_label>
22                          <response_label ident="B" rarea="Ellipse"> 320,40,1,1
23                          </response_label>
24                          <response_label ident="C" rarea="Ellipse"> 380,100,1,1
25                          </response_label>
26                          <response_label ident="D" rarea="Ellipse"> 300,180,1,1
27                          </response_label>
28                          <response_label ident="E" rarea="Ellipse"> 240,120,1,1
29                          </response_label>
30                          <response_label ident="F" rarea="Ellipse"> 280,40,1,1
31                          </response_label>
32                      </render_hotspot>
33                  </response_xy>
34              </presentation>
35          </item>
36      <questestinterop>
```

The equivalent XML (with response processing):

```
<questestinterop>
    <qticomment>
        This is a standard connect-the-points example.
        The XY response-type is used.
    </qticomment>
    <item title="standard connect-the-points item" ident="IMS_V01_I_BasicExample019">
        <presentation label="BasicExample019">
            <response_xy ident="CTP02" rcardinality="Multiple" rtiming="No">
                <material>
                    <mattext>Connect the appropriate number of points to create a single
                            right-angled triangle.
                    </mattext>
                </material>
                <render_hotspot showdraw="Yes">
                    <material>
                        <matimage imagtype=”image/gif” uri=”ctpoint.gif” x0="0" width="400"
                            y0="0" height="200">
                        </matimage>
                    <material>
                    <response_label ident="A" rarea="Ellipse">300,20,1,1
                    </response_label>
                    <response_label ident="B" rarea="Ellipse">320,40,1,1
                    </response_label>
                    <response_label ident="C" rarea="Ellipse">380,100,1,1
                    </response_label>
                    <response_label ident="D" rarea="Ellipse">300,180,1,1
                    </response_label>
                    <response_label ident="E" rarea="Ellipse">240,120,1,1
                    </response_label>
                    <response_label ident="F" rarea="Ellipse">280,40,1,1
                    </response_label>
                </render_hotspot>
            </response_xy>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar varname="CTPCHOICE" vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition>
                <qticomment>Scoring for the correct answer.</qticomment>
                <conditionvar>
                    <and>
                        <varinside respident="CTP02" areatype="Ellipse">300,20,1,1
                        </varinside>
                        <varinside respident="CTP02" areatype="Ellipse">300,180,1,1
                        </varinside>
                        <or>
                            <varinside respident="CTP02" areatype="Ellipse">320,40,1,1
                            </varinside>
                            <varinside respident="CTP02" areatype="Ellipse">380,100,1,1
                            </varinside>
                            <varinside respident="CTP02" areatype="Ellipse">240,120,1,1
                            </varinside>
                            <varinside respident="CTP02" areatype="Ellipse">280,40,1,1
                            </varinside>
                        </or>
                    </and>
                </conditionvar>
                <setvar action="Add" varname="CTPCHOICE">3</setvar>
                <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
            </respcondition>
        </resprocessing>
        <itemfeedback ident="Correct" view="Candidate">
            <material><mattext>Correct.</mattext></material>
        </itemfeedback>
    </item>
<questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/ctpt_i_002/ctpt_i_002.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/ctpt_ir_002/ctpt_ir_002.xml'.

4.3 String

4.3.1 Standard Fill-in-Blank (Text)

Figure 4.15 shows a typical FIB text. The corresponding XML is listed after the figure. The user is required to type the answer into the allocated space.

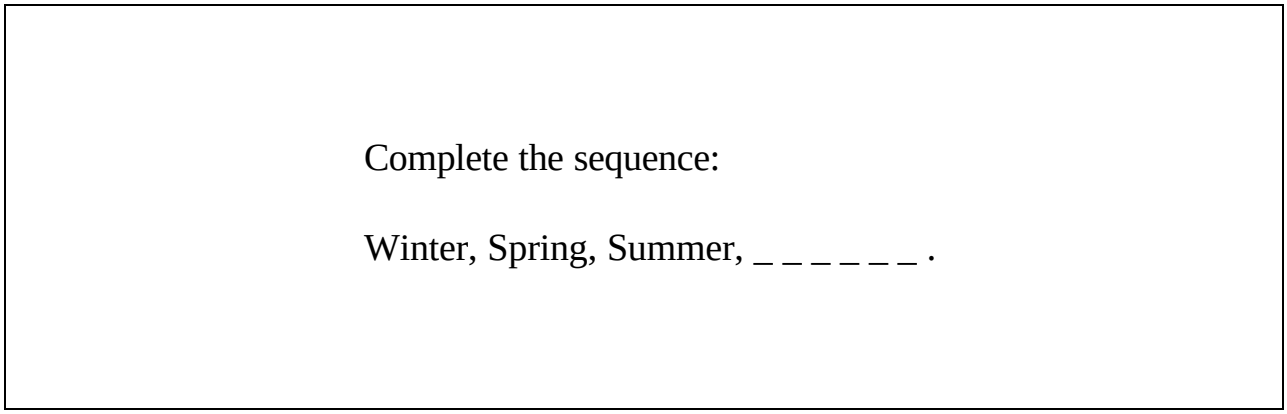

Figure 4.15 Standard fill-in-blank (text) item.

The equivalent XML (without response processing) is:

```
1       <questestinterop>
2           <qticomment>
3               This is a standard fill-in-blank (text) example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB string Item" ident="IMS_V01_I_BasicExample012">
7               <presentation label="BasicExample012">
8                   <response_str ident="FIB01" rcardinality="Single" rtiming="No">
9                       <render_fib fibtype="String" prompt="Dashline" maxchars="6">
10                          <material>
11                              <mattext>Complete the sequence: Winter, Spring, Summer, </mattext>
12                          </material>
13                          <response_label ident="A">
14                          </response_label>
15                          <material>
16                              <mattext>.</mattext>
17                          </material>
18                      </render_fib>
19                  </response_str>
20              </presentation>
21          </item>
22      <questestinterop>
```

The equivalent XML (with response processing) is:

```
1       <questestinterop>
2           <qticomment>
3               This is a standard multiple fill-in-blank (text) example.
4           </qticomment>
5           <item title="Standard FIB string Item" ident="IMS_V01_I_BasicExample013b">
6               <presentation label="BasicExample013b">
7                   <material>
8                           <mattext>Fill-in-the blanks in this text from Richard III:</mattext>
9                   </material>
10                  <response_str ident="FIB01" rcardinality="Single" rtiming="No">
11                      <render_fib fibtype="String" prompt="Dashline" maxchars="6">
12                          <material>
13                              <mattext>Now is the </mattext>
```

```
                    </material>
                    <response_label ident="A">
                    </response_label>
                </render_fib>
            </response_str>
            <response_str ident="FIB02" rcardinality="Single" rtiming="No">
                <render_fib fibtype="String" prompt="Dashline" maxchars="6">
                    <material>
                        <mattext>of our discontent made glorious </mattext>
                    </material>
                    <response_label ident="A">
                    </response_label>
                </render_fib>
            </response_str>
            <response_str ident="FIB03" rcardinality="Single" rtiming="No">
                <render_fib fibtype="String" prompt="Dashline" maxchars="4">
                    <material>
                        <mattext>by these sons of </mattext>
                    </material>
                    <response_label ident="A">
                    </response_label>
                </render_fib>
            </response_str>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar varname="FIBSCORE1" vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition>
                <qticomment>Scoring for the correct answer. </qticomment>
                <conditionvar>
                    <varequal respident="FIB01" case="Yes"> Winter </varequal>
                    <varequal respident="FIB02" case="Yes"> Summer </varequal>
                    <varequal respident="FIB03" case="Yes"> York </varequal>
                </conditionvar>
                <setvar action="Add" varname="FIBSCORE1"> 3 </setvar>
                <displayfeedback feedbacktype="Response" linkrefid="AllCorrect"/>
            </respcondition>
        </resprocessing>
        <resprocessing>
            <outcomes>
                <decvar varname="DUMMY"/>
            </outcomes>
            <respcondition>
                <qticomment>Detecting incorrect answers for feedback. </qticomment>
                <conditionvar>
                    <not><varequal respident="FIB01" case="Yes"> Winter </varequal></not>
                </conditionvar>
                <displayfeedback feedbacktype="Response" linkrefid="Incorrect1"/>
            </respcondition>
        </resprocessing>
        <resprocessing>
            <outcomes>
                <decvar varname="DUMMY"/>
            </outcomes>
            <respcondition>
                <qticomment>Detecting incorrect answers for feedback. </qticomment>
                <conditionvar>
                    <not><varequal respident="FIB02" case="Yes">Summer</varequal></not>
                </conditionvar>
                <displayfeedback feedbacktype="Response" linkrefid="Incorrect2"/>
            </respcondition>
        </resprocessing>
        <resprocessing>
            <outcomes>
                <decvar varname="DUMMY"/>
            </outcomes>
            <respcondition>
                <qticomment>Detecting incorrect asnwers for feedback. </qticomment>
                <conditionvar>
                    <not><varequal respident="FIB03" case="Yes">York</varequal></not>
```

```
85                      </conditionvar>
86                      <displayfeedback feedbacktype="Response" linkrefid="Incorrect3"/>
87                  </respcondition>
88              </resprocessing>
89              <itemfeedback ident="AllCorrect" view="Candidate">
90                  <material><mattext>All correct.  Well done.</mattext></material>
91              </itemfeedback>
92              <itemfeedback ident="InCorrect1" view="Candidate">
93                  <material><mattext>No. The correct first answer is "Winter".</mattext>
94                  </material>
95              </itemfeedback>
96              <itemfeedback ident="InCorrect2" view="Candidate">
97                  <material><mattext>No. The correct second answer is "Summer".</mattext>
98                  </material>
99              </itemfeedback>
100             <itemfeedback ident="InCorrect3" view="Candidate">
101                 <material><mattext>No. The correct thrd answer is "York".</mattext>
102                 </material>
103             </itemfeedback>
104         </item>
105     </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_i_001/fibs_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_ir_001/fibs_ir_001.xml'.

4.3.2 Standard Multiple Fill-in-Blank (Text)

Figure 4.16 shows a typical FIB text with multiple entries. The corresponding XML is listed after the figure. The user is expected to type the answers in the allocated spaces.

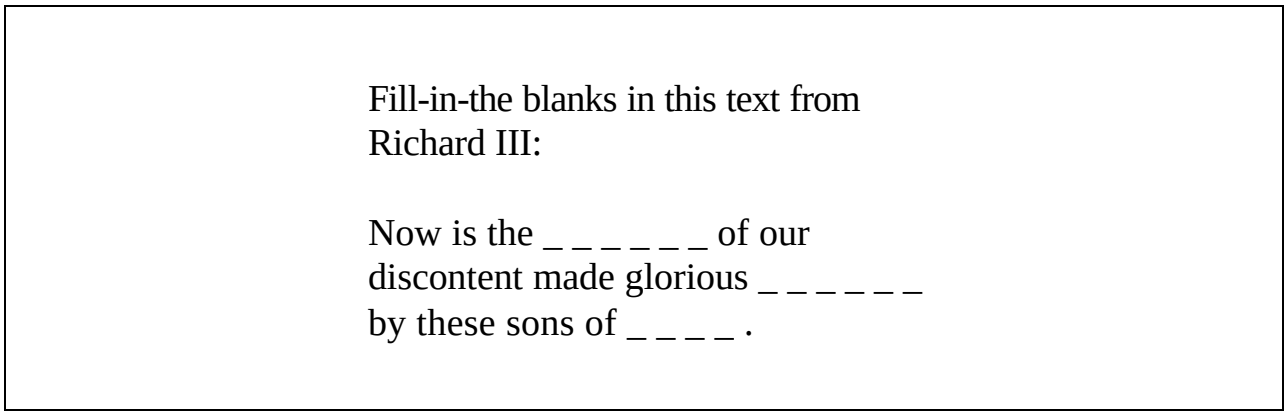

Figure 4.16 Multiple fill-in-blank (text) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard multiple fill-in-blank (text) example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB string Item" ident="IMS_V01_I_BasicExample013">
7               <presentation label="BasicExample013">
8                   <material>
9                       <mattext>Fill-in-the blanks in this text from Richard III:</mattext>
10                  </material>
11                  <response_str ident="FIB01" rcardinality="Single" rtiming="No">
12                      <render_fib fibtype="String" prompt="Dashline" maxchars="6">
13                          <material><mattext>Now is the </mattext></material>
14                          <response_label ident="A"></response_label>
15                      </render_fib>
16                  </response_str>
17                  <response_str ident="FIB02" rcardinality="Single" rtiming="No">
18                      <render_fib fibtype="String" prompt="Dashline" maxchars="6">
19                          <material>
20                              <mattext>of our discontent made glorious </mattext>
21                          </material>
22                          <response_label ident="A">
23                          </response_label>
24                      </render_fib>
25                  </response_str>
26                  <response_str ident="FIB03" rcardinality="Single" rtiming="No">
27                      <render_fib fibtype="String" prompt="Dashline" maxchars="4">
28                          <material>
29                              <mattext>by these sons of </mattext>
30                          </material>
31                          <response_label ident="A">
32                          </response_label>
33                      </render_fib>
34                  </response_str>
35              </presentation>
36          </item>
37      <questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_i_002/fibs_i_002.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_ir_002/fibs_ir_002.xml'.

4.3.3 Standard Short Answer

Figure 4.17 shows a typical short answer question. The corresponding XML is listed after the figure. The user is expected to type text into the space supplied.

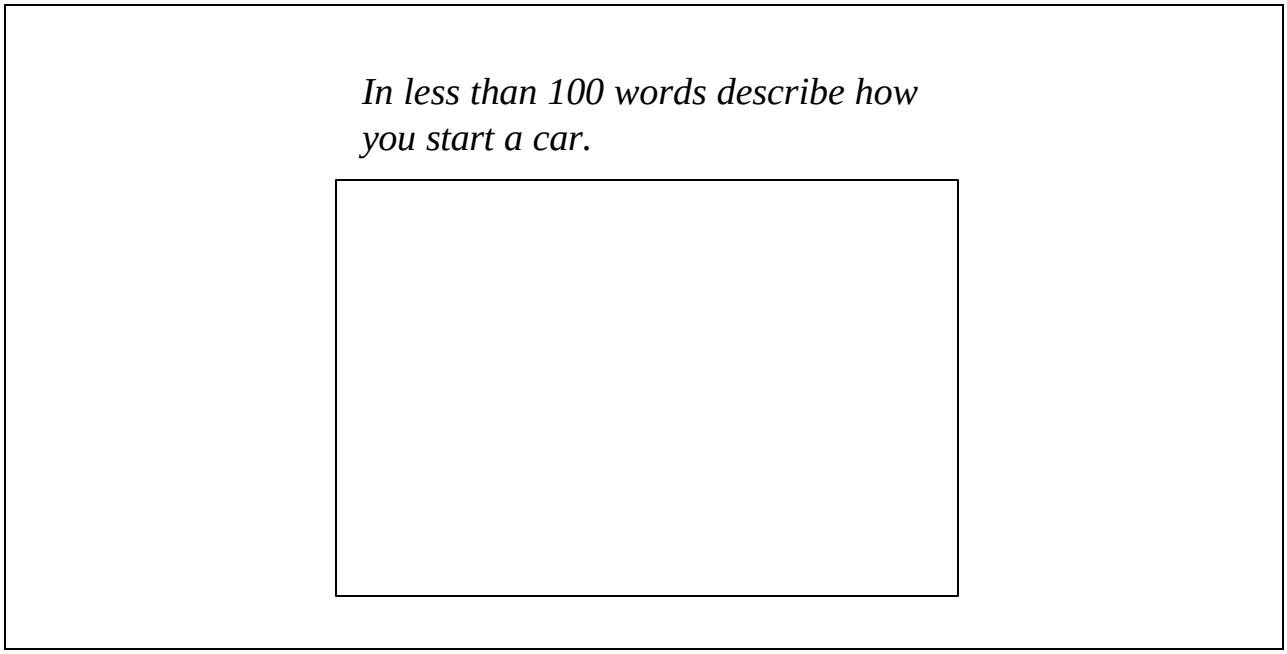

Figure 4.17 Standard short answer item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard fill-in-blank short answer example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB short answer Item" ident="IMS_V01_I_BasicExample014">
7               <presentation label="BasicExample014">
8                   <response_str ident="FIB91" rcardinality="Ordered" rtiming="No">
9                       <render_fib fibtype="String" prompt="Box" rows="20" columns="80">
10                          <material>
11                              <mattext>In less than 100 words describe how you start a car.
12                              </mattext>
13                          </material>
14                          <response_label ident="A">
15                          </response_label>
16                      </render_fib>
17                  </response_str>
18              </presentation>
19          </item>
20      <questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_i_003/fibs_i_003.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/fibs_ir_003/fibs_ir_003.xml'.

4.4 Numerical

4.4.1 Standard Fill-in-Blank (Decimal)

Figure 4.18 shows a typical FIB number. The corresponding XML is listed after the figure. The user is expected to type the appropriate number into the supplied box.

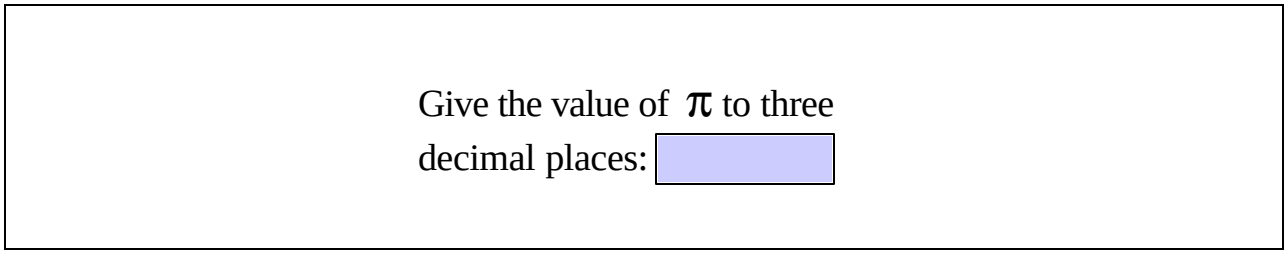

Figure 4.18 Standard numerical fill-in-blank item.

The equivalent XML (without response processing) is:

```
1       <questestinterop>
2           <qticomment>
3               This is a standard numerical fill-in-blank (decimal) example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB numerical Item" ident="IMS_V01_I_BasicExample015">
7               <presentation label="BasicExample015">
8                   <response_num ident="NUM01" rcardinality="Single" rtiming="No">
9                       <render_fib fibtype="Decimal" prompt="Box" maxchars="6">
10                          <material>
11                              <mattext charset="ascii/us">Give the value of</mattext>
12                              <mattext charset="greek"> p </mattext>
13                              <mattext charset="ascii/us"> to three decimal places: </mattext>
14                          </material>
15                          <response_label ident="A">
16                          </response_label>
17                      </render_fib>
18                  </response_str>
19              </presentation>
20          </item>
21      </questestinterop>
```

The equivalent XML (without response processing) is:

```
1       <questestinterop>
2           <qticomment>
3               This is a standard numerical fill-in-blank (decimal) example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB numerical Item"
7                   ident="IMS_V01_I_BasicExample015b">
8               <presentation label="BasicExample015b">
9                   <response_num ident="NUM01" rcardinality="Single" rtiming="No"
10                          numtype="Decimal">
11                      <render_fib fibtype="Decimal" prompt="Box" maxchars="6">
12                          <material>
13                              <mattext charset="ascii/us">Give the value of </mattext>
14                              <mattext charset="greek"> p </mattext>
15                              <mattext charset="ascii/us"> to three decimal places: </mattext>
16                          </material>
17                          <response_label ident="A"/>
18                      </render_fib>
19                  </response_num>
20              </presentation>
```

```
21            <resprocessing>
22                <outcomes>
23                    <decvar varname="REALSCORE" vartype="Integer" defaultval="0"/>
24                </outcomes>
25                <respcondition>
26                    <qticomment>Scoring for the correct answer. </qticomment>
27                    <conditionvar>
28                        <vargte respident="NUM01"> 3.141 </vargte>
29                        <varlte respident="NUM01"> 3.149 </varlte>
30                    </conditionvar>
31                    <setvar action="Add" varname="REALSCORE"> 1 </setvar>
32                    <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
33                </respcondition>
34                <respcondition>
35                    <qticomment>Scoring for the incorrect answer. </qticomment>
36                    <conditionvar>
37                        <not><vargte respident="NUM01"> 3.141 </vargte></not>
38                        <not><varlte respident="NUM01"> 3.149 </varlte></not>
39                        </not>
40                    </conditionvar>
41                    <setvar action="Subtract" varname="REALSCORE"> 1 </setvar>
42                    <displayfeedback feedbacktype="Response" linkrefid="Incorrect"/>
43                </respcondition>
44            </resprocessing>
45            <itemfeedback ident="Correct" view="Candidate">
46                <material><mattext>Yes, you are correct.  Well done. </mattext></material>
47            </itemfeedback>
48            <itemfeedback ident="Incorrect" view="Candidate">
49                <material><mattext>No.  The correct answer is 3.142. </mattext></material>
50            </itemfeedback>
51        </item>
52    </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/fibn_i_004/fibn_i_004.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/fibn_ir_004/fibn_ir_004.xml'.

4.4.2 Standard Fill-in-Blank (Integer)

Figure 4.19 shows a typical numerical entry using FIB rendering. The corresponding XML is listed after the figure. The user must enter an integer.

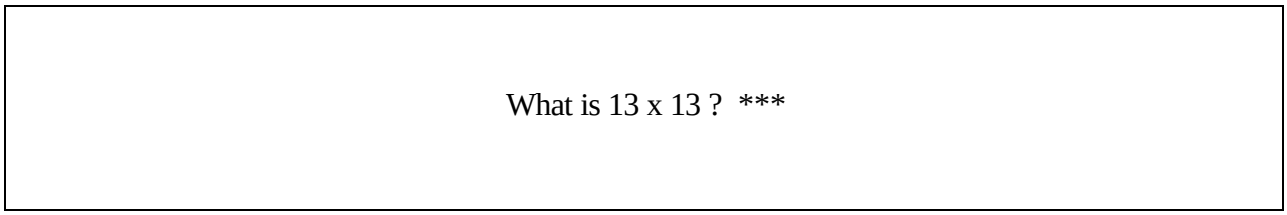

Figure 4.19 Standard FIB (integer) item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard numerical fill-in-blank (integer) example.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB numerical Item" ident="IMS_V01_I_BasicExample016">
7               <presentation label="BasicExample016">
8                   <response_num ident="NUM02" rcardinality="Single" rtiming="No">
9                       <render_fib fibtype="Integer" prompt="Asterisk" maxchars="3">
10                          <material>
11                              <mattext charset="ascii/us">What is 13 x 13 ?</mattext>
12                          <material>
13                          <response_label ident="A">
14                          </response_label>
15                      </render_fib>
16                  </response_str>
17              </presentation>
18          </item>
19      </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Basic/fibi_i_005/fibi_i_005.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Basic/fibi_ir_005/fibi_ir_005.xml'.

4.4.3 Numerical Entry with Slider

Figure 4.20shows a typical numerical FIB with slider rendering. The corresponding XML is listed after the figure. The user must move the slider until it displays the required answer.

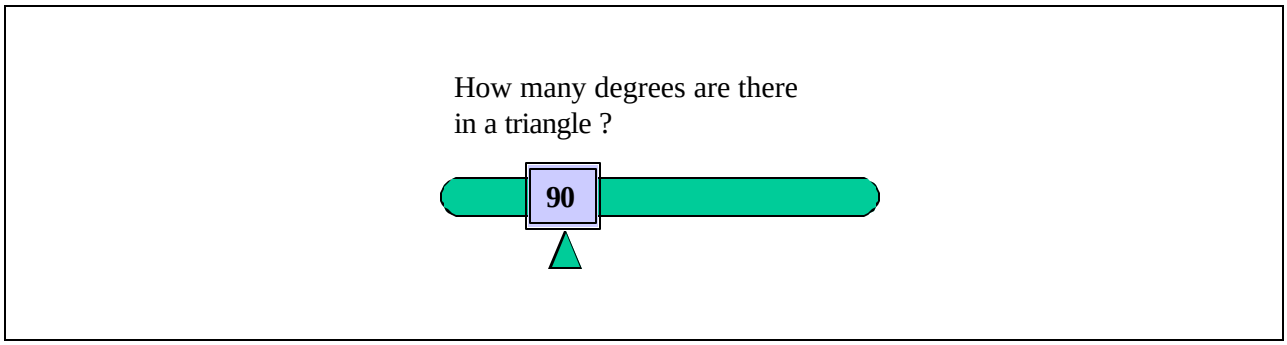

Figure 4.20 Numerical fill-in-blank with slider item.

```
1       <questestinterop>
2           <qticomment>
3               This is a standard numerical fill-in-blank using slider rendering.
4               No response processing is incorporated.
5           </qticomment>
6           <item title="Standard FIB numerical Item with Slider rendering"
7               ident="IMS_V01_I_BasicExample017">
8               <presentation label="BasicExample017">
9                   <response_num ident="NUM04" rcardinality="Single" rtiming="No">
10                      <material>
11                          <mattext>How many degree are there in a triangle ?</mattext>
12                      <material>
13                      <render_slider lowerbound="1" upbound="360" step="1" startval="90"
14                          steplabel="no" orientation="horizontal">
15                          <response_label ident="A">
16                          </response_label>
17                      </render_slider>
18                  </response_str>
19              </presentation>
20          </item>
21      </questestinterop>
```

This XML code is available in the file: ‘IMS_QTIv1/Valid/Basic/ihsp_i_006/ihsp_i_006.xml’. The equivalent example with the response processing incorporated is available in the file: ‘IMS_QTIv1/Valid/Basic/ihsp_ir_006/ihsp_ir_006.xml’.

5. Example Composite Item Types

The composite examples supplied are:

- Multiple choice derivatives
  - Multiple Choice with Fill-in-Blank;
  - Matrix-based Multiple Response.

5.1 Multiple Choice Derivatives

5.1.1 Multiple Choice with Fill-in-Blank

Figure 5.1 shows the multiple-choice question with an additional FIB response opportunity. The corresponding XML is listed after the figure.

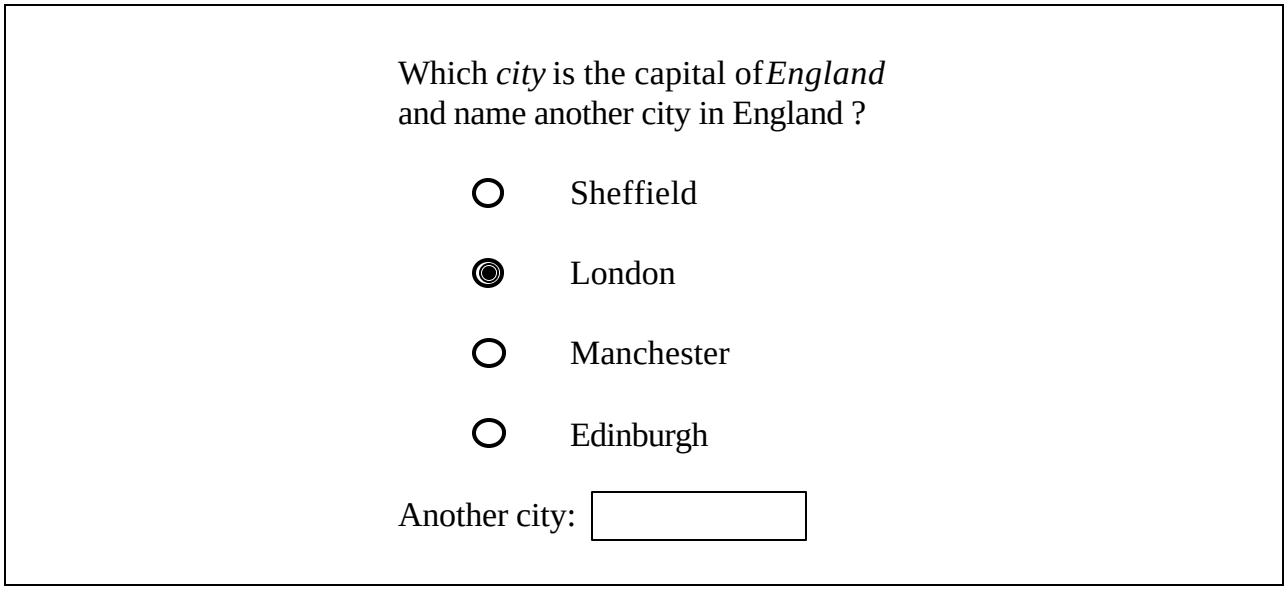

Figure 5.1 Composite multiple choice with FIB item.

The equivalent XML (without response processing) is:

```
1       <questestinterop>
2           <qticomment>
3               This is a composite item.
4               It consist of a standard multiple choice question
5               and a fill-in-blank alternative.
6               No response processing is incorporated.
7           </qticomment>
8           <item title="Composite Item" Ident="IMS_V01_I_CompExample001">
9               <presentation label="CompExample001">
10                  <material>
11                      <mattext>Which city is the capital of England and name another city in
12                          England ?<mattext>
13                  </material>
14                  <response_lid ident="Comp_MC01" rcardinality="Single" rtiming="No">
15                      <render_choice shuffle="Yes">
16                          <response_label ident="A">
17                              <material><mattext>Sheffield</mattext></material>
18                          </response_label>
19                          <response_label ident="B">
20                              <material><mattext>London</mattext></material>
21                          </response_label>
22                          <response_label ident="C">
23                              <material><mattext>Manchester</mattext></material>
24                          </response_label>
25                          <response_label ident="D">
26                              <material><mattext>Edinburgh</mattext></material>
27                          </response_label>
28                      </render_choice>
29              </response_lid>
30                  <response_str ident="Comp_FIB01" rcardinality="Single" rtiming="No">
31                      <render_fib fibtype="String" prompt="Box">
32                          <material>
```

```
                        <mattext>Another city:</mattext>
                    </material>
                    <response_label ident="A">
                    </response_label>
                </render_fib>
            </response_str>
        </presentation>
    </item>
</questestinterop>
```

The equivalent XML (with response processing) is:

```
<questestinterop>
    <qticomment>
        This is a composite item. It consists of a standard multiple choice question
        and a fill-in-blank alternative. Response processing fro the Multiple chpice part
        is supplied. The free format material in the FIB should be returned for non-
        computer based marking.
    </qticomment>
    <item title="Composite Item" ident="IMS_V01_I_CompExample001b">
        <presentation label="CompExample001b">
            <material>
                <mattext>
                    Which city is the capital of England and name another city in England ?
                </mattext>
            </material>
            <response_lid ident="Comp_MC01" rcardinality="Single" rtiming="No">
                <render_choice shuffle="Yes">
                    <response_label ident="A">
                        <material><mattext>Sheffield</mattext></material>
                    </response_label>
                    <response_label ident="B">
                        <material><mattext>London</mattext></material>
                    </response_label>
                    <response_label ident="C">
                        <material><mattext>Manchester</mattext></material>
                    </response_label>
                    <response_label ident="D">
                        <material><mattext>Edinburgh</mattext></material>
                    </response_label>
                </render_choice>
            </response_lid>
            <response_str ident="Comp_FIB01" rcardinality="Single" rtiming="No">
                <render_fib fibtype="String" prompt="Box">
                    <material>
                        <mattext>Another city:</mattext>
                    </material>
                    <response_label ident="A">
                    </response_label>
                </render_fib>
            </response_str>
        </presentation>
        <resprocessing>
            <outcomes>
                <decvar  varname="MCSCORE" vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition title="MCorrect">
                <conditionvar>
                    <varequal respident="Comp_MC01">B</varequal>
                </conditionvar>
                <setvar action="Set" varname="MCSCORE">1</setvar>
                <displayfeedback feedbacktype="Response" linkrefid="MCorrect"/>
            </respcondition>
        </resprocessing>
        <itemfeedback ident="Correct" view="Candidate">
            <material><mattext>Yes, you are right.</mattext></material>
        </itemfeedback>
    </item>
</questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Composite/mchc_i_001/mchc_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Composite/mchc_ir_001/mchc_ir_001.xml'.

5.1.2 Matrix-based Multiple Response

Figure 5.2 shows a composite multiple choice question arranged as a matrix i.e. only one answer per row is permitted. The corresponding XML is listed after the figure. The user must choose one option per row.

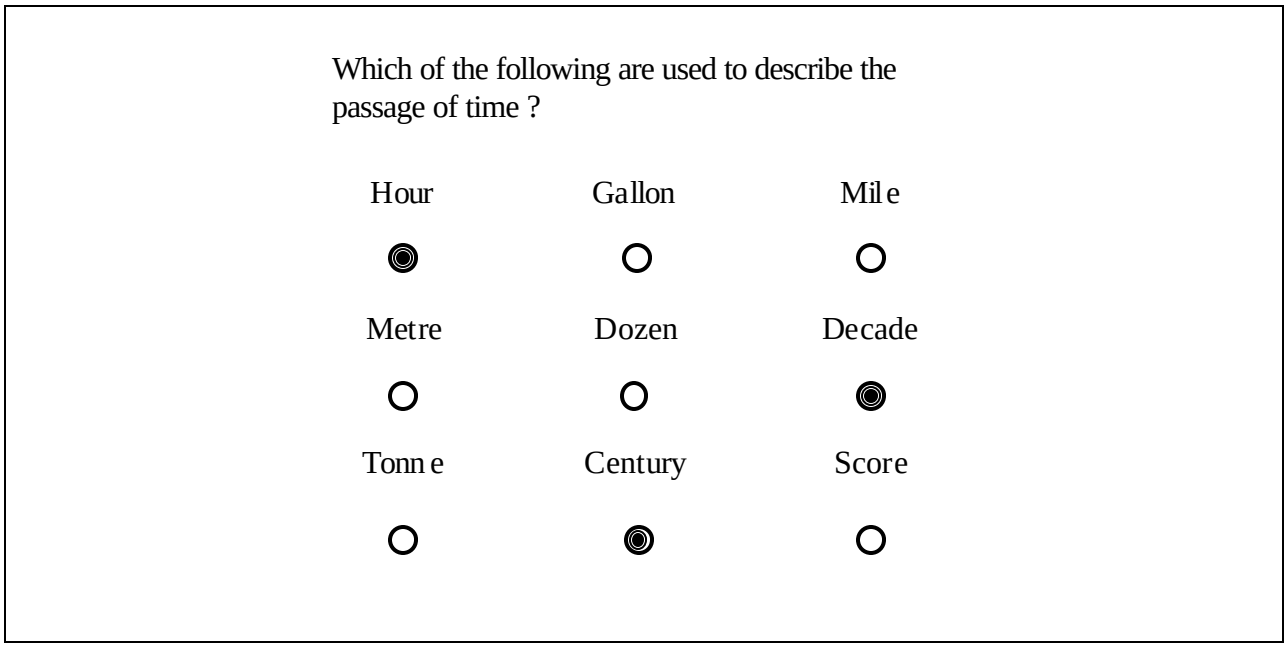

Figure 5.2 Composite matrix-based multiple response item.

```
1       <questestinterop>
2           <qticomment>
3               This is a composite item.
4               It consists of a matrix of multiple choice questions.
5               No response processing is incorporated.
6           </qticomment>
7           <item title="Composite Item" Ident="IMS_V01_I_CompExample002">
8               <presentation label="CompExample002">
9                   <material>
10                      <mattext>Which of the following are used to describe the passage of
11                              time ?
12                      <mattext>
13                  </material>
14                  <response_lid ident="Comp_MC01" rcardinality="Single" rtiming="No">
15                      <render_choice shuffle="Yes">
16                          <response_label ident="A">
17                              <material><mattext>Hour</mattext></material>
18                          </response_label>
19                          <response_label ident="B">
20                              <material><mattext>Galleon</mattext></material>
21                          </response_label>
22                          <response_label ident="C">
23                              <material><mattext>Mile</mattext></material>
24                          </response_label>
25                      </render_choice>
26                  </response_lid>
27                  <response_lid ident="Comp_MC02" rcardinality="Single" rtiming="No">
28                      <render_choice shuffle="Yes">
29                          <response_label ident="A">
30                              <material><mattext>Metre</mattext></material>
31                          </response_label>
32                          <response_label ident="B">
33                              <material><mattext>Dozen</mattext></material>
34                          </response_label>
```

```
35                      <response_label ident="C">
36                          <material><mattext>Decade</mattext></material>
37                      </response_label>
38                  </render_choice>
39              </response_lid>
40              <response_lid ident="Comp_MC03" rcardinality="Single" rtiming="No">
41                  <render_choice shuffle="Yes">
42                      <response_label ident="A">
43                          <material><mattext>Tonne</mattext></material>
44                      </response_label>
45                      <response_label ident="B">
46                          <material><mattext>Century</mattext></material>
47                      </response_label>
48                      <response_label ident="C">
49                          <material><mattext>Score</mattext></material>
50                      </response_label>
51                  </render_choice>
52              </response_lid>
53          </presentation>
54      </item>
55  </questestinterop>
```

This XML code is available in the file: 'IMS_QTIv1/Valid/Composite/mrsp_i_001/mrsp_i_001.xml'. The equivalent example with the response processing incorporated is available in the file: 'IMS_QTIv1/Valid/Composite/mrsp_ir_001/mrsp_ir_001.xml'.

6. Example XML Schema

6.1 Item Examples

6.1.1 Minimum Definition

The XML for the minimal single useful 'Item' is as follows (this is also shown schematically as Figure 4.1):

```
<questestinterop>
    <item ident="IMS_V01_I_BasicExample001a">
    <presentation label="BasicExample001a">
        <material>
               <mattext>Paris is the Capital of France ?</mattext>
        </material>
        <response_lid ident="TF01">
               <render_choice>
                   <response_label ident="T">
                   <material><mattext>True</mattext></material>
                   </response_label>
                   <response_label ident="F">
                   <material><mattext>False</mattext></material>
                   </response_label>
               </render_choice>
        </response_lid>
        </presentation>
    </item>
</questestinterop>
```

6.1.2 Full Definition

The XML for a complete single useful 'Item' is as follows (this is also shown schematically as Figure 4.1 and is an extension of the XML code for the minimal single useful Item example):

```
<questestinterop>
    <qticomment>
        This is a simple True/False multiple choice example.
        The rendering is a standard radio button style.
        No response processing is incorporated.
    </qticomment>
    <item ident="IMS_V01_I_BasicExample001b">
        <duration minute="1"/>
        <itemmetadata>
        </itemmetadata>
        <objectives view="Candidate">
            <material>
                <mattext>To test your understanding of French cities.</mattext>
            </material>
        </objectives>
        <objectives view="Scorer">
            <material>
                <mattext>Award marks for the right answer only.</mattext>
            </material>
        </objectives>
        <itemcontrol hintswitch="Yes"/>
    <itemrubric>
            <material>
                <mattext>Attempt all questions.</mattext>
            </material>
        </itemrubric>
        <presentation label="BasicExample001b">
            <material>
                <mattext>Paris is the Capital of France ?</mattext>
            </material>
            <response_lid ident="TF01" rcardinality="Single" rtiming="No">
                <render_choice>
```

```
                <response_label ident="T">
                    <material><mattext>True</mattext></material>
                </response_label>
                <response_label ident="F">
                    <material><mattext>False</mattext></material>
                </response_label>
            </render_choice>
        </response_lid>
    </presentation>
    <resprocessing>
        <outcomes><decvar/></outcomes>
        <respcondition title="Correct">
            <conditionvar>
                <varequal respident="TF01">T</varequal>
            </conditionvar>
            <SETVAR ACTION="Set" >1</setvar>
            <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
        </respcondition>
    </resprocessing>
    <itemfeedback ident="Correct" view="Candidate">
        <material><mattext>Yes, you are right.</mattext></material>
    </itemfeedback>
  </item>
</questestinterop>
```

This example is given in the example file: IMS_QTIv1/Valid/Basic/trfl_ir_001/trfl_ir_001.xml

6.2 Section Examples

6.2.1 Minimum Definition

The XML for the minimal single useful ‘Section’ is as follows:

```
<questestinterop>
  <section ident=”IMS_V01_S_Example001”>
    <item ident="IMS_V01_I_BasicExample001">
    <presentation label="RS01">
        <material>
            <mattext>Paris is the Capital of France ?</mattext>
        </material>
        <response_lid ident="TF01">
            <render_choice>
                <response_label ident="T">
                    <material><mattext>True</mattext></material>
                </response_label>
                <response_label ident="F">
                    <material><mattext>False</mattext></material>
                </response_label>
            </render_choice>
        </response_lid>
        </presentation>
    </item>
  </section>
</questestinterop>
```

The XML for the minimal multiple ‘Section’ is as follows:

```
<questestinterop>
  <section ident=”IMS_V01_S_Example001”>
    <item ident="IMS_V01_I_BasicExample001">
    <presentation label="RS01">
        <material>
            <mattext>Paris is the Capital of France ?</mattext>
        </material>
        <response_lid ident="TF01">
```

```
            <render_choice>
                <response_label ident="T">
                    <material><mattext>True</mattext></material>
                </response_label>
                <response_label ident="F">
                    <material><mattext>False</mattext></material>
                </response_label>
            </render_choice>
        </response_lid>
        </presentation>
    </item>
</section>
<section ident=”IMS_V01_S_Example002”>
    <item ident="IMS_V01_I_BasicExample200">
    <presentation label="RS20">
        <material>
            <mattext>London is the Capital of Germany ?</mattext>
        </material>
        <response_lid ident="TF02">
            <render_choice>
                <response_label ident="T">
                    <material><mattext>True</mattext></material>
                </response_label>
                <response_label ident="F">
                    <material><mattext>False</mattext></material>
                </response_label>
            </render_choice>
        </response_lid>
        </presentation>
    </item>
</section>
</questestinterop>
```

6.2.2 Full Definition

The XML for a complete useful ‘Section’ example (this contains two Items) is as follows:

```
<questestinterop>
    <qticomment>
        This example consists of two Sections.
    </qticomment>
    <section title="European Capitals" ident="IMS_V01_S_Example201">
        <sectionmetadata/>
        <objectives view="Candidate">
            <material>
                <mattext>To assess your knowledge of the capital cities in Europe.</mattext>
            </material>
        </objectives>
        <objectives view="Tutor">
            <material>
                <mattext>
                    To ensure that the student knows the difference between the Capital cities of
                    France, UK, Germany, Spain and Italy.
                </mattext>
            </material>
        </objectives>
        <item title="Capital of France" ident="I01" maxattempts="6">
            <qticomment>
                This Item is also avalibale in the accompanying example files.
            </qticomment>
            <itemmetadata/>
            <itemrubric view="Candidate">
                <material>
                    <mattext>Choose only one of the choices available.</mattext>
                </material>
            </itemrubric>
            <presentation label="Resp001">
                <response_lid ident="LID01">
```

```
                <material><mattext>What is the Capital of France ?</mattext></material>
                <render_choice shuffle="Yes">
                    <response_label ident="LID01_A">
                        <material><mattext>London</mattext></material>
                    </response_label>
                    <response_label ident="LID01_B">
                        <material><mattext>Paris</mattext></material>
                    </response_label>
                    <response_label ident="LID01_C">
                        <material><mattext>Washington</mattext></material>
                    </response_label>
                    <response_label ident="LID01_D" rshuffle="No">
                        <material><mattext>Berlin</mattext></material>
                    </response_label>
                </render_choice>
            </response_lid>
        </presentation>
        <resprocessing>
            <qticomment/>
            <outcomes>
                <decvar vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition>
                <qticomment> Scoring for the correct answer. </qticomment>
                <conditionvar>
                    <varequal respident="LID01">LID01_B</varequal>
                </conditionvar>
                <setvar action="Set" varname="SCORE"> 10 </setvar>
                <displayfeedback feedbacktype="Response" linkrefid="I01_IFBK01"/>
            </respcondition>
        </resprocessing>
        <itemfeedback title="Correct answer" ident="I01_IFBK01">
            <material>
                <mattext>Correct answer.</mattext>
            </material>
        </itemfeedback>
        <itemfeedback ident="I01_IFBK02">
            <solution>
                <solutionmaterial>
                    <material>
                        <mattext>
                            London is the Capital of England.
                            Paris is the Capital of France.
                            Washington is in the USA.
                            Berlin is the Capital of Germany.
                        </mattext>
                    </material>
                </solutionmaterial>
            </solution>
        </itemfeedback>
        <itemfeedback ident="I01_IFBK03" view="All">
            <hint feedbackstyle="Multilevel">
                <hintmaterial>
                    <material>
                        <mattext>One of the choices is not in Europe.</mattext>
                    </material>
                </hintmaterial>
                <hintmaterial>
                    <material>
                        <mattext>Berlin is the Capital of Germany.</mattext>
                    </material>
                </hintmaterial>
                <hintmaterial>
                    <material>
                        <mattext>The Eiffel tower is in the Capital of France.</mattext>
                    </material>
                </hintmaterial>
            </hint>
        </itemfeedback>
    </item>
</section>
```

```
    <section title="European Rivers" ident="IMS_V01_S_Example202">
        <sectionmetadata/>
        <objectives view="Candidate">
            <material>
                <mattext>To assess your knowledge of the rivers in Europe.</mattext>
            </material>
        </objectives>
        <objectives view="Assessor">
            <material>
                <mattext>Questions on the rivers in Germany, Spain, Italy and France.</mattext>
            </material>
        </objectives>
        <item title="Rivers in France question" ident="I02">
            <itemrubric view="Candidate">
                <material><mattext>Choose all of the correct answers.</mattext></material>
            </itemrubric>
            <presentation label="Resp002">
                <response_lid ident="LID02" rcardinality="Multiple">
                    <material><mattext>Which rivers are in France ?</mattext>
                    </material>
                    <render_choice shuffle="Yes" minnumber="1" maxnumber="2">
                        <response_label ident="LID02_A">
                            <material><mattext>Seine</mattext></material>
                        </response_label>
                        <response_label ident="LID02_B">
                            <material><mattext>Thames</mattext></material>
                        </response_label>
                        <response_label ident="LID02_C">
                            <material><mattext>Danube</mattext></material>
                        </response_label>
                        <response_label ident="LID02_D">
                            <material><mattext>Loire</mattext></material>
                        </response_label>
                    </render_choice>
                </response_lid>
            </presentation>
        </item>
        <item title="Rivers in Germany" ident="I03"/>
            <duration> 0000:00:4T02:30:00 </duration>
            <itemrubric view="Candidate">
                <material><mattext>Choose all of the correct answers.</mattext></material>
            </itemrubric>
            <presentation label="Resp003">
                <response_lid ident="LID03" rcardinality="Multiple">
                    <material>
                        <matimage imagtype=”image/jpeg” uri=”rivers.jpg”></matimage>
                        <mattext>Which rivers are in Germany ?</mattext>
                    </material>
                    <render_hotspot x0="500" y0="500" height="200" width="200">
                        <response_label ident="LID03_A" rarea="Ellipse"> 10,10,2,2
                        </response_label>
                        <response_label ident="LID03_B" rarea="Ellipse"> 15,15,2,2
                        </response_label>
                        <response_label ident="LID03_C" rarea="Ellipse"> 30,30,2,2
                        </response_label>
                        <response_label ident="LID03_D" rarea="Ellipse"> 60,60,2,2
                        </response_label>
                        <response_label ident="LID03_E" rarea="Ellipse"> 70,70,2,2
                        </response_label>
                    </render_hotspot>
                </response_lid>
            </presentation>
        </item>
    </section>
</questestinterop>
```

This example is given in the example file: IMS_QTIv1/Valid/Advanced/mchc_smimr_104/mchc_smimr_104.xml

6.3 Assessment Examples

6.3.1 Minimum Definition

The XML for the minimal single useful 'Assessment' is as follows:

```
<questestinterop>
    <assessment ident="IMS_V01_A_Example301">
        <section ident="IMS_V01_S_Example301">
            <item ident="IMS_V01_I_BasicExample301">
            <presentation label="RS01">
                <material><mattext>Paris is the Capital of France ?</mattext></material>
                <response_lid ident="TF01">
                    <render_choice>
                        <response_label ident="T">
                            <material><mattext>True</mattext></material>
                        </response_label>
                        <response_label ident="F">
                            <material><mattext>False</mattext></material>
                        </response_label>
                    </render_choice>
                </response_lid>
                </presentation>
            </item>
        </section>
    </assessment>
</questestinterop>
```

The XML for the minimal multiple 'Assessment' is as follows:

```
<questestinterop>
    <assessment ident="IMS_V01_A_Example301">
        <section ident="IMS_V01_S_Example301">
            <item ident="IMS_V01_I_BasicExample301">
                <presentation label="RS01">
                    <material><mattext>Paris is the Capital of France ?</mattext></material>
                    <response_lid ident="TF01">
                        <render_choice>
                            <response_label ident="T">
                                <material><mattext>True</mattext></material>
                            </response_label>
                            <response_label ident="F">
                                <material><mattext>False</mattext></material>
                            </response_label>
                        </render_choice>
                </response_lid>
                </presentation>
            </item>
        </section>
    </assessment>
    <assessment ident="IMS_V01_A_Example302">
        <section ident="IMS_V01_S_Example302">
            <item ident="IMS_V01_I_BasicExample302">
                <presentation label="RS20">
                    <material><mattext>London is the Capital of Germany ?</mattext></material>
                    <response_lid ident="TF02">
                        <render_choice>
                            <response_label respident="T">
                                <material><mattext>True</mattext></material>
                            </response_label>
                            <response_label respident="F">
                                <material><mattext>False</mattext></material>
                            </response_label>
                        </render_choice>
                    </response_lid>
                </presentation>
            </item>
```

```
        </section>
    </assessment>
</questestinterop>
```

6.3.2 Full Definition

The following example consists of one Assessment with two Sections. One Section contains two Items and the other contains one Item.

```
<questestinterop>
    <assessment title="European Geography" ident="A01">
        <qticomment>A Complex Assessment example.</qticomment>
        <assessmentmetadata/>
        <objectives view="Candidate">
            <material><mattext>To test your knowledge of European geography.</mattext>
            </material>
        </objectives>
        <objectives view="Assessor">
            <material>
                <mattext>Tests the candidate's knowledge of European geography.</mattext>
            </material>
        </objectives>
        <assessprocessing>
            <qticomment>Processing of the final accumulated assessment.</qticomment>
            <scores>
                <decvar/>
            </scores>
            <scorecondition>
                <conditionvar>
                    <varlte respident="SCORE">9</varlte>
                </conditionvar>
                <displayfeedback feedbacktype="Response" linkrefid="Failed"/>
            </scorecondition>
            <scorecondition>
                <conditionvar>
                    <vargt respident="SCORE">10</vargt>
                </conditionvar>
                <displayfeedback feedbacktype="Response" linkrefid="Passed"/>
            </scorecondition>
        </assessprocessing>
        <assessfeedback title="Failed" ident="Failed">
            <material><mattext>You failed the test.</mattext></material>
        </assessfeedback>
        <assessfeedback title="Passed" ident="Passed">
            <material><mattext>You passed the test.</mattext></material>
        </assessfeedback>
        <section title="European Capitals" ident="S01">
            <sectionmetadata/>
            <objectives view="Candidate">
                <material>
                    <mattext>To assess your knowledge of the capital cities in Europe.
                    </mattext>
                </material>
            </objectives>
            <objectives view="Tutor">
                <material>
                    <mattext>
                        To ensure that the student knows the difference between the Capital cities
                        of France, UK, Germany, Spain and Italy.
                    </mattext>
                </material>
            </objectives>
            <item title="Capital of France" ident="I01" maxattempts="6">
                <itemrubric view="Candidate">
                    <material><mattext>Choose only one of the choices available.</mattext>
                    </material>
                </itemrubric>
                <presentation label="Resp001">
```

```
            <response_lid ident="LID01">
                <material><mattext>What is the Capital of France ?</mattext></material>
                <render_choice shuffle="Yes">
                    <response_label ident="LID01_A">
                        <material><mattext>London</mattext></material>
                    </response_label>
                    <response_label ident="LID01_B">
                        <material><mattext>Paris</mattext></material>
                    </response_label>
                    <response_label ident="LID01_C">
                        <material><mattext>Washington</mattext></material>
                    </response_label>
                    <response_label ident="LID01_D" rshuffle="No">
                        <material><mattext>Berlin</mattext></material>
                    </response_label>
                </render_choice>
            </response_lid>
        </presentation>
        <resprocessing>
            <qticomment/>
            <outcomes>
                <decvar vartype="Integer" defaultval="0"/>
            </outcomes>
            <respcondition>
                <qticomment>Scoring for the correct answer.</qticomment>
                <conditionvar>
                    <varequal respident="LID01">LID01_B</varequal>
                </conditionvar>
                <setvar action="Set" varname="SCORE">10</setvar>
                <displayfeedback feedbacktype="Response" linkrefid="I01_IFBK01"/>
            </respcondition>
        </resprocessing>
        <itemfeedback title="Correct answer" ident="I01_IFBK01">
            <material><mattext>Correct answer.</mattext></material>
        </itemfeedback>
        <itemfeedback ident="I01_IFBK02">
            <solution>
                <solutionmaterial>
                    <material>
                        <mattext>
                            London is the Capital of England.
                            Paris is the Capital of France.
                            Washington is in the USA.
                            Berlin is the Capital of Germany.
                        </mattext>
                    </material>
                </solutionmaterial>
            </solution>
        </itemfeedback>
        <itemfeedback ident="I01_IFBK03" view="All">
            <hint feedbackstyle="Multilevel">
                <hintmaterial>
                    <material><mattext>One of the choices is not in Europe.</mattext>
                    </material>
                </hintmaterial>
                <hintmaterial>
                    <material><mattext>Berlin is the Capital of Germany.</mattext>
                    </material>
                </hintmaterial>
                <hintmaterial>
                    <material>
                        <mattext>The Eiffel tower is in the Capital of France.</mattext>
                    </material>
                </hintmaterial>
            </hint>
        </itemfeedback>
    </item>
</section>
<section title="European Rivers" ident="SO2">
    <sectionmetadata/>
    <objectives view="Candidate">
```

```
            <material><mattext>To assess your knowledge of the rivers in Europe.</mattext>
            </material>
        </objectives>
        <objectives view="Assessor">
            <material><mattext>Questions on rivers in Germany, Spain, Italy and France.
                </mattext>
            </material>
        </objectives>
        <item title="Rivers in France question" ident="I02">
            <itemrubric view="Candidate">
                <material><mattext>Choose all of the correct answers.</mattext>
                </material>
            </itemrubric>
            <presentation label="Resp002">
                <response_lid ident="LID02" rcardinality="Multiple">
                    <material><mattext>Which rivers are in France ?</mattext>
                    </material>
                    <render_choice shuffle="Yes" minnumber="1" maxnumber="2">
                        <response_label ident="LID02_A">
                            <material><mattext>Seine</mattext></material>
                        </response_label>
                        <response_label ident="LID02_B">
                            <material><mattext>Thames</mattext></material>
                        </response_label>
                        <response_label ident="LID02_C">
                            <material><mattext>Danube</mattext></material>
                        </response_label>
                        <response_label ident="LID02_D">
                            <material><mattext>Loire</mattext></material>
                        </response_label>
                    </render_choice>
                </response_lid>
            </presentation>
        </item>
        <item title="Rivers in Germany” ident="I03"/>
            <duration> 0000:00:10T04:00:00 </duration>
            <itemrubric view="Candidate">
                <material><mattext>Choose all of the correct answers.</mattext>
                </material>
            </itemrubric>
            <presentation label="Resp003">
                <response_lid ident="LID03" rcardinality="Multiple">
                    <material>
                        <matimage imagtytpe=”image/gif” uri=”rivers.gif”></matimage>
                        <mattext>Which rivers are in Germany ?</mattext>
                    </material>
                    <render_hotspot x0="500" y0="500" height="200">
                        <response_label ident="LID03_A" rarea="Ellipse"> 10,10,2,2
                        </response_label>
                        <response_label ident="LID03_B" rarea="Ellipse"> 15,15,2,2
                        </response_label>
                        <response_label ident="LID03_C" rarea="Ellipse"> 30,30,2,2
                        </response_label>
                        <response_label ident="LID03_D" rarea="Ellipse"> 60,60,2,2
                        </response_label>
                        <response_label ident="LID03_E" rarea="Ellipse"> 70,70,2,2
                        </response_label>
                    </render_hotspot>
                </response_lid>
            </presentation>
        </item>
    </section>
</assessment>
</questestinterop>
```

This example is given in the example file:
‘IMS_QTIv1/Valid/Advanced/mchc_asmimr_106/mchc_asmimr_106.xml’.

6.4 The XML Example Files

The full set of example files, as referred to in Sections 4 and 6 are available as part of the Q&TI Resource Kit. These files are listed in Table 6.1 (correct basic examples), 6.2 (correct advanced examples), 6.3 (correct composite examples), 6.4 (examples with a single known syntactic error) and 6.5 (examples with a known semantic error). The files with known errors are supplied to provide test case failure conditions in the parsers and rendering systems. Each XML example directory contains the files necessary to support an example Assessment, Section and/or Item. The XML files are denote by an '.xml' extension (the adopted naming convention is described in Appendix C). The following tables list the name of each example directory, the nature of the example in terms of data structures i.e. Assessment (A), Section (S) and/or Item (I) and a brief description of the example.

Table 6.1 The Q&TI XML basic example (correct) files.

Directory Name	Nature	Description
ctpt_i_001	I(1)	Standard connect-the-points using logical identifier.
ctpt_i_002	I(1)	Standard connect-the-points using XY co-ordinates.
ctpt_ir_001	I(1)	Standard connect-the-points using logical identifier with response processing.
ctpt_ir_002	I(1)	Standard connect-the-points using XY co-ordinates with response processing.
dobj_ir_001	I(1)	Standard drag object source with image-based objects.
fibi_i_001	I(1)	Numerical (integer) using FIB rendering.
fibi_i_002	I(1)	Numerical question with SLIDER rendering.
fibi_ir_001	I(1)	Numerical (integer) using FIB rendering (as per fibi_i_001.xml) with new response processing.
fibi_ir_002	I(1)	Numerical question with SLIDER rendering (as per fibi_i_002.xml) with new response processing.
fibn_i_001	I(1)	Numerical (real) using FIB rendering.
fibn_ir_001	I(1)	Numerical (real) using FIB rendering (as per fibn_i_001.xml) with new response processing.
fibs_i_001	I(1)	Standard FIB (string) using FIB rendering.
fibs_i_002	I(1)	Multiple FIB (strings) using FIB rendering.
fibs_i_003	I(1)	Short answer (string) using FIB rendering.
fibs_ir_001	I(1)	Standard FIB (string) using FIB rendering (as per fibs_i_001.xml) with new response processing.
fibs_ir_002	I(1)	Multiple FIB (strings) using FIB rendering (as per fibs_i_002.xml) with new response processing.
fibs_ir_003	I(1)	Short answer (string) using FIB rendering (as per fibs_i_003.xml) with new response processing.
ihsp_i_001	I(1)	Standard IHS using XY rendering.
ihsp_ir_001	I(1)	Standard IHS using XY rendering (as per ihsp_i_001.xml) with new response processing.
mchc_i_001	I(1)	Multiple-choice (text) with CHOICE rendering.

Directory Name	Nature	Description
mchc_i_001	I(1)	A standard multiple-choice question with a FIB extension.
mchc_i_001	I(1)	A standard multiple-choice question with a FIB extension (as per mchc_i_001.xml) with new response processing.
mchc_i_002	I(1)	Multiple-choice (image) with CHOICE rendering.
mchc_i_003	I(1)	Multiple-choice (audio) with CHOICE rendering.
mchc_i_005	I(1)	Multiple-choice (image) with HOTSPOT rendering.
mchc_i_006	I(1)	Multiple-choice (text) with SLIDER rendering.
mchc_ir_001	I(1)	Multiple-choice (text) with CHOICE rendering and response processing.
mchc_ir_002	I(1)	Multiple-choice (image) with CHOICE rendering (as per mchc_i_002.xml) with response processing.
mchc_ir_003	I(1)	Multiple-choice (audio) with CHOICE rendering (as per mchc_i_003.xml) with response processing.
mchc_ir_004	I(1)	Multiple-choice (image) with HOTSPOT rendering.
mchc_ir_004	I(1)	Multiple-choice (image) with HOTSPOT rendering (as per mchc_i_004.xml) with response processing.
mchc_ir_005	I(1)	Multiple-choice (image) with HOTSPOT rendering (as per mchc_i_005.xml) with response processing.
mchc_ir_006	I(1)	Multiple-choice (text) with SLIDER rendering (as per mchc_i_006.xml) with response processing.
mrsp_i_001	I(1)	Multiple response (text) with CHOICE rendering.
mrsp_ir_001	I(1)	Multiple response (text) with CHOICE rendering (as per mrsp_i_001.xml) with response processing.
mtch_i_001	I(1)	Matching of items using a logical group.
mtch_ir_001	I(1)	Matching of items using a logical group with response processing.
oitm_i_001	I(1)	Partial ordering of events using logical identifier.
oitm_ir_001	I(1)	Partial ordering of events using logical identifier with response processing.
oobj_i_001	I(1)	Standard order objects (text) with proprietary (OBJECT) rendering.
oobj_i_002	I(1)	Standard order objects (image) with proprietary (OBJECT) rendering.
oobj_i_002	I(1)	Standard order objects (image) with proprietary (OBJECT) rendering (as per oobj_i_002.xml) with new response processing.
oobj_ir_001	I(1)	Standard order objects (text) with proprietary (OBJECT) rendering (as per oobj_i_001.xml) with response processing.
trfl_i_001	I(1)	True/false with CHOICE rendering.
trfl_ir_001	I(1)	True/false with CHOICE rendering and response processing

The directory for these files is: IMS_QTIv1/Valid/Basic/…

Table 6.2 The Q&TI XML advanced example (correct) files.

Directory Name	Nature	Description
mchc_ir_007	I(1)	Multiple-choice Chinese writing example - I with response processing.
mchc_ir_008	I(1)	Multiple-choice Chinese writing example - II with response processing.
fibs_ir_101a	I(1)	Example of using the logical AND/OR elements.
fibs_ir_101b	I(1)	Same example as fibs_ir_101a.xml but without the usage of AND/OR elements.
mchc_imr_101	I(8)	Various multiple-choice (text) questions with response processing.
mchc_ir_102	I(3)	Reading passage with 3 multiple-choice questions with response processing.
fibs_ir_102_prespext	I(1)	Essay with Human or Machine scoring options with response processing.
mchc_ir_103_prespext	I(1)	Multiple-choice (text) with two scoring models.
mchc_smimr_104	S(1)I(2)	A single Section with two multiple-choice (text) questions.
mchc_amsmim_105	A(2) [S(1)I(1)]	Two assessment each with one section and one item.
mchc_asmimr_106	A(1) [S(2) I(2) I(1)]	A complex single Assessment containing several multiple-choice questions.
mchc_amsmimr_105	A(2) [S(1)I(1)]	Two assessments each with one section and one item (as per mchc_amsmim_105.xml). Add the response processing parts.
mchc_ir_009	I(1)	Multiple-choice Chinese writing example - III with response processing.
mrsp.ir_001a	I(1)	Multiple response using ‘,rtf’ material with response processing.

The directory for these files is: IMS_QTIv1/Valid/Advanced/…

Table 6.3 The Q&TI XML composite example (correct) files.

Directory Name	Nature	Description
mrsp.ir_001b	I(1)	Multiple response using ‘,rtf’ material with alternative response processing.
mrsp.ir_001c	I(1)	Multiple response using ‘,rtf’ material with alternative response processing.

The directory for these files is: IMS_QTIv1/Valid/Composite/…

Table 6.4 The Q&TI XML example (syntactically incorrect) files.

Directory Name	Nature	Description
elmt_ir_001	I(1)	An invalid <material> element is used at the <item> level.
elmt_ir_002	I(1)	An invalid element is used within the <presentation> element.
elmt_ir_003	I(1)	An invalid element is used within the <response_lid> element.
elmt_ir_004	I(1)	An invalid element is used within the <response_xy> element.
elmt_ir_005	I(1)	An invalid element is used within the <response_str> element.
elmt_ir_006	I(1)	An invalid element is used within the <response_num> element.
elmt_ir_007	I(1)	An invalid element is used within the <response_grp> element.
elmt_ir_008	I(1)	An invalid element is used within the <outcomes> element.
elmt_ir_009	I(1)	An invalid element is used within the <itemfeedback> element.
elmt_ir_010	I(1)	More than one <presentation> element is used at the <item> level.
atrb_ir_001	I(1)	An invalid enumerated type is used for the *view* attribute with the <objective> element.
atrb_ir_002	I(1)	An invalid enumerated type is used for the *feedbackswitch* attribute with the <itemcontrol> element.
atrb_ir_003	I(1)	An invalid enumerated type is used for the *hintswitch* attribute with the <itemcontrol> element.
atrb_ir_004	I(1)	An invalid enumerated type is used for the *solutionswitch* attribute with the <itemcontrol> element.
atrb_ir_005	I(1)	An invalid enumerated type is used for the *rcardinality* attribute with the <response_lid> element.
atrb_ir_006	I(1)	An invalid enumerated type is used for the *rtiming* attribute with the <response_lxy> element.
atrb_ir_007	I(1)	An invalid enumerated type is used for the *feedbackstyle* attribute with the <solution> element.
atrb_ir_008	I(1)	An invalid enumerated type is used for the *feedbacktype* attribute with the <displayfeedback> element.
atrb_ir_009	I(1)	An invalid enumerated type is used for the *numtype* attribute with the <response_num> element.
atrb_ir_010	I(1)	An invalid enumerated type is used for the *areatype* attribute with the <varinside> element.

The directory for these files is: IMS_QTIv1/Invalid/Syntax/…

Table 6.5 The Q&TI XML example (semantically incorrect) files.

File Name	Nature	Description
elmt_ir_001	I(1)	An incorrect response_label is used as part of the standard multiple choice response processing.
elmt_ir_002	I(1)	An incorrect response_label is used as part of the standard true/false response processing.
elmt_ir_003	I(1)	Duplicated images are displayed as part of a multiple choice question using images.
atrb_ir_001	I(1)	The incorrect *rcardinality* attribute value is used for a standard multiple choice question.
atrb_ir_002	I(1)	The incorrect *rcardinality* attribute value is used for a standard multiple response question.
atrb_ir_003	I(1)	The incorrect *minnumber* attribute value is used for a standard multiple response question.
atrb_ir_004	I(1)	The incorrect *maxnumber* attribute value is used for a standard multiple response question.
elmt_ir_004	I(1)	The <response_label> values are incorrect for the rendered slider defined ranges.
elmt_ir_005	I(1)	The <response_label> values are incorrect for the rendered image and the associated hot-spot.
elmt_ir_006	I(1)	The response processing values are incorrect for the rendered slider defined ranges.

The directory for these files is: IMS_QTIv1/Invalid/Semantic/…

7. Implementation Guidance

7.1 Assessments

7.1.1 Elements and their Attributes

Assessmentmetadata

The *assessmentmetadata* element is used to contain the description of the Assessment hosting the declaration. A wide range of meta-data entities are available and these are either optional or mandatory depending on their nature. These entities will be used by the distributed learning systems to ascertain where the Assessment is useful/relevant to the target environment.

Objectives

The *objectives* element should be used to define the objectives of the Assessment 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.

Assessmentcontrol

The *assessmentcontrol* element should be used to define the default conditions for the display of different types of feedback to users. The Assessment level definitions of the *feedbackswitch*, *hintswitch* and *solutionswitch* take precedence if no lower level definition is encountered i.e. within a Section or Item. This means that the Assessment level definition acts as the default state.

Assessprocessing

At present the Assessment processing reflects the model employed within Items. Sum of scores is the only explicitly support scoring aggregation. This area will be developed in V2.0 of the specification.

7.1.2 Groups of Elements

Selection & Sequencing

The selection and sequencing of Sections is supported through two mechanisms, both of which are for further study in V2.0. The first is a pre-selection mechanism using the *sectionselection* and *sectionsequence* elements that define the Sections selected from the pool of available Sections and the order in which those Sections are activated respectively. The second is an invocation mechanism that uses the *sectionprecondition* and *sectionpostcondition* elements to confirm that the section can be activated at the moment of activation.

Variable Manipulation

In this version of the specification the scoring model at the Assessment level, more formally termed Assessment Accumulated Processing, is incomplete (this will be rectified in V2.0). The current approach is identical in nature to that at the Section and Item levels. The appropriate scoring variables are declared using the decvar element and grouped in the scores element. A series of tests are then applied (within the conditionvar element) to the responses identified through the *respident* attributes. This means that the Assessment scoring mechanism is directly tied to the responses, as is the case for Item level response processing. The later algorithms will not use such a directly coupled approach.

7.2 Sections

7.2.1 Elements and their Attributes

Sectionmetadata

The *sectionmetadata* element is used to contain the description of the Section hosting the declaration. A wide range of meta-data entities are available and these are either optional or mandatory depending on their nature. These entities will be used by the distributed learning systems to ascertain where the Section is useful/relevant to the target environment.

Objectives

The *objectives* element should be used to define the objectives of the Section 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.

Sectioncontrol

The *sectioncontrol* element should be used to define the default conditions for the display of different types of feedback to users. The Section level definitions of the *feedbackswitch*, *hintswitch* and *solutionswitch* take precedence if no lower level definition is encountered i.e. within an Item. In the case of a clash with Assessment level definitions the Section level ones take precedence.

Sectionprocessing

At present the Section processing reflects the model employed within Items. Sum of scores is the only explicitly support scoring aggregation. This area will be developed in V2.0 of the specification.

7.2.2 Groups of Elements

Selection & Sequencing

The process for selecting and sequencing Sections within a Section is as per the Assessment mechanism. A similar approach is used when selecting and sequencing Items. The selection and sequencing of Items is supported through the pre-selection mechanism (using the *itemselection* and *itemsequence* elements that define the Items selected from the pool of available Sections and the order in which those Sections are activated respectively) and an invocation mechanism (using the *itemprecondition* and *itempostcondition* elements to confirm that the Item can be activated at the moment of activation).

7.3 Items

7.3.1 Elements and their Attributes

Itemmetadata

The *itemmetadata* element is used to contain the description of the Item hosting the declaration. A wide range of meta-data entities are available and these are either optional or mandatory depending on their nature. These entities will be used by the distributed learning systems to ascertain where the Item is useful/relevant to the target environment.

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.

Itemcontrol

The *itemcontrol* element should be used to define the default conditions for the display of different types of feedback to users. The Item level definitions of the *feedbackswitch*, *hintswitch* and *solutionswitch* take precedence over all other levels of definition.

7.3.2 Groups of Elements

Response Types

There are five basic response types, as listed in Section 3.2 of the IMS Q&TI Information Model. It is important to realise that the response-type is determined by the manner in which the response is to be internally processed. A single item can have more than one response-type i.e. to create a composite response-type. If a new type of response is identified then this can be added using the proprietary extension element *response_extension*. The classical multiple-choice and multiple response question types can all be 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. If a new type of rendering is identified then this can be added using the

proprietary extension element *render_extension*. The classical multiple-choice and multiple response 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. The itemfeedback element can contain multiple sets of hints, solutions and standard feedback responses. Differentiation between these is achieved using the *feedbacktype* and *feedbackstyle* attributes. The former defines the type of feedback to be displayed (hint, solution or response) while the latter indicates how the feedback material is to be revealed e.g. incrementally, etc.

Variable Manipulation

The manipulation of the scoring variables declared in the *outcomes*/*decvar* combination is contained within the *conditionvar* element. The variable comparisons are made individually using the elements defined as *var****; the state of these comparisons can be inverted using the logical 'NOT' element. The analysis of the period of the response activity is supported using the *dur**** elements (an associated default variable is not assumed). This mechanism will be developed further in V2.0 of the specification. The combination of the individual *var**** elements is possible using two techniques:

- Implicit - the sequence of *var**** elements within a *conditionvar* element is by definition an 'AND' condition. The usage of multiple *conditionvar* elements is also treated as an 'AND' condition. The 'OR' condition is achieved through the use of multiple *respcondition* elements. The sequence of these is equivalent to a logical inclusive 'OR' condition;
- Explicit - the usage of the logical *and* and *or* elements that combine the outcomes of each separate comparison and combine them in one consolidated state declaration for the *conditionvar* element.

It is recommended that the Implicit approach be used whenever possible. This approach results in more interoperable code. Examples of both approaches are given in the files:
IMS_QTIv1/Valid/Advanced/fibs_ir_101a.xml (usage of the *and*/*or* elements) and
IMS_QTIv1/Valid/Advanced/fibs_ir_101b.xml (implicit approach).

The response processing variables are declared using the *decvar* element. Each implementation of the QTI 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 are two special conditions that need to be trapped:

- When the response has not been answered - this can be supported using two techniques namely the *unanswered* element or the *response_na* element. The *unanswered* element is placed within *conditionvar* and is activated whenever that response has not been attempted. *The response_na* element is a proprietary extension facility that is hosted within the each of the rendering elements i.e. *render_choice*, *render_hotspot*, etc.
- When none of the conditions within the *conditionvar* element are satisfied and so some catch-all state needs to be declared. This default state is captured using the *other* element. When included in the *conditionvar* then 'true' will be returned if none of the other conditions have been invoked (this does not mean being invoked when all of the tests return the false state or when no response is offered).

7.4 Aggregated Scoring and Response Processing

The QTI specifications support scoring at three levels:

- Item - response processing (this is fully formed and functional);
- Section - section processing (at present this is similar to the Item response-processing and as such does not support aggregated processing/scoring dependent on more than one Item or Section). A more complete model for Section-level processing will be addressed in Version 2.0;
- Assessment - assessment processing (at present this is similar to the Section-processing and as such does not support aggregated processing/scoring dependent on more than one Item or Section). A more complete model for Assessment-level processing will be addressed in Version 2.0.

The response processing for Items is rich in features and as such it can support a very wide range of response-types. In the QTI Information Model the cardinality of the responses is described in Figure 3.1 and consists of Single, Multiple and Ordered responses. The relationship between these, the five core response-types (*response_lid*, etc.) and the types of question is shown in Table 3.1 of the same QTI Information Model. This gives rise to the data set as described in Section 4.2 of the QTI Information Model, that must be held internally by the test engine. It is this sequence of responses that is then applied to the response processing to determine the correctness or otherwise of the response, the corresponding scoring and the subsequent feedback (if any).

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

The response processing test on line 36 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 6 of the *rcardinality* attribute i.e. ‘Single’.

The next stage is to consider a multiple response Item in which five responses are required. The question is "Which days are NOT the week-end ?". The resulting QTI XML code is shown below:

```
1       <questestinterop>
2           <item title="Single response" ident="A">
3               <presentation label="BasicExample002a">
4                   <material><mattext>Which days are NOT the week-end ?</mattext>
5                   </material>
6                   <response_lid ident="MCb_01" rcardinality="Multiple" rtiming="No">
7                       <render_choice>
8                           <response_label ident="A">
9                               <material><mattext>Saturday</mattext></material>
10                          </response_label>
11                          <response_label ident="B">
12                              <material><mattext>Monday</mattext></material>
13                          </response_label>
14                          <response_label ident="C">
15                              <material><mattext>Wednesday</mattext></material>
16                          </response_label>
17                          <response_label ident="D">
18                              <material><mattext>Tuesday</mattext></material>
19                          </response_label>
20                          <response_label ident="E">
21                              <material><mattext>Sunday</mattext></material>
22                          </response_label>
23                          <response_label ident="F">
24                              <material><mattext>Friday</mattext></material>
25                          </response_label>
26                          <response_label ident="G">
27                              <material><mattext>Thursday</mattext></material>
28                          </response_label>
29                      </render_choice>
30                  </response_lid>
31              </presentation>
32              <resprocessing>
33                  <outcomes><decvar/></outcomes>
34                  <respcondition title="Correct">
35                      <conditionvar>
36                          <varequal respident="MCb_01">B</varequal>
37                          <varequal respident="MCb_01">C</varequal>
38                          <varequal respident="MCb_01">D</varequal>
39                          <varequal respident="MCb_01">F</varequal>
40                          <varequal respident="MCb_01">G</varequal>
41                      </conditionvar>
42                      <setvar action="Set" >1</setvar>
43                      <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
44                  </respcondition>
45              </resprocessing>
46              <itemfeedback ident="Correct" view="Candidate">
47                  <material><mattext>Yes, you are right.</mattext></material>
48              </itemfeedback>
49          </item>
50      </questestinterop>
```

The response processing test on lines 36-40 (inc.) do the check to see if the responses are 'Monday', 'Tuesday', 'Wednesday', 'Thursday' and 'Friday', the correct answer. The system is aware that several responses are required because of the value in line 6 of the *rcardinality* attribute i.e. 'Multiple'. The key point to note is that the set of responses must be checked against the list of responses without relying on the order of the responses or the sequence in which the tests are applied - remember this is a 'Multiple' response only and NOT an 'Ordered' one. In the *conditionvar* element (lines 35-41) all of the tests MUST be true for the *setvar* condition to be triggered (line 42).

The final stage is to consider an ordered response Item in which seven responses in the right order are required. The question is "Give the correct order of the days of the week starting with Sunday ?". The resulting QTI XML code is shown below:

```
1       <questestinterop>
2           <item title="Single response" ident="A">
3               <presentation label="BasicExample002a">
4                   <material><mattext>Which days are NOT the week-end ?</mattext>
5                   </material>
6                   <response_lid ident="MCb_01" rcardinality="Ordered" rtiming="No">
7                       <render_choice>
8                           <response_label ident="A">
9                               <material><mattext>Saturday</mattext></material>
10                          </response_label>
11                          <response_label ident="B">
12                              <material><mattext>Monday</mattext></material>
13                          </response_label>
14                          <response_label ident="C">
15                              <material><mattext>Wednesday</mattext></material>
16                          </response_label>
17                          <response_label ident="D">
18                              <material><mattext>Tuesday</mattext></material>
19                          </response_label>
20                          <response_label ident="E">
21                              <material><mattext>Sunday</mattext></material>
22                          </response_label>
23                          <response_label ident="F">
24                              <material><mattext>Friday</mattext></material>
25                          </response_label>
26                          <response_label ident="G">
27                              <material><mattext>Thursday</mattext></material>
28                          </response_label>
29                      </render_choice>
30                  </response_lid>
31              </presentation>
32              <resprocessing>
33                  <outcomes><decvar/></outcomes>
34                  <respcondition title="Correct">
35                      <conditionvar>
36                          <varequal respident="MCb_01">E</varequal>
37                          <varequal respident="MCb_01">B</varequal>
38                          <varequal respident="MCb_01">D</varequal>
39                          <varequal respident="MCb_01">C</varequal>
40                          <varequal respident="MCb_01">G</varequal>
41                          <varequal respident="MCb_01">F</varequal>
42                          <varequal respident="MCb_01">A</varequal>
43                      </conditionvar>
44                      <setvar action="Set" >1</setvar>
45                      <displayfeedback feedbacktype="Response" linkrefid="Correct"/>
46                  </respcondition>
47              </resprocessing>
48              <itemfeedback ident="Correct" view="Candidate">
49                  <material><mattext>Yes, you are right.</mattext></material>
50              </itemfeedback>
51          </item>
52      </questestinterop>
```

The response processing tests on lines 36-42 (inc.) do the check to see if the responses are ‘Sunday’, ‘Monday’, ‘Tuesday’, ‘Wednesday’, ‘Thursday’, ‘Friday’ and ‘Saturday’, the correct answer. The system is aware that several responses are required in the correct order because of the value in line 6 of the *rcardinality* attribute i.e. 'Ordered'. The key point to note is that the set of responses must be checked against the list of responses in the order of the responses and the sequence in which the tests are applied - the test system must ensure that each response is tested for only once and with respect to the correctly sequenced test. In the *conditionvar* element (lines 35-43) all of the tests MUST be true for the *setvar* condition to be triggered (line 44).

7.5 Meta-data

The QTI has its own set of specific meta-data that is contained within the data structures themselves i.e. the ASIs. This meta-data has been given its own set of names, all of which start with the characters 'qmd_' thereby ensuring there is no clash with any of the IMS Meta-data, IEEE LOM or Dublin Core elements. These QTI specific mete-data elements are designed to add value to the current IMS Meta-data Specifications; both techniques should be used to complement each other.

7.6 Naming Conventions

7.6.1 Identities and Labels

Many elements within the Q&TI specifications require unique identities and labels. Uniqueness is particularly important when importing from multiple sources. To support unique naming we propose the following 32 character convention is adopted by all organisations creating assessments, sections and items. Namely:

ABCDEFGH_V**_?_***…***

- 'ABCDEFGH' denotes the 1-8 character code for the organisation responsible for creating the names;
- 'V**' is the three letter code identifying the version of the specification to which the names conform;
- '?' is either 'A' for *Assessment*, 'S' for *Section* or 'I' for *Item*;
- '***…***' is the 1-17 character code for the name itself.

An example of this for an assessment identifier created by 'ETS' under the Q&TI v1.0 specification is:

ETS_V01_A_TESTSECTIONv001[2]

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

7.6.2 Proprietary Extensions

There are several places where proprietary extensions are made possible. It would be useful if an appropriate naming convention for these extensions can be agreed and followed by the different organisations that will wish to make them.

It is proposed that all of the proprietary elements created having the following naming convention[3]:

x_abcdefgh_v**_lmn…xyz

Where:

- 'x_' denotes the fact that the element is a proprietary extension;
- 'abcdefgh' is the 1-8 character code for the organisation responsible for creating the proprietary extension (this would be the same as the code assigned for the identifier and label naming convention);
- 'v**' is the three character code identifying the version of the specification to which the extension conform;
- 'lmn…xyz' denotes the name given to the proprietary extension itself.

An example of the naming convention for a new rendering element within Q&TI version 1.0 by Question Mark Corporation becomes:

<render_extension (x_qusmarkc_v01_synthspeech)>[4]

7.7 Scoping Rules

7.7.1 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&TI specification this global uniqueness was considered too constraining and so the scoping rules listed in Table 7.1 should be followed whenever possible:

[2] Note that the usage of the 'ETS' identifier is a fictional allocation used for the purposes of the example.
[3] The elements should conform to the W3C XHTML specification in which all elements and attributes use lower case characters.
[4] Note that the usage of the 'QUSMARKC' identifier is a fictional allocation used for the purposes of the example.

Table 7.1 Scoping rules for identifiers.

Identifier Name	Role of the Identifier	Scoping Rule
assessfeedback	Identifier for each of the feedback responses given to the users.	Unique within the assessment.
section	Identifier for each Section.	Unique across the Assessment and for all sections within a Section. Should be unique to the organisation creating it.
item	Identifier for each Item.	Unique across the Assessment and Sections which contain or reference it. Should be unique to the organisation creating it.
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 *must* be unique across all of the five *response_*** elements.
response_num	Identifier for the response_num response-type.	This identifier is used to link the response supplied by the user and the corresponding response processing. This identifier *must* be unique across all of the five *response_*** elements.
response_str	Identifier for the response_str response-type.	This identifier is used to link the response supplied by the user and the corresponding response processing. This identifier *must* be unique across all of the five *response_*** elements.
response_xy	Identifier for the response_xy response-type.	This identifier is used to link the response supplied by the user and the corresponding response processing. This identifier *must* be unique across all of the five *response_*** elements.
response_grp	Identifier for the response_grp response-type.	This identifier is used to link the response supplied by the user and the corresponding response processing. This identifier *must* be unique across all of the five *response_*** elements.
response_label	Identifier for each of the possible responses to be made by the user.	This identifier is used as part of the *respcondition* element processing within each *resprocessing* element. This identifier must be unique within the *render_*** element it is used.
mattext	Identifier for text-based material.	This allows the text to be defined once and then referenced from any other location using the *matref* element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.
matimage	Identifier for image-based material.	This allows the image to be defined once and then referenced from any other location using the *matref* element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.
mataudio	Identifier for audio-based material.	This allows the audio to be defined once and then referenced from any other location using the

Identifier Name	Role of the Identifier	Scoping Rule
		matref element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.
matvideo	Identifier for video-based material.	This allows the video to be defined once and then referenced from any other location using the *matref* element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.
matapplication	Identifier for application-based material.	This allows the application to be defined once and then referenced from any other location using the *matref* element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.
matapplet	Identifier for applet-based material.	This allows the applet to be defined once and then referenced from any other location using the *matref* element. This identifier must be unique to all of the *mat**** elements used internally or from the Section and Item pool.

8. Proprietary Extensions

The proprietary extensions facilities listed in Table 8.1 are supported as elements within the specifications:

Table 8.1 List of proprietary extension elements.

Extension Element Name	Host Element	Description
assessproc_extension	assessment	Inclusion for alternative Accumulated Assessment level response processing as distinct from that supported in the defined assessprocessing element.
condition_extension	sectionprocessing, assessprocessing	Inclusion for alternative Section and Assessment condition processing models.
itemproc_extension	item	Inclusion for alternative Item level response processing algorithms as distinct from that supported in the defined *resprocessing* element.
mat_extension	material	Inclusion for styles of material that cannot be support using the supplied elements of *mattext*, *matvideo*, *matimage*, *mataudio*, *matapplet* and *matapplication*.
render_extension	response_lid, response_xy, response_str, response_num, response_grp.	Inclusion for styles of Item rendering that cannot be supported using the current elements: *render_choice*, *render_hotspot*, *render_slider* and *render_fib*.
respcond_extension	resprocessing	An alternative to the defined *respcondition* element that is responsible for the conditions applied to the activated response.
response_extension	presentation	Inclusion of an alternative Response format as distinct from the elements defined: *response_lid*, *response_xy*, *response_str*, *response_num*, *response_grp*.
response_na	render_choice, render_hotspot, render_slider, render_fib.	Inclusion of an alternative processing mechanism to the *unanswered* element.
scorecondition_extension	scorecondition	Inclusion of an alternative scoring process that can be used by Sections and Assessments.
sectionproc_extension	section	Inclusion for alternative Section level accumulated processing as distinct from that supported in the defined *sectionprocessing* element.
var_extension	conditionvar	Inclusion of alternative condition tests that can be applied to the user responses for an Item.

Note:

These elements are only used if the suppliers of the ASIs require proprietary features that are not supported by the available range of elements. It is recommended that these elements are used sparingly. Backwards compatibility with proprietary extensions will NOT be guaranteed in future versions of these specifications.

9. V2.0 Issues & Compatibility

The elements listed in Table 9.1 are used to indicate where new functionality will be added in V2.0 of the specifications:

Table 9.1 List of V2.0 specific elements.

Extension Element Name	Host Element	Description
durequal	conditionvar	Equivalence test for the time taken to produce the response.
durgt	conditionvar	Greater than test for the time taken to produce the response.
durgte	conditionvar	Greater than or equal to test for the time taken to produce the response.
durlt	conditionvar	Less than test for the time taken to produce the response.
durlte	conditionvar	Less than or equal to test for the time taken to produce the response.
itemfeedback	item	Presentation of Item level feedback, particularly for Hints and Solutions.
itempostcondition	item	Postconditions to be applied to test the conditions under which the next Item can be activated.
itemprecondition	item	Preconditions to be applied to test if the Item can be activated.
itemselection	section	The mechanism by which the appropriate Items can be selected from the range of Items.
itemsequence	section	The mechanism by which the selected Items are sequenced for activation.
sectionfeedback	section	Presentation of Section level feedback.
sectionpostcondition	section	Postconditions to be applied to test the conditions under which the next Section can be activated.
sectionprecondition	section	Preconditions to be applied to test if the Section can be activated.
sectionprocessing	section	Section level response processing.
sectionselection	assessment	The mechanism by which the appropriate Sections can be selected from the range of Sections.
sectionsequence	assessment	The mechanism by which the selected Sections are sequenced for activation.

Note:

The structure of these elements will change in V2.0 of these specifications. Their main role is to indicate the type of functions to be included in later releases of these specifications and as such vendors are encouraged NOT to make use of these in V1.0 implementations.

9.1 Function Requirements

The functional developments to be considered in V2.0 are:

- The provision of a full results reporting method. This results reporting method will consider the return of the Item, Section and Assessment level results, the selection and responses supplied to each of the Assessments, Sections and Items, and the appropriate tracking information that will provide state recovery features;
- An integrated scoring model across the assessment, section and item data structures. This model will allow a variety of aggregation methods between Sections and Items, and Assessments and Sections. No numeric scoring methods and the corresponding feedback will also be considered;
- Full provision for static and dynamic selection and sequencing of sections and items. This will enable each Assessment to select and sequence the Sections within it, each Section to select and sequence the Sections, and Section to select and sequence the Items within it. Dynamic selection and sequencing will be supported i.e. the next Section/Item is dependent on the user responses to the previous Section/Item;
- The completion of the V1.0 functions that were removed for further study. For example, this includes the provision of the <style> element and consideration of some of the extensions that have been developed within the examples.

The results reporting method is a way of describing the results of Items, Sections and/or Assessments in XML. In particular it will communicate the response made by the user answering each Item and the scores or other responses processing performed.

9.2 Constraints

The constraints to be applied to the v2.0 specifications are:

- The core functions of the v1.0 specifications will be maintained and will be compatible with the new functionality supplied within v2.0. The exceptions are:
  - The extension features described within v1.0
  - The v2.0 features described in v1.0

  The only changes to the V1.0 DTD for Items will be the addition of optional tags for the V2.0 features. Software that is compliant with the V1.0 DTD will be able to import V2.0 Items providing it ignores the optional tags.

  Items exported from software compliant with the V1.0 DTD will be able to be imported without change into software that is compliant with the V2.0 spec.
- Harmonisation of the IMS QTI with the other IMS specifications, in particular Meta-data, Content & Packaging and Profiles. Harmonisation with specifications from other organisations such as the ADL will also be addressed;
- Items as defined and implemented under the IMS QTI V1.0 specification MUST operate in the same manner in the V2.0 specifications. No alteration or new pre-processing must be required.

10. Conformance

The purpose of this conformance statement is to provide a mechanism for customers to fairly compare vendors of assessment tools and content. It is not required for a vendor to support every feature to claim conformance, however, a vendor must detail their level of conformance with a "Conformance Statement". For example vendors may choose to supply:

- Data files (source materials), export features, import features and/or repackaging (the collation of multiple files) features;
- Data files and/or tools that support items, sections and/or assessments.

10.1 Valid Data Issues

Vendors claiming conformance shall produce, parse, provide and/or "pass through" valid Q&TI data as defined by the DTD including proprietary extensions where applicable. Vendors claiming their system tools export Q&TI shall export valid Q&TI data. Vendors claiming their system tools import Q&TI data shall be able to parse and recognize valid Q&TI data. Vendors claiming their system tools repackage Q&TI data shall be able "pass through" valid Q&TI data whether the tool recognises the optional elements or not. Vendors claiming their assessment content conforms to this specification shall provide valid Q&TI data.

10.2 Conformance Statement

Vendors claiming conformance must provide a "Conformance Statement", detailing their level of conformance, substantially similar to the information shown below, upon a reasonable request from a member of the IMS, a prospective customer(s).

Table 10.1 Conformance functionality grid.

Functionality		**Level of Support**				
		Data File	Import	Export	Repackage	Explanation of optional functionality supported.
Assessment level element support	*Mandatory*	Y or N	Y or N	Y or N	Y or N	
	Optional	Y or N	Y or N	Y or N	Y or N	
Section level element support	*Mandatory*	Y or N	Y or N	Y or N	Y or N	
	Optional	Y or N	Y or N	Y or N	Y or N	
Item level element support	*Mandatory*	Y or N	Y or N	Y or N	Y or N	
	Optional	Y or N	Y or N	Y or N	Y or N	Optional elements listed here.
	Response types	Y or N	Y or N	Y or N	Y or N	Response types listed here, such as multiple choice, multiple response, string response, numeric, xy co-ordinates, etc.
	Material	Y or N	Y or N	Y or N	Y or N	e.g. text, video, audio, image, applet, application, alternative material, referenced material.
	Scoring	Y or N	Y or N	Y or N	Y or N	e.g. scoring models.
	Feedback	Y or N	Y or N	Y or N	Y or N	

Appendix A – QTI DTDs & XDRs

A1 Overview

The Version 1.0 IMS Question & Test Interoperability Document Type Definitions (DTDs) and XML Data Representation (XDRs) files are contained in a directory that has:

- **xmla** - the directory that contains all of the DTDs and XDRs in native XML Authority format. XML Authority (v1.1e) is a product supplied by eXtensibility Inc;
- **mac** - the directory that contains all of the DTDs and XDRs in text file format. These text files are designed for usage with Macintosh systems;
- **ibm**- the directory that contains all of the DTDs and XDRs in text file format. These text files are designed for usage with PC systems;
- **unix**- the directory that contains all of the DTDs and XDRs in text file format. These text files are designed for usage with Unix systems.

The further directory structure under each of these directories is identical. This further structure is :

- dtds
  - QTIcorencdtd - non-commented core DTD
  - QTIfulldtd - full commented DTD
  - QTIfullncdtd - full non-commented DTD
  - QTIItemncdtd - non-commented Item specific DTD
  - QTISectionncdtd - non-commented Section specific DTD
- xdrs
  - QTIcorencxdr - non-commented core XDR
  - QTIfullxdr - full commented XDR
  - QTIfullncxdr - full non-commented XDR
  - QTIItemncxdr - non-commented Item specific XDR
  - QTISectionncxdr - non-commented Section specific XDR

Within each of the DTD directories is the file: **IMS_QTIv1p0.dtd** and within each XDR directory is the file **IMS_QTIv1p0.xdr**.

This approach means that's the different types of DTD/XDR can be applied without requiring any editing of the associated source XML files. The full directory structure is given in Appendix A of this file.

A2 Features of the Different DTDs/XDRs

The key features of the different DTD/XDR implementations are:

- QTIfulldtd/xdr - this is the full DTD/XDR with all of the corresponding comments. The comments correspond to those given in the IMS QTI XML Binding Specification v1.0;
- QTIfullncdtd/xdr - this is the full DTD/XDR but with all of the comments removed. This makes the file smaller and it is easier to see the internal structure;
- QTIcorencdtd/xdr - the core features of the DTD/XDR are as per the full versions but with ALL extension and Version 2.0 specific elements removed. This will ensure that the 'xml' files using this DTD/XDR will be compatible with future releases of the specification;

- QTISectionncdtd/xdr - only those core elements that are used by the Section data structure are available. This means that ALL Assessment specific elements have been removed. This DTD/XDR is a further refinement of the core features;
- QTIItemncdtd/xdr- only those core elements that are used by the Item data structure are available. This means that ALL Assessment and Section specific elements have been removed. This DTD/XDR is a further refinement of the core features;

A3 Recommended Usage of the DTDs/XDRs

The recommended uses of the different DTDs/XDRs are:

- Select the set of XDRs/DTDs that suit your system. All of the mac/unix/ibm text versions are derived from the XML Authority version and created using the Alpha text processing application;
- In most case only the non-commented versions need to be used. The fully commented files are intended to be informative;
- The core set should be used if you are concerned with compatibility with later versions of the specification. However, this will prohibit the usage of proprietary extension features. The Version 2.0 specific elements are there to show where new developments to the specification are intended and you are recommended to AVOID using any xml files that require their inclusion;
- The Section and Item specific DTDs/XDRs should be used if you intend to import/export only Sections and/or Items. These simpler files structures will also make it easier to understand the structure of the full and core DTDs/XDRs.

A4 Full Directory Structure

The full directory structure is:

xmla
- dtds
  - QTIcorencdtd
    - IMS_QTIv1p0.dtd
  - QTIfulldtd
    - IMS_QTIv1p0.dtd
  - QTIfullncdtd
    - IMS_QTIv1p0.dtd
  - QTIitemncdtd
    - IMS_QTIv1p0.dtd
  - QTIsectionncdtd
    - IMS_QTIv1p0.dtd
- xdrs
  - QTIcorencxdr
    - IMS_QTIv1p0.xdr
  - QTIfullxdr
    - IMS_QTIv1p0.xdr
  - QTIfullncxdr
    - IMS_QTIv1p0.xdr
  - QTIitemncxdr
    - IMS_QTIv1p0.xdr
  - QTIsectionncxdr
    - IMS_QTIv1p0.xdr

mac
- dtds
  - QTIcorencdtd
    - IMS_QTIv1p0.dtd
  - QTIfulldtd
    - IMS_QTIv1p0.dtd
  - QTIfullncdtd
    - IMS_QTIv1p0.dtd
  - QTIitemncdtd
    - IMS_QTIv1p0.dtd
  - QTIsectionncdtd
    - IMS_QTIv1p0.dtd
- xdrs
  - QTIcorencxdr
    - IMS_QTIv1p0.xdr
  - QTIfullxdr
    - IMS_QTIv1p0.xdr
  - QTIfullncxdr
    - IMS_QTIv1p0.xdr
  - QTIitemncxdr
    - IMS_QTIv1p0.xdr
  - QTIsectionncxdr
    - IMS_QTIv1p0.xdr

unix

- dtds
  - QTIcorencdtd
    - IMS_QTIv1p0.dtd
  - QTIfulldtd
    - IMS_QTIv1p0.dtd
  - QTIfullncdtd
    - IMS_QTIv1p0.dtd
  - QTIitemncdtd
    - IMS_QTIv1p0.dtd
  - QTIsectionncdtd
    - IMS_QTIv1p0.dtd
- xdrs
  - QTIcorencxdr
    - IMS_QTIv1p0.xdr
  - QTIfullxdr
    - IMS_QTIv1p0.xdr
  - QTIfullncxdr
    - IMS_QTIv1p0.xdr
  - QTIitemncxdr
    - IMS_QTIv1p0.xdr
  - QTIsectionncxdr
    - IMS_QTIv1p0.xdr

ibm

- dtds
  - QTIcorencdtd
    - IMS_QTIv1p0.dtd
  - QTIfulldtd
    - IMS_QTIv1p0.dtd
  - QTIfullncdtd
    - IMS_QTIv1p0.dtd
  - QTIitemncdtd
    - IMS_QTIv1p0.dtd
  - QTIsectionncdtd
    - IMS_QTIv1p0.dtd
- xdrs
  - QTIcorencxdr
    - IMS_QTIv1p0.xdr
  - QTIfullxdr
    - IMS_QTIv1p0.xdr
  - QTIfullncxdr
    - IMS_QTIv1p0.xdr
  - QTIitemncxdr
    - IMS_QTIv1p0.xdr
  - QTIsectionncxdr
    - IMS_QTIv1p0.xdr

Appendix B – Glossary of Terms

B1 General Terms

adaptive testing	A sequential form of individual testing in which successive items in the test are chosen based primarily on the psychometric properties and content of the items and the test taker's response to previous items.
ADL	The Advanced Distributed Learning group was started by the United States White House in 1997 which aims to advance the use of online training.
AICC	Aviation Industry CBT Committee is a membership-based international forum that develops recommendations on interoperable learning technologies.
answer key	The key that describes the scoring scenario for a question or test.
arrange objects response	A response style in which the test taker arranges one of more objects.
assessment	Any systematic method of obtaining evidence from tests, examinations, questionnaires, surveys and collateral sources used to draw inferences about characteristics of people, objects, or programs for a specific purpose.
assessment engine	The process that supports the evaluation of the responses to produce scores and feedback.
authoring system	A generic name for one or more computer programs that allow a user to author, and edit items (i.e. questions, choices, correct answer, scoring scenarios and outcomes) and maintain test definitions (i.e. how items are delivered with a test).
battery	A set of tests standardized on the same population, so that norm-referenced scores on the several tests can be compared or used in combination for decision making.
bilingual	The characteristic of being relatively proficient in two languages.
candidate	A person that participates in a test, assessment or exam by answering questions.
candidate data repository	The database of candidate specific information.
certification	A form of credentialing, usually used to refer to voluntary credential not involving governmental sanction. See also *licensing*.
certification processing	The process of matching an individual's accomplishments against the requirements for a certification program, and awarding certifications when all requirements have been met.
character set	The characters used by a computer to display information.
choice	One of the possible responses that a test taker might select. Choices contain the correct answer/s and distracters.
composite response	A combination of response styles presented within a single item.
composite score	A score that combines several scores by a specified formula.
computerized adaptive test	An adaptive test administered by computer. See *adaptive test.*
conditional measurement error variance	The variance of measurement efforts that affect the scores of examinees at a specified test score level; the square of the conditional standard error of measurement.
conformance statement	A conformance statement provides a mechanism for customers to fairly compare vendors of assessment tools and content.

construct response item	An exercise for which examinees must create their own responses or products rather than choose a response from an enumerated set.
cut score	A specified point on a score scale, such that scores at or above that point are interpreted differently from scores below that point. Sometimes there is only one cut score, dividing the range of possible scores into "passing" and "failing" or “mastery” and “nonmastery” regions. Sometimes two or more cut scores may be used to define three or more score categories, as in establishing performance standards. See also, *performance standards.*
database	A collection of information/data, often organized within tables, within a computer's mass storage system. Databases are structured in a way to provide for rapid search and retrieval by computer software. The following databases are used by testing systems; item, test definition, scheduling and results.
delivery channel	One of more testing centers, usually managed by a delivery provider (i.e. an organization that provides candidate scheduling services, computers, proctoring services, and the space in which to conduct a computerized test).
delivery provider	An organization that provides candidate scheduling services, computers, proctoring services, and the space in which to conduct a computerized test.
distracter	One of the choices, that a test taker may select, that is not the correct answer.
difficulty	A statistical property, sometimes known as facility, indicating the level of a question, from 0.0 to 1.0. Calculated as the average score for the question divided by the maximum achievable score. A facility of 0.0 means that the question is very hard (no-one got it right) and 1.0 means that it is very easy (no-one got it wrong). 0.5 ideal.
drag and drop response	A response style where the test taker indicates their choice by dragging an image from one place to another.
DTD	Document Type Definition.
dynamic sequencing	The sequencing of items or sections is based upon previous responses from a test taker.
element	An XML term that defines a component within an XML document that has been identified in a way a computer can understand.
element contents	An XML term used to describe the the content of the element.
element attributes	Provides additional information about an element.
essay response	A response style where the test taker enters an essay in response to the stimulus.
facility	A statistical property, indicating the level of a question, from 0.0 to 1.0. Calculated as the average score for the question divided by the maximum achievable score.
feedback	Information provided to a participant to aid the learning process.
fill-in-the-blank(s)	A response style where the test taker completes a phrase by entering a word, words or a number.
gain score	The difference between the score on a test and the score on an earlier administration of the same or an equivalent test.
grade equivalent score	The school grade level for which a given score is the real or estimated median or mean.
high-stakes test	A test whose result has important, direct consequences for examinees, program, or institutions tested.
holistic scoring	A method of obtaining a score on a test, or a test item, that results from an overall

	judgement of performance using specified criteria.
hotspot response	A response style where the test taker indicates their selection by using a mouse or pointing device on a graphic display.
IEEE	Institute of Electrical and Electronics Engineers that provides a forum for developing specifications and standards.
image hotspot response	A response style where the test taker indicates their selection by using a mouse or pointing device on a graphic display.
IMS	An organization dedicated to developing specifications for distributed learning.
intelligence test	A psychological or educational test designed to measure intellectual processes in accord with some evidence-based theory of intelligence.
invigilator	A person who proctors a test.
item	The questions, choices, correct answer, scoring scenarios and outcomes used within a test.
item analysis	The process of studying the responses to questions delivered in the pilot study or prototype in order to select the best questions in terms of facility and discrimination.
item pool	The aggregate of items from which a test or test scale's items are selected during test development, or the total set of items from which a particular test is selected for test taker during adaptive testing.
item prompt	The question, stimulus, or instructions that direct the efforts of examinees in formulating their responses to a constructed-response exercise.
item response theory (IRT)	A theory of test performance that emphasises the relationship between mean item score (P) and level (0) of the ability or trait measured by the item. In the case of an item scored 0 (incorrect response) or 1 (correct response), the mean item score equals the proportion of correct responses. In most applications, the mathematical function relating P to 0 is assumed to be a logistic function that closely resembles the cumulative normal distribution.
job analysis	Any of several methods of identifying the tasks performed on a job or the knowledge, skills, abilities, and other personal characteristics relevant to job performance.
licensing	The issuing, usually by a government agency, of a credential indicating competence in some profession or client-centered activity. See also *certification*.
logical identifier	A category of response styles that presents various choices and provides a mechanism for the test taker to select one or more choices.
logical group	A category of response styles that allows a test taker to group objects together to indicate their choice.
low-stakes test	A test whose result has only minor or indirect consequences for examinees, programs, or institutions tested.
LTSC	Learning Technology Standards Committee
LMS	Learning Management System which is the system responsible for the management of the learning experience.
mandated tests	Tests that are administered because of a mandate from an external authority.
mastery test	A test designed to indicate that the test taker has or has not mastered some domain or knowledge or skill. Mastery is generally indicated by a passing score or cut score. See *cut score*.

mean	Arithmetic average of some scores, i.e. the sum of the scores divided by the number of scores.
meta-data	Tags that described the content of the associated data.
multiple choice	A response style where the test taker selects one choice from several to indicate their opinion as to the correct answer.
multiple response	A response style where the test taker selects more than one choice from several to indicate their opinion as to the correct answers. Multiple response questions have answer keys that describe various combinations of choices being right or wrong with different possible outcome for the different combination of selections.
normalized standard score	A derived test score in which a numerical transformation has been chosen so that the score distribution closely approximates a normal distribution, for some specific population.
numeric response	A response style where the test taker enters a number to indicate their choice
outcome	The event that will occur after a question or questions have been answered (i.e. the item is scored, feedback is provided, etc.)
outcome evaluation	The activity of a practitioner that evaluates the efficacy of an intervention.
participant	A person that participates in a testing, assessment or survey process by answering questions.
participant mean	The mean of the percentage score achieved by candidates. Used to determine validity of choices, within an item, by examining the choices selected by the higher and/or lower scoring candidates.
platform	The computing environment that hosts the assessment system.
pilot test	A test administered to a representative sample of test takers solely for the purpose of determining the properties of the test. See *field test*.
pretest	An administration of test items to a representative sample of test takers solely for the purpose of determining the characteristics of the item.
proctor	A person who invigilates a test.
program evaluation	The collection of systematic evidence to determine the extent to which a planned set of procedures obtains particular effects.
psychometric	Properties of the item(s) and test(s) such as the distribution of item difficulty and discrimination indices.
psychometrician	A qualified person who analyses the psychometrics of a test or item.
publish test	To release a test from the development system to the production or release system.
questionnaire	One or more questions presented and answered together.
Question and Test (Q&TI)	The formal title for the IMS team dealing with specification for Question (item) and Test (assessment) Interoperability.
reliability	The degree to which the scores of every individual are consistent over repeated applications of a measurement procedure and hence are dependable, and repeatable; the degree to which scores are free of errors of measurement.
rendering	The process by which an item is presented on a computer screen.
respondent	A person that participates in a survey process by answering questions
response processing	The process of evaluating the test takers responses.
response type	The method by which the test taker provides their answer to the question.

scheduling system	The generic name for one or more computer programs that allows a user to track candidate appointments. Scheduling systems may also provide bill collection information, testing centre resource scheduling and candidate demographics.
score	Any specific number resulting from the assessment of an individual; a generic term applied for convenience to such diverse measures as test scores, estimates of latent variables, production counts, absence records, course grades, ratings, and so forth.
scoring formula	The formula by which the raw score on a test is obtained. The simplest scoring formula is "raw score equals number correct." Other formulas differentially weight item responses, sometimes in an attempt to correct for guessing or non-response, by assigning zero weights to non-responses and negative weights to incorrect responses.
scoring protocol	The established criteria, including rules, principles, and illustrations, used in scoring responses to individual items and clusters of items. The term usually refers to the scoring procedures for assessment tasks that do not provide enumerated responses from which test-takers make a choice.
scoring rubric	The principles, rules, and standards used in scoring an examinee performance, product, or constructed response to a test item. Scoring rubrics vary in the degree of judgement entailed, in the number of distinct score levels defined, in the latitude given scorers for assigning intermediate or fractional score values, and in other ways.
section	A collection of items (generated either statically or dynamically) normally focused at a particular objectives.
selection response	A response style where the test taker selects from a pull-down list.
sequence response	A response style where the test taker orders a list of objects or text to formulate their response.
speeded test	A test in which performance is measured primarily or exclusively by the time to perform a specified task, or the number of tasks performed in a given time, such as tests of typing speed and reading speed.
speededness	A test characteristic, dictated by the test's time limits, that result in a test-taker's score being dependent on the rate at which work is performed as well as the correctness of the responses.
standard deviation	A statistical measure of the spread of results. The higher the standard deviation, the greater the spread of data.
standards-based assessment	Assessments intended to represent systematically described content and performance standards.
static sequencing	The sequencing of items or sections is fixed and does not vary with previous responses from a test taker.
string response	A category of response styles that allows the test taker to enter text and/or numbers.
technical manual	A publication prepared by test authors and publishers to provide technical and psychometric information on a test.
test center	A facility that provides computers and proctoring services in which to conduct tests.
test center administration system	The generic name for one or more computer programs used by a test center to administer tests to candidates. This may include, but is not limited to, starting tests, stopping tests and communicating item, test and results data back and forth.
test developer	The person(s) or agency responsible for the construction of a test and for the documentation regarding its technical quality for an intended purpose.

test development	The process through which a test is planned, constructed, evaluated and modified, including consideration of content, format, administration, scoring, item properties, scaling, and technical quality for its intended purpose.
test development system	A generic name for one or more computer programs that allow a user to author, and edit items (i.e. questions, choices, correct answer, scoring scenarios and outcomes) and maintain test definitions (i.e. how items are delivered with a test).
test driver	A generic name for one or more computer programs that displays test items to a computer screen, collects candidate's responses score and stores the results.
test taker	The participant or candidate taking a test.
test sponsor	The person(s) or agency responsible for the choice and administration of a test, the interpretation of test scores produced in a given context, and for any decisions or actions that are based, in part, on test scores.
topic	The subject matter of a question.
true/false	A response style where the test taker selects from two choices, one labeled “true” and one “false”.
validation	The process of investigation by which the validity of the proposed interpretation of test scores is evaluated.
validity	An overall evaluation of the degree to which accumulated evidence and theory support specific interpretations of test scores.
W3C	World Wide Web Consortium.
weight	The number of points awarded for a given response.
weighted scoring	A method of scoring a test in which the number of points awarded for a correct (or diagnostically relevant) response is not the same for all items in the test. In some cases, the scoring formula awards more points for one response to an item than for another.
word response	A response style where the test taker enters a word to indicate their choice.
XML	Extensible Mark-up Language is a specification, produced by the World Wide Web Consortium.
XY Co-ordinate	A category of response styles that presents an image, or various images, for the test taker to select a position on the image or images to indicate their choice.

B2 Q&TI Elements and Attributes

action The *action* attribute is used by the *setvar* element to define the manner in which the identified variable is to be modified. The potential actions are: set, add, subtract, divide and multiply.

altmaterial The *altmaterial* 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 *altmaterial* can support the same types as the *material* element it would normally be conservative in manner i.e. typically consist of text only as an alternative to video, etc.

and The *and* element provides the mechanism by which multiple condition comparison elements (e.g. *varequal*, *varlt*, etc.) can be combined in the form of a logical 'and' condition. If each comparison element returns a 'True' state then the composite returned state is 'True' otherwise it is 'False'.

apptype The *apptype* attribute is used by the *matapplication* element to define the type of application to be invoked. The format of the string field for the *apptype* attribute is 'application/***' (the MIME format is to be used).

areatype The *areatype* attribute is used by the *varinside* element to define the type of area within which the location of the co-ordinate is to be tested. Three types of area are defined: ellipse (including the circle), rectangle (including the square) and a general bounded area.

assessfeedback The *assessfeedback* element is the sub-element of *assessment* that is responsible for containing the feedback material that is to be presented to the users. This element will be further developed in V2.0 when a clearer model of assessment level scoring has been developed.

assessment The *assessment* element is one of the three primary interoperable data structures. An Assessment is equivalent to a test and as such it contains all of the information necessary to present, process and feedback information relevant to the Assessment. Assessments can be imported/exported.

assessmentcontrol The *assessmentcontrol* element is responsible for establishing the state of the available types of feedback with respect to the Assessment. Each of the *hintswitch*, *feedbackswitch* and *solutionswitch* attributes can be set as the default condition within the enclosed/referenced Sections and Items.

assessmentmetadata This is the container for the meta-data descriptions of the corresponding Assessment. This meta-data is QTI specific and should be used in conjunction with the standard IMS Meta-data Specifications.

assessproc_extension The *assessproc_extension* element is a sub-element of the *assessment* element that provides a proprietary extension capability for alternatives to the supplied assessment level processing element *assessprocessing*.

assessprocessing This element enables the inclusion of proprietary alternative Accumulated Assessment level response processing models. It is contained within the *assessment* element. The sub-elements should adopt the naming convention described in Section 7.6.

audiotype This attribute is used in the *MATAUDIO* element to define the type of audio file. The file type is assumed to take the form of 'audio/***' where '***' is indicative of the file source e.g. 'aicc' etc (the MIME format is to be used). The mapping of this to an actual file type is beyond the scope of this specification and will be vendor specific.

case	The *case* attribute is used to define if the response evaluation processing tests have to be case sensitive or not.
charset	The *charset* 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.
columns	The *columns* attribute is used by the *render_fib* element to indicate the maximum number of text columns within which a user can type their answer. This attribute is normally used when a free-form short answer or essay type of response is expected.
condition_extension	This element enables the inclusion of proprietary alternative Section and Assessment condition processing models. It is contained within the *sectionprocessing* element. The sub-elements should adopt the naming convention described in Section 7.6.
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 *and* and *or* elements whereas the implicit approach assumes the elements contained with the *conditionvar* element are combined using the logical AND.
continue	The *continue* attribute is used to control the flow of the response processing for each of the data structures. It is applied to the *respcondition* and *scorecondition* elements. When the attribute is set as 'No' then the condition processing terminates if that condition has been evaluated as 'True'.
decvar	The *decvar* 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 and default value of the variables is determined by the *varname*, *vartype* and *defaultval* attributes.
defaultval	The *defaultval* attribute is used by the *decvar* element to define the start value for the variable. This value is bounded by the *minvalue* and *maxvalue* attributes, when used. The type of value is determined by the type of variable declared i.e. it should conform to the *vartype* attribute as used by the *decvar* element.
displayfeedback	The *displayfeedback* element is the trigger for the presentation of feedback to the users. The type of feedback to be displayed is determined by the *feedbacktype* attribute. The *linkrefid* attribute is used to identify the element containing the feedback.
duration	The *duration* element is used within the *item*, *section* and *assessment* elements to define the permitted duration for the enclosed activity. The duration is defined as the period between the activity's activation and completion. The ISO 8601 format is used: YYYY:MM:DDTHH:MM:SS.
durequal	The *durequal* element is the equivalence test for the time taken to produce the response. This element will be further developed as part of the V2.0 specification.
durgt	The *durgt* element is the greater than test for the time taken to produce the response. This element will be further developed as part of the V2.0 specification.
durgte	The *durgte* element is the greater than or equal to test for the time taken to produce the response. This element will be further developed as part of the V2.0 specification.
durlt	The *durlt* element is the less than test for the time taken to produce the response. This element will be further developed as part of the V2.0 specification.
durlte	The *durlt* element is the less than or equal to test for the time taken to produce the

response. This element will be further developed as part of the V2.0 specification.

embedded The *embedded* attribute is used by the *material* and *mat**** 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.

encoding The *encoding* attribute is used to define the type of character encoding used for the entered text under the *render_fib* element. The default value is 'UTF-8'.

feedbackstyle The *feedbackstyle* attribute is used by the *solution* and *hint* elements to denote the type of feedback information contained within the corresponding element. The range of values are complete (default), incremental, multilevel and proprietary. These values inform the display system on how to use the material e.g. multilevel means that several alternative complete hints/solutions are available whereas incremental means that one solution in several distinct parts is available.

feedbackswitch The *feedbackswitch* attribute is used to define the permission state of the response feedback in the associated data structure i.e. Item, Section and Assessment. This switch is used within the *itemcontrol*, *sectioncontrol* and *assessmentcontrol* elements.

feedbacktype The *feedbacktype* attribute is used by the *displayfeedback* element to denote the type of feedback that is to be invoked. The possible values for the *feedbacktype* 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.

fibtype The *fibtype* attribute is used with the *render_fib* element to denote the type of response expected from the user. The range of values is: String, Integer, Decimal and Scientific. The last three are types of number input whereas the first denotes free format text entry.

height The *height* attribute is used to denote the total y-axis size, in pixels, of the material being presented. This attribute is normally used by the *matimage* element but is available to other *mat**** elements. If this attribute is not used then the system should determine the y-axis size from the defined x-axis size (given by the *width* 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.

hint The *hint* element is the container within the *itemfeedback* element that contains all of the information relevant to hints that can be revealed to the users. The nature of the hints is defined according to the *feedbackstyle* attribute and the content for each type of hint is contained within the *hintmaterial* sub-element.

hintmaterial The *hintmaterial* element is the container for the actual hint content presented in the enclosed *material* sub-elements. Each *hint* can contain multiple *hintmaterial* sub-elements and the manner in which this sequence is treated is defined according to the *feedbackstyle* attribute.

hintswitch The *hintswitch* attribute is used to define the permission state of the availability of hint feedback to the associated data structure i.e. Item, Section and Assessment. This switch is used within the *itemcontrol*, *sectioncontrol* and *assessmentcontrol* elements.

ident The *ident* 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 1-32 characters of length.

imagtype	This attribute is used in the *matvideo* element to define the type of video file. The file type is assumed to take the form of 'video/***' where '***' is indicative of the file source e.g. 'mpeg' 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.
interpretvar	The *interpretvar* element is used to describe statistical features about the associated variable. The variable that is associated is identified by the *varname* attribute. The primary usage for this element is to enable descriptions about the significance of the variables to be associated with their declaration - statistical parameters will be of particular importance.
item	The *item* element is one of the three primary interoperable data structures. An Item is a combination of the question, the rendering form, the processing of the subsequent response and the corresponding feedback (including hints and solutions). Items can be imported/exported.
itemcontrol	The *itemcontrol* element is responsible for establishing the state of the available types of feedback with respect to the Item. Each of the *hintswitch*, *feedbackswitch* and *solutionswitch* attributes can be set.
itemfeedback	The itemfeedback 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. *hint* for hints, *solution* for solutions and *material* for responses. The *view* attribute is used to define the actors to whom the material can be displayed.
itemmetadata	This is the container for the meta-data descriptions of the corresponding Item. This meta-data is QTI specific and should be used in conjunction with the standard IMS Meta-data Specifications.
itempostcondition	The *itempostcondition* element is responsible for determining if the next selected and sequenced Item can be activated i.e. the current postconditions are true. This element will be developed in the V2.0 specification.
itemprecondition	The *itemprecondition* element is responsible for determining if the selected and sequenced Item can be activated i.e. its preconditions are true. This element will be developed in the V2.0 specification.
itemproc_extension	This element enables the inclusion of proprietary alternative Item level response processing models as distinct from that supported in the *resprocessing* element. It is contained within the *item* element. The sub-elements should adopt the naming convention described in Section 7.5.
itemref	The *itemref* element enables Items that have not been included in the associated Assessment/Section data structures to be 'pulled into scope'. This means that Items not explicitly defined can be utilised. The *linkrefid* attribute is used to identify the associated Item.
itemrubric	The *itemrubric* element is a specialised form of the *material* element available as a sub-element to *item*. This provides a mechanism to supply content that can be used to describe a range of information about the Item as a whole. The content is made available as defined through the *View* attribute.
itemselection	The *itemselection* element contains the instructions on how to statically/dynamically select the Items from those that are either defined or referenced in the containing Assessment/Sections. This element will be developed in the V2.0 specification.
itemsequence	The *itemsequence* element contains the instructions that sequence the items selected under the *itemselection* element. This element will be developed in the

	V2.0 specification.
label	The *label* 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 *labelrefid* attribute is used by the *response_label* element to allow the different response labels to be linked to more complex scoring models.
linkrefid	The *linkrefid* attribute is used to associate Sections, 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.
lowerbound	The *lowerbound* attribute is used by the *render_slider* element to define the lowest value to be displayed by the slider rendering engine.
mat_extension	This is the extension facility defined as a sub-element under *material*. It can be used to support proprietary types of content to be presented to users as distinct from that already supported i.e. *mattext*, *matimage*, *mataudio*, *matvideo*, *matapplet* and *matapplication*.
matapplet	This element is responsible for the presentation of a JAVA applet. Further study is required in V2.0 to address issues such as run time implications.
matapplication	This element is responsible for the presentation of executable content. The type of application must be described using the *apptype* attribute. It is always a sub-element of *material*.
mataudio	This element is responsible for the presentation of audio content. The type of audio must be described using the *audiotype* attribute. It is always a sub-element of *material*. Further study is required in V2.0 to address issues such as sampling rate.
material	This is the container for all types of content that is to be presented to the users. Each *material* element can have multiple types of content as defined by *mattext*, *matimage*, *mataudio*, *matvideo*, *matapplet* and *matapplication*.
matimage	This element is responsible for the presentation of graphics/image content. The type of image must be described using the *imagtype* attribute. It is always a sub-element of *material*.
matref	The *matref* 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 utilised. The *linkrefid* 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 *texttype, charset and encoding* attributes. It is always a sub-element of *material*.
matvideo	This element is responsible for the presentation of video content. The type of video must be described using the *videotype* attribute. It is always a sub-element of *material*. Further study is required in V2.0 to address issues such as framing and control panel.
maxattempts	The *maxattempts* attribute is used by the *item* element to denote the maximum number of attempts that a user can have at answering the item. The action once this number is exceeded is not defined by this specification.
maxchars	This attribute is used by the *render_fib* element to define the maximum number of

	characters that a user can enter as their response to the FIB question posed. This mechanism is one way of defining the maximum amount of information that can be submitted by a user.
maxnumber	The *maxnumber* attribute is used to limit the maximum number of responses that a user can enter for a particular response - the response-type must have been defined as either multiple or ordered using the *rcardinality* attribute. One example of using the *maxnumber* attribute is in limiting the number of permitted answers to a multiple response question.
maxvalue	The *maxvalue* attribute is used with the *decvar* element to define the maximum value that can be assigned to the declared variable. This also limits the maximum value that can be assigned as part of the processing.
members	The members attribute is use by the decvar element to define the members of the set being defined. The members of the set will be enclosed by '{}' and will be separated by commas.
minnumber	The *minnumber* attribute is used to state the minimum number of responses that a user must enter for a particular response - the response-type will usually have been defined as either multiple or ordered using the *rcardinality* attribute. One example of using the *minnumber* attribute is in defining the minimum number of answers required from a multiple response question.
minvalue	The *minvalue* attribute is used with the *decvar* element to define the minimum value that can be assigned to the declared variable. This also limits the minimum value that can be assigned as part of the processing.
not	The *not* element provides the mechanism by which the result of a comparison elements (e.g. *varequal*, *varlt*, etc.) can be inverted. If the comparison element returns a 'True' state then the *NOT* returned state is 'False' and vice-versa.
numtype	The *numtype* attribute is used with the *response_num* element to define the type of number to be input by the user. The potential types are: Integer, Decimal and Scientific. The default type is Integer.
objectives	This *objectives* element is the container for the description of the objectives of the Assessments, Sections and Items - the context is defied by the usage. These objectives are defined with respect to the actor as defined by the *view* attribute.
or	The *or* element provides the mechanism by which multiple condition comparison elements (e.g. *varequal*, *varlt*, etc.) can be combined in the form of a logical 'or' condition. If at least one comparison element returns a 'True' state then the composite returned state is 'True' otherwise it is 'False'.
orientation	The *orientation* attribute is used with the *render_slider* element to define whether it is rendered vertically or horizontally. The actual graphic used in either case is question-engine specific.
other	The *other* element is a condition test, a sub-element within the *conditionvar* element, used to denote any other condition i.e. a response (otherwise the *unanswered* element would be used) not specifically evaluated in the condition.
outcomes	The *outcomes* element is used to contain the declaration of the variables for the Item-level response processing. The *outcomes* element contains the *decvar* and *interpretvar* sub-elements to define the scoring variables.
presentation	The *presentation* 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.
prompt	The *prompt* attribute is used with the *render_fib* element to define the type of input prompt to be rendered to the user. The types of prompt available are: Box

	(surrounding the area for text input), Dashline (underlining the text input position with one line per character), Asterisk (denoting the location for each character) and Underline (underlining the position for input).
qmd_absolutescore	This QTI specific meta-data element is used to define the maximum and minimum absolute scores that can be obtained.
qmd_assessmenttype	This QTI specific meta-data element is used to define the nature of the Assessment i.e. Examination, Survey, Tutorial, Self-assessment.
qmd_computerscored	This QTI specific meta-data element is used to define if the Item can be scored automatically.
qmd_feedbackpermitted	This QTI specific meta-data element is used to define if feedback is available (Yes/No).
qmd_hintspermitted	This QTI specific meta-data element is used to define if hints are available (Yes/No).
qmd_itemselection	This QTI specific meta-data element is used to define if Item selection is available (Yes/No) in the Assessment and/or Section.
qmd_itemsequence	This QTI specific meta-data element is used to define if Item sequencing is available (Yes/No) in the Assessment and/or Section.
qmd_itemtype	This QTI specific meta-data element is used to define the type of the Item e.g. Multiple choice, connect-the-points, etc.
qmd_levelofdifficulty	This QTI specific meta-data element is used to define the level of difficulty in terms of the anticipated educational level e.g. K-12, vocational training, ec.
qmd_material	This QTI specific meta-data element is used to define the nature of the material that is used within the data structures e.g. video, images, text, etc.
qmd_maximumscore	This QTI specific meta-data element is used to define the maximum score that can be obtained from that Item.
qmd_numberofitems	This QTI specific meta-data element is used to define the number of Items contained within the Section and/or Assessment.
qmd_renderingtype	This QTI specific meta-data element is used to define the type of rendering used by the Item i.e. render_choice, render_hotspot, render_slider and render_fib.
qmd_responsetype	This QTI specific meta-data element is used to define the type of responses expected from the user i.e. Single, Multiple and Ordered.
qmd_scoretype	This QTI specific meta-data element is used to define the type of scoring available at the Assessment i.e. Absolute, Percentage, Unscored and Multidimensional.
qmd_scoringpermitted	This QTI specific meta-data element is used to define if scoring is available (Yes/No).
qmd_sectionselection	This QTI specific meta-data element is used to define if Section selection is available (Yes/No) in the Assessment and/or Section.
qmd_sectionsequence	This QTI specific meta-data element is used to define if Section sequencing is available (Yes/No) in the Assessment and/or Section.
qmd_sectionsincluded	This QTI specific meta-data element is used to define if the Section contains other Sections or references to other Sections.
qmd_solutionspermitted	This QTI specific meta-data element is used to define if solutions are available (Yes/No).
qmd_status	This QTI specific meta-data element is used to define the status of the Item i.e.

	Experimental, Normal and Retired.
qmd_timedependence	This QTI specific meta-data element is used to define the if the responses are time dependent (Yes/No).
qmd_timelimit	This QTI specific meta-data element is used to define the number of minutes that the Assessment/Item will normally take to complete.
qmd_toolvendor	This QTI specific meta-data element is used to define the vendor of the tools creating the Assessment and/or Item.
qmd_topic	This QTI specific meta-data element is used to define the topic being addressed within the Item.
qmd_typeofsolution	This QTI specific meta-data element is used to define the type of solutions that are available i.e. Complete, Incremental, Multilevel and Proprietary.
qmd_weighting	This QTI specific meta-data element is used to define the weighting of the Item to be applied in aggregated scoring.
qticomment	This is the commenting facility within the XML schemas. The comments can take any form supported as #PCDATA. The key difference between this comment style and the standard '<!-- *** -->' is that the former is passed through the XML parser to the host system.
questestinterop	This is the root element. The top-most elements are either the *assessment*, *section* or *item* elements (excluding the comment element *qticomment*). Multiple Assessments, Section or Items are supported.
rarea	This attribute is used with the *response_label* element to define the area that is to act as a hot-spot. The area can take the form of an ellipse (including a circle), rectangle (including a square) and bounded area.
rcardinality	The *rcardinality* 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.
render_choice	The *render_choice* 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.
render_extension	This element enables the inclusion of proprietary alternative Item rendering methods that cannot be provided by the *render_choice*, *render_hotspot*, *render_slider* and *render_fib* elements. It is contained within the *response_lid*, *response_xy*, *response_str*, *response_num* and *response_grp* elements. The sub-elements should adopt the naming convention described in Section 7.6.
render_fib	The *render_fib* element is used to indicate that the response-type is to be presented to the user as a text entry. The specific attributes used to support this element are *fibtype*, *rows*, *columns*, *maxchars* and *prompt*.
render_hotspot	The *render_hotspot* element is used to indicate that the response-type is to be rendered as graphic with hotspots. This rendering mechanism is well-suited to responses in which the xy-coordinate is important. The attributes used with this element are *minnumber* and *maxnumber* to indicate the expected range of hotspots to be identified by the user, and *showdraw* that is used to display the user actions.
render_slider	The *render_slider* element is used to indicate that the response_type is to be presented as a slider graphic. This rendering mechanism is well-suited to numerical input or as an alternative to the classical multiple choice selection technique. The rendering is controlled by the *orientation*, *lowerbound*,

	upperbound, *step*, *startval*, *steplabel*, *minnumber* and *maxnumber* attributes.
respcond_extension	This element enables the inclusion of proprietary alternative *respcondition* elements that are responsible for conditions applied to the activated response. It is contained within the *resprocessing* element. The sub-elements should adopt the naming convention described in Section 7.6.
respcondition	The *respcondition* element is used as the container for the application of the conditions to the Item-level responses. It also contains the associated score elements (*setvar*) and the feedback trigger (*displayfeedback*). The *continue* attribute is used to denote if further *respcondition* elements should be processed.
respident	The *respident* attribute is used by all of the condition test elements (*varequal*, *varlt*, *varlte*, etc.) to identify the response-type as defined by its *ident* attribute. This identification acts as the scope for the *response_label*(s) to be considered by the condition tests.
response_extension	This element enables the inclusion of proprietary alternative response formats as distinct from the elements defined as *response_lid*, *response_xy*, *response_str*, *response_num*, *response_grp*. It is contained within the *presentation* element. The sub-elements should adopt the naming convention described in Section 7.6.
response_grp	The *response_grp* element is the response-type associated with the selection of responses that are used to denote relationships between two or more objects. This element is used when matching/pairing objects. The key point is that this element is used whenever the responses have to be processed as groups.
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: *rshuffle* (used for the *render_choice* element), *rarea* (used for the *render_hotspot* element), *rrange* (used for the *render_num* element), and *labelrefid*. The *response_label* element is available to all of the response-types and render-types.
response_lid	The *response_lid* 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.
response_na	This element is used to provide a proprietary extension facility within each of the rendering elements (*render_choice*, etc.). This element is used to define the proprietary processing of an unanswered response. The alternative mechanism is to use the *unanswered* attribute within the *conditionvar* processing.
response_num	The *response_num* element is the response-type associated with numerical-based questions. The user will be expected to supply a number in response to the question. This type of this number is defined by the *numtype* attribute and can be: integer, decimal or scientific.
response_str	The *response_str* element is the response-type associated with string-based questions. The user will be expected to supply a string in response to the question (this can range from a single word to a long essay). The length of the string expected is defined using a combination of the *rows*, *columns* and *maxchars* attributes.
response_xy	The *response_xy* element is the response-type associated with co-ordinate-based questions. The user will be expected to identify an xy-point on the display (using a mouse or other pointer) as the response to the question.
resprocessing	The *resprocessing* 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.

rows	The *rows* attribute is used by the *render_fib* element to indicate the maximum number of text rows within which a user can type their answer. This attribute is used when a free-form short answer or essay type of response is expected.
rrange	The *rrange* attribute is used with the *response_label* element to define the expected accuracy required of the response - it is enumerated as either Exact or Range. This could be used with the *response_num* element to indicate that the number had to be exact or within a particular range.
rshuffle	The *rshuffle* attribute is used with the *response_label* element in the context of the *response_choice* element. The *rshuffle* 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 *response_choice* element.
rtiming	This attribute is used with the response-type elements (*response_lid*, *response_xy*, 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.
scorecondition	The *scorecondition* element is used to contain the actual Assessment and Section-level condition testing and scoring. The current method is identical to that used at the Item-level and as such this approach may be too simple to support the required accumulated processing. Further work will be undertaken on this element in V2.0.
scorecondition_extension	This element enables the inclusion of proprietary alternative scoring processes that can be used by Sections and Assessments. It is contained within the *scorecondition* element. The sub-elements should adopt the naming convention described in Section 7.6.
scoremodel	The *scoremodel* attribute is used by the *resprocessing* and *assessprocessing* elements to identify that type of scoring model being used. The current default value is 'SumOfScores'. This attribute allows vendors to clearly identify when they are using proprietary scoring models.
scores	The *scores* element is used to contain the declaration of the variables for the Assessment and Section-level accumulated processing. The *scores* element is the Assessment/Section level equivalent of the *outcomes* element to Item response processing.
section	The *section* element is one of the three primary interoperable data structures. A Section is a group construct and as such may contain other Sections and one or more Items. Sections can be imported/exported.
sectioncontrol	The *sectioncontrol* element is responsible for establishing the state of the available types of feedback with respect to the Section. Each of the *hintswitch*, *feedbackswitch* and *solutionswitch* attributes can be set as the default condition within the enclosed/referenced Sections and Items.
sectionfeedback	The *sectionfeedback* attribute will be developed as part of the V2.0 specification. It will be used to display feedback that results from section level accumulated response processing.
sectionmetadata	This is the container for the meta-data descriptions of the corresponding Section. This meta-data is QTI specific and should be used in conjunction with the standard IMS Meta-data Specifications.
sectionpostcondition	The *sectionpostcondition* element is responsible for determining if the next selected and sequenced Section can be activated i.e. the current postconditions are

	true. This element will be developed in the V2.0 specification.
sectionprecondition	The *sectionprecondition* element is responsible for determining if the selected and sequenced Section can be activated i.e. its preconditions are true. This element will be developed in the V2.0 specification.
sectionproc_extension	This element enables the inclusion of proprietary alternative Section level response processing instead of the *sectionprocessing* element. It is contained within the *section* element. The sub-elements should adopt the naming convention described in Section 7.6.
sectionprocessing	The sectionprocessing element will be developed more thoroughly as part of the V2.0 specification. At present it uses a scheme similar to that adopted as the Item level processing. The type of scoring model is defined by the *scoremodel* attribute.
sectionref	The *sectionref* element enables Sections that have not been included in the associated Assessment/Section data structures to be 'pulled into scope'. This means that Sections not explicitly defined can be utilised. The *linkrefid* attribute is used to identify the associated Section.
sectionselection	The *sectionselection* element contains the instructions on how to statically/dynamically select the Items from those that are either defined or referenced in the containing Assessment/Sections. This element will be developed in the V2.0 specification.
sectionsequence	The *sectionsequence* element contains the instructions that sequence the Items selected under the *sectionselection* element. This element will be developed in the V2.0 specification.
setmatch	The *setmatch* attribute is used with the *varsubset* element to define the nature of the set-based comparison. The options for the attribute are Exact (the default) and Partial.
setvar	The *setvar* element is used to control the allocation of the Item level scoring as a consequence of the response condition processing within the *conditionvar* element. The scoring is controlled by the *action* attribute and the corresponding variable is identified through the *varname* attribute.
showdraw	The *showdraw* attribute is used by the *render_hotspot* element to control the display of the points selected by the user e.g. as required in connecting-the-points. The default value is 'No'.
shuffle	The *shuffle* attribute is used with the *response_choice* element. The *shuffle* attribute is enumerated as either Yes/No and it is used to denote whether the following *response_label* elements i.e. the allowed options to be chosen by the user, can be shuffled before the question is actually rendered and displayed.
solution	The *solution* element is the container within the *itemfeedback* element that contains all of the information relevant to solutions that can be revealed to the users. The nature of the solutions is defined according to the *feedbackstyle* attribute and the content for each type of hint is contained within the *solutionmaterial* sub-element.
solutionmaterial	The *solutionmaterial* element is the container for the actual solution content presented in the enclosed *material* sub-elements. Each *solution* can contain multiple *solutionmaterial* sub-elements and the manner in which this sequence is treated is defined according to the *feedbackstyle* attribute.
solutionswitch	The *solutionswitch* attribute is used to define the permission state for the availability of the solution feedback to the associated data structure i.e. Item, Section and Assessment. This switch is used within the *itemcontrol*,

	sectioncontrol and *assessmentcontrol* elements.
startval	The *startval* attribute is used by the *render_slider* attribute to define the start value for the slider graphic. This value must be between the lower and upper bounds of the slider and should be aligned to the increment value as defined by the *step* attribute.
step	The *step* attribute is used by the *render_slider* element to define the increment step of the axis. This increment should allow multiple points to be labelled between the lower and upper bounds supported by the slider.
steplabel	The *steplabel* attribute is used by the *render_slider* element to indicate whether or not the values on the slider should be displayed at all times (the value being indicated by the slider must be displayed).
texttype	This attribute is used in the *mattext* element 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.
title	The *title* 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 *unanswered* element is a sub-element of *conditionvar.* It is used to indicate that the state of the condition test should take into account the situation when the question is not attempted.
upperbound	The *upperbound* attribute is used by the *render_slider* element to define the maximum value to be displayed by the slider rendering engine.
uri	The *uri* attribute is used to identify an external resource i.e. typically a file name or URL.
var_extension	This element enables the inclusion of proprietary alternative condition tests that can be applied to the user responses of an Item. It is contained within the *conditionvar* element. The sub-elements should adopt the naming convention described in Section 7.5.
varcontains	The *varcontains* element is used to test if an entity is a member of a set of predefined entities. If the entities are based on text then the *case* attribute is used to state if the comparison should be case sensitive.
varequal	The *varequal* element is the equivalence test for the response identified by the accompanying *respident* attribute. If equivalent the value 'True' is returned. This element may be modified by the *not* element and could be enclosed within the *and* and *or* elements. The comparison value is contained as the #PCDATA for the element body.
vargt	The *vargt* element is the greater than test for the response identified by the accompanying *respident* attribute. If greater then the value 'True' is returned. This element may be modified by the *not* element and could be enclosed within the *and* and or elements. The comparison value is contained as the #PCDATA for the element body.
vargte	The *vargte* element is the greater than or equal to test for the response identified by the accompanying *respident* attribute. If greater than or equal to then the value 'True' is returned. This element may be modified by the *not* element and could be enclosed within the *and* and or elements. The comparison value is contained as the #PCDATA for the element body.
varinside	The *varinside* element is used to test if an xy co-ordinate is within a defined area.

	The area is defined by the *areatype* attribute and can be an ellipse, rectangle or irregular bounded area.
varlt	The *varlt* element is the less than test for the response identified by the accompanying *respident* attribute. If less than then the value 'True' is returned. This element may be modified by the *not* element and could be enclosed within the *and* and or elements. The comparison value is contained as the #PCDATA for the element body.
varlte	The *varlte* element is the less than or equal to test for the response identified by the accompanying *respident* attribute. If less than or equal then the value 'True' is returned. This element may be modified by the *not* element and could be enclosed within the *and* and or elements. The comparison value is contained as the #PCDATA for the element body.
varname	The *varname* attribute is used to define the name of the variable. This attribute is used by the *decvar* and *interpretvar* elements. The name can consist of any lowercase and uppercase characters i.e. a-z, A-Z, 0-9 and underscore (0-9 should not be used for the first character).
vartype	The *vartype* attribute is used by the *decvar* 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 *members* attribute that must accompany the declaration.
videotype	This attribute is used in the *matvideo* element to define the type of video file. The file type is assumed to take the form of 'video/***' where '***' is indicative of the file source e.g. 'mpeg' etc (the MIME format is to be used). The mapping of this to an actual file type is beyond the scope of this specification and will be vendor specific.
view	The view 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 *width* attribute is used to denote the total x-axis size, in pixels, of the material being presented. This attribute is normally used by the *matimage* element but is available to other *mat**** elements. If this attribute is not used then the system should determine the x-axis size from the defined y-axis size(given by the *height* 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.
x0	The *x0* attribute is used to define the 'x' co-ordinate location of the top left-hand corner of an image. This value is in terms of pixels and is expected to be in the range 0-799 i.e. 800 pixels for the x-axis.
y0	The *y0* attribute is used to define the 'y' co-ordinate location of the top left-hand corner of an image. This value is in terms of pixels and is expected to be in the range 0-599 i.e. 600 pixels for the y-axis.

Appendix C – Examples Information

C1 Proposed Naming Convention

A request has been made to introduce a naming convention for the QTI example files that is more informative. The request is for the name to reflect the nature of the content of the file.

A new naming convention is proposed. This convention is based upon two facets:

- Directory - this will reflect the class of the example;
- File - this will reflect the nature of the contents of the '.xml' file.

C1.1 Directory Naming

The directory naming convention is proposed to be (the bold words are the corresponding directory names):

IMS_QTIv1

Valid - Valid Examples

Basic - Basic Examples

Advanced - Advanced Examples

Composite - Composite Examples

Invalid - Invalid Examples

Syntax - Syntactically Incorrect Examples

Semantic - Semantically Incorrect Examples

C1.2 File Naming

The file naming can mow focus on the nature of the content in the files. The following convention is proposed:

[ABCD]_[A…Z]_[XYZ][*]_[<free form>]

where:

[ABCD] A four character string that reflects the type of question contained. The range of options includes -

mchc	multiple choice
mrsp	multiple response
fibi	integer fill-in-blank
fibn	numeric fill-in-blank (not integer)
fibs	string fill-in-blank
trfl	true/false
ihsp	image hotspot
ctpt	connect the points

[A…Z] A variable character string that describes the type of data structures contained. The basic convention is -

a	single assessment without score processing
ar	single assessment with score processing
am	multiple assessments with score processing
amr	multiple assessments with score processing
s	single section without score processing
sr	single section with score processing
sm	multiple section without score processing
smr	multiple section with score processing
i	single item without score processing
ir	single item with score processing

	im multiple item without response processing iir multiple item with response processing
	Any appropriate combination is used.
[XYZ]	A number in the range 001-999 (000 is reserved for later usage).
	The number of the example. This should follow some form of numbering system.
[*]	A single character in the range 'a' to 'z'.
	This is used to denote differences between files that are alternative solutions to the same Assessment/Section/Item.
[<free form>]	Free format that can be used to add any other descriptive information. It should be of alphanumeric form.

Index

E

F

H

I

M

O

P

Q

R

S

T

X