Using IMS Caliper Analytics™, Question and Test Interoperability™ and Learning Tools Interoperability™ with EPUB3™: EDUPUB Best Practices

Title: Using IMS Caliper, Question & Test Interoperability (QTI) and Learning Tools Interoperability (LTI) with EPUB3: EDUPUB Best Practices

Authors: Lisa Mattson (IMS Global, USA)  
Colin Smythe (IMS Global, UK)  
Stephen Vickers (IMS Global, UK)  
Thor Anderson (IMS Global, USA)  
Chris Vento (Intellify Learning, USA)  
Prashant Nayak (Intellify Learning, USA)

Version: DRAFT

Version Date: 6th June 2014

Release: 8.0

Status: Discussion Document

Summary: This document presents the best practice recommendations for using IMS Learning Tools Interoperability™ (LTI™), IMS Caliper Analytics™ and IMS Question & Test Interoperability™ (QTI™) based content within an EPUB3 context. These form a key part of the EDUPUB standard. Examples of these best practices are presented using the Readium open source EPUB3 reader but are generally applicable to other EPUB3 readers provided they follow the conformance requirements.
Executive Summary

This document contains the best practice recommendations for the use of the IMS Learning Tools Interoperability™ (LTI), IMS Caliper Analytics™ and IMS Question and Test Interoperability™ (QTI) with an International Digital Publishing Forum™ EPUB3™ formatted document. This combination of EPUB3 electronic book format with the QTI, LTI and Caliper standards is called the EDUPUB Standard. The use of LTI, QTI and Caliper provides the capability to use third party tools/apps/systems, quizzes for formative assessment and to log the raw data for the creation of learning analytics metrics respectively within the context of an EPUB3-based eBook.

This best practice document provides the key architectural use-cases that can be supported by the new functionality. A detailed description for the use of LTI, Caliper and QTI within EPUB3 is supplied along with annotated examples to demonstrate the key features. Examples and the source code (including the IMS fork of the Readium™ source code) are available at: http://www.imsglobal.org/edupub/. The key architectural use cases are:

a) The use of the eBook in a typical ‘consumer mode’ in which there is no source learning management system or remote learning analytics server. The eBook is used as a standalone entity and so the formative assessment/quiz must be contained in the eBook itself;

b) The use of the eBook as part of an organized learning activity moderated by some institutional services such as a learning management system and a learning analytics service/repository. In this case network access is, in general, required for access to the learning materials. The eBook can now contain links to external tools (launched using the IMS LTI protocol) and can use the Sensor API™ to report the learning analytics. Furthermore a quiz can be either embedded or accessed as a remote tools using LTI.

The key best practice recommendations detailed within this document are:

a) The eBook Reader, whether a specialist device, or software that runs on a laptop, tablet, etc. should be IMS LTI v1.x Tool Provider and Tool Consumer compliant. This will allow an EPUB3™ resource to use any third party LTI-compliant tool/service/app;

b) The eBook Reader, whether a specialist device, or software that runs on a laptop, tablet, etc. should be IMS Caliper compliant i.e. use the Sensor API™. This will allow all EPUB3™ resources to report the appropriate leaning analytics;

c) The eBook should support QTI-runtime compliant formative tests (quizzes) either as a remote service accessed using an LTI Tool Consumer interaction or as an embedded resource. When embedded the QTI-runtime capability is supplied as a combination of QTI-JSON files, QTI-XHTML files and QTI JavaScript libraries.

It should be noted that this is a work in progress. Over the next 12 months, IMS Global and the IDPF will complete further refinement of this work and will publish updates to this best practices document. In particular, IMS will define and introduce the EDUPUB Conformance Program.
# Table of Contents

TRADEMARK NOTICE ........................................................................................................... 6

ACKNOWLEDGEMENTS ......................................................................................................... 6

1  INTRODUCTION .................................................................................................................. 7
   1.1  SCOPE & CONTEXT ....................................................................................................... 7
   1.2  EPUB3 OVERVIEW ....................................................................................................... 7
   1.3  IMS CONFORMANCE CERTIFICATION PROGRAM ................................................. 8
   1.4  STRUCTURE OF THIS DOCUMENT ........................................................................... 9
   1.5  ACRONYMS .................................................................................................................. 10
   1.6  REFERENCES ............................................................................................................... 11

2  ARCHITECTURAL FRAMEWORKS ...................................................................................... 13
   2.1  EPUB ARCHITECTURE ............................................................................................... 13
       2.1.1  Readium EPUB Reader ...................................................................................... 15
   2.2  LTIv1.X ARCHITECTURE .......................................................................................... 15
   2.3  LTIv2.0 ARCHITECTURE .......................................................................................... 16
   2.4  CALIPER ANALYTICS ARCHITECTURE .................................................................... 16
       2.4.1  Sensor API Design and Implementation .............................................................. 18
   2.5  QTIV2.1 CONTENT ARCHITECTURE ....................................................................... 18
       2.5.1  Tests & Packages .............................................................................................. 18
       2.5.2  QTI Item Features ............................................................................................. 21
       2.5.3  Metadata .......................................................................................................... 22
       2.5.4  APIP Accessibility Features .............................................................................. 23
   2.6  THE EDUPUB ARCHITECTURE ................................................................................. 23
       2.6.1  The Core EDUPUB Use-cases .......................................................................... 24
       2.6.2  Supporting Learning Analytics .......................................................................... 24

3  EPUB3 AND LEARNING TOOLS INTEROPERABILITY V1.X ................................................. 26
   3.1  LTIv1.X TOOL PROVIDER SUPPORT IN EPUB3 READERS ..................................... 26
       3.1.1  Configuring Access for a Tool Consumer ......................................................... 26
       3.1.2  Processing Launch Request Messages .............................................................. 26
   3.2  LTIv1.X TOOL CONSUMER SUPPORT IN EPUB3 READERS .................................. 28
       3.2.1  Configuring Access to a Tool Provider ............................................................. 28
       3.2.2  LTI Launch Parameters ................................................................................... 29
   3.3  LTIv1.X USAGE IN THE EPUB3 BOOKS .................................................................... 30
   3.4  EDUPUB AND LTIv1.X CONFORMANCE .................................................................. 31
   3.5  READIUM AND LTIv1.X COMPLIANCE ................................................................... 31

4  EPUB3 AND CALIPER .......................................................................................................... 32
   4.1  CALIPER SUPPORT IN EPUB3 READERS ............................................................... 32
### 4.1.1 Reader Sensor(s) to Collect Learning Metrics

#### 5.2.2 The Supported QTI Features

#### 5.3.3 The Quiz Interface & User Experience

#### 5.3.4 Mapping from QTI to EDUPUB Quiz Format

### 5.4 Metadata

### 5.5 Using Caliper with EDUPUB/QTI-Based Content

### 5.6 Remote Data Synchronization

### 5.7 EDUPUB and QTI Conformance

### 5.8 Readium and Assessment Support Compliance

### APPENDIX A – TERMS & DEFINITIONS

### APPENDIX B – ANNOTATED LTI/EDUPUB EXAMPLES

**B1** Using LTI to Launch a Tool from Within an EPUB3 eBook

**B2** Launching the Questionmark System

**B3** Launching the YouSeeU System

### APPENDIX C – ANNOTATED CALIPER/EDUPUB EXAMPLES

### APPENDIX D – ANNOTATED EMBEDDED QTI/EDUPUB EXAMPLE

**D1** Overview of the Example

**D2** The User Experience for the Quiz

**D3** The Definitive Source XML for the Quiz

**D4** The Embedded XHTML File(s)

**D5** The Embedded JSON File(s)

**D6** Other Files Changes in the EPUB3

---

**IMS Global**  
Draft 8 / 6th June, 2014  
EPUB3/LTI/QTI/Caliper Best Practices  
CS
D7 THE SUPPORT JAVA SCRIPT FILES ................................................................. 126
D8 THE NEW EPUB3/EDUPUB CONTENT STRUCTURE ................................. 126

APPENDIX E – METADATA IN EDUPUB ......................................................... 128
E1 METADATA INFORMATION MODEL ..................................................... 128
E2 BINDING MAPPINGS IN IEEE LOM AND SCHEMA.ORG ...................... 132
E3 EXAMPLES OF THE BINDINGS ............................................................. 136

APPENDIX F – READIUM SUPPORT FOR EDUPUB ................................... 141
F1 SOURCE CODE ADDITIONS TO SUPPORT LTIv1.x TOOL PROVIDER FUNCTIONALITY .................................................. 141
F2 SOURCE CODE ADDITIONS TO SUPPORT LTIv1.x TOOL CONSUMER FUNCTIONALITY .............................................. 141
F3 SOURCE CODE ADDITIONS TO SUPPORT IMS SENSOR API FUNCTIONALITY ................................................................. 141
F4 THE READIUM SOURCE CODE DISTRIBUTION .................................... 143

APPENDIX G – BEST PRACTICES CHECKLIST ............................................. 145
G1 EDUPUB CONTENT ................................................................................ 145
  G1.1 General Recommendations ............................................................ 145
  G1.2 LTI-Specific Recommendations ....................................................... 145
  G1.3 Caliper-Specific Recommendations ................................................. 145
  G1.4 QTI-Specific Recommendations ..................................................... 145
  G1.5 Metadata Recommendations ........................................................ 146
G2 EDUPUB READERS ............................................................................. 146
  G2.1 General Recommendations ............................................................ 146
  G2.2 LTI-Specific Recommendations ....................................................... 146
  G2.3 Caliper-Specific Recommendations ................................................. 146
  G2.4 QTI-Specific Recommendations ..................................................... 146
  G2.5 Metadata Recommendations ........................................................ 147
G3 EDUPUB AUTHORING SYSTEMS, TOOLS & APPLICATIONS .......... 147
  G3.1 General Recommendations ............................................................ 147
  G3.2 LTI-Specific Recommendations ....................................................... 147
  G3.3 Caliper-Specific Recommendations ................................................. 147
  G3.4 QTI-Specific Recommendations ..................................................... 147
  G3.5 Metadata Recommendations ........................................................ 147

APPENDIX H – LTI/CALIPER/QTI EDUPUB CONFORMANCE .................... 148
Trademark Notice

APIP™ and Accessible Portable Item Protocol™ are trademarks of IMS Global Learning Consortium.

QTI™ and Question and Test Interoperability™ are trademarks of IMS Global Learning Consortium.

LTI™ and Learning Tools Interoperability™ are trademarks of IMS Global Learning Consortium.

CC™ and Common Cartridge™

AccessForAll™ is a trademark of IMS Global Learning Consortium.

Caliper Analytics™ and the Sensor API™ are trademarks of IMS Global Learning Consortium.

EPUB is a registered trademark of the International Digital Publishing Forum.

International Digital Publishing Forum™ and the IDPF™ are trademarks of International Digital Publishing Forum.

Acknowledgements

IMS Global wish to thank Carson Dellosa Publishing for the use of their EPUB3 publication “INTENSE! Machines” as the basis for the examples used in prototyping this approach and described in this document.
1 Introduction

1.1 Scope & Context

This document contains the best practice recommendations for the use of the IMS Learning Tools Interoperability (LTI) [LTI, 12], [LTI, 14], IMS Caliper Analytics [Caliper, 13] and IMS Question and Test Interoperability (QTI) [QTI, 12a] with an EPUB3 [EPUB3, 11a] document: the International Digital Publishing Forum (IDPF) is responsible for EPUB3. This combination of EPUB3 electronic book format with the QTI, LTI and Caliper standards is called the EDUPUB Standard. The use of LTI, QTI and Caliper provides the capability to use third party tools/apps/systems, quizzes for formative assessment and to log the data for the creation of learning analytics metrics respectively within the context of an EPUB3-based eBook.

This best practice document provides the key architectural use-cases that can be supported by the new functionality. A detailed description for the use of LTI, Caliper and QTI within EPUB3 is supplied along with annotated examples to demonstrate the key features. Examples and the source code (including the IMS fork of the Readium source code) are available at: http://www.imsglobal.org/edupub/.

It should be noted that this is a work in progress. Over the next 12 months, IMS and the IDPF will complete further refinement of this work and will publish updates to this best practices document. In particular, IMS will define and introduce the EDUPUB Conformance Program.

1.2 EPUB3 Overview

The EPUB specification is a distribution and interchange format standard for digital publications and documents. EPUB defines a means of representing, packaging and encoding structured and semantically enhanced Web content including HTML5, CSS, SVG, images, and other resources, for distribution in a single-file format. EPUB 3, the third major release of the standard, consists of a set of four specifications, each defining an important component of an overall EPUB Publication:

- EPUB Publications 3.0 [EPUB3, 11a] that defines publication-level semantics and overarching conformance requirements for EPUB Publications;
- EPUB Content Documents 3.0 [EPUB3, 11b] that defines profiles of XHTML, SVG and CSS for use in the context of EPUB Publications;
- EPUB Open Container Format (OCF) 3.0 [EPUB3, 11c] that defines a file format and processing model for encapsulating a set of related resources into a single-file (ZIP) EPUB Container;
- EPUB Media Overlays 3.0 [EPUB3, 11d] that defines a format and a processing model for synchronization of text and audio.
EPUB has been widely adopted as the format for digital books (eBooks), and these new specifications significantly increase the format’s capabilities in order to better support a wider range of publication requirements, including complex layouts, rich media and interactivity, and global typography features. The expectation is that EPUB 3 will be utilized for a broad range of content, including books, magazines and educational, professional and scientific publications.

1.3 IMS Conformance Certification Program

Over the past fifteen years IMS has published an extensive set of e-learning interoperability standards. During the past five years IMS has focused on introducing a Conformance Certification Program for these standards so that vendors can have their products that use these standards recognised through certification. Whenever a new standard is produced by IMS, it is accompanied a Conformance Program so that vendors’ systems, applications, tools and content can be certified.

Each Conformance Program consists of a defined conformance process that reflects the nature of the standard: it must be stressed that IMS is concerned only with information interoperability and not how the information may be subsequently used within an end system. Self-certification is the basis of the process with IMS moderating the subsequent documentation submission. IMS standards break into two types: data exchange format definition and service exchange definition. The Conformance Programme for a data exchange format based standard consists of the use of the:

- IMS Online Validator to verify that content complies with the required formats. This is used to certify standalone content or to confirm that a system exports data in the right format;

- Reference Test Data Set to verify that systems that import the data are capable of handling all of the various features within the standard.

For service-based standards the Conformance Programme typically consists of:

- A service test harness that is used to confirm that a service provider operates as required. This uses the corresponding service API that is also made available as part of the specification;

- A service test harness to confirm that a service consumer operates as required.

All of the IMS conformance test tools supply a conformance test report that describes the findings of the test tools. The conformance process requires a vendor to complete a functionality matrix that describes the features of the standard that the product supports: in many cases a compliant product is not required to support all the features of a standard. The vendor then uses the appropriate set of conformance testing tools and submits the set of test reports to IMS along with the functionality matrix. IMS inspect and validate the submission.

EDUPUB is based upon two IMS service-based (LTI and Caliper) and one data format (QTI) standards. Therefore the Conformance Programme is based on testing both the service and data functionality.
# 1.4 Structure of this Document

The structure of the rest of this document is:

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.</td>
<td>Architectural Frameworks</td>
<td>Summary of the individual architectural frameworks for EPUB3, LTI, Caliper and QTI and the integration of these to support the various EDUPUB use-cases.</td>
</tr>
<tr>
<td>3.</td>
<td>EPUB3 and Learning Tools Interoperability V1.x</td>
<td>Describes the best practices for using the LTIV1.x specification to provide data exchange between tools and systems;</td>
</tr>
<tr>
<td>4.</td>
<td>EPUB3 and Caliper</td>
<td>Describes how the Caliper learning analytics framework and Sensor API can be used to support analytics reporting from eBooks;</td>
</tr>
<tr>
<td>5.</td>
<td>EPUB3 and Question &amp; Test Interoperability</td>
<td>Describes the best practices for providing quizzes that are embedded with an EPUB3-based eBook;</td>
</tr>
<tr>
<td>Appendix A</td>
<td>Terms &amp; Definitions</td>
<td>Definition of the key concepts used throughout this document;</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Annotated LTI/EDUPUB Examples</td>
<td>The detailed, annotated examples showing how to use LTI to support EDUPUB including the use of an external quizzing/assessment tool;</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Annotated Caliper/EDUPUB Examples</td>
<td>The detailed, annotated examples showing how to use Caliper to support EDUPUB;</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Annotated Embedded QTI/EDUPUB Examples</td>
<td>The detailed, annotated examples showing how to use embedded QTI to support EDUPUB;</td>
</tr>
<tr>
<td>Appendix E</td>
<td>Metadata in EDUPUB</td>
<td>The usage of the various metadata features in EPUB3 and the relational between the various binding forms that can be used;</td>
</tr>
<tr>
<td>Appendix F</td>
<td>Readium Support for EDUPUB</td>
<td>Collation and summary of all the changes to the Readium source code required to support the LTI, Caliper and QTI functionality;</td>
</tr>
<tr>
<td>Appendix G</td>
<td>Best Practices Checklist</td>
<td>Collation and summary of all the best practices summarized throughout this document;</td>
</tr>
<tr>
<td>Appendix H</td>
<td>QTI/Caliper/LTI and EDUPUB Conformance</td>
<td>Collation and summary of the conformance steps required for compliance to LTI, Caliper and QTI for EDUPUB.</td>
</tr>
</tbody>
</table>
1.5 Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMD</td>
<td>Asynchronous Module Definition</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>APIP</td>
<td>Accessible Portable Item Protocol</td>
</tr>
<tr>
<td>CP</td>
<td>Content Package</td>
</tr>
<tr>
<td>CSM</td>
<td>Curriculum Standards Metadata</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascading Style Sheet</td>
</tr>
<tr>
<td>DRM</td>
<td>Digital Rights Management</td>
</tr>
<tr>
<td>EPUB</td>
<td>Electronic Publication</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Mark-up Language</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer protocol</td>
</tr>
<tr>
<td>IDPF</td>
<td>International Digital Publishing Forum</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronic Engineers</td>
</tr>
<tr>
<td>IMS</td>
<td>IMS Global Learning Consortium</td>
</tr>
<tr>
<td>JSON</td>
<td>Java Script Object Notation</td>
</tr>
<tr>
<td>LMS</td>
<td>Learning Management System</td>
</tr>
<tr>
<td>LOM</td>
<td>Learning Object Metadata</td>
</tr>
<tr>
<td>LRMI</td>
<td>Learning Resource Metadata Initiative</td>
</tr>
<tr>
<td>LTI</td>
<td>Learning Tools Interoperability</td>
</tr>
<tr>
<td>MP</td>
<td>Metric Profiles</td>
</tr>
<tr>
<td>OCF</td>
<td>Open Container Format</td>
</tr>
<tr>
<td>ODF</td>
<td>Open Document Format</td>
</tr>
<tr>
<td>OEBPS</td>
<td>Open eBook Publication Structure</td>
</tr>
<tr>
<td>OPF</td>
<td>Open Package Format</td>
</tr>
<tr>
<td>QTI</td>
<td>Question &amp; Test Interoperability</td>
</tr>
<tr>
<td>RDF</td>
<td>Resource Description Framework</td>
</tr>
<tr>
<td>REST</td>
<td>Representational State Transfer</td>
</tr>
<tr>
<td>SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>SIS</td>
<td>Student Information Systems</td>
</tr>
<tr>
<td>SVG</td>
<td>Scalable Vector Graphics</td>
</tr>
<tr>
<td>URI</td>
<td>Uniform Resource Identifier</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
</tbody>
</table>
W3C World Wide Web Consortium
XHTML Extensible Hypertext Mark-up Language
XML Extensible Mark-up Language
XSD XML Schema Definition

1.6 References


2 Architectural Frameworks

2.1 EPUB Architecture

EPUB is a standardised format for the electronic publication of books. Therefore, it is a directly rendered format when using an EPUB Reader. EPUB Readers are available in many device and software-only forms. The structure of an EPUB3 file is shown in Figure 2.1.

Figure 2.1 – The EPUB3 OCF zip file structure.
Essentially, an EPUB3-compliant eBook is supplied as a zip file with the file extension ‘.epub’. This is termed the Open Container Format (OCF). Within the zip file are three root-level structures:

a) The ‘mimetype’ file containing the string ‘application/epub+zip’. This must be the first file;

b) The META-INF folder. The contents of the META-INF folder are:

   • ‘container.xml’ [required] – identifies the file that is the point of entry for each embedded Publication. This identifies the media type of, and paths to, the root files for the EPUB publications included within the container

   • ‘signatures.xml’ [optional] – contains digital signatures for various assets

   • ‘encryption.xml’ [optional] – contains information about the encryption of publication resources (this file is required if font obfuscation is used). If this file is not present, the OCF container provides no information indicating any part of the container is digitally signed at the container level. However, it is possible that digital signing exists within any alternate contained rendering

   • ‘metadata.xml’ [optional] – used to store metadata about the container. This file, if present, must be used for container-level metadata. This version of the OCF specification does not specify any container-level metadata.

   • ‘rights.xml’ [optional] – used to store information about digital rights. This file is reserved for digital rights management (DRM) information for trusted exchange of Publications among rights holders, intermediaries, and users. This version of the OCF specification does not specify a required format for DRM information, but a future version may specify a particular format for DRM information

   • ‘manifest.xml’ [allowed] – manifest of container contents as allowed by the Open Document Format (ODF);

c) The OEBPS folder that is used to contain the core content of the eBook. The typical contents are:

   • The Package Document (this is the file with the file extension ‘.opf’). The Package Document carries bibliographic and structural metadata about an EPUB Publication, and is thus the primary source of information about how to process and display it. The Package Document is an XML document consisting of a set of container elements, each dedicated to housing information about a particular aspect of the Publication. These containers effectively centralize metadata for the Publication, detail the individual resources that compose it and provide reading order and other information for rendering the Publication to a User

   • The set of XHTML files that contain the actual HTML5 content
- The set of supporting assets such as the CSS, images, audio, etc. files.

### 2.1.1 Readium EPUB Reader

All of the examples in this document are based upon the use of the Readium EPUB3 Reader. Readium is an open source EPUB Reader under development by the Readium Foundation ([http://www.readium.org](http://www.readium.org)). The Readium Foundation develops technology to accelerate adoption of EPUB 3 and the Open Web Platform by the global digital publishing industry. Readium.org was formed in February, 2013 as a non-profit membership organization. Current Readium.org projects include Readium Web (an EPUB 3 rendering engine for browser-based cloud readers) and Readium SDK (an EPUB 3 rendering engine for native apps).

For these best practices, Readium has been used in two modes:

- As a browser-based cloud reader with the Readium engine hosted in IMS Cloud;


### 2.2 LTIv1.x Architecture

With LTI 1.x, a Tool Consumer is configured using a Launch URL, consumer key and shared secret provided by the Tool Provider (see Figure 2.2). This is sufficient to enable a user to be able to launch into the Tool Provider’s service. The Tool Consumer may be configured to permit (or deny) the passing of different launch parameters, especially personal data such as a user’s name and email address. It may also make available the LTI Outcomes service to permit the Tool Provider to return grades associated with the launch link.

![Figure 2.2 – The LTI 1.x architecture.](image)
2.3 LTIv2.0 Architecture\(^1\)

With LTI 2.0, the configuration of the Tool Consumer occurs as part of a dialog between the System Administrator and the Tool Provider. During this process the Tool Consumer makes a profile available which describes its properties, capabilities and supported services. The Tool Provider selects from those elements on offer to generate a Tool Proxy describing the connection details required to configure the Tool Consumer. This process allows a Tool Provider to configure multiple tools (resource handlers) with separate launch URLs all at the same time.

![Figure 2.3 – The LTI 2.0 architecture.](image)

2.4 Caliper Analytics Architecture

The IMS Caliper Framework [Caliper, 13] provides a high-level recommendation of how Learning Systems should capture and share data around learning interactions, using existing and upcoming IMS specifications, which can be used to support and advance Learning Measurement and Analytics. Figure 2.4 shows the components of the Caliper Framework. IMS Caliper is built around the following concepts:

- IMS Learning Metric Profiles that provide a Learning Activity centric focus to standardize on metrics (actions and related context) captured across consumer and producer learning tool’s delivery activities and delivery platforms that consume and orchestrate activity based curriculum, while providing for custom extensions and future additions to the profiles;

\(^1\) LTIv2.0 support within EDUPUB is not defined in this release of the best practices. Information about LTIv2.0 is given to supply context for a later functional addition to EDUPUB.

\(^2\) CSS 3.0 is now available and later releases of the QTIv2.1 specification will address its usage.
• IMS Learning Sensor Application Programming Interface (API) and Learning Events drive standardized instrumentation and metric capture and marshal between tools and their delivery platforms and/or associated analytics service solution aggregating metrics;

• IMS LTI/LIS/QTI leverage and extensions enhance and integrate granular, standardized learning measurement with tools interoperability and the underlying learning information models, inclusive of course, learner, outcomes and other critical associated context.

Figure 2.4 – The IMS Caliper Framework.

IMS Learning Metric Profiles are intended to define a standardized, structured collection of Learning Activity Metrics that represent granular measurements specific to actions within each genre of activity including all relevant context for any given action on an activity. In addition, there are pan-genre Foundational Metrics such as engagement, performance, etc., that support common measurement of more generally applicable metrics across all activities. Any given Learning Activity can have (one or more) Metric Profiles associated with it to collect and exchange whatever measurements which need to captured and managed.
In this best practice document the focus is on the use of the IMS Sensor API to report the raw data activity from within the EPUB3 Reader/document. The corresponding learning activity metrics for the use of the EPUB3 reader, in general, and the formative assessment, in particular, are not addressed.

2.4.1 Sensor API Design and Implementation

The IMS Sensor API is intended to support the instrumentation, collection and exchange of data from Learning Tools/Systems and associated Learning Content elements. This enables the availability of standard metrics accessed via any given Analytics Store and associated APIs. At its core, the Sensor API supports the exchange of Learning Events based on interactions with/on Learning Activities.

Learning Events are expressed in the form of a data triple - “Action (Learning) Context” - “Action” - “Activity Context”. This is based on the W3C RDF Triple form of "subject/predicate/object" - i.e. an expression linking one object (subject) to another object (object) or a literal via a predicate. In this case, the Learning Event will allow for the expression of the measurement of an interaction with a Learning Activity. Learning Events are consumed by Sensor endpoints. These can be implemented by any application that wishes to consume Learning Events and is expected to support popular protocols like HTTP/REST. Each element of the Learning Event triple defines one or more of the entities in a learning interaction. Together, these entities make up the Learning Graph based on the connectedness of the entities within a learning environment. Each of these elements is based upon existing IMS specifications and vocabulary (for EDUPUB this includes the QTI specification), as well as, the in-progress IMS Learning Activity Metric profiles.

The Learning Events and Metric Profiles are intended to strike a balance between a completely open mechanism and a rigid schema for measuring and analyzing learning in the scope of the Learning Graph. The Metric Profiles are intended to apply a very lightweight schema while allowing for marshaling of custom data specific to a particular App/System with a goal of enhancing the profiles in a very iterative way in order to support the needs of the educational technology community.

2.5 QTIv2.1 Content Architecture

The Question & Test Interoperability (QTI) Specification provides assessment programs and question item developers a data model for standardizing the file format of digital test items [QTI, 12a], [QTI, 12b]. The QTI specification makes use of the IMS Content Packaging (CP) v1.2 specification [CP, 07a]. QTI is based on AssessmentItem and AssessmentTest models that allow question developers to specify a variety of information for a test (an AssessmentTest consists of one or more AssessmentItems). The APIP/QTI model provides a comprehensive framework that encompasses the requirements for creating accessible tests.

2.5.1 Tests & Packages

Each Quiz/Test is contained within its own Quiz/Test Package (as a zip file). Each assessmentTest must contain one or more testParts and each testPart must contain one or more
**assessmentSections.** Each testPart definition is contained within the assessmentTest XML instance. It is recommended that each assessmentSection in a testPart is defined by reference i.e. it is contained in a separate XML instance file.

Within a package, the set of XML instances and asset files should be collected in directories based upon the corresponding Item/Section. A separate directory may be used to store assets that are used by more than one Item. The directory composition in a content package is used to ease the packaging of the information. It is not required for end systems to sustain this set of relationships however it must be possible for the end system to obtain all of the files identified by an XML instance. A schematic representation of the composition of a QTI-based Quiz/Test is shown in Figure 2.5.

![Diagram of QTI-based Quiz/Test composition](image)

**Figure 2.5 – The logical and file composition of a QTI-based assessment.**

The key features in Figure 2.5 are:

- *AssessmentTest, assessmentSection and assessmentItem objects are contained in their own XML instance files;*

- *AssessmentSections are permitted to contain assessmentSections i.e. these are the recursive objects used to build complex assessmentTests;*
• The links to `assessmentSections` in `assessmentTests` and `assessmentSections` is provided using the `assessmentSectionRef` feature;

• `AssessmentItems` are linked to the `assessmentTest` using the `assessmentItemRef` object within the corresponding `assessmentSection(s)`.

Figure 2.6 The QTI-based quiz/test package.

A Quiz/Test consists of one or more Sections and one or more Items. In fact, a Quiz/Test consists of all of the Items that are, or may be used by the Quiz/Test. Figure 2.6 shows the relationship between the logical data model for a quiz/test and the construction of the Package.
The appropriate ‘imsmanifest.xml’ file must be included in the Package. The zip file must comply with the IMS Content Packaging requirements and the corresponding profile constraints.

### 2.5.2 QTI Item Features

The QTIv2.1 specification defines the behavioural expectations for the various QTI features. All of the QTI interactions are permitted within a package. Other QTI *assessmentItem* features available are:

- **Composite Items** – these are items that contain more than one type of interaction. Composite items may contain multiple instances of the same type of interaction or have a mixture of interaction types. When it isn’t ‘composite’, an item is considered ‘simple’;

- **Use of CSS 2.1 tags for layout and formatting** – QTI 2.1 allows for the inclusion of a CSS file reference within the item. APIP enforces the specific use of CSS as the layout specification. The full CSS 2.1 specification can be found at [http://www.w3.org/TR/CSS2/](http://www.w3.org/TR/CSS2/).

- **Templates** – Item templates are templates that can be used for producing large numbers of similar items. Such items are often called cloned items. Item templates can be used to produce items by special purpose cloning engines or, where delivery engines support them, be used directly to produce a dynamically chosen clone at the start of an item session;

- **Adaptive Items** – adaptive items are a new feature of version 2 that allows an item to be scored adaptively over a sequence of attempts. This allows the candidate to alter their answer following feedback or to be posed additional questions based on their current answer;

- **Feedback** – feedback consists of material presented to the candidate conditionally based on the result of *responseProcessing*. In other words, feedback is controlled by the values of outcome variables. There are two types of feedback material, modal and integrated. Modal feedback is shown to the candidate after response processing has taken place and before any subsequent attempt or review of the item. Integrated feedback is embedded into the *itemBody* and is only shown during subsequent attempts or review;

- **Response Processing Templates** – response processing templates are used to simplify response processing. When such templates are used these may be included in the Package as a separate resource. The key open issue for each template is the set of response processing variables that should be defined for each type of interaction;

- **Outcomes Declarations** – the outcomes defined for an Item can be associated with the learning standards to which they apply. This is achieved using the curriculum standards metadata annotation for a resource in a Package (in the manifest). For each outcome defined in an *assessmentItem*, a separate Outcomes Declaration XML instance file is created. These outcomes declaration instance files are now defined as manifest resources and the appropriate

---

2 CSS 3.0 is now available and later releases of the QTIv2.1 specification will address its usage. APIP compliant systems are recommended to support CSS 3.0.
CSM annotations are included. Finally, the dependency between the Outcome Declaration resource and the parent *assessmentItem* resource is identified using the resource *dependency* feature in the manifest.

### 2.5.3 Metadata

There are many types of metadata used within a QTI Package, namely:

- **Manifest Metadata** – manifest metadata is normally encoded using IEEE LOM [LOM, 02] (all LOM parts are available for use as required). An importing system is expected to make this metadata available for observation to appropriately authorised user roles. An exporting system must ensure that this information can be accurately supplied;

- **Resource Metadata** – the resource metadata is normally encoded using IEEE LOM (all LOM parts are available for use as required). An importing system is expected to make this metadata available for observation to appropriately authorised user roles. An exporting system must ensure that this information can be accurately supplied;

- **QTI-specific Metadata** – this metadata describes if the Item is a composite or not i.e. uses more than one interaction form and identifies the type of interaction used. Other QTI metadata is available to describe the tool that was responsible for the creation of the Item i.e. tool name, tool version and tool vendor. The metadata is encoded using the QTI metadata XML binding and this is supplied within the LOM descriptions for the associated resource. An importing system is expected to make this metadata available for observation to appropriately authorised user roles. An exporting system must ensure that this information can be accurately supplied;

- **Curriculum Standards Metadata (CSM) [CC, 11]** – curriculum standards metadata is supported at the package and resource levels. The design of this support allows for at least the following:
  
  - Any curriculum standard, learning standards or learning map definition can be used, as long as it supports unique identifiers. The provider of a specific standard is designated by the string-valued ‘provider’ element. Note that the provider can be accompanied by region and version string-valued attributes. These are intended to be for descriptive value only
  
  - Any cartridge and resource (hence Item, Section, Test and Outcome Declaration) can be associated with zero or more curriculum standards from one or more providers
  
  - The provider value and GUID should be sufficient to unambiguously identify a standard
  
  - An optional GUID label is supported to make standards more readily apparent when examining the cartridge;

- **File Metadata** – the file metadata is normally encoded using IEEE LOM (all LOM parts are available for use as required). An importing system is expected to make this metadata
available for observation to appropriately authorised user roles. An exporting system must ensure that this information can be accurately supplied;

- Access For All Digital Resource Description (AfA DRD) Metadata [DRD, 09] – AfA DRDv2.0 metadata can be used to described variant QTI Item structures e.g. when an alternative QTI Item based on simplified language is also available. There is no particular recommendation for how to use AfA DRD to support variant resources.

### 2.5.4 APIP Accessibility Features

The following accessibility features [APIPQ, 14] are available:

- Assessment Item (Item Body) – the principle manner for the description of the alternative accessible forms of the content for the question;
- Assessment Item (Rubric Block) – Item-level content that is presented as rubric information (the specification is silent on best practices for how a system should present this information);
- Assessment Item (Modal Feedback) – Item-level feedback (the specification is silent on best practices for how a system should present this information);
- Assessment Section (Rubric Block) – Section-level content that is presented as rubric information including use as a common stimulus for several Items (the specification is silent on best practices for how a system should present this information);
- Assessment Test (Test Feedback) – Test-level feedback (the specification is silent on best practices for how a system should present this information).

It should be noted that the QTI-based accessibility features are not supported in this first version of EDUPUB.

### 2.6 The EDUPUB Architecture

The underlying architectural model for EDUPUB is shown in Figure 2.7. The key features are:

- EPUB Reader – the device and/or software that is used to render the eBook for use by the user. The Reader will, in general, contain many eBooks;
- Learning Management System (LMS) – an institutions LMS through which the learning activity is moderated;
- Assessment System – the system responsible for presenting and scoring an online assessment (this can be formative and summative assessment);
- Analytics System – the learning analytics repository plus the services through which the learning metrics are made available;
• Student Information System (SIS) – the institutions system that stores all of the reference information about the learner.

![Overarching EDUPUB Architectural Model](image)

Figure 2.7 – Overarching EDUPUB Architectural Model

### 2.6.1 The Core EDUPUB Use-cases

The core usage scenarios for this architecture are:

• Standalone Consumer Access – in this scenario the eBook Reader is used without any of the use of institutional or other third party services. This is the typical usage for a consumer;

• Institutional-oriented Access – in this scenario the eBook is part of a larger educational and/or training environment. A number of institutional services such as the LMS, SIS, assessment and analytics systems are used;

• LTI-Mediated Remote Content Access & Reporting – a software-based eReader may be launched via LTI using a pre-configured consumer key and shared secret (see Section 3). A specific eBook may be opened automatically by passing its name in a custom LTI launch parameter. LTI-mediated content can be used in both standalone and institutional-oriented access.

### 2.6.2 Supporting Learning Analytics

Support for learning analytics requires various systems and the content itself to have the ability to report the appropriate data to the analytics server/repository. This gives rise to the use of the IMS Sensor API as shown in Figure 2.8.
There is no co-ordination between each of the Sensor deployments. Therefore each data log sent by each sensor must have sufficient semantic information so that the data from the different streams can be collated and aggregated as appropriate to the learning metrics. A sensor can send to any endpoint and multiple end points can be supported by one physical analytics repository. Endpoint differentiation can be used for data segregation e.g. different levels of data privacy may be handled at different endpoints.

Figure 2.8– Supporting learning analytics within the EDUPUB Architectural Model
3  EPUB3 and Learning Tools Interoperability v1.x

IMS has developed the LTI specifications to allow remote tools and content to be integrated into a LMS. LTI has two main components:

- What is traditionally named as the LMS is referred to as the “Tool Consumer” (TC) as it “consumes” the tool;
- The external tool or content is called the “Tool Provider” (TP) as it “provides” the tool for use in the Tool Consumer. Example Tool Providers might include an externally hosted testing system or a server that contains externally hosted premium content.

3.1  LTIv1.x Tool Provider Support in EPUB3 Readers

3.1.1  Configuring Access for a Tool Consumer

A Tool Consumer may be given the ability to launch Readium by issuing it with a unique consumer key and shared secret. These properties are configured in Readium by adding an element to the LTI_CONSUMERS array variable in server/main.js; for example:

```javascript
{
   "key" : "imsglobal.org",
   "secret" : "ASecureSecret"
}
```

The consumer key and shared secret should be securely submitted to the Tool Consumer for use in their configuration process of the LTI tool; the launch URL is the address of the Readium server with a path of /ltilaunch.

3.1.2  Processing Launch Request Messages

When a launch request message is received by the /ltilaunch path, the following steps are followed:

1. Check that a consumer key, OAuth signature and timestamp have been included in the request;

2. Check that the timestamp is within 5 minutes of the Readium server time;

3. Check that the consumer key has been configured in the Readium server (see Subsection 3.1.1);

4. Verify the OAuth signature using the shared secret configured for the consumer key;

---

3  A hard-coded JavaScript array is currently used to record these values, but should be replaced by a more persistent form of storage in a future version.
5. Check that the following required launch parameters are present:
   a. context_id
   b. required_link_id
   c. user_id

6. Display the Readium home page exposing the following variables:
   a. LTI – an object containing all of the LTI launch parameters for this session (see below)
   b. LTI_TOOL_TARGETS – containing the display target for each available tool provider (see Subsection 3.2.1);

   If a parameter named custom_epub is included in the launch request, then its value will be used as the name of the eBook to be opened;

7. If the launch request is not valid then the user is redirected to the return URL with an error message, or if no return URL is provided, an error page is displayed.

The LTI variable is an array with one element named “session” containing an object with the following structure:

```
{
  "id" : "1453556211996114",
  "user" : {
    "id" : "29123",
    "firstname" : "John",
    "lastname" : "Baird",
    "email" : "jbaird@uni.ac.uk"
  },
  "context" : {
    "id" : "S3294476"
  }
}
```

The session ID (LTI['session'].id) is used to keep each launch request received separate and should be passed when performing any launches to an LTI Tool Provider (see Subsection 3.3). The other properties can be accessed via JavaScript in a similar way; for example, a user’s email address is LTI['session'].user.email. Each property is guaranteed to exist and will be filled with dummy data when the corresponding launch parameters have not been passed from the tool consumer. The following dummy values are used:

- user.firstname: User;
- user.lastname: same as user.id;
- user.email: empty.
3.2 LTIf1.x Tool Consumer Support in EPUB3 Readers

3.2.1 Configuring Access to a Tool Provider

A Tool Provider may be made available for launch from within Readium by adding an element to the LTI_PROVIDERS array variable in server/main.js, for example:

```javascript
{
   "id" : "test",
   "name" : "Test launch",
   "key" : "testing.edu",
   "secret" : "secret",
   "target" : "window",
   "presentation_document_target" : "popup",
   "displayWidth" : 800,
   "displayHeight" : 600,
   "custom" : {
      "debug" : "true",
      "one" : "1"
   }
}
```

For each Tool Provider there are the following attributes:

- **id** – a unique id used in launch links to identify the tool to be launched (see );
- **name** – a human readable name for the tool;
- **key** – the consumer key issued for use by the tool provider;
- **secret** – the secret issued for use by the tool provider;
- **url** – the launch URL;
- **presentation_document_target** – where the tool should be opened (“window” and “popup” are currently supported);
- **displayWidth** – the width of the window to be opened when the target is set to “popup”, a default value of “400” is used when this attribute is not specified;
- **displayHeight** – the height of the window to be opened when the target is set to “popup”, a default value of “400” is used when this attribute is not specified;
- **custom** – a set of custom parameters which will be included in launch requests.

---

4 A hard-coded JavaScript array is currently used to record these values, but should be replaced by a more persistent form of storage in a future version.
3.2.2 LTI Launch Parameters

The parameters listed in Table 3.1 are currently included in launch requests to available Tool Providers:

**Table 3.1 – The parameters available for an LTI launch request.**

<table>
<thead>
<tr>
<th>Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>context_id</td>
<td>C</td>
</tr>
<tr>
<td>context_title</td>
<td>C</td>
</tr>
<tr>
<td>context_type</td>
<td>C</td>
</tr>
<tr>
<td>launch_presentation_document_target</td>
<td>P</td>
</tr>
<tr>
<td>launch_presentation_width</td>
<td>P</td>
</tr>
<tr>
<td>launch_presentation_height</td>
<td>P</td>
</tr>
<tr>
<td>launch_presentation_locale</td>
<td>C</td>
</tr>
<tr>
<td>lis_person_contact_email_primary</td>
<td>C</td>
</tr>
<tr>
<td>lis_person_name_family</td>
<td>C</td>
</tr>
<tr>
<td>lis_person_name_full</td>
<td>C</td>
</tr>
<tr>
<td>lis_person_name_given</td>
<td>C</td>
</tr>
<tr>
<td>lis_person_sourcedid</td>
<td>C</td>
</tr>
<tr>
<td>lti_message_type</td>
<td>S</td>
</tr>
<tr>
<td>lti_version</td>
<td>S</td>
</tr>
<tr>
<td>oauth_callback</td>
<td>S</td>
</tr>
<tr>
<td>oauth_consumer_key</td>
<td>S</td>
</tr>
<tr>
<td>oauth_nonce</td>
<td>S</td>
</tr>
<tr>
<td>oauth_signature</td>
<td>S</td>
</tr>
<tr>
<td>oauth_signature_method</td>
<td>S</td>
</tr>
<tr>
<td>oauth_timestamp</td>
<td>S</td>
</tr>
<tr>
<td>oauth_version</td>
<td>S</td>
</tr>
<tr>
<td>resource_link_id</td>
<td>Q</td>
</tr>
<tr>
<td>resource_link_title</td>
<td>Q</td>
</tr>
<tr>
<td>roles</td>
<td>C</td>
</tr>
<tr>
<td>tool_consumer_info_product_family_code</td>
<td>P</td>
</tr>
</tbody>
</table>
### Key to source values:

- **C** – passed on from the original LTI launch request received from the Tool Consumer (omitted if the parameter was not included in the original request);
- **P** – based on data held within Readium;
- **Q** – value specified in the construction of the launch link within Readium;
- **S** – values generated automatically as part of the launch process.

Any custom parameters specified in the Tool Provider configuration will also be added to the launch request, and the parameters named `custom_caliper_service_url` and `custom_caliper_api_key` will also be passed on if present in the original launch request.

### 3.3 LTIv1.x Usage in the EPUB3 Books

One of the available Tool Providers may be launched by sending an HTTP request to the Readium server with a path of `/ltitool` and including the following query parameters:

- `session` – the id for the user session (as passed in LTI["session"].id);
- `tool` – the id for the tool to be launched;
- `id` – the id of the link from which the tool is being launched (each link should have a unique id which is passed as the `resource_link_id` parameter on an LTI launch;

The URL is constructed by a JavaScript function so a launch link can be created by inserting a hyperlink similar to the following into an eBook:

```html
<a href="" onclick="return ltiLaunchTool(this, 'test', '54935405')"
title="Test launch">Test launch</a>
```

Clicking this link will attempt to launch the tool which has the ID of `test` with a resource link ID of 54935405. The link text in this example is taken from the Tool Provider configuration but may be changed as desired for each link. The title attribute is optional but, when present, will be used as the value of the `resource_link_title` launch parameter.
3.4 EDUPUB and LTIV1.x Conformance

At the time of publication the EDUPUB Conformance Programme has not been finalised. This Section will be completed in later versions of this document.

3.5 Readium and LTIV1.x Compliance

At the time of publication the Readium tool has not undergone LTIV1.x conformance testing. This Section will be completed in later versions of this document.
4 EPUB3 and Caliper

4.1 Caliper Support in EPUB3 Readers

Caliper support in EPUB3 readers is enabled by

- (Ideally) having the EPUB3 Reader also be a LTI Tool Provider;

- As part of the LTI Launch into the EPUB3 reader, propagating both the LTI context (userId, contextId) and the Caliper Context (passed as custom LTI launch parameters at the current time) to any installed Caliper sensors;

- Installing a Caliper Reading Sensor to the EPUB3 Reader and initializing it with both the Caliper and LTI Contexts during an LTI launch;

- Using the Caliper Readium Sensor to instrument key events in the EPUB3 reader via the Caliper Reading Metric Profile (see Subsection 4.1.2).

4.1.1 Reader Sensor(s) to Collect Learning Metrics

Most modern application, including Readium have event handling code/callbacks to deal with key user interactions with the functionality/content provided by the application. By using the appropriate Caliper Metric Profile, an application developer can plug into the existing event handling code and send Caliper Events to the endpoint (propagated during the LTI Launch).

4.1.2 Applicable Metric Profiles

In general, the Caliper Metric Profiles (MP) define the ontology and vocabulary representative of both learning activity common as well as specific measures or metrics and their related contextual attributes. Specifically, for the EDPUB Reading activity, the following Caliper Metric Profiles initially apply and are extensible as required per profile elements as well as in the ability to apply additional or new profiles as the Reading activity is further enhanced with more activity related functionality. The key MPs are:

- Reading MP – covers the metrics and contextual elements for the core reading activity functionality inclusive of but not limited to e-text related navigation, e-text interactions, and engagement;

- Annotation MP – covers the metrics and contextual elements for the typical annotations capabilities made available for e-text reading activity functionality inclusive of but not limited to highlighting, notes and marks;

- Media MP – covers the metrics and contextual elements for any embedded media elements contained within the EDPUB e-text content such as images, audio and video;
Assessment MP – covers the metrics and contextual elements for an embedded formative or otherwise quiz activity contained within the EDUPUB e-text content such as quiz level interactions and attributes, item level interactions and attributes, and performance related item and quiz results (note that a separate but related Performance MP can be leveraged to accommodate more comprehensive detail for scores, results and/or outcomes).

4.1.3 Caliper Sensor Endpoint Use/Propagation

When the Readium EPUB3 Reader/LTI Tool Provider is launched, custom LTI parameters are passed as part of the launch (the Caliper Context). This enables propagation of a Caliper EventStore endpoint URL and a shared API Key from the LTI Consumer into the LTI Provider. Therefore, when the Readium EPUB3 Reader/LTI Provider launches a book as part of an LTI launch, it can propagate the Caliper Context using the code shown in Code 4.1:

```
0000 var loadEbook = function (packageDocumentURL)
0001     {
0002         readium.openPackageDocument(
0003             packageDocumentURL,
0004             function(packageDocument){
0005             // Initialize the Readium Caliper Sensor for this LTI Session
0006             // We now have a fresh context for the sensor for each book
0007             if (window.LTI) {
0008                 // Extract required Caliper and LTI context
0009                 var host     = LTI["session"].caliper.serviceurl;
0010                 var apiKey   = LTI["session"].caliper.apikey;
0011                 var personId = LTI["session"].user.id;
0012                 var courseId = LTI["session"].context.id;
0013                 // Initialize the Readium Sensor
0014                 ReadiumSensor.initSensor(apiKey,
0015                     host,
0016                     courseId,
0017                     personId,
0018                     packageDocument.getMetaData());
0019             }
0020             ...
0021         ...
0022     }
```

Code 4.1 – Caliper context provision from LTI.
### 4.2 Caliper Support in EPUB3 Books

The method calls available from the Sensor API are listed in Table 4.1.

**Table 4.1 – The list of Sensor API methods.**

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialize()</td>
<td>Initializes the API instance. This method must be called before the API can be used. The method parameters are:</td>
</tr>
<tr>
<td></td>
<td>• API Key – the API authorization key;</td>
</tr>
<tr>
<td></td>
<td>• Server URL – the URL for the target analytics repository;</td>
</tr>
<tr>
<td></td>
<td>• Sensor Identifier – the identifier for the source sensor instance.</td>
</tr>
<tr>
<td>describe()</td>
<td>Sends a description of the context in which the measurements are to be supplied. The method parameters are:</td>
</tr>
<tr>
<td></td>
<td>• Type – the type of object being described;</td>
</tr>
<tr>
<td></td>
<td>• Entity Identifier – identifier of the entity producing the measurements;</td>
</tr>
<tr>
<td></td>
<td>• Properties – metadata about the measurements to be reported;</td>
</tr>
<tr>
<td></td>
<td>• Time Stamp – date/time log for the description report.</td>
</tr>
<tr>
<td>measure()</td>
<td>Sends a measurement to the learning analytics repository. The method parameters are:</td>
</tr>
<tr>
<td></td>
<td>• Action – the name of the action (from a predefined vocabulary);</td>
</tr>
<tr>
<td></td>
<td>• Learning-context – information about the learning context for the measurement</td>
</tr>
<tr>
<td></td>
<td>• Activity-context – the information about the learning activity i.e. the activity measurement itself;</td>
</tr>
<tr>
<td></td>
<td>• Time Stamp – date/time log for the measurement report.</td>
</tr>
</tbody>
</table>

See Subsection 4.1 for more details on the form of the measurement triple.

An example of the Sensor API being used in the context of a quiz within an EPUB3 is provided in Appendix D4.1.
4.3 **EDUPUB and Caliper Conformance**

At the time of publication the EDUPUB Conformance Programme has not been finalised. This Section will be completed in later versions of this document.

4.4 **Readium and Caliper Compliance**

At the time of publication the EDUPUB Conformance Programme has not been finalised. This Section will be completed in later versions of this document.
5  EPUB3 and Question & Test Interoperability

5.1  Open Approaches for Formative Assessment in EPUB3

Two approaches for the support of formative assessment within EPUB3 are presented in this best practices document:

- LTI-mediated – the use of LTI links placed within the EPUB3 eBook. When a link is activated the corresponding LTI launch request to the remote quizzing/assessment tool is made. The tool now presents the quiz (in this scenario the quizzing tool does not have to be compliant to QTIv2.1). For this scenario the eBook Reader must be connected to a network;

- Embedded – the quiz content is embedded within the EPUB3. This scenario is ideal when the eBook is used in standalone mode without access to a network. In this approach it is also possible to use remote services to support the embedded quiz e.g. for the response processing and so avoid the answer to the question being stored in the EPUB3 file.

In both cases the QTIv2.1 XML form of the quiz/test is the definitive content. All other forms must be derived from this reference baseline. This ensures that any identification/mapping information that is required by other services e.g. the learning analytics using the Sensor API, is available through any of the intermediate forms because they are all derived from the single reference source.

Both approaches are demonstrated using the examples described in Appendices B and D.

5.2  LTI-Mediated Assessment Content Delivery

A schematic representation of the LTI-mediated approach for assessment delivery is shown in Figure 5.1. For LTI-mediated delivery the only information contained within the EPUB3 file is the LTI-link and the associated launch request. The launch request requires that either the eBook Reader be an LTI-compliant Tool Consumer or the EPUB3 file itself contain all of the required code (for example as JavaScript) and the Reader permit direct network access by the EPUB3 content.

In this scenario, the remote quiz/test tool is responsible for any learning analytics data logging i.e. this data does not go via the eBook Reader. The LTI launch passes all of the relevant user identification code and so this can be used to provide the user-context for the analytics data to be logged. Once the quiz has been completed, control is returned to the eBook along with an outcome (the latter requires the use of LTIv1.1).

An example of the relevant EPUB3 page is shown in Figure 5.2. This page has two LTI links: one to the Questionmark assessment tool (which is LTI compliant but which uses QTIv1.2, and which supports the Sensor API; the other to the video-centric LMS YouSeeU. The use of these two LTI links is explained in Appendix B.
Figure 5.1 LTI-mediated delivery of a quiz/test from an EPUB3.

Figure 5.2 – LTI links embedded within an EPUB3 page.
5.3 Assessment Content Embedded in the EPUB3 Book

5.3.1 Overview of the Approach

In the scenario in which the quiz/test is embedded within the EPUB3, not only must the actual content for the quiz be contained, all of the associated code to support the rendering, response processing, feedback and outcomes processing and analytics reporting must be supplied. A schematic representation of the embedded content within the EPUB3 is shown in Figure 5.3.

![Figure 5.3 Embedded QTI-based delivery of a quiz/test in an EPUB3.](image)

A workflow is required to start from the QTI-XML and to end with the quiz embedded within the EPUB3. This workflow is\(^5\):

a) The QTIv2.1 quiz package is unzipped. All of the QTI files are supplied in a profiled form of IMS Content Package (essentially this profile requires the `<organizations>` element in the ‘imsmanifest.xml’ file to be empty and prohibits sub-manifests). The ‘imsmanifest.xml’ file explains the structure of the QTI package with the essential features being:

---

\(^5\) This workflow is presented for discussion and feedback. A number of modified versions are under evaluation and so IMS make no claims that this is the final and definitive mechanism to embed the quizzes. However, this approach produces a valid, operational solution.
• Each QTI Item is contained within its own XML instance file

• The Quiz QTI Test file is contained within its own XML instance file

• Every file in the package has its own resource definition. The <dependency> element is used to define the dependencies between these resource e.g. the test resource has a dependency on each of the item resource;

• Each resource has accompanying metadata that is encoded using the IMS Metadata v1.3 binding of the IEEE LOM v1.0 standard;

• The manifest-level metadata is also encoded using the IMS Metadata v1.3 binding of the IEEE LO v1.0 standard;

• The manifest, as a while, and each resource, can be annotated with the associated curriculum standards identifiers (this uses the IMS Curriculum Standards Metadata, CSM, standard metadata approach).

b) The set of XML files are now transformed from their XML format into the JSON equivalent (each XML file becomes a separate JSON file). This conversion is achieved using the NodeJS tool. The configuration settings required for this tool are:

```javascript
var parser = new xml2js.Parser({
  attrkey : "_",
  mergeAttrs : true,
  charkey : "__text",
  normalize : true,
  explicitArray : false,
  explicitRoot : true
});
```

Before this conversion is undertaken, all of the embedded HTML should be escaped. For example, if the original mark-up is:-

```xml
<modalFeedback
  outcomeIdentifier="FEEDBACK"
  showHide="show"
  identifier="Incorrect"
  title="Incorrect Answer Feedback">
  <p>No, wrong answer. A <em>Combine</em> is used to collect the kernels of corn.</p>
</modalFeedback>
```

should become

```xml
<modalFeedback
  outcomeIdentifier="FEEDBACK"
  showHide="show"
  identifier="Incorrect"
  title="Incorrect Answer Feedback">
  &lt;p&gt;No, wrong answer. A &lt;em&gt;Combine&lt;/em&gt; is used to collect the kernels of corn. &lt;/p&gt;
</modalFeedback>
```
c) The next stage is to create the XHTML file(s). At present this is a manual process using a set of question layout templates, a corresponding CSS file and insertion of the appropriate AngularJS primitives. The AngularJS primitives are used to obtain, in real-time, the relevant parts of JSON encoded data and to insert it into the HTML5 layout;

d) The Sensor API calls are inserted in the AngularJS controller code. These controllers are invoked the handle the associated AngularJS calls within the XHTML file. These controllers plus all of the associated JavaScript libraries to the Sensor API, the AngularJS runtime, etc. are now inserted into the EPUB3 file structure layout;

e) The OPF file for the EPUB3 has to be amended to include the references for all of the new files inserted to support the embedded quiz i.e. the contents of the <manifest> element are extended;

f) The final change is that the navigation file (‘nav.xhtml’ has to be amended to include the new link to the embedded Quiz within the Table of Contents for the EPUB3 (see Figure D1.4);

g) The set of files can now be zipped to create the new EPUB3-compliant eBook. This can now be accessed using an eReader such as Readium.

An example of this workflow is detailed in Appendix D.

5.3.2 The Supported QTI Features

The full QTIv2.1 standard is an extensive and powerful representation mark-up for tests and items. A considerably reduced subset has been adopted as the starting point for EDUPUB quizzes. EDUPUB is primarily concerned with formative assessment. The QTI features that are supported by this initial version of EDUPUB are:

- The Quiz/Test will be constrained to a single Testpart, will contain a single Section which will have at least one rubric block. The Section will not contain other Sections. The Section will contain any number of Items (but at least one) and each Item will have at least one rubric block. The Test will include support for test-level feedback;

- A predefined set of test-level outcome variables will be used i.e. ‘TOTAL_SCORE’, ‘TOTAL_MAXSCORE’, ‘TOTAL_MINSCORE’, ‘MASTERY_SCORE’, ‘TOTAL_NUMBERINCORRECT’, ‘TOTAL_NUMBERCORRECT’, ‘TOTAL_NUMBERPRESENTED’, ‘TOTAL_NUMBERSELECTED’, ‘TOTAL_NUMBERRESPONDED’ and ‘FEEDBACK’;

- Test-level outcomes processing will be defined using an implied template. The actual outcomes processing rules will be supplied in the QTI-XML but these do not need to be parsed;
• Choice interactions (used for True/False, Multiple Choice/Single Response an Multiple Choice/Multiple Response types of question). All of the associated content must be appropriately rendered. Support for shuffled answer options will be provided;

• Text entry interactions (used for Fill-in-Blank type of question). Inline support will be supported because the question(s) will, typically, be embedded in a larger paragraph of text. Only simple string comparison will be used for the response processing;

• Simple Hotspot interactions. This will include the use of a single background image with circular hotspot areas. The user will be expected to select a single region, or hotspot;

• Response processing to provide the appropriate feedback to the reader. The response processing will be based upon fixed, predefined, response processing templates that detect the correct answer and trigger the corresponding feedback statement, and which detect an incorrect answer and provide a single form of corresponding feedback (this is item-level modal feedback only);

• A predefined set of item-level outcome variables will be used i.e. ‘SCORE’ and ‘FEEDBACK’. ‘SCORE’ will be used to contain the score assigned by the response processing and ‘FEEDBACK’ to control the item-level feedback to be shown to the user. Item will have the maximum and minimum scores defined;

• The content HTML will be restricted to the use of <p>, <em> and <img> elements;

• Shared stimulus, usage statistics and results reporting are not supported.

The CP-XML features that must be addressed are:

• The quiz QTI-XML instance file will also contain the XML for the single QTI-Section. Each Item QTI-XML will be in its own file. The associated manifest will include all of the dependency definitions;

• Each resource will include only one physical file reference. The dependency element will be used to link together the relevant resources e.g. the quiz file will have a dependency on each of the item files;

• Access and use of the resource-level metadata including the CSM, and the QTI-specific metadata contained within the LOM-encoded resource-level metadata (use of the AfA DRD will not be addressed). This metadata may be supplied in the examples but usage/processing is not required;

• Access and use of the manifest-level metadata including the CSM. This metadata may be supplied in the examples but usage/processing is not required;

• The response processing templates may be included in the QTI-package but these will not be parsed to create the response processing. Instead the response processing template identifier will be used to identify the corresponding runtime software embedded in the EPUB3.
5.3.3 The Quiz Interface & User Experience

The original QTIv2.1 specification had an abstract user interface that defined the context interactions for a question. In this case the QTI is to be rendered and so an outline user experience has been established for the framing and interactions of the quiz and its questions as a whole. The opening page layout is shown in Figure 5.4 (an example of an the actual layout used in the examples is shown in Figure D2.1).

The opening page displays the title of the quiz, the section-level rubric (as defined for the user) and provides buttons to start or end the quiz. If the ‘Start’ button is pressed the user progress to the layout shown in Figure 5.5 else if the ‘End’ button is pressed the layout shown in Figure 5.6 is shown. The general layout for a question is shown in Figure 5.5 (this is amended if the first or last questions are encountered by removing either the previous or next item buttons respectively).

The Item layout has a region for the display of the Item-level rubric, the body of the Item i.e. the actual question and the associated content, and the feedback once an answer has been submitted. The ‘Next’ and ‘Previous’ buttons move to another Item (the sequential order is defined by the order the Items are referenced in the Section and not from the order of the resources in the QTI-Package). Once the user has answered the question, this attempt is scored only when the ‘Submit’ button is pressed. Examples of the layout for each of the supported question types are shown in Figures D2.2, D2.3, D2.4 and D2.5.

When the final question has been answered the end page is presented. The layout for the test summary page is shown in Figure 5.6.

Figure 5.4 – The entry page for the quiz.
Figure 5.5 – The page layout for an item.

Figure 5.6 – The quiz completion page.

The layout for the quiz completion page has a region for the test-level feedback and the quiz statistics (the actual statistics to be presented has not be finalized). An example of this layout is shown in Figure D2.6.
5.3.4 Mapping from QTI to EDUPUB Quiz Format

Currently, mapping from the QTI-XML to embedded EDUPUB format is achieved by hand. This is to enable a range of approaches be evaluate without wasting effort on needless computer transformations. The details of the mappings are derived by examining the source QTI-XML files in Appendix D3 with the equivalent embedded files contained in Appendices D4 (XHTML) and D5 (JSON). The mapping details are summarised in Tables 5.1 and 5.2.

<table>
<thead>
<tr>
<th>QTI-XML Feature</th>
<th>Equivalent EDUPB Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Assigned in the XHTML via application of AngularJS model ‘qti.assessmentTest.title’ to the ‘qti.assessmentTest.title’ object in the quiz JSON file.</td>
</tr>
<tr>
<td>Outcome Variables</td>
<td>No transformation. Implicit variables used in the outcomes processing library.</td>
</tr>
<tr>
<td>Test Part</td>
<td>No transformation. A single testPart is assumed.</td>
</tr>
<tr>
<td>Assessment Section</td>
<td>No transformation. A single assessmentSection is assumed.</td>
</tr>
<tr>
<td>Item References</td>
<td>Insertion of a set of AngularJS question handlers for the True/False, Multiple Choice-Single Response, Multiple Choice-Multiple Response, Fill-in-Blank and Hotspot question types.</td>
</tr>
<tr>
<td>Rubric Block</td>
<td>Assigned in the XHTML via application of AngularJS binding to ‘qti.assessmentTest.testPart.assessmentSection.rubricBlock.__text’</td>
</tr>
<tr>
<td>Outcomes Processing</td>
<td>No transformation. Implicit outcomes processing code used in the outcomes processing library.</td>
</tr>
<tr>
<td>Feedback</td>
<td>Assigned in the XHTML with the AngularJS ‘data-ng-show’ value of ‘showResponsePage’.</td>
</tr>
</tbody>
</table>
Table 5.2 – QTI-XML to embedded EDUPUB mappings for an Item.

<table>
<thead>
<tr>
<th>QTI-XML Feature</th>
<th>Equivalent EDUPUB Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Assigned in the XHTML via application of AngularJS model ‘qti.currItem.assessmentItem.title’ to the ‘qti.currItem.assessmentItem.title’ object in the quiz JSON file.</td>
</tr>
<tr>
<td>Response Variables</td>
<td>No transformation. Implicit default variable names used in the response processing library.</td>
</tr>
<tr>
<td>Outcome Variables</td>
<td>No transformation. Implicit variables used in the outcomes processing library.</td>
</tr>
<tr>
<td>Rubric</td>
<td>Assigned in the XHTML via application of AngularJS binding to ‘qti.currItem.assessmentItem.itemBody.rubricBlock.__text’.</td>
</tr>
<tr>
<td>Presentation</td>
<td>Insertion of a set of AngularJS question handlers for the True/False, Multiple Choice-Single Response, Multiple Choice-Multiple Response, Fill-in-Blank and Hotspot question types.</td>
</tr>
<tr>
<td>Response Processing</td>
<td>No transformation. Implicit response processing code used in the response processing library.</td>
</tr>
<tr>
<td>Feedback</td>
<td>Assigned in the XHTML with the AngularJS ‘data-ng-show’ value of ‘itemFeedback’.</td>
</tr>
</tbody>
</table>

### 5.4 Metadata

All of the metadata from the QTI-Package is contained within the manifest of the content package. The metadata used within a QTI Package:

- Manifest Metadata – about the package as a whole;
- Resource Metadata – about the resource as a whole;
- QTI-specific Metadata – supplied for assessment-based resources;
- Curriculum Standards Metadata – to indicate the curriculum standards that are addressed by the use of the resource or manifest
- File Metadata – metadata specific to a file used as part of a resource. This metadata is not used in this version of EDUPUB;
- AfA DRD Metadata – to provide information about the accessibility features of a resource. Support for this metadata is not addressed in this version of EDUPUB.
This metadata has to be mapped into the equivalent binding form before it can be stored within an EPUB3. The above metadata is to be stored within the new QTI-XHTML file, one for each quiz that is used that must be stored within the EPUB3. Therefore, Schema.Org based micro-data binding must be used. A full description of the range of metadata that can be supplied from a QTI-Package and the corresponding mapping to the Schema.Org-based representation is explained in Appendix E.

A simple example of the mapping of the QTI-XML to Schema.Org metadata for the quiz resource is supplied in Code inserts 5.1 and 5.3. In Code 5.1 the resource-level metadata is supplied in lines 0003-0044 (in this example all of the metadata has been supplied within the context of LOM i.e. using LOM extensions to contain the QTI-specific and the CSM). Within this the core LOM metadata is supplied in lines 0005-0023, the QTI-specific in lines 0024-0027 and the CSM in lines 0028-0042.

**Code 5.1 – An example of resource-level XML-based metadata for a quiz.**

```
0000 0001 0002 0003 0004 0005 0006 0007 0008 0009 0010 0011 0012 0013 0014 0015 0016 0017 0018 0019 0020 0021 0022 0023 0024 0025 0026 0027 0028 0029 0030 0031 0032 0033 0034 0035 0036 0037 0038 0039
0000 0001 0002 0003 0004 0005 0006 0007 0008 0009 0010 0011 0012 0013 0014 0015 0016 0017 0018 0019 0020 0021 0022 0023 0024 0025 0026 0027 0028 0029 0030 0031 0032 0033 0034 0035 0036 0037 0038 0039
   ...<resource type="imsqti_test_xmlv2p1" identifier="ResourceTest01"
        href="...">
   <metadata>
   <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
   <general>
   <identifier>
   <entry>MACHINE-QUIZ01</entry>
   </identifier>
   <title>
   <string>A Short Quiz on Intense Machines</string>
   </title>
   <lifeCycle>
   <contribute/>
   <version>
   <string>Final 1.0</string>
   </version>
   </lifeCycle>
   <educational>
   <description>
   <string>EPUB3/QTI Example Quiz 1</string>
   </description>
   </educational>
   <qti:qtiMetadata>
   <qti:feedbackType>nonadaptive</qti:feedbackType>
   <qti:solutionAvailable>false</qti:solutionAvailable>
   </qti:qtiMetadata>
   <csm:curriculumStandardsMetadataSet
        resourceLabel="..." resourcePartId="...">
   <csm:curriculumStandardsMetadata
        providerId="...">
   <csm:setOfGUIDs region="Florida" version="1.0">
   <csm:labelledGUID>
   <csm:label>GUID01_Label</csm:label>
   <csm:GUID>GUID01</csm:GUID>
   </csm:labelledGUID>
   <csm:label>GUID02_Label</csm:label>
   <csm:GUID>GUID02</csm:GUID>
   </csm:labelledGUID>
```

IMS Global
The, partial, Schema.Org microdata encoded equivalent of the quiz Code 5.1 is shown in Code 5.2. This microdata identifies the object as a ‘Quiz’ (line 0009) with the version encoded in line 0012, the description encoded in lines 0013-0014, and the title encoded in line 0015. If the available content is not defined within HTML flow then the <meta> element would have to be inserted to associate the relevant metadata with the corresponding <div> structure.

In Code 5.3 the resource-level metadata for the example Item is supplied in lines 0004-0048 (in this example all of the metadata has been supplied within the context of LOM i.e. using LOM extensions to contain the QTI-specific and the CSM). Within this the core LOM metadata is supplied in lines 0006-0025, the QTI-specific in lines 0026-0031 and the CSM in lines 0032-0046.

---

**Code 5.2 – Mapping of the test XML metadata to its Schema.Org microdata equivalent.**

```html
<html xmlns="http://www.w3.org/1999/xhtml" data-ng-app="myApp">
<head>
    <title>Intense Machines Quiz</title>
    <meta name="viewport" content="width=1100, height=1650"/>
    ...
</head>
<body id="quiz_body" itemscope itemtype="http://schema.org/Quiz">
    <div class="Basic-Text-Frame" data-ng-controller="MainCtrl">
        <div class="qti-item" data-ng-hide="hideIntro">
            <meta itemprop="version" content="Final V1.0"/>
            <meta itemprop="description" content="EPUB3/QTI Example Quiz 1"/>
            <div data-ng-model="qti.assessmentTest.testPart.assessmentSection.rubricBlock.__text">
                Quiz:
                {{qti.assessmentTest.title}}</div>
        </div>
        ...
    </div>
</body>
</html>
```
Code 5.3 – An example of resource-level metadata for an Item.

```
<resource type="imsqti_test_xmlv2p1" identifier="ResourceTest01"
    href="...">
    <metadata>
        <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
            <general>
                <identifier>
                    <entry>MACHINE-ITEM01</entry>
                </identifier>
                <title>
                    <string>Number of Engines on a Jumbo Jet</string>
                </title>
                <contributor/>
                <version>
                    <string>Final 1.0</string>
                </version>
            </general>
            <lifeCycle>
                <description>
                    <string>True/False Question on Number of Engines on a Jumbo Jet</string>
                </description>
            </lifeCycle>
            <educational>
                <qti:qtiMetadata>
                    <qti:interactionType>choiceInteraction</qti:interactionType>
                    <qti:feedbackType>nonadaptive</qti:feedbackType>
                    <qti:solutionAvailable>false</qti:solutionAvailable>
                </qti:qtiMetadata>
                <csm:curriculumStandardsMetadataSet
                    resourceLabel="..." resourcePartId="...">
                    <csm:curriculumStandardsMetadata
                        providerId="...">
                        <csm:setOfGUIDs region="Florida" version="1.0">
                            <csm:labelledGUID>
                                <csm:label>GUID01_Label</csm:label>
                                <csm:GUID>GUID01</csm:GUID>
                            </csm:labelledGUID>
                            <csm:labelledGUID>
                                <csm:label>GUID02_Label</csm:label>
                                <csm:GUID>GUID02</csm:GUID>
                            </csm:labelledGUID>
                        </csm:setOfGUIDs>
                    </csm:curriculumStandardsMetadata>
                </csm:curriculumStandardsMetadataSet>
            </educational>
        </lom>
    </metadata>
</resource>
```
Code 5.4 – Mapping of the item XML metadata to its microdata equivalent.

```
0000 ...<div id="wrapper" data-ng-hide="hideQuiz">
0002 <div itemscope itemtype="http://schema.org/Question"
0003 class="qti-item" data-ng-hide="showResponsePage">
0004 <div class="qti-item-header">
0005 <div itemprop="name"
0006 data-ng-model="qti.currItem.assessmentItem.title"
0007 class="title">{{qti.currItem.assessmentItem.title}}</div>
0008 </div>
0009 <div class="points">1 pt</div>
0010 </div>
0011 <div class="qti-item-rubric">
0012 <div class="qti-item-prompt"
0013 data-ng-bind-html="qti.currItem.assessmentItem.itemBody.rubricBlock.__text">
0015 </div>
0016 </div>
0017 <div class="item-content">
0019 <div class="item-content">
0020 ...
0021 </div>
0022 ...
0023 </div>
```

The, partial, Schema.Org microdata encoded equivalent of the Item Code 5.3 is shown in Code 5.4. This microdata identifies the object as a ‘Question’ (line 0002) with the version encoded in line 0012 and the title encoded in line 0005. Again, if the required content is not available within HTML flow then the <meta> element would have to be inserted to associate the relevant metadata with the corresponding <div> structure.

5.5 Using Caliper with EDUPUB/QTI-based Content

The Sensor API is used to report the analytics measurements as the quiz is used. The measurements to be reported are defined using the Caliper Quiz Activity Metric Profile. The measurements reports are triggered by the corresponding user actions within the quiz. The state diagram for this activity is shown in Figure 5.7. In Figure 5.7 the seven states are linked using transitions that are marked with the user action required to trigger the transition and the corresponding list of learning analytics actions to be reported (the logical user interface is shown in Figure 5.4, 5.5 and 5.6). The seven state definitions are:

- S1 (In Content) – the user is interacting with the other, non-assessment, content. This is the active initial and end states;
- S2 (In Test) – the user has clicked on the test and started an assessment activity. Background information will be displayed e.g. rubric;
- S3 (Active Test) – the test has been started and further information may be displayed e.g. rubric informing what the user do to complete the test;
Figure 5.7 – The state diagram for the quiz interaction and Caliper reporting.

- S4 (Active Item) – the Item has been entered and the question presented to the user. Any Item rubric is also presented;
- S5 (Answered Item) – the user has started to interact with the question and has made one or more answer selections but has not yet submitted these answers for scoring;
- S6 (Completed Item) – the user has submitted the answers for scoring i.e. pressed the ‘Submit’ button. The corresponding feedback has been presented and the score for the question stored;
- S7 (Completed Test) – the user has completed the test. The score for the quiz is presented along with the appropriate test-level feedback.

The set of User Actions are:
• Enter Test – the user clicks on the test to initiate the assessment activity

• Start Test – the user confirms that they wish to start the test using an appropriate button e.g. ‘Start Quiz’;

• Exit Test – the user leaves the quiz using an appropriate button e.g. ‘End Quiz’;

• Start Item – the user enters the first, next or previous Item by pressing the appropriate button;

• Answer Item – the user selects or uses the various options available to select an answer e.g. press and radio button for a multiple choice-single response question, etc;

• Submitted Item – the user presses the ‘Submit’ button and so invokes the response processing and associated Item-level feedback;

• Submitted Test – once the full set of Item have been viewed, and perhaps answered and scored, the test itself can be completed by pressing the ‘End Quiz’ button and the corresponding quiz score and feedback presented.

The set of ‘Learning Analytics Action List’ definitions is (the order of the actions is significant and reflects the sequence of event processing):

• T1 – { ‘VIEW TEST’, ‘VIEW TESTPART’, ‘VIEW SECTION’ }. This list shows that the Quiz, Test Part and Section have been entered (at present this is a single action but when multiple test parts and multiple assessment sections are supported a more complex sequence will be created);

• T2 – { ‘ATTEMPT TEST’, ‘ATTEMPT TESTPART’, ‘ATTEMPT SECTION’ }. This indicates that the test has been started (as opposed to just entered);

• T3 – { ‘EXIT SECTION’, ‘EXIT TESTPART’, ‘EXIT TEST’ }. This list indicates that the test has been left without any questions being entered and so exit actions from the Section, Test Part and Quiz are reported;

• T4 – { ‘EXIT ITEM’, ‘SUBMIT SECTION’, ‘SUBMIT TESTPART’, ‘SUBMIT TEST’ }. This list indicates that the last Item has been completed and the quiz itself submitted for final scoring and feedback;

• T5 – { ‘EXIT SECTION’, ‘EXIT TESTPART’, ‘EXIT TEST’ }. This list indicates that the test has been completed and so exit actions from the Section, Test Part and Quiz are reported;

• I1 – { ‘VIEW ITEM’ }. This list indicates that the first question has been displayed;

• I2 – { ‘ATTEMPT ITEM ’}. This list indicates that an Item has been answered but not submitted for scoring’;
• I3 – {‘SUBMIT ITEM’}. This list indicates that the item has been scored and Item-level feedback presented;

• I4 – {‘EXIT ITEM’, ‘VIEW ITEM’}. This list indicates that the next/previous Item has been displayed

In current implementation, only a small part of this state diagram is implemented.

## 5.6 Remote Data Synchronization

Most people expect their computer applications to work as well offline as they do online. Offline use is a common scenario for EPUB documents. Data that is presented to the learner can often be stored as part of the EPUB file. But what about a scenario where the learner takes a quiz and each part of their movement through the quiz and their answers to questions need to be captured and stored offline?

There are a number of technologies emerging that can answer that question. All of the major browsers now support some form of local database storage accessible through JavaScript. To make things even easier for developers, technologies such as PouchDB have emerged that allow for easy offline storage of data and later synchronization with a compatible server whenever the application is online again.

**Code 5.1 – Example of storing learning events offline to synchronize later**

```javascript
0000
0001  var db = new PouchDB(quizresults);
0003
0004  db.put({
0005      activityId: "quiz-22321",
0006      percentageScore: ($scope.qti.finalScore / $scope.qti.MAX_ITEMS) * 100,
0007      achievedScore: $scope.qti.finalScore,
0008      Date.now()});
0010
0011
0012  db.replicate.to('http://example.com/mydb');
0013
```

The sample above is very simple but also very powerful. On line 0001 a new database instance of PouchDB is created. On line 0004 the put() method is used to store a Javascript object that contains the key/value pairs for activityId, percentageScore, achievedScore, and a current time stamp. Line 00012 is where the magic happens. PouchDB will synchronize or replicate its stored data to an online CouchDB compatible database whenever it is online again.
5.7 EDUPUB and QTI Conformance

At the time of publication the EDUPUB Conformance Programme has not been finalised. This Section will be completed in later versions of this document.

5.8 Readium and Assessment Support Compliance

At the time of publication the EDUPUB Conformance Programme has not been finalised. This Section will be completed in later versions of this document.
Appendix A – Terms & Definitions

Content Package
The key exchange object defined in the IMS Content Package v1.2 standard. A profiled version of CP is used in the IMS QTIv2.2 standard for the exchange of assessmentItems and assessmentTests.

IMS Curriculum Standards Metadata (CSM)
This is the IMS standard that is used to provide the curriculum standards annotation metadata. This metadata is used to identify the set of globally unique identifiers (this GUID can take any form e.g. URI, etc.) that are relevant to the parent resource to which the CSM is attached. The context for the GUID is also supplied and so together this information can be used to identify the external standards definition documents.

EDUPUB
From the perspective of this best practices, the EDUPUB standard is the combination IMS Learning Tools Interoperability (LTI), IMS Caliper and IMS Question and Test Interoperability (QTI) with the EPUB3 eBook format (as defined by the International Digital Publishing Forum).

EPUB
The EPUB specification is a distribution and interchange format standard for digital publications and documents create and distributed by the IDPF. EPUB defines a means of representing, packaging and encoding structured and semantically enhanced Web content including HTML5, CSS, SVG, images, and other resources, for distribution in a single-file format.

Feedback
Within QTI, this is the information that is displayed to the user once they have answered a question. Within EDUPUB the available feedback is for correct and incorrect answers (feedback for partially correct answers is not supported – this is possible within the QTI-XML). The appropriate feedback is selected as part of the QTI response processing rules.

Fill-in-Blank (FIB) Question
One of the question-types that is supported by EDUPUB. This question contains a space in which the user is expected to type a short response (typically one word but short phrases are also common). These words/phrases are then scored using the associated set of response processing rules i.e. only simple computer-based scoring is used.

Hotspot
One of the question-types that is supported by EDUPUB. This question is based upon the use of an underlying image on which there are overlaid a number of hotspot regions that the user can select. The response processing and scoring is based upon the
hotspot selected by the user.

**IDPF**

The International Digital Publishing Forum (IDPF) is the global trade and standards organization dedicated to the development and promotion of electronic publishing and content consumption. The work of the IDPF promotes the development of electronic publishing applications and products that will benefit creators of content, makers of reading systems, and consumers. The IDPF develops and maintains the EPUB content publication standard that enables the creation and transport of reflowable digital books and other types of content as digital publications that are interoperable between disparate EPUB-compliant reading devices and applications.

**IMS Caliper**

The IMS Caliper Framework provides a high-level recommendation of how Learning Systems should capture and share data around learning interactions, using existing and upcoming IMS specifications, which can be used to support and advance Learning Measurement and Analytics.

**IMS Global**

The IMS Global Learning Consortium (IMS Global) is a global, nonprofit, member organization that strives to enable the growth and impact of learning technology in the education and corporate learning sectors worldwide. IMS Global members provide leadership in shaping and growing the learning industry through community development of interoperability and adoption practice standards and recognition of the return on investment from learning and educational technology.

**IMS Learning Tools Interoperability (LTI)**

The LTI specification to allow remote tools and content to be integrated into a Learning Management System (LMS). LTI has two main components: what is traditionally named as the LMS is referred to as the “Tool Consumer” (TC) as it “consumes” the tool; and the external tool or content is called the “Tool Provider” (TP) as it “provides” the tool for use in the Tool Consumer. Example Tool Providers might include an externally hosted testing system or a server that contains externally hosted premium content.

**IMS Question & Test Interoperability (QTI)**

An information model describing data structures that are used to provide interoperability of assessments/tests and questions/items. This specification was created by IMS and the latest version available is V2.1 (released in 2012).

**IEEE LOM Metadata**

LOM, or IEEE 1484.12.1:2002, is the IEEE standard for the describing of “learning objects”. There is a long list of metadata attributes many of which have little relevance and common
usage. LOM and the IMS Global Metadata have a community origin and either the IEEE or IMS Global XML Bindings can be used to create LOM metadata instances. IMS use the IMS LOM binding.

**LTI Tool Consumer**
A Web application (such as an LMS or portal) from which users connect to LTI Tool Providers.

**LTI Tool Provider**
A Web application that provides access to resources or other functionality that can be integrated with LTI Tool Consumers.

**Manifest**
Within the IMS CP context this is the XML structure contained within the file “imsmanifest.xml” that must be included within a content package. A QTI Package is described by its manifest.

Within the EPUB3 the manifest is the container of content as allowed by the Open Document Format. This manifest has its own XML structure.

**Manifest Metadata**
This is the metadata that is defined at the top level of the manifest. LOM is used as the required format and manifest metadata is required in a QTI Content Package. This metadata is used to describe the QTI content package as a whole and not just the manifest within the package.

**Multiple Choice-Multiple Response (MC-MR) Question**
One of the question-types that is supported by EDUPUB. This question contains a set of answer options that are presented to the user who can select any combination of options. The associated response processing in EDUPUB provides feedback for the correct selection and a different feedback for the wrong answers (this feedback is the same for all wrong answers i.e. feedback for partial correctness is not supported).

**Multiple Choice-Single Response (MC-SR) Question**
One of the question-types that is supported by EDUPUB. This question contains a set of answer options that are presented to the user who must select only one. The associated response processing in EDUPUB provides feedback for the correct selection and a different feedback for the wrong answers (this feedback is the same for all wrong answers).

**Open Container Format (OCF)**
The OCF is the EPUB3 file format and processing model for encapsulating the sets of related resources that comprise one or more EPUB Publications into a single-file container.

**Open Container Package (OCP)**
This is the EPUB3 Package Document that carries bibliographic and structural metadata about an EPUB Publication, and is thus the primary source of information about how to process and display it. The Package Document is an XML document
consisting of a set of container elements, each dedicated to housing information about a particular aspect of the Publication. These containers effectively centralize metadata for the Publication, detail the individual resources that compose it and provide reading order and other information for rendering the Publication to a User.

**QTI Item**

This is the representation of an Item using QTI. Each `assessmentItem` is contained in its own XML instance file.

**QTI Section**

This is the representation of a Section using QTI. In the current version of EDUPUB only one `assessmentSection` is permitted and this is contained in the same XML instance as the `assessmentTest`.

**QTI Test**

This is the representation of a Test using QTI. Each `assessmentTest` is contained in its own XML instance file. The associated `assessmentItems` are contained in their own XML instance files.

**QTI TestPart**

An `assessmentTest` is composed of one or more `testPart`. In the current version of EDUPUB only one `testPart` is permitted and this is contained in the same XML instance as the `assessmentTest`.

**Readium**

Readium is an open source EPUB Reader under development by the Readium Foundation. The Readium Foundation develops technology to accelerate adoption of EPUB 3 and the Open Web Platform by the global digital publishing industry. Readium.org was formed in February 2013 as a non-profit membership organization. Current Readium.org projects include Readium Web (an EPUB 3 rendering engine for browser-based cloud readers) and Readium SDK (an EPUB 3 rendering engine for native apps).

**Resource**

A resource is a structure in a content package that is used to identify the physical resources that are used. In QTI the resource types are the original QTI XML instance, the set of related asset files (such as images, audio files, etc.) and the set of control files (such as XSDs) contained within the content package. Every physical file contained within the content package must be identified in a resource.

**Resource Metadata**

This is the metadata defined for a resource within the `manifest` of a content package. Each resource is expected to have its own metadata description. The metadata for a resource is based upon a profile of the LOM metadata that also includes an extension to support the QTI Metadata.

**Response Processing**

This is the set of instructions contained within the QTI-XML that
define the rules that are to be used to convert the response received from a user, when interacting with a question, into the corresponding score. Within EDUPUB the specific rules are replaced by the equivalent JavaScript response processing library that performs the task defined by the corresponding QTI-XML response processing rules.

**Schema.Org Metadata**

Schema.org is an initiative launched on 2 June 2011 by the operators of the then world’s largest search engine to “create and support a common set of schemas for structured data markup on web pages.” They propose using their ontology and Microdata in HTML5 in to mark up website content with metadata about itself. Such markup can be recognized by search engine spiders and other parsers thus gaining access to the meaning of the websites. The initiative started with a small number of formats, but the long-term goal is to support a wider range of schemas including those for education.

**Sensor API**

The IMS Sensor API is intended to support the instrumentation, collection and exchange of data from Learning Tools/Systems and associated Learning Content elements. This enables the availability of standard metrics accessed via any given Analytics Store and associated APIs. At its core, the Sensor API will support the exchange of Learning Events based on interactions with/on Learning Activities.

**True/False Question**

One of the question-types that is supported by EDUPUB. The user is presented with an either/or selection (this is equivalent to a multiple choice-single response question with just two choices).
Appendix B – Annotated LTI/EDUPUB Examples

B1 Using LTI to Launch a Tool from within an EPUB3 eBook

When the ‘QTI Assessment’ link is clicked in the ‘INTENSE! Machines’ table of contents the page shown in Figure B1.1 is displayed.

![Quiz: A Short Quiz on Intense Machines](image)

![Launch LTI Tool: Questionmark](image)

![Launch LTI Tool: YouSeeU](image)

**Figure B1.1 – The quiz launch options.**

The two LTI-based links are for:

- Questionmark – to launch into the Questionmark assessment tool;
- YouSeeU – to launch into the YouSeeU video-centric LMS.

Before this approach can be used, the Radium tool must be configured to support the Questionmark and YouSeeU tools i.e. LTI cannot be used to launch a tool that has not been registered within the Readium system. This configuration is described in Section 3.2. In the case of the Questionmark and YouSeeU systems the code in Code B1.1 is inserted into the ‘LTI_PROVIDERS’ variable in the file main.js in the ‘server’ folder.
Code B1.1 – Code to use LTI to launch the Questionmark system.

```
0000  ... 
0001  {
0003   "id": "qm",
0004   "name": "Questionmark",
0005   "key": "{...}",
0006   "secret": "...",
0007   "url": "...",
0008   "target": "window"
0009 }, {
0010   "id": "youseeu",
0011   "name": "YouSeeU",
0012   "key": "...",
0013   "secret": "...",
0014   "url": "...",
0015   "target": "window"
0016 } 
0017  ...
```

B2  Launching the Questionmark System

The code within the EPUB3 for the LTI-launch link (in the file ‘qti_assessment.xhtml’ with the OEBPS folder inside of the EPUB3 zip file) to the Questionmark system is listed in Code B2.1.

Code B2.1 – Code to use LTI to launch the Questionmark system.

```
0000 <div class="qti-item" data-ng-hide="hideIntro">
0003   <div class="qti-item-header">Launch LTI Tool: Questionmark</div>
0004   <div class="navigation">
0005     <h2>
0006       <a href="" onclick="parent.ltiLaunchTool(this, 'qm', 'qm001');" title="Questionmark">Questionmark</a>
0007     </h2>
0008   </div>
0009 </div>
0010 </div>
0011 </div>
0012```
Once the launch is completed the user is presented with the page shown in Figure B2.1.

![Figure B2.1 – The page displayed when launched into the Questionmark assessment system.](image)

This is the first question of the quiz (it should be noted that Questionmark does not support QTIv2.1 and so the same set of questions were imported into Questionmark using the QTIv1.2.1 format. The context for the LTI Tool Consumer code within Readium is described in Subsection 3.3 and Appendix F2 i.e. the ‘ltiLaunchTool()’ method call.
### B3  Launching the YouSeeU System

The code within the EPUB3 for the LTI-launch link (in the file ‘qi_assessment.xhtml’ with the OEBPS folder inside of the EPUB3 zip file) to the YouSeeU system is listed in Code B3.1.

**Code B3.1 – Code to use LTI to launch the YouSeeU system.**

```html
0000 <div class="qti-item" data-ng-hide="hideIntro">
0003  <div class="qti-item-header">Launch LTI Tool: YouSeeU</div>
0005  <h2>
0006    <a href="" onclick="parent.ltiLaunchTool(this, 'youseeu', 'ysu');" title="YouSeeU">YouSeeU</a>
0010  </h2>
0012 </div>
```

Once the launch is completed the user is presented with the page shown in Figure B3.1.

![Figure B3.1 – The page displayed when launched into the YouSeeU system.](image-url)
Appendix C – Annotated Caliper/EDUPUB Examples

The Readium Caliper Sensor is used to instrument key events inside Readium, while the user is engaging with the EPUB3 content. Some typically examples follow, from the file lib/EpubReader.js, in Code C1.1:

**Code C1.1 – Using the Sensor API in Readium.**

```javascript
var installReaderEventHandlers = function(){
  // Set handlers for click events
  $('#previous-page-btn').unbind("click");
  $('#previous-page-btn').on("click", function () {
    readium.reader.openPageLeft();
    ReadiumSensor.measureActivity(ReadingActions.READ_PAGE,
    {'turnDirection' : 'LEFT'});
    return false;
  });
  $('#next-page-btn').unbind("click");
  $('#next-page-btn').on("click", function () {
    readium.reader.openPageRight();
    ReadiumSensor.measureActivity(ReadingActions.READ_PAGE,
    {'turnDirection' : 'RIGHT'});
    return false;
  });
  $('.icon-full-screen').on('click', toggleFullScreen);
  $('.icon-library').on('click', function(){
    $(window).trigger('loadlibrary');
    ReadiumSensor.measureActivity(ReadingActions.VIEW_LIBRARY,
    {});
    return false;
  });
};
```

As depicted in Appendix F, the ReadiumSensor establishes a common LearningContext for the LTI Session and uses it in all Caliper measure calls. It should be noted that the userId and courseId are made available to the Sensor by the LTI Provider module/code in the Readium EPUB3 Reader.

**Code C1.2 – The ReadiumSensor learning context code.**

```javascript
var measureActivity = function(action, activityContextDetail) {
  var learningContext = {
    'personId': window.readiumSensor.context.userId,
    'courseId': window.readiumSensor.context.courseId
  };
  ... 
  measure(action, learningContext, activityContextDetail, Date.now());
};
```
Appendix D – Annotated Embedded QTI/EDUPUB Example

D1 Overview of the Example

The example used for the embedded quiz is based upon on the EPUB3 publication “INTENSE! Machines” from Carson Dellosa Publishing. The original EPUB3 eBook was supplied to IMS Global. This EPUB3 was then processed to add the LTI links and the embedded quiz. The extended EPUB3 was then made available through: i) the IMS Cloud and the hosted Moodle installation; and ii) as a standalone EPUB3.

When the user logs into the Moodle instance, the list of available EPUB3 resources is displayed as shown in Figure D1.1.

When one of the icons for the ‘INTENSE! Machines’ eBook is clicked, the Readium Reader is activated (each EPUB3 resource is an LTI link to the Readium Reader which then opens the identified EPUB3 file) at the page shown in Figure D1.2.
If the TOC icon is pressed in Figure D1.2 then the view shown in Figure D1.4 is displayed. Alternatively, the ‘INTENSE! Machines’ EPUB3 can be downloaded and opened using the Readium app plug-in for the Chrome browser. When this app is used your local library is displayed: an example is sown in Figure D1.3. When the ‘INTENSE! Machines’ EPUB3 is clicked and the eBook is opened in Table of Contents (TOC) mode the book is displayed as shown in Figure D1.4.

In the TOC, the link for the embedded quiz is shown fourth from the bottom i.e. ‘QTI Assessment’. Interaction with the eBook is now through either the TOC or the next (‘>’) or previous (‘<’) page icons.
Figure D1.3 – Desktop Readium library view.

Figure D1.4 – Table of contents for the ‘INTENSE! Machines’ eBook.
When the original ‘INTENSE! Machines’ EPUB3 file is unzipped the contents structure is (italics denote a folder):

mimetype

META-INF
   container.xml

OEBPS
   audio
      bcover.xhtml
      config.js
      credit.xhtml
   css
      fcover.xhtml
   fonts
      images
      js
      license.xhtml
   nav.xhtml
   package.opf
   page_000.xhtml page_001.xhtml page_002.xhtml page_003.xhtml
   page_004.xhtml page_005.xhtml page_006.xhtml page_007.xhtml
   page_008.xhtml page_009.xhtml page_010.xhtml page_011.xhtml
   page_012.xhtml page_013.xhtml page_014.xhtml page_015.xhtml
   page_016.xhtml page_017.xhtml page_018.xhtml page_019.xhtml
   page_020.xhtml page_021.xhtml page_022.xhtml page_023.xhtml
   page_024.xhtml page_025.xhtml page_026.xhtml page_027.xhtml
   page_028.xhtml page_029.xhtml page_030.xhtml page_031.xhtml
   page_032.xhtml
   smil
   submarine
   title.xhtml
D2 The User Experience for the Quiz

When the ‘QTI Assessment’ link is pressed, the page shown in Figure D2.1 is displayed.

![Quiz: A Short Quiz on Intense Machines](image)

**Figure D2.1** – The opening quiz page.

When the ‘Start Quiz’ button is pressed the first question as shown in Figure D2.2 is displayed. The initial question rendering for this True/False question is shown in Figure D2.2a. When the user selects the ‘True’ option and presses the ‘Submit Answer’ the display is changed to provide the feedback for the attempt as shown in Figure D2.2b.

![Number of Engines on a Jumbo Jet](image)

(a) Question 1 ready to be answered. (b) Question 1 feedback for a correct answer.

**Figure D2.2** – Display of question 1 (True/False).
(a) Question 2 ready to be answered.  

(b) Question 5 answered.

**Figure D2.3 – Display of questions 2 and 5 (Multiple Choice-Single Response).**

The formats for the two multiple choice-single response questions (2 and 5) are shown in Figure D2.3. Figure 2.3a shows the situation when question 2 has been displayed and the ‘Submit Answer’ button pressed without making a selection. Figure 2.3b shows the display for question 5 when one selection has been made.

(a) Question 3 feedback for a correct answer.  

(b) Question 4 feedback for a wrong answer.

**Figure D2.4 – Display of question 3 (Multiple Choice-Single Response) and question 4 (Fill-in-Blank).**
The formats for questions 3 (multiple choice-multiple response) and 4 (fill-in-blank) are shown in Figure D2.4. Figure D2.4a shows the feedback when the user has submitted the correct answer. Figure D2.4b shows the feedback when the user has submitted the wrong answer to the fill-in-blank question.

**Figure D2.5 – Display of question 6 (Hotspot).**

The display for the Question 6 (hotspot) is shown in Figure D2.5. When the ‘Finish’ button is pressed the final test summary page is displayed as shown in Figure D2.6.

**Figure D2.6 – Display of the completion page for the quiz.**
The test summary page provides the overall percentage score for the quiz (expressed as the percentage of the number correct against total number of questions).

**D3 The Definitive Source XML for the Quiz**

The QTI-based quiz is supplied as an IMS Content Package. The required ‘imsmanifest.xml’ file explains the structure of the QTI package with the essential features being:

- Each QTI Item is contained within its own XML instance file;
- The Quiz QTI Test file is contained within its own XML instance file;
- Every file in the package has its own resource definition. The `<dependency>` element is used to define the dependencies between these resource e.g. the test resource has a dependency on each of the item resource.

Once the example file (‘machinesepub3qyuiz01v1.zip’) has been unzipped the following file structure is revealed:

*Images*

    - Chopperv1.jpg

    imsmanifest.xml

*Items*

    - machinequiz01_item01.xml
    - machinequiz01_item02.xml
    - machinequiz01_item03.xml
    - machinequiz01_item04.xml
    - machinequiz01_item05.xml
    - machinequiz01_item06.xml

*Tests*

    - machineepub3quiz01_test01.xml

The key points about this structure are:

- A total of eight XML files and one image file are supplied;
- The required manifest file (“imsmanifest.xml”) is present (this conforms to the IMS CP XML);
- Each of the six Item QTI-XML instance files are collected in the ‘Items’ folder;
- The single Test QTI-XML instance file is collected in the ‘Tests’ folder;
- A single image file (used with the hotspot Item) is stored in the ‘Images’ folder.
D3.1 The Manifest CP-XML

The contents of the “imsmanifest.xml” file are listed in Code D3.1.

Code D3.1 – The CP manifest CP-XML.

```xml
<manifest identifier="Manifest_MACHINEEPUB3QUIZ01"
  xmlns="http://www.imsglobal.org/xsd/imscp_v1p1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="...">
  <metadata>
    <schema>QTIv2.1 Package</schema>
    <schemaversion>1.0.0</schemaversion>
    <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
      <educational>
        <learningResourceType>
          <source>IMS</source>
          <value>QTI Package</value>
        </learningResourceType>
      </educational>
      <general>
        <identifier>Manifest_MACHINEEPUB3QUIZ01</identifier>
        <title>EPUB3/QTI Quiz 1 Proof-of-Concept Example. This quiz is added to the Carson Dellosa Publishing eBook 'INTENSE! Machines'</title>
      </general>
      <lifeCycle>
        <contribute/>
        <version>
          <string>Final 1.0</string>
        </version>
      </lifeCycle>
      <rights>
        <copyrightAndOtherRestrictions>
          <source>LOMv1.0</source>
          <value>yes</value>
        </copyrightAndOtherRestrictions>
        <description>
          <string>2014 IMS Global Learning Consortium Inc.</string>
        </description>
      </rights>
    </metadata>
    <organizations/>
    <resources>
      <resource type="imsqti_test_xmlv2p1" identifier="ResourceTest01"
        href="Tests/machineepub3quiz01_test01v1.xml">
        <metadata>
          <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
```

IMS Global
<general>
  <identifier>
    <entry>MACHINE-QUIZ01</entry>
  </identifier>
  <title>
    <string>A Short Quiz on Intense Machines</string>
  </title>
</general>

<lifeCycle>
  <contribute/>
  <version>
    <string>Final 1.0</string>
  </version>
</lifeCycle>

<educational>
  <description>
    <string>EPUB3/QTI Example Quiz 1</string>
  </description>
</educational>

<qtiMetadata
  xmlns="...">
  <feedbackType>nonadaptive</feedbackType>
  <solutionAvailable>false</solutionAvailable>
</qtiMetadata>

</lom>

<file href="Tests/machinepub3quiz01_test01v1.xml"/>

<dependency identifierref="ResourceItem01"/>
<dependency identifierref="ResourceItem02"/>
<dependency identifierref="ResourceItem03"/>
<dependency identifierref="ResourceItem04"/>
<dependency identifierref="ResourceItem05"/>
<dependency identifierref="ResourceItem06"/>

<resource type="imsqti_item_xmlv2p1" identifier="ResourceItem01"
  href="Items/machinequiz01_item01v1.xml">
  <metadata>
    <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
      <general>
        <identifier>
          <entry>MACHINE-ITEM01</entry>
        </identifier>
        <title></title>
      </general>
      <lifeCycle>
        <contribute/>
        <version>
          <string>Final 1.0</string>
        </version>
      </lifeCycle>
      <description>
        <string>True/False Question on Number of Engines on a Jumbo Jet.</string>
      </description>
      <educational>
        <description>
          <string>...</string>
        </description>
      </educational>
    </lom>
  </metadata>
</resource>
<feedbackType>nonadaptive</feedbackType>  
<solutionAvailable>false</solutionAvailable>  
</qtiMetadata>
</lom>
</metadata>
</file>

<resource type="imsqti_item_xmlv2p1" identifier="ResourceItem02" href="Items/machinequiz01_item02v1.xml">
  <metadata>
    <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
      <general>
        <identifier>
          <entry>MACHINE-ITEM02</entry>
        </identifier>
        <title>
          <string>Machines Used to Collect Kernels of Corn</string>
        </title>
      </general>
      <lifeCycle>
        <contribute/>
        <version>
          <string>Final 1.0</string>
        </version>
      </lifeCycle>
      <educational>
        <description>
          <string>MC-SR question on machines used to collect kernels of corn.</string>
        </description>
        <qtiMetadata xmlns="">
          <interactionType>choiceInteraction</interactionType>
          <feedbackType>nonadaptive</feedbackType>
          <solutionAvailable>false</solutionAvailable>
        </qtiMetadata>
      </educational>
    </lom>
  </metadata>
</resource>

<resource type="imsqti_item_xmlv2p1" identifier="ResourceItem03" href="Items/machinequiz01_item03v1.xml">
  <metadata>
    <lom xmlns="http://ltsc.ieee.org/xsd/LOM">
      <general>
        <identifier>
          <entry>MACHINE-ITEM03</entry>
        </identifier>
        <title>
          <string>Machines Used to Fly</string>
        </title>
      </general>
      <lifeCycle>
        <contribute/>
        <version>
        </version>
      </lifeCycle>
      <educational>
        <description>
          <string>MC-SR question on machines used to collect kernels of corn.</string>
        </description>
        <qtiMetadata xmlns="">
          <interactionType>choiceInteraction</interactionType>
          <feedbackType>nonadaptive</feedbackType>
          <solutionAvailable>false</solutionAvailable>
        </qtiMetadata>
      </educational>
    </lom>
  </metadata>
</resource>
<string>Final 1.0</string>

MC-MR question on machines used to fly.

<interactionType>choiceInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>chooseSelectInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailable>

<interactionType>textEntryInteraction</interactionType>
<feedbackType>nonadaptive</feedbackType>
<solutionAvailable>false</solutionAvailability>
<title>Machines Using Railway Tracks</title>

<description>MC-SR question on machines using railway tracks.</description>

<interactionType>choiceInteraction</interactionType>

<feedbackType>nonadaptive</feedbackType>

<solutionAvailable>false</solutionAvailable>
The key points are:

a) The manifest-level metadata is given in lines 0009-0042. This metadata is the learning resource type (lines 0011-0014), the manifest identifier (lines 0017-0019), the package title (0020-0023), the version number (0026-0031) and the copyright statement (0032-0040);

b) The test resource is given in lines 0048-0083. This includes the dependencies on the resources for the six Items (lines 0078-0083). The metadata for the resource is defined in lines (0050-0076) and includes the identifier for the Item (lines 0053-0055), the title (lines 0056-0058), the version (lines 0062-0064), the description (lines 0066-0070) and the QTI-specific information about feedback and solution availability (lines 0071-0074). Line 0077 defines the actual Test XML instance;

c) The resources for question 1 are given in lines 0086-0117. The associated resource metadata is defined in lines (0088-0115) and includes the identifier for the Item (lines 0091-0055), the title (line 0094), the version (lines 0098-0100), the description (lines 0103-0108) and the QTI-specific information about feedback and solution availability (lines 0109-0113). Line 0116 defines the actual Item XML instance;
d) The resources for question 2 are given in lines 0119-0152, for question 3 in lines 0154-0185, for question 4 in lines 0187-0219 and for question 5 in lines 0221-0253

e) There are two resources for question 6 (the Item XML instance and the background image). These resources are given in lines 0255-0316.

While the manifest is required to define the total composition of the QTI-package, none of the information is used in the generation of the rendered test i.e. none of the information in the manifest is displayed in Figures D2.2, D2.3, D2.4, D2.5 and D2.6.

D3.2 The Test QTI-XML

The contents of the assessmentTest XML file are listed in Code D3.2.

<table>
<thead>
<tr>
<th>Code D3.2 – Test QTI-XML.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
| 0001 |   <outcomeDeclaration identifier="TOTAL_SCORE" cardinality="single" baseType="integer">
| 0002 |       <defaultValue>
| 0003 |         <value>0</value>
| 0004 |       </defaultValue>
| 0005 |   </outcomeDeclaration>
| 0006 |   <outcomeDeclaration identifier="TOTAL_MAXSCORE" cardinality="single" baseType="integer">
| 0007 |       <defaultValue>
| 0008 |         <value>1</value>
| 0009 |       </defaultValue>
| 0010 |   </outcomeDeclaration>
| 0011 |   <outcomeDeclaration identifier="TOTAL_MINSCORE" cardinality="single" baseType="integer">
| 0012 |       <defaultValue>
| 0013 |         <value>0</value>
| 0014 |       </defaultValue>
| 0015 |   </outcomeDeclaration>
| 0016 |   <outcomeDeclaration identifier="MASTERY_SCORE" cardinality="single" baseType="integer">
| 0017 |       <defaultValue>
| 0018 |         <value>3</value>
| 0019 |       </defaultValue>
| 0020 |   </outcomeDeclaration>
| 0021 |   <outcomeDeclaration identifier="TOTAL_NUMBERINCORRECT" cardinality="single" baseType="integer">
| 0022 |       <defaultValue>
| 0023 |         <value>0</value>
| 0024 |       </defaultValue>
| 0025 |   </outcomeDeclaration> |
<outcomeDeclaration identifier="TOTAL_NUMBERCORRECT"
cardinality="single" baseType="integer">
  <defaultValue>
    <value>0</value>
  </defaultValue>
</outcomeDeclaration>

<outcomeDeclaration identifier="TOTAL_NUMBERPRESENTED"
cardinality="single" baseType="integer">
  <defaultValue>
    <value>0</value>
  </defaultValue>
</outcomeDeclaration>

<outcomeDeclaration identifier="TOTAL_NUMBERSELECTED"
cardinality="single" baseType="integer">
  <defaultValue>
    <value>0</value>
  </defaultValue>
</outcomeDeclaration>

<outcomeDeclaration identifier="TOTAL_NUMBERRESPONDED"
cardinality="single" baseType="integer">
  <defaultValue>
    <value>0</value>
  </defaultValue>
</outcomeDeclaration>

<outcomeDeclaration identifier="FEEDBACK" cardinality="single"
  baseType="identifier"/>

<testPart identifier="MACHINE-QUIZ01-TP1" navigationMode="linear"
  submissionMode="individual">
  <assessmentSection identifier="MACHINE-QUIZ01-TP1-S1" title="Section 1"
    visible="true">
    <rubricBlock view="candidate">
      Try all of the following questions.
    </rubricBlock>
    <assessmentItemRef identifier="MACHINE-QUIZ01-ITEM01"
      required="true" fixed="true"
      href="../Items/machinequiz01-item01v1.xml"/>
    <assessmentItemRef identifier="MACHINE-QUIZ01-ITEM02"
      required="true" fixed="true"
      href="../Items/machinequiz01-item02v1.xml"/>
    <assessmentItemRef identifier="MACHINE-QUIZ01-ITEM03"
      required="true" fixed="true"
      href="../Items/machinequiz01-item03v1.xml"/>
    <assessmentItemRef identifier="MACHINE-QUIZ01-ITEM04"
      required="true" fixed="true"
      href="../Items/machinequiz01-item04v1.xml"/>
    <assessmentItemRef identifier="MACHINE-QUIZ01-ITEM05"
      required="true" fixed="true"
      href="../Items/machinequiz01-item05v1.xml"/>
  </assessmentSection>
</testPart>
<assessmentItemRef identifier="MACHINE-QUIZ01-ITEM06" required="true" fixed="true" href="../Items/machinequiz01-item06v1.xml"/>

</assessmentSection>
</testPart>

<outcomeProcessing>
  <setOutcomeValue identifier="TOTAL_SCORE">
    <sum>
      <testVariables variableIdentifier="SCORE" />
    </sum>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_MAXSCORE">
    <sum>
      <testVariables variableIdentifier="MAXSCORE" />
    </sum>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_MINSCORE">
    <sum>
      <testVariables variableIdentifier="MINSCORE" />
    </sum>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_NUMBERINCORRECT">
    <numberIncorrect/>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_NUMBERCORRECT">
    <numberCorrect/>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_NUMBERRESPONDED">
    <numberResponded/>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_NUMBERSELECTED">
    <numberSelected/>
  </setOutcomeValue>

  <setOutcomeValue identifier="TOTAL_NUMBERPRESENTED">
    <numberPresented/>
  </setOutcomeValue>
</outcomeProcessing>

<testFeedback access="atEnd" outcomeIdentifier="FEEDBACK" showHide="show" identifier="Passed" title="Passed test feedback">

  <p>Well done. You have passed the quiz.</p>
  <p>You scored <printedVariable identifier="TOTAL_SCORE"/> marks out of a maximum of <printedVariable identifier="TOTAL_MAXSCORE"/> marks.</p>
</testFeedback>
The key points are:

a) The title for the test is given in line 0007. This is the information that is displayed as the ‘Quiz Title’ as shown in Figure D2.1;

b) A set of outcomes variables is defined for the test (see lines 0009-0073). This consists of 10 variables. The three variables ‘TOTAL_SCORE’ (lines 0009-0014), ‘TOTAL_MAXSCORE’ (lines 0016-0021) and ‘TOTAL_MINSCORE’ (lines 0023-0028) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The variable ‘MASTERY_SCORE’ (lines 0030-0035) is used to define the score at or above mastery is achieved. The five variables ‘TOTAL_NUMBERINCORRECT’ (lines 0037-0042), ‘TOTAL_NUMBERCORRECT’ (lines 0044-0049), ‘TOTAL_NUMBERPRESENTED’ (lines 0051-0056), ‘TOTAL_NUMBERSELECTED’ (lines 0058-0063) and ‘TOTAL_NUMBERRESPONDED’ (Lines 0065-0070) provide contextual information about the questions presented to the user. The final variable ‘FEEDBACK’ (lines 0072-0073) is used to determine the nature of the test-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any quiz and so do not need to be defined. They have been included in the QTI-XML for completeness);

c) The single testPart is defined in lines 0075-0105. This contains the single assessmentSection that is defined in lines 0078-0104;

d) The test-level rubricBlock is given in lines 0081-0083. This is the information that must be displayed in the test-level rubric block space on the opening page of the quiz as per Figure D2.1;

e) The test-level outcomes processing is presented in lines 0107-0146. This is the set of processing rules that must be used to populate the appropriate values in the set of outcomes variables. See the QTIv2.1 specification for the interpretation of the actual processing rules themselves. In the current version of EDUPUB these processing rules are assumed and are not parsed during the transformation. They have been included in the QTI-XML for completeness;

f) The test-level feedback is given in lines 0150-0168. If the test has been passed then the feedback given in lines 0150-0157 is presented. If the test has been failed the feedback given
in lines 0161-0168 is presented. This is the feedback that is presented in the final page displayed as shown in Figure D2.6.

**D3.3 The Six Item QTI-XML**

The contents of the `assessmentItem` for the question 1 XML file are listed in Code D3.3.

**Code D3.3 – True/false Item QTI-XML (question 1).**

```
<assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"
                 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                 xsi:schemaLocation="...">
  <identifier>MACHINE-ITEM01</identifier>
  <title>Number of Engines on a Jumbo Jet</title>
  <label>True/False Question on Number of Engines on a Jumbo Jet</label>
  <xml:lang>en-US</xml:lang>
  <timeDependent>true</timeDependent>
  <adaptive>false</adaptive>
  <responseDeclaration identifier="RESPONSE" cardinality="single">
    <correctResponse>
      <value>true</value>
    </correctResponse>
  </responseDeclaration>
  <outcomeDeclaration identifier="SCORE" cardinality="single">
    <defaultValue>
      <value>0.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="MAXSCORE" cardinality="single">
    <defaultValue>
      <value>1.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="MINSCORE" cardinality="single">
    <defaultValue>
      <value>-1.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="FEEDBACK" cardinality="single"/>

  <itemBody>
    <rubricBlock view="candidate">
      <p>Choose <em>one</em> of the two options.</p>
    </rubricBlock>
  </itemBody>
  <choiceInteraction responseIdentifier="RESPONSE" shuffle="false">
    <maxChoices>1</maxChoices>
  </choiceInteraction>
</assessmentItem>
```
A Boeing 747, also known as a Jumbo Jet, has four engines.

True

False

Correct answer. Two engines under each wing.

No, wrong answer. A Jumbo Jet does have four engines. Two under each wing.

The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0017. The associated correct answer is defined a ‘true’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0019-0024), ‘MAXSCORE’ (lines 0026-0031 and ‘MINSCORE’ (lines 0033-0038) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0040-0041) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

d) The actual question is defined in lines 0043-0063. This consists of the Item-level rubric (lines 0044-0046), the choice interaction (lines 0048-0060) with the prompt given in lines 0051-0054 and the two choices in lines 0052-0055;

e) The response-processing template to be used for this Item is given in line 0064. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;
f) The Item-level feedback is given in lines 0066-0076. If the Item has been answered correctly, the feedback given in lines 0066-0069 is presented otherwise the feedback given in lines 0071-0075 is presented.

The contents of the assessmentItem for the question 2 XML file are listed in Code D3.4.

Code D3.4 – Multiple choice-single response Item QTI-XML (question 2).

```
<assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="...">
  <identifier>MACHINE-ITEM02</identifier>
  <title>Machines Used to Collect Kernels of Corn</title>
  <label>MC-SR question on machines used to collect kernels of corn.</label>
  <xml:lang>en-US</xml:lang>
  <timeDependent>true</timeDependent>
  <adaptive>false</adaptive>

  <responseDeclaration identifier="RESPONSE" cardinality="single"
    baseType="identifier">
    <correctResponse>
      <value>B</value>
    </correctResponse>
  </responseDeclaration>

  <outcomeDeclaration identifier="SCORE" cardinality="single"
    baseType="float" masteryValue="0.0">
    <defaultValue>
      <value>0.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="MAXSCORE" cardinality="single"
    baseType="float">
    <defaultValue>
      <value>1.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="MINSCORE" cardinality="single"
    baseType="float">
    <defaultValue>
      <value>-1.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="FEEDBACK" cardinality="single"
    baseType="identifier"/>

  <itemBody>
    <rubricBlock view="candidate">
      <p>Choose <em>one</em> of the five options.</p>
    </rubricBlock>

    <choiceInteraction responseIdentifier="RESPONSE"
      shuffle="false" maxChoices="1">
      <prompt>
```
Which machine is used to collect the kernels of corn in a field?

- Submarine.  
- Combine.  
- Chopper.  
- Locomotive.  
- Helicopter.

The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0017. The associated correct answer is defined as ‘B’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0019-0024), ‘MAXSCORE’ (lines 0026-0031 and ‘MINSCORE’ (lines 0033-0038) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0040-0041) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

d) The actual question is defined in lines 0043-0068. This consists of the Item-level rubric (lines 0044-0046), the choice interaction (lines 0048-0066) with the prompt given in lines 0050-0053 and the five choices in lines 0055-0064;
e) The response-processing template to be used for this Item is given in line 0070. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

f) The Item-level feedback is given in lines 0072-0080. If the Item has been answered correctly, the feedback given in lines 0072-0075 is presented otherwise the feedback given in lines 0077-0081 is presented.

The contents of the assessmentItem for question 3 XML file are listed in Code D3.5.

**Code D3.5 – Multiple choice-multiple response Item QTI-XML (question 3).**

```xml
<assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"
                 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                 xsi:schemaLocation="..." identifier="MACHINE-ITEM03"
                 title="Machines Used to Fly"
                 label="MC-MR question on machines used to fly."
                 xml:lang="en-US"
                 timeDependent="true"
                 adaptive="false">
  <responseDeclaration identifier="RESPONSE" cardinality="single"
                         baseType="identifier">
    <correctResponse>
      <value>B</value>
      <value>E</value>
    </correctResponse>
  </responseDeclaration>
  <outcomeDeclaration identifier="SCORE" cardinality="single"
                       baseType="float" masteringValue="0.0">
    <defaultValue>
      <value>0.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="MAXSCORE" cardinality="single"
                        baseType="float">
    <defaultValue>
      <value>1.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="MINSCORE" cardinality="single"
                        baseType="float">
    <defaultValue>
      <value>-1.0</value>
    </defaultValue>
  </outcomeDeclaration>
  <outcomeDeclaration identifier="FEEDBACK" cardinality="single"
                        baseType="identifier"/>
  <itemBody>
```
The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0018. The associated correct answer is defined as ‘B’ and ‘E’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0020-0025), ‘MAXSCORE’ (lines 0027-0032 and ‘MINSCORE’ (lines 0034-0039) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0041-0042) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;
d) The actual question is defined in lines 0044-0069. This consists of the Item-level rubric (lines 0046-0049), the choice interaction (lines 0051-0067) with the prompt given in lines 0053-0055 and the five choices in lines 0056-0065;

e) The response-processing template to be used for this Item is given in line 0071. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

f) The Item-level feedback is given in lines 0073-0082. If the Item has been answered correctly, the feedback given in lines 0073-0076 is presented otherwise the feedback given in lines 0078-0082 is presented.

The contents of the assessmentItem for question 4 XML file are listed in Code D3.6.

```xml
<assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="...">
  identifier="MACHINE-ITEM04"
  title="Chopper is a type of?"
  label="Fill-In Blank Question on a Chopper"
  xml:lang="en-US"
  timeDependent="true"
  adaptive="false">
  <responseDeclaration identifier="RESPONSE" cardinality="single"
    baseType="identifier">
    <correctResponse>
      <value>motorcycle</value>
    </correctResponse>
  </responseDeclaration>

  <outcomeDeclaration identifier="SCORE" cardinality="single"
    baseType="float" masteryValue="0.0">
    <defaultValue>
      <value>0.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="MAXSCORE" cardinality="single"
    baseType="float">
    <defaultValue>
      <value>1.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="MINSCORE" cardinality="single"
    baseType="float">
    <defaultValue>
      <value>-1.0</value>
    </defaultValue>
  </outcomeDeclaration>

  <outcomeDeclaration identifier="FEEDBACK" cardinality="single">
    ...
  </outcomeDeclaration>
</assessmentItem>
```
The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0017. The associated correct answer is defined as ‘motorcycle’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0019-0024), ‘MAXSCORE’ (lines 0026-0031) and ‘MINSCORE’ (lines 0033-0038) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0040-0041) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

d) The actual question, FIB, is defined in lines 0043-0053. This consists of the Item-level rubric (lines 0045-0047), the text entry interaction (lines 0049-0051);

e) The response-processing template to be used for this Item is given in line 0055. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;
f) The Item-level feedback is given in lines 0057-0066. If the Item has been answered correctly, the feedback given in lines 0057-0060 is presented otherwise the feedback given in lines 0062-0066 is presented.

The contents of the assessmentItem for question 5 XML file are listed in Code D3.7.

**Code D3.7 – Multiple choice-single response Item QTI-XML (question 5).**

```xml
0000 0001 <assessmentItem xmlns="http://www.imsglobal.org/xsd/qti_v2p1"
0002  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
0003  xsi:schemaLocation="..."
0004  identifier="MACHINE-ITEM05"
0005  title="Machines Using Railway Tracks"
0006  label="MC-SR question on machines using railway tracks."
0007  xml:lang="en-US"
0008  timeDependent="true"
0009  adaptive="false">
0010  <responseDeclaration identifier="RESPONSE" cardinality="single"
0011    baseType="identifier">
0012    <correctResponse>
0013      <value>C</value>
0014    </correctResponse>
0015  </responseDeclaration>
0016  <outcomeDeclaration identifier="SCORE" cardinality="single"
0017    baseType="float" masteryValue="0.0">
0018    <defaultValue>
0019      <value>0.0</value>
0020    </defaultValue>
0021  </outcomeDeclaration>
0022  <outcomeDeclaration identifier="MAXSCORE" cardinality="single"
0023    baseType="float">
0024    <defaultValue>
0025      <value>1.0</value>
0026    </defaultValue>
0027  </outcomeDeclaration>
0028  <outcomeDeclaration identifier="MINSCORE" cardinality="single"
0029    baseType="float">
0030    <defaultValue>
0031      <value>-1.0</value>
0032    </defaultValue>
0033  </outcomeDeclaration>
0034  <outcomeDeclaration identifier="FEEDBACK" cardinality="single"
0035    baseType="identifier"/>
0036  <itemBody>
0037    <rubricBlock view="candidate">
0038      <p>Choose <em>one</em> of the five options.</p>
0039    </rubricBlock>
0040    <choiceInteraction responseIdentifier="RESPONSE"
0041      shuffle="false" maxChoices="1">
0042      <prompt>
```
Which machine moves on a railway track?

A) Indy Car.
B) Combine.
C) Locomotive.
D) Chopper.
E) Front-End Loader.

Correct answer. Well done.

No, wrong answer. A Locomotive uses railway tracks.

The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0017. The associated correct answer is defined as ‘C’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0019-0024), ‘MAXSCORE’ (lines 0026-0031 and ‘MINSCORE’ (lines 0033-0038) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0040-0041) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

d) The actual question is defined in lines 0043-0067. This consists of the Item-level rubric (lines 0044-0046), the choice interaction (lines 0048-0065) with the prompt given in lines 0050-0052 and the five choices in lines 0054-0063;
e) The response-processing template to be used for this Item is given in line 0069. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

f) The Item-level feedback is given in lines 0071-0080. If the Item has been answered correctly, the feedback given in lines 0071-0074 is presented otherwise the feedback given in lines 0076-0080 is presented.

The contents of the *assessmentItem* for question 6 XML file are listed in Code D.3.8.

**Code D.3.8 – Hotspot Item QTI-XML (question 6).**

```
0000 0001  <assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p1"
0002 0003     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
0004 0005     identifier="MACHINE-ITEM06"
0006 0007     title="Chopper Gas Tank"
0008 0009     label="Hotspot question on the Chopper."
0010 0011     xml:lang="en-US"
0012 0013     timeDependent="true"
0014 0015     adaptive="false">
0016 0017     <responseDeclaration identifier="RESPONSE" cardinality="single"
0018 0019     baseType="identifier">
0020 0021     <correctResponse>
0022 0023     <value>A</value>
0024 0025     </correctResponse>
0026 0027     </responseDeclaration>
0028 0029     <outcomeDeclaration identifier="SCORE" cardinality="single"
0030 0031     baseType="float" masteryValue="0.0">
0032 0033     <defaultValue>
0034 0035     <value>0.0</value>
0036 0037     </defaultValue>
0038 0039     </outcomeDeclaration>
0040 0041     <outcomeDeclaration identifier="MAXSCORE" cardinality="single"
0042 0043     baseType="float">
0044 0045     <defaultValue>
0046 0047     <value>1.0</value>
0048 0049     </defaultValue>
0050 0051     </outcomeDeclaration>
0052 0053     <outcomeDeclaration identifier="MINSCORE" cardinality="single"
0054 0055     baseType="float">
0056 0057     <defaultValue>
0058 0059     <value>-1.0</value>
0060 0061     </defaultValue>
0062 0063     </outcomeDeclaration>
0064 0065     <outcomeDeclaration identifier="FEEDBACK" cardinality="single"
0066 0067     baseType="identifier"/>
0068 0069     <itemBody>
0070 0071     <rubricBlock view="candidate">
```
Choose one of the four options.

Where is the gas tank?

Correct answer. Well done.

No, wrong answer.

The key points are:

a) The title of the Item is given in line 0006;

b) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0017. The associated correct answer is defined as ‘A’;

c) Four outcomes variables are defined for the Item (see lines 0019-0041). The three variables ‘SCORE’ (lines 0019-0024), ‘MAXSCORE’ (lines 0026-0031 and ‘MINSCORE’ (lines 0033-0038) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0040-0041) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;
d) The actual question, hotspot, is defined in lines 0043-0067. This consists of the Item-level rubric (lines 0045-0047), the hotspot interaction (lines 0049-0065) with the prompt given in line 0052, the four choices in lines 0056-0063 and the background image in lines 0053-0054;

e) The response-processing template to be used for this Item is given in line 0069. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

f) The Item-level feedback is given in lines 0071-0079. If the Item has been answered correctly, the feedback given in lines 0071-0074 is presented otherwise the feedback given in lines 0076-0079 is presented.

D3.4 The Response Processing Templates

[[ ED NOTE: To be completed in a later draft. ]]
D4  The Embedded XHTML File(s)

The contents of the quiz XHTML embedded in the EPUB3 file are listed in Code D4.1.

Code D4.1 – The embedded XHTML file for the quiz.

```xml
<html xmlns="http://www.w3.org/1999/xhtml" data-ng-app="myApp">
<head>
<title>Intense Machines Quiz</title>
<meta name="viewport" content="width=1100, height=1650"/>
<link rel="stylesheet" type="text/css" href="/../../../lib/thirdparty/thickbox.css"/>
<script src="/uvu_dgm/js/lib/angular/angular.js"></script>
<script src="/uvu_dgm/js/lib/angular/angular-sanitize.js"></script>
<script src="/uvu_dgm/js/lib/angular/pouchdb-1.1.0.min.js"></script>
<script src="/uvu_dgm/js/lib/angular/pouchdb.js"></script>
<script src="/uvu_dgm/js/controllers.js"></script>
<script src="/uvu_dgm/js/services.js"></script>
<script src="/uvu_dgm/js/directives.js"></script>
<script src="/uvu_dgm/js/lib/jquery-1.11.0.js"></script>
<script src="/uvu_dgm/js/lib/intellisense.min.js"></script>
<script src="/../../../lib/thirdparty/lti.js" type="text/javascript"></script>
<script src="/../../../lib/thirdparty/jquery-migrate-1.2.1.min.js" type="text/javascript"></script>
<script src="/../../../lib/thirdparty/thickbox-compressed.js" type="text/javascript"></script>
</head>
<body id="quiz_body">
  <div class="Basic-Text-Frame" data-ng-controller="MainCtrl">
    <div class="qti-item" data-ng-hide="hideIntro">
      <div class="qti-item-header">
        Quiz: {{qti.assessmentTest.title}}
      </div>
      <div class="item-content">
        Launch LTI Tool:
      </div>
    </div>
  </div>
</body>
</html>
```
<!-- MultiChoice Multi Response -->
<div data-ng-if="qti.currItem.quizType=='choiceMulti'">
  <div class="qti-item-prompt" data-ng-bind-html="qti.currItem.assessmentItem.itemBody.choiceInteraction.prompt"></div>
  <ul class="qti-item-choices">
    <li data-ng-repeat="choice in qti.currItem.assessmentItem.itemBody.choiceInteraction.simpleChoice" data-ng-class="{correct: choice.correctResponse}">
      <label>
        <input type="checkbox" value="{{choice.identifier}}" check-list='checked_choices'/>
        {{choice.__text}}
      </label>
    </li>
  </ul>
</div>

<!-- Fill In The Blank -->
<div data-ng-if="qti.currItem.quizType=='fillInBlank'">
  <div class="qti-item-prompt" data-ng-bind-html="qti.currItem.assessmentItem.itemBody.__text"></div>
  <input type="text" data-ng-model="qti.fib_userTyped"/>
</div>

<!-- Hot Spot questions -->
<div data-ng-if="qti.currItem.quizType=='hotSpot'">
  <div class="qti-item-prompt" data-ng-bind-html="qti.currItem.assessmentItem.itemBody.__text"></div>
  <div data-ng-bind-html="qti.currItem.assessmentItem.itemBody.hotspotInteraction.prompt"></div>
  <svg width="660" height="415" viewBox="0 0 660 415"
        xmlns="http://www.w3.org/2000/svg"
        xmlns:xlink="http://www.w3.org/1999/xlink"
        data-ng-click="chopperClick($event)">
    <image xlink:href="uvudgm/images/Chopperv1.jpg"
           height="415px" width="660px"/>
    <circle cx="270" cy="190" r="50" fill="none"
           stroke="white" stroke-width="2"/>
    <circle cx="532" cy="330" r="93" fill="none"
           stroke="white" stroke-width="2"/>
    <circle cx="240" cy="315" r="80" fill="none"
           stroke="white" stroke-width="2"/>
    <circle cx="385" cy="145" r="30" fill="none"
           stroke="white" stroke-width="2"/>
  </svg>
Quiz Results

Well done. You have passed the quiz.

You scored {{qti.finalScore}} marks out of a maximum of {{qti.MAX_ITEMS}} marks.

Try again and see if you can get more correct answers.

You scored {{qti.finalScore}} marks out of a maximum of {{qti.MAX_ITEMS}} marks.

Score: {{(qti.finalScore / qti.MAX_ITEMS) * 100 | number:2}}%

Points: {{qti.finalScore}} out of {{qti.MAX_ITEMS}}

Prev
<button class="btn-approve" data-ng-disabled="submitAnswerDisabled" data-ng-click="submitAnswer()"
  data-ng="hide=qti.currItem.assessmentItem.isSubmitted">
  Submit Answer
</button>

<button class="btn-primary btn-next" data-ng-click="nextItem()" data-ng-show="showNextBtn()">
  Next
</button>

<button class="btn-approve btn-save" data-ng-click="submitQuiz()" data-ng-show="(qti.currItemIndex + 1)===qti.MAX_ITEMS">
  Finish
</button>

</div>

</div>

<script type="text/javascript">
$(window).bind("load", function () { // Wait until window load is complete
  var LTI = window.parent.LTI;
  var host = LTI.session.caliper.serviceurl;
  var apiKey = LTI.session.caliper.apikey;
  var server = "http://" + host;

  var options = {
    'apiKey': apiKey,
    'server': server,
    'sensor_id': 'com.testorg.thorTek1'
  };

  intellisense.initialize({'intellisense': options}, {});
  console.log(LTI);
  console.log("init complete..");
});
</script>
The key points are:

a) This XHTML file has been hand-created using the QTI-XML files from the content package as the source material. A number of classes have been defined for the layout of the quiz and questions. A set of AngularJS actions have been defined and these are identified using the attribute ‘data-ng-*** e.g. ‘data-ng-model’, ‘data-ng-hide’, etc;

b) The links to the external JavaScript libraries are supplied in lines 0010-0026 (these include the libraries for LTI, the Sensor API and the AngularJS framework;

c) The scaffolding for the quiz (the quiz title, buttons, etc.) is provided in lines 0031-0049;

d) The links to the two external systems, launched using LTI, are given in lines 0050-0069 (to Questionmark in lines 0050-0059 and to YouSeeU in lines 0060-0069);

e) The code for the set of questions is shown in lines 0072-0171 with True/False and Multiple Choice-Single Response in lines 0090-113, Multiple Choice-Multiple Response in lines 0116-0135, FIB in lines 0137-0143 and Hotspot in lines 0145-0169. The title for the questions is given in lines 0076-0078;

f) Control of the Item level-feedback is supplied in lines 0173-0178;

g) Control of the Test level feedback is supplied in lines 0180-0220;

h) The set of navigation buttons for the questions is supplied in lines 0222-0255;

i) The JavaScript that is used to initialise the SensorAPI is supplied in lines 0259-0282.

**D4.1 The Embedded Caliper Calls**

The declaration and initialization of the LTI global variable and the Sensor API ‘person’ and ‘course’ describe learning events are sent as listed in Code D4.2.

```javascript
var LTI = $window.parent.LTI;

intellisense.describe('person', LTI.session.user.id,
  {"firstName": LTI.session.user.firstname,
   "lastName": LTI.session.user.lastname,
   "emailAddress": LTI.session.user.email}, Date.now());

intellisense.describe('course', LTI.session.context.id,
  {"title": $scope.qti.assessmentTest.title}, Date.now());
```

When an `assessmentTest` is begun, there may be Sensor API learning events sent using the measure method that will indicate the precise beginning of a test attempt, a test part attempt, and a section attempt as listed in Code D4.1.2.
Code D4.3 – Learning events sent for test, testPart and section attempts.

```
0001 intellisense.measure('ATTEMPT_TEST', {
0003     "courseId": LTI.session.context.id,
0004     "personId": LTI.session.user.firstname},
0005     {"activityId": "quiz-22321"}
0006    );
0007 intellisense.measure('ATTEMPT_TESTPART', {
0009     "courseId": LTI.session.context.id,
0010     "personId": LTI.session.user.firstname},
0011     {"activityId": "quiz-22321"}
0012    );
0013 intellisense.measure('ATTEMPT_SECTION', {
0015     "courseId": LTI.session.context.id,
0016     "personId": LTI.session.user.firstname},
0017     {"activityId": "quiz-22321"}
0018    );
```

As each assessmentItem is encountered by the learner within the assessmentTest, testPart, or assessmentSection, there may be Sensor API learning events sent using the measure method that will indicate the precise beginning of an assessmentItem attempt as listed in Code D4.3.

Code D4.3 – Learning event posted for each attempted quiz item.

```
0001 intellisense.measure('ATTEMPT_ITEM', {
0003     "courseId": LTI.session.context.id,
0004     "personId": LTI.session.user.firstname},
0005     {"activityId": "quiz-22321"}
0006    );
```

Upon submission of the quiz by the learner for grading and results reporting, there may be a Sensor API measure event sent that will convey the appropriate context, person, and activity information as well as provide scoring and results information as listed in Code D4.4.

Code D4.4 – Learning event posted upon quiz submission.

```
0001 intellisense.measure('SUBMIT_QUIZ', {
0003     "courseId": LTI.session.context.id,
0004     "personId": LTI.session.user.firstname},
0005     {"activityId": "quiz-22321",
0006     "percentageScore": ($scope.qti.finalScore / $scope.qti.MAX_ITEMS) * 100,
0007     "achievedScore": $scope.qti.finalScore},
0008     Date.now()
0009    );
```
D5  The Embedded JSON File(s)

The contents of the quiz JSON file embedded in the EPUB3 are listed in Code D5.1.

Code D5.1– The test/quiz JSON file.

```json
{
  "assessmentTest": {
    "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "xsi:schemaLocation": "...",
    "identifier": "MACHINE-QUIZ01",
    "title": "A Short Quiz on Intense Machines",
    "outcomeDeclaration": [
      {
        "identifier": "TOTAL_SCORE",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "0"
        }
      },
      {
        "identifier": "TOTAL_MAXSCORE",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "1"
        }
      },
      {
        "identifier": "TOTAL_MINSCORE",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "0"
        }
      },
      {
        "identifier": "MASTERY_SCORE",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "4"
        }
      },
      {
        "identifier": "TOTAL_NUMBERINCORRECT",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "0"
        }
      },
      {
        "identifier": "TOTAL_NUMBERCORRECT",
        "cardinality": "single",
        "baseType": "integer",
        "defaultValue": {
          "value": "0"
        }
      }
    ]
  }
}
```
"baseType": "integer",
"defaultValue": {
  "value": "0"
}
},
  "identifier": "TOTAL_NUMBERPRESENTED",
  "cardinality": "single",
  "baseType": "integer",
  "defaultValue": {
    "value": "0"
  }
},
  "identifier": "TOTAL_NUMBERSELECTED",
  "cardinality": "single",
  "baseType": "integer",
  "defaultValue": {
    "value": "0"
  }
},
  "identifier": "TOTAL_NUMBERRESPONDED",
  "cardinality": "single",
  "baseType": "integer",
  "defaultValue": {
    "value": "0"
  }
},
  "identifier": "FEEDBACK",
  "cardinality": "single",
  "baseType": "identifier"
],
"testPart": {
  "identifier": "MACHINE-QUIZ01-TP1",
  "navigationMode": "linear",
  "submissionMode": "individual",
  "assessmentSection": {
    "identifier": "MACHINE-QUIZ01-TP1-S1",
    "title": "Section 1",
    "visible": "true",
    "rubricBlock": {
      "__text": "<p>Try <em>all</em> of the following questions.</p>",
      "view": "candidate"
    },
    "assessmentItemRef": [
      { "identifier": "MACHINE-QUIZ01-ITEM01",
        "required": "true",
        "fixed": "true",
        "href": "/Items/machinequiz01-item01v1.xml"
      },
      { "identifier": "MACHINE-QUIZ01-ITEM02",
        "required": "true",
        "fixed": "true",
        "href": "/Items/machinequiz01-item02v1.xml"
      }
    ]
  }
}
"fixed": "true",
"href": "../Items/machinequiz01-item03v1.xml"
),

"identifier": "MACHINE-QUIZ01-ITEM04",
"required": "true",
"fixed": "true",
"href": "../Items/machinequiz01-item04v1.xml"
},

"identifier": "MACHINE-QUIZ01-ITEM05",
"required": "true",
"fixed": "true",
"href": "../Items/machinequiz01-item05v1.xml"
},

"identifier": "MACHINE-QUIZ01-ITEM06",
"required": "true",
"fixed": "true",
"href": "../Items/machinequiz01-item06v1.xml"
}
]

"outcomeProcessing": {

"setOutcomeValue": [

"identifier": "TOTAL_SCORE",
"sum": {
"testVariables": {
"variableIdentifier": "SCORE"
}
}
},

"identifier": "TOTAL_MAXSCORE",
"sum": {
"testVariables": {
"variableIdentifier": "MAXSCORE"
}
}
},

"identifier": "TOTAL_MINSCORE",
"sum": {
"testVariables": {
"variableIdentifier": "MINSCORE"
}
}
]

"identifier": "TOTAL_NUMBERINCORRECT",
The key points are:

a) The JSON in Code 5.1 is produced by transforming the equivalent QTI-XML for the quiz (see Code D3.2);
b) The title for the test is given in line 0007. This is the information that is displayed as the ‘Quiz Title’ as shown in Figure D2.1;

c) A set of outcomes variables is defined for the test (see lines 0008-0086). This consists of 10 variables. The three variables ‘TOTAL_SCORE’ (lines 0009-0016), ‘TOTAL_MAXSCORE’ (lines 0017-0024 and ‘TOTAL_MINSCORE’ (lines 0025-0032) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The variable ‘MASTERY_SCORE’ (lines 0033-0040) is used to define the score at or above mastery is achieved. The five variables ‘TOTAL_NUMBERINCORRECT’ (lines 0041-0048), ‘TOTAL_NUMBERCORRECT’ (lines 0044-0049), ‘TOTAL_NUMBERPRESENTED’ (lines 0049-0064), ‘TOTAL_NUMBERSELECTED’ (lines 0065-0072) and ‘TOTAL_NUMBERRESPONDED’ (Lines 0073-0080) provide contextual information about the questions presented to the user. The final variable ‘FEEDBACK’ (lines 0081-0085) is used to determine the nature of the test-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any quiz and so do not need to be defined. They have been included in the QTI-XML for completeness;

d) The single testPart is defined in lines 0087-0140. This contains the single assessmentSection that is defined in lines 0091-0139 and the associated set of Item references are given in lines 0100-0138;

e) The test-level rubricBlock is given in lines 0095-0099. This is the information that must be displayed in the test-level rubric block space on the opening page of the quiz as per Figure D2.1;

f) The test-level outcomes processing is presented in lines 0141-0188. This is the set of processing rules that must be used to populate the appropriate values in the set of outcomes variables. See the QTIv2.1 specification for the interpretation of the actual processing rules themselves. In the current version of EDUPUB these processing rules are assumed and are not parsed during the transformation. They have been included in the QTI-XML for completeness;

g) The test-level feedback is given in lines 0189-0218. If the test has been passed then the feedback given in lines 0190-0203 is presented. If the test has been failed the feedback given in lines 0204-0217 is presented. This is the feedback that is presented in the final page displayed as shown in Figure D2.6.
The contents of the question 1 JSON file embedded in the EPUB3 are listed in Code D5.2.

**Code D5.2 – The question 1 JSON file.**

```json
{
    "assessmentItem": {
        "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
        "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
        "xsi:schemaLocation": "...",
        "identifier": "MACHINE-ITEM01",
        "title": "Number of Engines on a Jumbo Jet",
        "label": "True/False Question on Number of Engines on a Jumbo
Jet",
        "xml:lang": "en-US",
        "timeDependent": "true",
        "adaptive": "false",
        "responseDeclaration": {
            "identifier": "RESPONSE",
            "cardinality": "single",
            "baseType": "identifier",
            "correctResponse": {
                "value": "true"
            }
        },
        "outcomeDeclaration": {
            "identifier": "SCORE",
            "cardinality": "single",
            "baseType": "float",
            "masteryValue": "0.0",
            "defaultValue": {
                "value": "0.0"
            }
        },
        "outcomeDeclaration": {
            "identifier": "MAXSCORE",
            "cardinality": "single",
            "baseType": "float",
            "defaultValue": {
                "value": "1.0"
            }
        },
        "outcomeDeclaration": {
            "identifier": "MINSCORE",
            "cardinality": "single",
            "baseType": "float",
            "defaultValue": {
                "value": "-1.0"
            }
        },
        "itemBody": {
            "rubricBlock": {
```
The key points are:

a) The JSON in Code 5.2 is produced by transforming the equivalent QTI-XML for the question 1 (see Code D3.3);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0013-0020. The associated correct answer is defined a ‘true’;
d) Four outcomes variables are defined for the Item (see lines 0021-0052). The three variables ‘SCORE’ (lines 0022-0030), ‘MAXSCORE’ (lines 0031-0038) and ‘MINSCORE’ (lines 0039-0046) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0047-0051) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question is defined in lines 0053-0077. This consists of the Item-level rubric (lines 0054-0057), the choice interaction (lines 0058-0076) with the prompt given in lines 0062-0063 and the two choices in lines 0064-0075;

f) The response-processing template to be used for this Item is given in lines 0078-0080. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

g) The Item-level feedback is given in lines 0081-0098. If the Item has been answered correctly, the feedback given in lines 0082-0089 is presented otherwise the feedback given in lines 0090-0097 is presented.

The contents of the question 2 JSON file embedded in the EPUB3 are listed in Code D5.3.

Code D5.3 – The question 2 JSON file.

```json
{
  "assessmentItem": {
    "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "xsi:schemaLocation": "",
    "identifier": "MACHINE-ITEM02",
    "title": "Machines Used to Collect Kernels of Corn",
    "label": "MC-SR question on machines used to collect kernels of corn",
    "xml:lang": "en-US",
    "timeDependent": "true",
    "adaptive": "false",
    "responseDeclaration": {
      "identifier": "RESPONSE",
      "cardinality": "single",
      "baseType": "identifier",
      "correctResponse": {
        "value": "B"
      }
    },
    "outcomeDeclaration": {
      "identifier": "SCORE",
      "cardinality": "single",
      "baseType": "float",
      "masteryValue": "0.0",
      "defaultValue": {
        "value": "0.0"
      }
    }
  }
}
```
Choose *one* of the five options.

Which machine is used to collect the kernels of corn in a field?

- Submarine.
- Combine.
- Chopper.
- Locomotive.
- Helicopter.
The key points are:

a) The JSON in Code 5.3 is produced by transforming the equivalent QTI-XML for the question 2 (see Code D3.4);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0013-0020. The associated correct answer is defined as ‘B’;

d) Four outcomes variables are defined for the Item (see lines 0021-0052). The three variables ‘SCORE’ (lines 0022-0030), ‘MAXSCORE’ (lines 0031-0038 and ‘MINSCORE’ (lines 0039-0046) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0047-0051) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question is defined in lines 0053-0093. This consists of the Item-level rubric (lines 0054-0058), the choice interaction (lines 0059-0092) with the prompt given in lines 0063-0064 and the five choices in lines 0065-0091;

f) The response-processing template to be used for this Item is given in lines 0094-0096. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;
g) The item-level feedback is given in lines 0097-00114. If the item has been answered correctly, the feedback given in lines 0098-0105 is presented otherwise the feedback given in lines 0106-0113 is presented.

The contents of the question 3 JSON file embedded in the EPUB3 are listed in Code D5.4.

**Code D5.4 – The question 3 JSON file.**

```json
{
    "assessmentItem": {
        "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
        "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
        "xsi:schemaLocation": "...",
        "identifier": "MACHINE-ITEM03",
        "title": "Machines Used to Fly",
        "label": "MC-MR question on machines used to fly.",
        "xml:lang": "en-US",
        "timeDependent": "true",
        "adaptive": "false",
        "responseDeclaration": {
            "identifier": "RESPONSE",
            "cardinality": "single",
            "baseType": "identifier",
            "correctResponse": {
                "value": ["B", "E"]
            }
        },
        "outcomeDeclaration": [
            {
                "identifier": "SCORE",
                "cardinality": "single",
                "baseType": "float",
                "masteryValue": "0.0",
                "defaultValue": {
                    "value": "0.0"
                }
            },
            {
                "identifier": "MAXSCORE",
                "cardinality": "single",
                "baseType": "float",
                "defaultValue": {
                    "value": "1.0"
                }
            },
            {
                "identifier": "MINSCORE",
                "cardinality": "single",
                "baseType": "float",
                "defaultValue": {
                    "value": "-1.0"
                }
            },
            {
                "identifier": "FEEDBACK",
                "cardinality": "single",
                "baseType": "string",
                "defaultValue": {
                    "value": "..."
                }
            }
        ]
    }
}
```
"cardinality": "single",
"baseType": "identifier"
],
"itemBody": {
  "rubricBlock": {
    "__text": "Choose as many of the options as you think are correct."",
    "view": "candidate"
  },
  "choiceInteraction": {
    "responseIdentifier": "RESPONSE",
    "shuffle": "false",
    "maxChoices": "5",
    "prompt": "Which machines can fly?"",
    "simpleChoice": [
      {
        "__text": "Submarine.",
        "identifier": "A",
        "fixed": "true"
      },
      {
        "__text": "Boeing 747.",
        "identifier": "B",
        "fixed": "true"
      },
      {
        "__text": "Chopper.",
        "identifier": "C",
        "fixed": "true"
      },
      {
        "__text": "Locomotive.",
        "identifier": "D",
        "fixed": "true"
      },
      {
        "__text": "Helicopter.",
        "identifier": "E",
        "fixed": "true"
      }
    ]
  },
  "responseProcessing": {
    "template": "mcmr-epub3rptemplate01"
  },
  "modalFeedback": {
    "__text": "Correct answer. Well done."",
    "outcomeIdentifier": "FEEDBACK",
    "showHide": "show",
    "identifier": "Correct",
    "title": "Correct Answer Feedback"
  },
  "__text": "No, wrong answer. Only the Boeing 747 and Helicopter can fly.""
}
The key points are:

a) The JSON in Code 5.4 is produced by transforming the equivalent QTI-XML for the question 2 (see Code D3.5);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0022. The associated correct answer is defined as ‘B’ and ‘E’;

d) Four outcomes variables are defined for the Item (see lines 0023-0054). The three variables ‘SCORE’ (lines 0024-0032), ‘MAXSCORE’ (lines 0033-0040) and ‘MINSCORE’ (lines 0041-0048) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0049-0053) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question is defined in lines 0055-0094. This consists of the Item-level rubric (lines 0046-0049), the choice interaction (lines 0051-0067) with the prompt given in line 0065 and the five choices in lines 0066-0092;

f) The response-processing template to be used for this Item is given in lines 0095-96. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

g) The Item-level feedback is given in lines 0098-0115. If the Item has been answered correctly, the feedback given in lines 0099-0106 is presented otherwise the feedback given in lines 0107-0114 is presented.
The contents of the question 4 JSON file embedded in the EPUB3 are listed in Code D5.5.

**Code D5.5 – The question 4 JSON file.**

```json
{  
  "assessmentItem": {  
    "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "xsi:schemaLocation": "...",
    "identifier": "MACHINE-ITEM04",
    "title": "Chopper is a type of?",
    "label": "Fill-In Blank Question on a Chopper",
    "xml:lang": "en-US",
    "timeDependent": "true",
    "adaptive": "false",
    "responseDeclaration": {  
      "identifier": "RESPONSE",
      "cardinality": "single",
      "baseType": "identifier",
      "correctResponse": {  
        "value": "motorcycle"
      }
    },
    "outcomeDeclaration": [  
      {  
        "identifier": "SCORE",
        "cardinality": "single",
        "baseType": "float",
        "masteryValue": "0.0",
        "defaultValue": {  
          "value": "0.0"
        }
      },  
      {  
        "identifier": "MAXSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {  
          "value": "1.0"
        }
      },  
      {  
        "identifier": "MINSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {  
          "value": "-1.0"
        }
      },  
      {  
        "identifier": "FEEDBACK",
        "cardinality": "single",
        "baseType": "identifier"
      }
    ],
    "itemBody": {  
      "rubricBlock": {  
        "view": "candidate",
        "text": "Chopper is a type of? Fill-In Blank Question on a Chopper.
        Chopper is a type of motorcycle."
      }
    }
  }
}
```
The key points are:

a) The JSON in Code 5.5 is produced by transforming the equivalent QTI-XML for the question 2 (see Code D3.6);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0019. The associated correct answer is defined as ‘motorcycle’;

d) Four outcomes variables are defined for the Item (see lines 0020-0051). The three variables ‘SCORE’ (lines 0021-0029), ‘MAXSCORE’ (lines 0030-0037 and ‘MINSCORE’ (lines 0038-0045) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0046-0050) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question, FIB, is defined in lines 0052-0064. This consists of the Item-level rubric (lines 0053-0057), the text entry interaction (lines 0058-0063);
f) The response-processing template to be used for this Item is given in lines 0065-0066. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

g) The Item-level feedback is given in lines 0068-0084. If the Item has been answered correctly, the feedback given in lines 0069-0075 is presented otherwise the feedback given in lines 0076-0083 is presented.

The contents of the question 5 JSON file embedded in the EPUB3 are listed in Code D5.6.

Code D5.6– The question 5 JSON file.

```json
{
  "assessmentItem": {
    "xmlns": "http://www.imsglobal.org/xsd/qti_v2p1",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "xsi:schemaLocation": "...",
    "identifier": "MACHINE-ITEM05",
    "title": "Machines Using Railway Tracks",
    "label": "MC-SR question on machines using railway tracks.",
    "xml:lang": "en-US",
    "timeDependent": "true",
    "adaptive": "false",
    "responseDeclaration": {
      "identifier": "RESPONSE",
      "cardinality": "single",
      "baseType": "identifier",
      "correctResponse": {
        "value": "C"
      }
    },
    "outcomeDeclaration": [
      {
        "identifier": "SCORE",
        "cardinality": "single",
        "baseType": "float",
        "masteryValue": "0.0",
        "defaultValue": {
          "value": "0.0"
        }
      },
      {
        "identifier": "MAXSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {
          "value": "1.0"
        }
      },
      {
        "identifier": "MINSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {
          "value": "-1.0"
        }
      }
    ]
  }
}
```
Choose <em>one</em> of the five options.

Which machine moves on a railway track?

- Indy Car.
- Combine.
- Locomotive.
- Chopper.
- Front-End Loader.

Correct answer. Well done.

Correct Answer Feedback
The key points are:

a) The JSON in Code 5.6 is produced by transforming the equivalent QTI-XML for the question 2 (see Code D3.7);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0019. The associated correct answer is defined as ‘C’;

d) Four outcomes variables are defined for the Item (see lines 0020-0051). The three variables ‘SCORE’ (lines 0021-0029), ‘MAXSCORE’ (lines 0030-0037 and ‘MINSCORE’ (lines 0038-0045) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0046-0050) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question is defined in lines 0052-0092. This consists of the Item-level rubric (lines 0053-0057), the choice interaction (lines 0058-0091) with the prompt given in lines 0062-0063 and the five choices in lines 0064-0090;

f) The response-processing template to be used for this Item is given in lines 0093-0094. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

g) The Item-level feedback is given in lines 0096-0113. If the Item has been answered correctly, the feedback given in lines 0097-0104 is presented otherwise the feedback given in lines 0105-0112 is presented.
The contents of the question 6 JSON file embedded in the EPUB3 are listed in Code D5.7.

**Code D5.7 – The question 6 JSON file.**

```json
{
  "assessmentItem": {
    "xmlns": "http://www.imsglobal.org/xsd/imsqti_v2p1",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "xsi:schemaLocation": "...",
    "identifier": "MACHINE-ITEM06",
    "title": "Chopper Gas Tank",
    "label": "Hotspot question on the Chopper.",
    "xml:lang": "en-US",
    "timeDependent": "true",
    "adaptive": "false",
    "responseDeclaration": {
      "identifier": "RESPONSE",
      "cardinality": "single",
      "baseType": "identifier",
      "correctResponse": {
        "value": "A"
      }
    },
    "outcomeDeclaration": [
      {
        "identifier": "SCORE",
        "cardinality": "single",
        "baseType": "float",
        "masteryValue": "0.0",
        "defaultValue": {
          "value": "0.0"
        }
      },
      {
        "identifier": "MAXSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {
          "value": "1.0"
        }
      },
      {
        "identifier": "MINSCORE",
        "cardinality": "single",
        "baseType": "float",
        "defaultValue": {
          "value": "-1.0"
        }
      },
      {
        "identifier": "FEEDBACK",
        "cardinality": "single",
        "baseType": "identifier"
      }
    ],
    "itemBody": {
      "rubricBlock": {
        "__text": "Choose <em>one</em> of the four
```
"options": {
  "view": "candidate",
  "hotspotInteraction": {
    "responseIdentifier": "RESPONSE",
    "maxChoices": "1",
    "prompt": "Where is the gas tank?",
    "object": {
      "__text": "Chopper",
      "type": "image/jpeg",
      "width": "720",
      "height": "495",
      "data": "Chopperv1.jpg"
    },
    "hotspotChoice": [
      {
        "identifier": "A",
        "shape": "circle",
        "coords": "300, 250, 25",
        "label": "gastank"
      },
      {
        "identifier": "B",
        "shape": "circle",
        "coords": "600, 100, 25",
        "label": "frontwheel"
      },
      {
        "identifier": "C",
        "shape": "circle",
        "coords": "400, 100, 25",
        "label": "engine"
      },
      {
        "identifier": "D",
        "shape": "circle",
        "coords": "500, 270, 25",
        "label": "light"
      }
    ],
    "responseProcessing": {
      "template": "hssr-epub3rptemplate01"
    }
  },
  "modalFeedback": [
    {
      "__text": "Correct answer. Well done."",
      "outcomeIdentifier": "FEEDBACK",
      "showHide": "show",
      "identifier": "Correct",
      "title": "Correct Answer Feedback"
    },
    {
      "__text": "No, wrong answer."",
      "outcomeIdentifier": "FEEDBACK",
      "showHide": "show",
      "identifier": "Incorrect",
      "title": "Wrong Answer Feedback"
    }
  ]
}
The key points are:

a) The JSON in Code 5.7 is produced by transforming the equivalent QTI-XML for the question 2 (see Code D3.8);

b) The title of the Item is given in line 0007;

c) The response processing variable, ‘RESPONSE’ is defined in lines 0012-0019. The associated correct answer is defined as ‘A’;

d) Four outcomes variables are defined for the Item (see lines 0020-0051). The three variables ‘SCORE’ (lines 0021-0029), ‘MAXSCORE’ (lines 0030-0037 and ‘MINSCORE’ (lines 0038-0045) are used to store the user’s score, the maximum possible score and the minimum possible score respectively. The final variable ‘FEEDBACK’ (lines 0046-0050) is used to determine the nature of the Item-level feedback to be presented. In the current version of EDUPUB these variables are assumed as the default set for any Item and so do not need to be defined. They have been included in the QTI-XML for completeness;

e) The actual question, hotspot, is defined in lines 0052-0096. This consists of the Item-level rubric (lines 0053-0057), the hotspot interaction (lines 0058-0095) with the prompt given in line 0061, the four choices in lines 0069-0094 and the background image in lines 0062-0068;

f) The response-processing template to be used for this Item is given in lines 0097-0098. This template provides a score of 1.0 when correct, a score of -1.0 when incorrect and a score of 0.0 when not answered;

g) The Item-level feedback is given in lines 0100-0116. If the Item has been answered correctly, the feedback given in lines 0101-0108 is presented otherwise the feedback given in lines 0109-0115 is presented.
D6 Other Files Changes in the EPUB3

D6.1 The Navigation File

The navigation file is amended to include the reference to the new QTI Quiz as shown in listing Code 5.8.

Code D5.8 – The EPUB3 ‘nav.html’ file.

```html
<html xmlns="http://www.w3.org/1999/xhtml"
     lang="en"
     xml:lang="en">
  <head>
    <title>INTENSE! Machines</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <section class="frontmatter TableOfContents"
             epub:type="frontmatter toc">
      <header>
        <h1>Table of Contents</h1>
      </header>
           epub:type="toc"
           id="toc">
          <ol>
            <li><a href="fcover.xhtml">Front Cover Page</a></li>
            <li><a href="page_000.xhtml">Parent Page</a></li>
            <li><a href="title.xhtml">Title Page</a></li>
            <li><a href="page_001.xhtml">Copyright Page</a></li>
            <li><a href="page_002.xhtml">Introduction</a></li>
            <li><a href="page_004.xhtml">Front-End Loader</a></li>
            <li><a href="page_006.xhtml">Giant Dump Truck</a></li>
            <li><a href="page_008.xhtml">Combine</a></li>
            <li><a href="page_010.xhtml">Indy Car</a></li>
            <li><a href="page_012.xhtml">Chopper</a></li>
            <li><a href="page_014.xhtml">Fire Truck</a></li>
            <li><a href="page_016.xhtml">Helicopter</a></li>
            <li><a href="page_018.xhtml">Locomotive</a></li>
            <li><a href="page_020.xhtml">Boeing 747</a></li>
            <li><a href="page_022.xhtml">Semitruck</a></li>
            <li><a href="page_024.xhtml">Cargo Ship</a></li>
            <li><a href="page_026.xhtml">Submarine</a></li>
            <li><a href="page_028.xhtml">Aircraft Carrier</a></li>
            <li><a href="page_030.xhtml">Space Shuttle</a></li>
            <li><a href="page_031.xhtml">INTENSE! Machines</a></li>
            <li><a href="qti_assessment.xhtml">QTI Assessment</a></li>
            <li><a href="page_033.xhtml">Interactive Submarine Activity</a></li>
            <li><a href="credit.xhtml">Credits</a></li>
            <li><a href="bcover.xhtml">Back Cover Page</a></li>
          </ol>
        <section class="comprehension-questions" id="comprehension-questions">
          Comprehension Questions
        </section>
      </nav>
    </section>
  </body>
</html>
```
The new entry into the EPUB3 table of contents is shown in lines 0038-0040 (see Figure D1.3).

**D6.2 The OPF File**

The amendments required in the OPF file are shown in listing Code 5.8.

**Code D5.8 – The EPUB3 ‘package.opf’ file.**

```xml
<package xmlns="http://www.idpf.org/2007/opf" version="3.0">
  <metadata>
    ...
  </metadata>
  <manifest>
    ...
    <!--QTI-->
    <item id="qti_01" href="qti_assessment.xhtml"
      media-type="application/xhtml+xml" properties="scripted"/>
    <item id="qtijs_01" href="uvu_dgm/js/lib/angular/angular.js"
      media-type="text/javascript"/>
    <item id="qtijs_02" href="uvu_dgm/js/lib/angular/angular-pouchdb.js"
      media-type="text/javascript"/>
    <item id="qtijs_03" href="uvu_dgm/js/lib/angular/angular-resource.js"
      media-type="text/javascript"/>
    <item id="qtijs_04" href="uvu_dgm/js/lib/angular/angular-sanitize.js"
      media-type="text/javascript"/>
    <item id="qtijs_05" href="uvu_dgm/js/lib/angular/pouchdb-1.1.0.min.js"
      media-type="text/javascript"/>
    <item id="qtijs_06" href="uvu_dgm/js/lib/intellisense.min.js"
      media-type="text/javascript"/>
    <item id="qtijs_07" href="uvu_dgm/js/lib/jquery-1.11.0.js"
      media-type="text/javascript"/>
    <item id="qtijs_08" href="uvu_dgm/js/app.js"
      media-type="text/javascript"/>
    <item id="qtijs_09" href="uvu_dgm/js/controllers.js"
      media-type="text/javascript"/>
    <item id="qtijs_10" href="uvu_dgm/js/services.js"
      media-type="text/javascript"/>
    <item id="qtijs_11" href="uvu_dgm/js/directives.js"
      media-type="text/javascript"/>
    <item id="qti_css01" href="uvu_dgm/css/assessment.css"
      media-type="text/css"/>
    <item id="qti_test01" href="uvu_dgm/quiz_items/machineepub3quiz01_test01v1.json"
      media-type="application/json"/>
    <item id="qti_item01" href="uvu_dgm/quiz_items/machinequiz01_item01v1.json"
      media-type="application/json"/>
  </manifest>
</package>
```
media-type="application/json" />
                <item id="qti_image01" href="uvu_dgm/images/Chopperv1.jpg"
                media-type="image/jpeg" />
                <item id="qti_icon01" href="uvu_dgm/images/icons/icon_add.png"
                media-type="image/png" />
                <item id="qti_icon02" href="uvu_dgm/images/icons/icon_approve.png"
                media-type="image/png" />
                <item id="qti_icon03" href="uvu_dgm/images/icons/icon_checkmark.png"
                media-type="image/png" />
                <item id="qti_icon04" href="uvu_dgm/images/icons/icon_checkmark_blue.png"
                media-type="image/png" />
                <item id="qti_icon05" href="uvu_dgm/images/icons/icon_close.png"
                media-type="image/png" />
                <item id="qti_icon06" href="uvu_dgm/images/icons/icon_closeWht.png"
                media-type="image/png" />
                <item id="qti_icon07" href="uvu_dgm/images/icons/icon_discard.png"
                media-type="image/png" />
                <item id="qti_icon08" href="uvu_dgm/images/icons/icon_download.png"
                media-type="image/png" />
                <item id="qti_icon09" href="uvu_dgm/images/icons/icon_selector_correct.png"
                media-type="image/png" />
                <item id="qti_icon10" href="uvu_dgm/images/icons/icon_selector_incorrect.png"
                media-type="image/png" />
                <item id="qti_icon11" href="uvu_dgm/images/icons/icon_selector_neutral.png"
                media-type="image/png" />
                <item id="qti_icon12" href="uvu_dgm/images/icons/icon_upload.png"
                media-type="image/png" />
                <item id="qti_icon13" href="uvu_dgm/images/icons/icon_next.png"
                media-type="image/png" />
                <item id="qti_icon14" href="uvu_dgm/images/icons/icon_previous.png"
                media-type="image/png" />
The changes to the <manifest> are shown in lines 0009-0106 i.e. the listing of all the new files that have been added due to the inclusion of the quiz. The changes to the <spine> are shown in lines 0113-0114 i.e. the pointer to the actual quiz XHTML <item> in the <manifest>.

D7 The Support Java Script Files

[[ ED NOTE: To be completed in a later draft. ]]

D8 The New EPUB3/EDUPUB Content Structure

When the new EDUPUB/EPUB3 file is unzipped the new contents structure is (italics denote a folder, shaded lines denote the new folders/files and bold denotes original files that have been amended):

    mimetype

    META-INF
        container.xml

    OEBPS
        audio
            bcover.xhtml
            config.js
            credit.xhtml
        css
            fcover.xhtml
        fonts
        images
        js
            license.xhtml
    nav.xhtml
    package.opf
        page_000.xhtml page_001.xhtml page_002.xhtml page_003.xhtml
        page_004.xhtml page_005.xhtml page_006.xhtml page_007.xhtml
        page_008.xhtml page_009.xhtml page_010.xhtml page_011.xhtml
Appendix E – Metadata in EDUPUB

E1 Metadata Information Model

The metadata fields available for use in EDUPUB are listed in Table E1.1. The key to Table E1.1 is:

- **Name** – the name of the field. The term enclosed in the accompanying square brackets denote the equivalent Dublin Core term;
- **Type** – the data type for the field;
- **Mult** – indicates the multiplicity of the field with values of ‘1’ or many (‘*’);
- **Description** – a short description of how the field is to be used;
- ‘*’ – denotes use in the context of a QTI-based assessment.

No attempt has been made to define whether or not an entry should/must be supplied. Instead, the aim is to provide mappings between the various binding form and to leave each sector to determine what metadata should be supplied.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Mult</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUID [ dc:identifier ]</td>
<td>String.</td>
<td>1</td>
<td>This field is the globally unique identifier of the object from the originator. This will be used to determine if the object already exists in the system in order to avoid duplicates. It will allow the owner of the object to update the metadata or URL when necessary. Systems can also allow direct access to the object within their system by allowing the field to be searched.</td>
</tr>
<tr>
<td>Version</td>
<td>String</td>
<td>1</td>
<td>The version of the object.</td>
</tr>
<tr>
<td>LTI URL</td>
<td>String (URL).</td>
<td>1</td>
<td>This field is the IMS Global Learning Tools Interoperability launch URL to access the tool within the tool provider. It contains either the direct URL to the tool or the authorization service of LTI while the actual tool or resource is at the Resource URL location.</td>
</tr>
<tr>
<td>Resource URL</td>
<td>String (URL).</td>
<td>1</td>
<td>This field is used as a parameter with LTI. It contains the redirect URL that the LTI tool provider requires to access the tool. The LTI URL authorizes and sets up the tool session while the resource URL accesses the specific tool or resource within the tool provider.</td>
</tr>
<tr>
<td>Language [ dc:language ]</td>
<td>Enumeration as per RFC3066/ISO639.</td>
<td>1</td>
<td>This is the primary language of the content of the resource. This takes the form of two- and three-letter primary language tags with optional subtags.</td>
</tr>
</tbody>
</table>

Table E1.1 – The set of logical metadata fields supported in EDUPUB.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Mult</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title [ dc:title ]</td>
<td>String</td>
<td>1</td>
<td>This field contains the metadata title of the object.</td>
</tr>
<tr>
<td>Description [ dc:description ]</td>
<td>String</td>
<td>1</td>
<td>This field contains the metadata description of the object.</td>
</tr>
<tr>
<td>Creator/Author [ dc:creator ]</td>
<td>String</td>
<td>1</td>
<td>An entity primarily responsible for making the content of the resource. Examples of a Creator include a person, an organization, or a service. Typically the name of the Creator should be used to indicate the entity.</td>
</tr>
<tr>
<td>Publisher [ dc: publisher ]</td>
<td>String</td>
<td>1</td>
<td>The entity responsible for making the resource available. Examples of a Publisher include a person, an organization, or a service. Typically, the name of a Publisher should be used to indicate the entity.</td>
</tr>
<tr>
<td>Date Created [ dc:date ]</td>
<td>Date (YYYY-MM-DD)</td>
<td>1</td>
<td>The date of the creation of the resource itself as a complete entity.</td>
</tr>
<tr>
<td>Date Modified [ dc:date ]</td>
<td>Date (YYYY-MM-DD)</td>
<td>1</td>
<td>The date at which the resource was modified.</td>
</tr>
<tr>
<td>Educational Role</td>
<td>Enumeration: { teacher</td>
<td>author</td>
<td>learner</td>
</tr>
<tr>
<td>Typical Learning Time</td>
<td>Duration (PTHH:MM:SS)</td>
<td>1</td>
<td>Approximate or typical time it takes to work with or through this learning resource for the typical intended target audience.</td>
</tr>
<tr>
<td>Interactivity Type</td>
<td>Enumeration: { active</td>
<td>expositive</td>
<td>mixed }</td>
</tr>
<tr>
<td>Interactivity Level</td>
<td>Enumeration: { very low</td>
<td>low</td>
<td>medium</td>
</tr>
<tr>
<td>Source [ dc:source ]</td>
<td>String</td>
<td>1</td>
<td>A Reference to a resource from which the present resource is derived. The present resource may be derived from the Source resource in whole or part. Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.</td>
</tr>
<tr>
<td>Keywords [ dc:subject ]</td>
<td>String (space delimited).</td>
<td>1</td>
<td>This field contains the single words separated by spaces related to the object. This helps organize the object within the system for easier searching and identification.</td>
</tr>
<tr>
<td>Typical Age Range</td>
<td>String</td>
<td>1</td>
<td>A free format field to provide the typical age range for which the content is suitable i.e. ‘From Grade’ – ‘To Grade’. Typical values are “k-12”, “7-9”, “0-5”, etc. Each range component is enumerated by: [ p</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Mult</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------</td>
<td>------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Copyright</td>
<td>Year (4 digit string).</td>
<td>1</td>
<td>This field contains the year in which the object was created. It represents when the copyright of the object was first introduced.</td>
</tr>
<tr>
<td>Use Rights</td>
<td>URL</td>
<td>1</td>
<td>The URL where the owner specifies permissions for using the resource.</td>
</tr>
<tr>
<td>Context</td>
<td>Enumeration: { school</td>
<td>higher education</td>
<td>training</td>
</tr>
<tr>
<td>Difficulty</td>
<td>Enumeration: { very easy</td>
<td>easy</td>
<td>medium</td>
</tr>
<tr>
<td>ISBN</td>
<td>String.</td>
<td>1</td>
<td>This field contains the ISBN (International Standard Book Number) of the object. This allows quick access to the object during a search.</td>
</tr>
<tr>
<td>Learning Resource Type</td>
<td>Enumeration: { video</td>
<td>audio, image</td>
<td>interactive</td>
</tr>
<tr>
<td>AccessPermission</td>
<td>Enumeration: { parent</td>
<td>student</td>
<td>teacher</td>
</tr>
</tbody>
</table>
| UseType                  | Enumeration: { primary | secondary | teacher | assessment | question | concept | assignment }.                                               * This field contains the use type of the object. It allows the object to be organized based on the purpose of the object. For example, if the object was a LTI link to resource for homework purposes, it would be set to “assignment”. That means the object would be available to a student during an assignment. The values can be:  
  - Primary: within a lesson it would be part of the primary lesson.  
  - Secondary: within a lesson it would be a secondary object and not necessarily used and isn’t the main focus of the lesson.  
  - Teacher: within a lesson it would be viewable only to a teacher.  
  - Assessment: within a lesson it would be an
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Mult</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visibility</td>
<td>Enumeration: { private</td>
<td>shared</td>
<td>public }.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Private: Only the person who imported the object has the right to view the object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Shared: Users within the school can view it.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Public: Everyone in the district can view it.</td>
</tr>
<tr>
<td>Thumbnail</td>
<td>String.</td>
<td>1</td>
<td>This field contains the URL to the location of a thumbnail of the object. This would be used in previewing the object while searching or viewing the object. Since the object is a URL and the tool provider may change the object at any time, the thumbnail would be located on the tool provider’s servers. Updated when the object is updated and accessible without an LTI launch.</td>
</tr>
<tr>
<td>Mapping to Curriculum Standards</td>
<td>List of GUIDs (strings).</td>
<td>*</td>
<td>This field (set of fields) contains each state standard or common core standard for the object. In this case the term GUID is uses in the sense of any globally unique identifier form e.g. number, URI, etc.</td>
</tr>
<tr>
<td>Item Template*</td>
<td>Boolean.</td>
<td>1</td>
<td>A Boolean to indicate it the Item is actually an Item template.</td>
</tr>
<tr>
<td>Time Dependent*</td>
<td>Boolean.</td>
<td>1</td>
<td>Whether or not the Item is time dependent.</td>
</tr>
<tr>
<td>Composite*</td>
<td>Boolean.</td>
<td>1</td>
<td>A composite contains more than one interaction. Set as ‘true’ if the Item is a composite.</td>
</tr>
<tr>
<td>Interaction Type*</td>
<td>Enumeration: { associateInteraction</td>
<td>choiceInteraction</td>
<td>customInteraction</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Mult</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Feedback Type*</td>
<td>Enumeration: { adaptive</td>
<td>nonadaptive</td>
<td>none }</td>
</tr>
<tr>
<td>Solution Available*</td>
<td>Boolean.</td>
<td>1</td>
<td>A statement of whether or not the Item includes the solution.</td>
</tr>
<tr>
<td>Scoring Mode*</td>
<td>Enumeration: { human</td>
<td>externalmachine</td>
<td>responseprocessing }</td>
</tr>
<tr>
<td>Tool Name*</td>
<td>String</td>
<td>1</td>
<td>The name of the tool used to create the quiz, test, question, etc.</td>
</tr>
<tr>
<td>Tool Version*</td>
<td>String</td>
<td>1</td>
<td>The version of the tool used to create the quiz, test, question, etc.</td>
</tr>
<tr>
<td>Tool Vendor*</td>
<td>String.</td>
<td>1</td>
<td>The vendor of the tool used to create the quiz, test, question, etc.</td>
</tr>
<tr>
<td>Extension</td>
<td>N/A</td>
<td>*</td>
<td>Support for an extension mechanism.</td>
</tr>
</tbody>
</table>

### E2 Binding Mappings in IEEE LOM and Schema.Org

The metadata binding mappings are defined in Table E2.1. The columns in Table E2.1 denote:

- **Name** – the name of the field. The term enclosed in the accompanying square brackets denote the equivalent Dublin Core term;
- **XML Mapping** – XML mapping when used with IMS specifications directly e.g. in Common Cartridge, APIP, etc. The actual binding formats are:-
  - ‘lom.’ – mapping to the IEEE LOM using the IMS Metadata v1.3.2 XSD binding
  - ‘csm’ – mapping to the IMS Curriculum Standards Metadata v1.0 XSD binding but contained within the LOM context
  - ‘qti’ – mapping to the IMS QTI Metadata v2.2 XSD binding but contained within the LOM context;
- **Micro Data Mapping** – mapping to the Schema.Org micro data format (including the LRMI support). In this column the value <object> denotes a creative work of: { Article | Blog |
If an explicit creative work is identified then the property is restricted to that creative work.

Table E2.1 – The metadata binding mappings.

<table>
<thead>
<tr>
<th>Name</th>
<th>IMS XML Mappings</th>
<th>Schema.Org Micro Data Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUID</td>
<td>lom.general.identifier .catalog=GUID .entry</td>
<td>– No mapping available –</td>
</tr>
</tbody>
</table>
| LTI URL      | lom.technical.installationRemarks                      | itemtype="http://schema.org/<object>" sameAs="???
| Resource URL | lom.technical.location                                 | itemtype="http://schema.org/<object>" url="???
| Language     | lom.general.language                                  | itemtype="http://schema.org/<object>" inLanguage="???
| Title        | lom.general.title                                     | itemtype="http://schema.org/<object>" name="???
| Description  | lom.general.description                               | itemtype="http://schema.org/<object>" description="???
| Creator      | lom.lifeCycle .contribute .role .value=creator .entity | itemtype="http://schema.org/<object>" creator="???
|             |                                                       | or                            |
|             |                                                       | itemtype="http://schema.org/<object>" author="???
| Publisher    | lom.lifeCycle .contribute .role .value=publisher .entity | itemtype="http://schema.org/<object>" publisher="???
| Date Created | lom.lifeCycle .contribute .role .value=creator .date .dateTime | itemtype="http://schema.org/<object>" dateCreated="???
<table>
<thead>
<tr>
<th>Name</th>
<th>IMS XML Mappings</th>
<th>Schema.Org Micro Data Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Modified</td>
<td>lom.lifeCycle.contribute.role.value=contributor.date</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dateModified=&quot;???&quot;</td>
</tr>
<tr>
<td>Educational Role</td>
<td>lom.educational.intendedEndUserRole.value</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>educationalRole=&quot;???&quot;</td>
</tr>
<tr>
<td>Typical Learning Time</td>
<td>lom.educational.itypicalLearningTime.duration</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timeRequired=&quot;???&quot;</td>
</tr>
<tr>
<td>Interactivity Type</td>
<td>lom.educational.interactivityType.value</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interactivityType=&quot;???&quot;</td>
</tr>
<tr>
<td>Interactivity Level</td>
<td>lom.educational.interactivityLevel.value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>– N/A –</td>
</tr>
<tr>
<td>Source</td>
<td>lom.relation.kind.value=isbasedon.resource.identifier.catalog=GUID.entry</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>isBasedOnUrl=&quot;???&quot;</td>
</tr>
<tr>
<td>Keywords</td>
<td>lom.general.keyword</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>keywords=&quot;???&quot;</td>
</tr>
<tr>
<td>Typical Age Range</td>
<td>lom.educational.typicalAgeRange</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>typicalAgeRange=&quot;???&quot;</td>
</tr>
<tr>
<td>Copyright</td>
<td>lom.lifeCycle.contribute.role.value = copyright.date</td>
<td>itemtype=&quot;<a href="http://schema.org/">http://schema.org/</a>&lt;object&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>copyrightYear=&quot;???&quot;</td>
</tr>
<tr>
<td>Use Rights</td>
<td>lom.rights.copyrightAndOtherRestrictions.value=yes.description.string</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>– N/A –</td>
</tr>
<tr>
<td>Context</td>
<td>lom.educational.context.value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>– N/A –</td>
</tr>
<tr>
<td>Difficulty</td>
<td>lom.educational.difficulty.value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>– N/A –</td>
</tr>
<tr>
<td>Name</td>
<td>IMS XML Mappings</td>
<td>Schema.Org Micro Data Mapping</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Learning Resource Type</td>
<td>lom.educational.learningresourcetype .type</td>
<td>itemtype=&quot;<a href="http://schema.org/object">http://schema.org/object</a>&quot; learningResourceType=&quot;???&quot;</td>
</tr>
<tr>
<td>AccessPermission</td>
<td>lom.educational.intendedEndUserRole .source=AccessPermissionVocab .value</td>
<td>itemtype=&quot;<a href="http://schema.org/EducationalAudience">http://schema.org/EducationalAudience</a>&quot; educationRole=&quot;???&quot;</td>
</tr>
<tr>
<td>UseType</td>
<td>lom.educational.intendedEndUserRole .source=UseTypeVocab .value</td>
<td>itemtype=&quot;<a href="http://schema.org/object">http://schema.org/object</a>&quot; educationalUse=&quot;???&quot;</td>
</tr>
<tr>
<td>Visibility</td>
<td>lom.educational.intendedEndUserRole .source=VisibilityVocab .value</td>
<td>- No mapping available</td>
</tr>
<tr>
<td>Thumbnail</td>
<td>lom.annotation.description</td>
<td>itemtype=&quot;<a href="http://schema.org/object">http://schema.org/object</a>&quot; thumbnailUrl=&quot;???&quot;</td>
</tr>
<tr>
<td>Mapping to Curriculum Standards</td>
<td>curriculumStandardsMetadataSet .@resourceLabel .@resourcePartId .curriculumStandardsMetadata .@providerId .setOfGUIDs .@region .@version .labelledGUID .label .GUID</td>
<td>item1 itemtype=&quot;<a href="http://schema.org/AlignmentObject">http://schema.org/AlignmentObject</a>&quot; alignmentType=&quot;???&quot; educationFramework=&quot;???&quot; targetUrl=&quot;???&quot; targetDescription=&quot;???&quot; targetName=&quot;???&quot;</td>
</tr>
<tr>
<td>Item Template</td>
<td>qtiMetadata.itemTemplate</td>
<td>- N/A</td>
</tr>
<tr>
<td>Time Dependent</td>
<td>qtiMetadata.timeDependent</td>
<td>- N/A</td>
</tr>
<tr>
<td>Composite</td>
<td>qtiMetadata.composite</td>
<td>- N/A</td>
</tr>
<tr>
<td>Interaction Type</td>
<td>qtiMetadata.interactionType</td>
<td>- N/A</td>
</tr>
<tr>
<td>Feedback Type</td>
<td>qtiMetadata.feedbackType</td>
<td>- N/A</td>
</tr>
<tr>
<td>Solution</td>
<td>qtiMetadata.solutionAvailable</td>
<td>- N/A</td>
</tr>
</tbody>
</table>
### E3 Examples of the Bindings

When using IMS-based content is typically supplied in the form of a content package and so the metadata is encoded in the “imsmanifest.xml” file using the IMS Metadata V1.3 binding of the IEEE LOM information model. An example of the full mapping of the IMS objects to their XML form is shown in Code E3.1.

**Code E3.1 – XML-based metadata binding for the IMS objects.**

```xml
0000  ...  
0001  <metadata>  
0003  <lom xmlns="http://ltsc.ieee.org/xsd/LOM"  
0004      xmlns:csm="http://www.imsglobal.org/xsd/imsccv1p3/imscsmd_v1p0"  
0005      xmlns:qti="http://www.imsglobal.org/xsd/imsqti_metadata_v2p2"  
0006      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
0007      xsi:schemaLocation="http://ltsc.ieee.org/xsd/LOM  
0008          http://www.imsglobal.org/xsd/imsmd_loose_v1p3p2.xsd  
0009          http://www.imsglobal.org/xsd/imsccv1p3/imscsmd_v1p0  
0010          http://www.imsglobal.org/profile/cc/ccv1p3/ccv1p3_imscsmd_v1p0.xsd  
0011          http://www.imsglobal.org/xsd/imsqti_metadata_v2p2  
0012          http://www.imsglobal.org/xsd/qtiv2p2/imsqti_metadata_v2p2.xsd">  
0013  <!-- GUID, TITLE, DESCRIPTION, KEYWORDS, ISBN, LANGUAGE -->  
0014  <identifier>  
0017      <catalog>GUID</catalog>  
0018      <entry>ABCDEFG...123456789...XYZ</entry>  
0020  </identifier>  
0021  <title>  
0022      <string>...Title...</string>  
0023  </title>  
0024  <description>  
0025      <string>...The description...</string>  
0026  </description>  
0027  <keyword>  
0028      <string>Value1 Value2 Value 2</string>  
0029  </keyword>  
0030  <identifier>  
0031      <catalog>ISBN</catalog>  
```
<entry>...ISBN Value...</entry>
</identifier>
<language>en-US</language>
</general>

<!-- ****************************
LTI URL, RESOURCE URL *******
--------------------------
**
-->

<!-- TYPICAL AGE RANGE, LEARNING RESOURCE TYPE, ACCESS PERMISSION, USE TYPE,
VISIBILITY, ED ROLE, INTERACTIVITY TYPE, INTERACTIVITY LEVEL, CONTEXT,
DIFFICULTY, TYPICAL LEARNING TIME **************************** -->

<technical>
<installationRemarks>
<string>http://...lti authentication URL ...</string>
</installationRemarks>
<location>http://...lti launch URL...</location>
</technical>

<!-- ****************************
TYPICAL AGE RANGE, LEARNING RESOURCE TYPE, ACCESS PERMISSION, USE TYPE,
VISIBILITY, ED ROLE, INTERACTIVITY TYPE, INTERACTIVITY LEVEL, CONTEXT,
DIFFICULTY, TYPICAL LEARNING TIME **************************** -->

<educational>
<typicalAgeRange>
<string>k-a</string>
</typicalAgeRange>
<learningResourceType>
<source>...Vocab Source...</source>
<value>...Type of Resource...</value>
</learningResourceType>
<intendedEndUserRole>
<source>Access Permission Vocab</source>
<value>...Access Permission...</value>
</intendedEndUserRole>
<intendedEndUserRole>
<source>UseType Vocab</source>
<value>...UseType...</value>
</intendedEndUserRole>
<intendedEndUserRole>
<source>Visibility Vocab</source>
<value>...Visibility...</value>
</intendedEndUserRole>
<intendedEndUserRole>
<source>Educational Role Vocab</source>
<value>learner</value>
</intendedEndUserRole>
<interactivityType>
<value>active</value>
</interactivityType>
<interactivityLevel>
<value>medium</value>
</interactivityLevel>
<context>
<source>LOMv1.0</source>
<value>school</value>
</context>
<difficulty>
<source>LOMv1.0</source>
<value>medium</value>
</difficulty>
<typicalLearningTime>
<duration>pT30M</duration>
</typicalLearningTime>
</educational>
<!-- ---------------------------------------------------------------------- -->
<!-- VERSION, CREATOR,DATE CREATED, PUBLISHER, DATE MODIFIED, COPYRIGHT -->
<!-- ---------------------------------------------------------------------- -->
<lifecycle>
  <version>
    <string>Final 1.0</string>
  </version>
  <contribute>
    <role>
      <value>creator</value>
    </role>
    <entity>...Name of Author...</entity>
    <date>
      <dateTime>20140527</dateTime>
    </date>
  </contribute>
  <contribute>
    <role>
      <value>publisher</value>
    </role>
    <entity>...Name of Publisher...</entity>
    <date>
      <dateTime>20140528</dateTime>
    </date>
  </contribute>
  <contribute>
    <role>
      <value>contributor</value>
    </role>
    <date>
      <dateTime>20140528</dateTime>
    </date>
  </contribute>
  <contribute>
    <role>
      <value>copyright</value>
    </role>
    <date>
      <dateTime>2014</dateTime>
    </date>
  </contribute>
</lifecycle>

<!-- SOURCE ---------------------------------------------------------------------- -->
<!-- SOURCE ---------------------------------------------------------------------- -->
<relation>
  <kind>
    <source>LOMv1.0</source>
    <value>isbasedon</value>
  </kind>
</relation>

<!-- USE RIGHTS ---------------------------------------------------------------------- -->
<!-- USE RIGHTS ---------------------------------------------------------------------- -->
<rights>
  <copyrightAndOtherRestrictions>
    <source>LOMv1.0</source>
  </copyrightAndOtherRestrictions>
</rights>
<value>yes</value>
</copyrightAndOtherRestrictions>
<description>
<string>...Use Rights URL...</string>
</description>

<right>

<!-- ********************* -->
<annotation>
<description>
<string>...URL for the Thumbnail...</string>
</description>
</annotation>

<!-- ********************* -->
<cs:curriculumStandardsMetadataSet>
<cs:labelledGUID>
<cs:label>...Human readable string...</cs:label>
<cs:GUID>GUID for Standard/Standard-part</cs:GUID>
</cs:labelledGUID>
<cs:labelledGUID>
<cs:label>...Human readable string...</cs:label>
<cs:GUID>GUID for Standard/Standard-part</cs:GUID>
</cs:labelledGUID>
</cs:curriculumStandardsMetadataSet>

</cs:curriculumStandardsMetadata>
</cs:curriculumStandardsMetadataSet>

</qti:qtiMetadata>

</lom>
</metadata>
…
An example of the full mapping of the IMS objects to their microdata form is shown in Code E3.2.

**Code E3.2 – Microdata-based metadata binding for the IMS objects.**

![Code E3.2](image)

NOTE: The code mappings of the IMS objects to the XML-based and Microdata-based bindings shown in Code E3.1 and E3.2 are for discussion. These are the current mapping recommendations but feedback to IMS on these is welcome.

[[ ED NOTE: To be completed in a later draft. ]]
Appendix F – Readium Support for EDUPUB

F1 Source Code Additions to Support LTIPv1.x Tool Provider Functionality

The server/main.js file defines an endpoint of /ltilaunch which represents the launch URL for use by tool consumers. Any key and secret issued to a tool consumer should be recorded in the LTI_CONSUMERS array variable (see Subsection 3.1.1 for an example). The processing of a launch request depends upon the oauth-signature module that should be installed on the Readium server. The process of handling an LTI launch request is described in Subsection 3.1.2. A successful launch causes the views/lti.jade to be rendered. If an invalid launch request is received then the user is redirected to the URL provided in the launch_presentation_return_url parameter with an lti_errormsg query parameter or, if this parameter is not available, then the views/lti-error.jade file is rendered. The lib/thirdparty/lti_lib.js file provides supporting functions for accessing tool consumer declarations and the request URL.

F2 Source Code Additions to Support LTIPv1.x Tool Consumer Functionality

The server/main.js file defines an endpoint of /ltitool which represents the URL to call when launching a tool provider. The tools available for launch are defined in the LTI_PROVIDERS array variable (see Subsection 3.2.1 for an example). The launch parameters included in a tool launch request are identified in Subsection 3.2.2. The location where the tool is opened is determined by the value of the presentation_document_target attribute in its declaration; a value of “popup” will cause the tool to be opened in a sized new window, whilst a value of “window” will open the tool in a new full-sized window (or tab). The size of the popup window will be 400 by 400 unless otherwise specified via the displayWidth and displayHeight attributes in the tool declaration. The oauth-signature module is used to generate the signature for the launch request data. The launch process requires 3 query parameters passed to the endpoint; if any of these is missing, or if the session parameter value does not represent a current session, or if the id parameter does not represent a current tool declaration, then the views/lti-error.jade file is rendered. Otherwise an attempt is made to launch the selected tool provider using the views/lti-launch.jade file. The lib/thirdparty/lti_lib.js file provides supporting functions for accessing tool provider declarations and for generating GUID values.

F3 Source Code Additions to Support IMS Sensor API Functionality

The lib/thirdPary/ReadiumSensor.js file provides an implementation of the Readium Caliper Sensor. The Readium Caliper Sensor is defined as a self-contained Asynchronous Module Definition (AMD) module that in turn leverages the IMS Base JavaScript Caliper Sensor.

Note that the code below is elided to remove some utility functions, leaving in place the key API calls that the sensor typically makes.
Code F3.1 – The IMS Base JavaScript Caliper Sensor API.

```javascript
define(['jquery', 'caliper'], function($, caliper) {
  // Initialize with the Caliper API key (one time)

  // initialize context
  var context = {
    'metadata': ''
  };

  var ReadingInfo = {
    'ID': 'readingId',
    'ISBN': 'isbn',
    'TYPE': 'reading',
    'LABEL': 'label',
    'TITLE': 'title',
    'DESCRIPTION': 'description'
  };

  var ReadingActions = {
    'LOAD_BOOK': 'LOAD_BOOK',
    'VIEW_TOC': 'VIEW_TOC',
    'SWITCHCHAPTER': 'SWITCHCHAPTER',
    'READ_PAGE': 'READ',
    'BOOKMARK_PAGE': 'BOOKMARK',
    'VIEW_LIBRARY': 'VIEW_LIBRARY',
    'SWITCHVIEWMODE': 'SWITCHVIEWMODE'
  };

  var describe = function(type, entityId, properties, timestamp) {
    caliper.describe(type, entityId, properties, timestamp);
  };

  var describeCurrentBook = function() {
    var type = ReadingInfo.TYPE;
    var entityId = window.readiumSensor.context.metadata.id;
    var properties = window.readiumSensor.context.metadata.
    attributes;
    describe(type, entityId, properties, Date.now());
  };

  var measure = function(action, learningContext, activityContext, timestamp) {
    caliper.measure(action, learningContext, activityContext, timestamp);
  };

  var measureActivity = function(action, activityContextDetail) {
    var learningContext = {
      'personId': window.readiumSensor.context.userId,
      'courseId': window.readiumSensor.context.courseId
    };
    var activityContext = merge({
      'activityId': window.readiumSensor.context.metadata.
    id,
```
F4  The Readium Source Code Distribution

The set of source code files distributed for Readium are shown below. Note that italics denote a folder, shaded lines denote the new folders/files and bold denotes original files that have been amended.

chrome-app
css
epub_content
   IntenseMachines
   extend_require_config.js
   Gruntfile.js
   i18n
   images
index.html

lib
  Dialog.js
  EpubLibrary.js
  **EpubReader.js**
  ReaderSettings.js
  Readium.js
  ReadiumViewer.js
  ReadiumViewer.Lite
  Require.js
  storage
  thirdparty
    ...
    intellisense.min.js
    ...
    jquery-1.11.0.min.js
    ...
    lti_lib.js
    lti.js
    ReadiumsSensor.js
    ...
  workers
  package.json
  readium.sublime-project
  readium.sub,ime-workspace
  README.md
  require_config.js

server
  main.js

simpleviewer.html

templates

views
  lti-error.jade
  lti-launch.jade
  lti.jade
Appendix G – Best Practices Checklist

G1  EDUPUB Content

G1.1  General Recommendations

The general best practice recommendations for EDUPUB content are:

• A design/implementation decision has to be made regarding the extent to which the eBook relies upon features available in the eReader against the provision of the same features within the eBook itself. If the eReader permits external Web service access then the eBook could be self sufficient with regard to LTI and Sensor support;

[[ ED NOTE: To be completed in a later draft. ]]

G1.2  LTI-Specific Recommendations

The LTI-specific best practice recommendations for EDUPUB content are:

• If the eBook is to make use of other resources using LTI, the user should receive an informative error message if the LTI launch fails. In general, the user will be unaware of the nature of LTI and its usage but even so this message should help the user locate and correct the problem.

G1.3  Caliper-Specific Recommendations

The Caliper-specific best practice recommendations for EDUPUB content are:

• Calls to the Sensor API should be used to report the appropriate measurements for the metric profiles as the eBook is used. Calls should be made to the Sensor API for every user action on the eBook;

• The user should be able to disable the sensor API so that measurements reporting is completely disabled for a particular eBook;

• The user should be able to define the target analytics repository (for both real-time and offline synchronisation modes).

G1.4  QTI-Specific Recommendations

The QTI-specific best practice recommendations for EDUPUB content are:

• In the QTI-XML for an Item the value of the assessmentItem/@identifier should be the same as the identifier value in the metadata for the QTI-Item’s resource in the manifest file. This should be globally unique;
• In the QTI-XML for the Test the value of the assessmentTest/@identifier should be the same as the identifier value in the metadata for the QTI-Item’s resource in the manifest file. This should be globally unique.

[[ ED NOTE: To be completed in a later draft. ]]  

G1.5 Metadata Recommendations  

[[ ED NOTE: To be completed in a later draft. ]]  

G2 EDUPUB Readers  

G2.1 General Recommendations  

The general best practice recommendations for an EDUPUB reader are:

• The reader must be LTIv1.x compliant both as a Tool Provider and as a Tool Consumer. Tool Provider compliance means the player can be launched by any LTI compliant Tool Consumer e.g. an LMS. Tool Consumer compliance means that the reader can launch other LTI compliant Tool Providers e.g. third party quizzing systems;

• The reader must be Caliper compliant both in terms of real-time data reporting and also using local caching and data synchronisation when the reader is not connected to a network;

• The reader must be EDUPUB/QTI compliant for the provision of formative assessments within the EPUB3 file. This provides the support for formative assessments that are embedded within the EPUB3 file. This compliance includes support for Caliper within the context of the formative assessment;

• If an eBook requires certain features of the eReader, e.g. LTI launch capability, analytics reporting, etc. and these are unavailable, then an appropriate warning/error message should be displayed to the user. If the eBook is still used, its behaviour is undefined when using the unavailable features.

G2.2 LTI-Specific Recommendations  

The LTI-specific best practice recommendations for an EDUPUB reader are:

• All of the supported Tools must be registered Tool Providers within the reader i.e. it must have the required authentication information. Therefore, the eReader must provide a simple interface (this could be a part of the normal eBook management software) through which such tool registrations can be established;

G2.3 Caliper-Specific Recommendations  

The Caliper-specific best practice recommendations for an EDUPUB reader are:
The user should be able to disable the sensor API so that measurements reporting is completely disabled;

G2.4 QTI-Specific Recommendations

[[ ED NOTE: To be completed in a later draft. ]]

G2.5 Metadata Recommendations

[[ ED NOTE: To be completed in a later draft. ]]

G3 EDUPUB Authoring Systems, Tools & Applications

G3.1 General Recommendations

The general best practice recommendations for an EDUPUB authoring system, etc. are:

• The tool must allow the eBook to return the appropriate information when invoked using an LTI launch;

• The tool must allow LTI-enabled launch links to be inserted into the EPUB3 file. This includes the insertion of the required code libraries;

• The tool must enable the use of the Sensor API to report the appropriate measures. This includes the insertion of the required code libraries.

G3.2 LTI-Specific Recommendations

[[ ED NOTE: To be completed in a later draft. ]]

G3.3 Caliper-Specific Recommendations

[[ ED NOTE: To be completed in a later draft. ]]

G3.4 QTI-Specific Recommendations

[[ ED NOTE: To be completed in a later draft. ]]

G3.5 Metadata Recommendations

[[ ED NOTE: To be completed in a later draft. ]]
Appendix H – LTI/Caliper/QTI EDUPUB Conformance

At the time of publication the EDUPUB Conformance Program has not been finalised. This Section will be completed in later versions of this document.