An Investigation of the Enhancement and the Formal Description of IMS/QTI Specification for use in a Programming Course

Ahlem HARCHAY, Lilia CHENITI-BELCADHI, Rafik BRAHAM
PRINCE Research Group, ISITCom, University of Sousse, Tunisia

Hammam Sousse from 05 to 07 July 2010
Outline

- Introduction
- Overview of the IMS/QTI specification
- Enrichment of the IMS/QTI specification
- Conclusion and future work
One of the major e-Learning problems is **online assessment**. Assessment is a fundamental step in any learning process leading to a qualitative judgment of the progress of the learner.

**SCORM, IMS/LD**: allow the organization of the educational contents. **LOM**: concerns the expression of metadata.

**IMS/QTI** is a **de-facto** standard for expression of assessment content and the construction of items and tests.

Beneficial to study the QTI specification and to detail its principal functionalities.
Introduction (2/2)

For the description of the QTI specification, the IMS consortium uses a set of guides, in a form of a *text description*, which explains its main functionalities.

Development of a standardized formalism is required to clarify the structure of a QTI item.

The QTI tries to cover different types of questions and implements different learning situations. But some types of questions for many disciplines still not completely covered.

Extension of QTI by adding other types of questions that would be applicable to our domain: *A Programming Course*. 
Research objectives

- Study the IMS / QTI specification.
- Develop a normalized formalism for the representation of a QTI item.
- Study the possibility to enrich the specification by adding other types of questions applied to a Programming Course.
- Examine the possibility to integrate the added item to a QTI editor.

[CHENITI, BRAHAM, HENZE] PhD :
“Assessment Personalization in the semantic web”
Overview of the IMS/QTI

Presentation
QTI Item Structure
Ontological Model
IMS/ QTI (Instructional Management Systems/Question and Test Interoperability) is used to represent the data structure of a question and a test and the correspondent results.

QTI is based on a data presentation using the XML language.

The basic elements : ASI acronym

- **Item** : the smallest exchangeable unit used by QTI. It relates to the question, its presentation, the response processing…
- **Section** : a set of section or items.
- **Assessment** : a set of one or more sections.
QTI Item Structure

Model for the representation of questions and possible answers that can explain *the structure* of an item.

**QTI Item version 2.1:**

- Attributs and variables
- Item Body
- Response processing
- Modal Feedback
Ontological model for a QTI item (1/3)

Standardized formalism for the QTI information model

- Ontology for the Information Model of a QTI item version 2.1, IMQTIO (Information Model QTI Ontology).

- IMQTIO brings various concepts keys (classes, associated classes, derived classes, attribute, etc…) charactering a QTI item. Its role is to provide a formal representation of a QTI item and to validate its structure.

- The role of IMSQTIO is to ensure a formal representation of a QTI item and to verify its structure and its compliance with the specification.

- For the development of IMQTIO, several tools are available such as: SWOOP, Protégé, WebOnto.

  Protégé provides a complete graphical formalism that facilitates the conception work.
Ontological model for a QTI item (2/3)
Advantages for the uses of IMQTIO:

- An ontology is a semantic representation of metadata for a rich description of resources, including software components.

→ IMQTIO can serve as a future support for software development.

- The strength of the ontological model is consistent with the logical description and its coupling with inference engines allowing expert systems to make logical reasoning to derive conclusions from a base of facts and a knowledge base.

→ IMQTIO can be used for the detection of different components that implement the QTI specification.

- Such ontology can also be used to verify the conformity of an item with the QTI specification.
Adding a new type of questions

Enrichment of the IMS/QTI specification

Validation approach

The composite-Text Interaction
Adding a new type of questions

- We propose to extend the QTI specification by adding new types of questions related to a programming course.
- New question type: “detect and correct errors in a program”.

Example of item:

```cpp
#include <iostream>
using namespace std;
main ()
{
    int nt ;
    int som;
    int i;
    do
        ( cout >> "terms number:" ;
        cin << nt; )
    while (nt > 1)
    for (i= 1, som= 0 ; i <= nt; i - -)
        som += (float) 1/j;
    cout<< "the sum"<< som <<"first term"<< ;
}
The **compositeText Interaction** (1/2)

**Interactions**

- **hotTextInteraction**
  Presents a set of choices represented as selectable runs of text embedded within a surrounding context.
  ➣ **hotTextInteraction**: is used to identify errors and to solve the first part of our question.

- **textEntryInteraction**
  Obtains a simple piece of text from the learner.
  ➣ **textEntryInteraction**: is used to solve the second part of our question which aims to correct the identified errors.

- **compositeTextInteraction**
  ➣ Helps to provide a context that contains selectable words. Once the word is selected, the learner is supposed to specify a new word in report.
Consider the following program that calculates the sum of the first $n$ terms of harmonic series given as follows: $1 + 1/2 + 1/3 + \ldots + 1/n$ where $n$ is an integer. **Try to detect and then correct errors.**
We examine the possibility to incorporate the added question into one of the editors supporting IMS/QTI specification.

Several editors are available to generate QTI items and tests such as Q-Player, QuestionWriter, QAed, etc.

We choose to employ Recourse Learning Design QTI editor:
- Developed by the European project TENCOnpetence.
- JAVA language, Eclipse 3.4.
- Open source.
Validation approach (2/4)

- Choice the type of question
- The environment of the question
Validation approach (3/4)

Enter the code of the item

Specify the assessment instruction

Specify the correct instruction
Validation approach (4/4)

Generate the XML/QTI code

Example

```xml
<assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqi_v2p0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.imsglobal.org/xsd/imsqi_v2p0 imsqi_v2p0.xsd" identifier="AI-719e350c-66ec-4fed-8dd4-6c13006831db-6" title="" adaptive="false" timeDependent="false">
  <outcomeDeclaration identifier="RD-78a44a56-a128-4b5b-aef4-59ca5ccc14f1-21.SCORE" cardinality="single" baseType="Integer" />
  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="Integer" />
  <responseDeclaration identifier="RD-78a44a56-a128-4b5b-aef4-59ca5ccc14f1-21" cardinality="single" baseType="string" />
  <correctResponse>
    <value>istream</value>
    <value>False</value>
    <value>iostream</value>
  </correctResponse>
</assessmentItem>
```
Conclusion

- Study the IMS / QTI allows the expression of questions and tests.

- Develop a standardized formalism for learning assessment by the design and the development of the Information Model QTI Ontology.

- Extend the QTI specification by adding a new type of questions.

- Integrate the question to a QTI editor.
Future works

- Examine the possibility to integrate this new type of questions to a LMS or a tool supporting the QTI specification.

- Study the possibility of communication of a QTI question with a domain ontology in order to automate some treatments.

- Examine the possibility of merging other types of interactions and to formulate new types of programming questions.

- Study the implementation of QTI for various assessment domains and learner profiles.