

# IMS QTI Engine on Android to Support Offline Mobile Learning

Pei Zhang<sup>a\*</sup>, Gary B. Wills<sup>a</sup>, Lester Gilbert<sup>a</sup>, Bill Warburton<sup>b</sup>, David Bacigalupo<sup>b</sup>

<sup>a</sup>School of Electronic and Computer Science, University of Southampton,  
LSL, Southampton SO17 1BJ, UK

<sup>b</sup>iSolutions, University of Southampton,  
iSolutions, Southampton SO17 1BJ, UK

\*Corresponding Author: [pz@ecs.soton.ac.uk](mailto:pz@ecs.soton.ac.uk)

## ABSTRACT

The IMS Question and Test Interoperability (QTI) specification defines a standard for representation of assessment content and results, supporting these materials to be transferred and delivered in multiple IT systems. The ability to import these learning questions and assessment results in mobile contexts offers greater flexibility and interactivity for learning. With the development of mobile technology, some recent mobile devices have the capability to provide a platform on which more complex applications could be implemented and therefore have the potential to be used as powerful offline learning tools. In this paper, we describe a mobile QTI rendering architecture and tools built on top of the core software library of “ASDEL” to deliver an assessment consisting of an assembly of QTI items and retrieve assessment results. A mobile QTI engine is deployed upon an Android system that provides services for rendering and processing QTI xml. It allows learners to play QTI items natively and get feedback effectively which enables better support for offline mobile learning.

**Keyword:** QTI, E-assessment, Android, Mobile learning

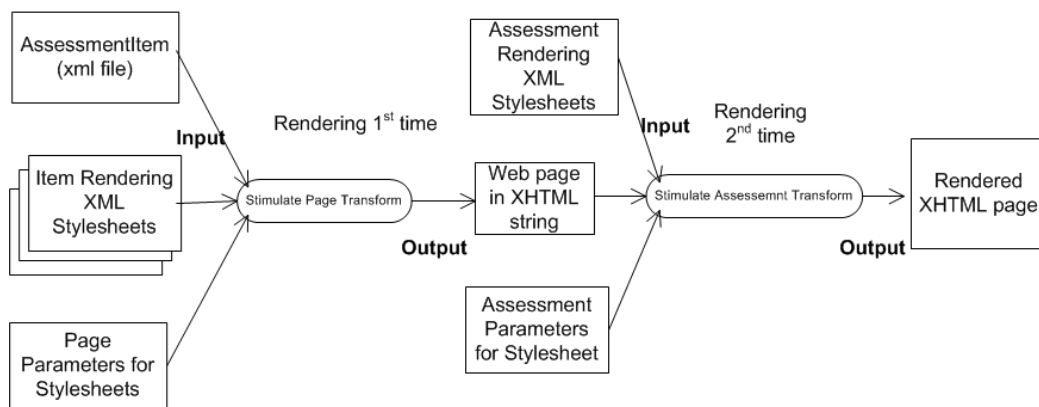
## 1. Introduction

E-assessment aims to improve reliability and provide appropriate feedback to learners that help them understand the material set more accurately and instantly by the use of technology enhanced systems such as web-based assessment tools or tutor software. The importance of e-assessment for higher education is widely recognized and some research communities work on the assessment standards. One of the most popular standards is Question and Test Interoperability (QTI) which defines a specification for representing questions and tests and the reporting of results, thereby allowing the exchange of data (item, test, and results) between tools such as authoring tools, item banks, test constructional tools, learning environments, and assessment delivery systems [1]. The ASDEL [2] project developed a QTI engine and a serial of tools based on a core software library to support formative on-line assessment. Importing

this engine to play these learning questions and assessment results in mobile context provides greater flexibility for assessment. At present, most mobile learning assessment tools are on-line applications that are limited by restrictions in network capacity or functional shortfalls such as learning content browsers which lack interactivity. This paper presents a mobile QTI rendering architecture and a native QTI player running on Android platform that allows users to play QTI questions and to get effective feedback.

## 2. Design and Implementation

In ASDEL, a core Java library called JQTI was built and an instantiation of the library into a system called playr was also developed to enable valid QTI assessment XML documents to be interpreted and executed [3]. However, in mobile platform because of the lack of necessary Java library support, it's not straightforward either to compile JQTI or to import playr on mobile. The rendering function in ASDEL is largely depended on xml stylesheet transformation, which is related to some java package not supported by Android 1.5. In ASDEL two rendering steps are made to transform an original assessment item xml file to the final display page. In the first step, JQTI render takes the assessment item xml file and the stylesheet (.xsl file) as input. Some parameters are defined in the stylesheet. The transform function provided by J2SE library will generate the rough XHTML page. In the second step, the transform programme will combine a certain look-and-feel template (also an xml stylesheet) with the XHTML string. The final output is a well-rendered XHTML page.



**Figure1. Rendering Architecture**

To make full use of the library and tools which have already been build in ASDEL, the rendering programme is re-written to stimulate the two transform behaviors according to Android SDK as shown in Figure1. In addition, a template is re-designed for mobile browsers. To be specific, Android SDK 1.5 does not have

javax.xml.transform package and some of the functions in org.w3c.dom package is different from the package in J2SE 1.5. So in the re-designed programme, all the xml stylesheets and the templates defined the stylesheets are converted into Java classes. The methods of these classes stimulate how an xml stylesheet renders an assessment item xml file. What's more, due to the difference between org.w3c.dom package in Android SDK and J2SE 1.5, some of the functions in QTI 2.0 library have to be modified so that it can be compiled and function well in Android environment.

Figure2 and Figure3 show the implementation of the mobile version of QTI player. A mobile web server I-Jetty (version 2.0) was installed on an Android mobile device and the QTI engine is ported into the device. Users can get questions/assessments from a QTI item banking system or a repository onto their local devices and then play these contents off-line without worrying about the location or network coverage. Instead of just a learning content browser, this leaning tool offers more interactivity by giving different feedback depending on users' different responses which is demonstrated on Figure4 and Figure5.



**Figure2**

**Figure3**

**Figure4**

**Figure5**

### 3. Conclusion

During the development work, we found that on the one hand, latest mobile technologies supported more and more various applications running on mobile devices, on the other hand, mobile learning software cannot act as powerful as desktop e-learning software because of both software and hardware limitations. For example, at this moment 4 out of 16 question interactions defined in QTI were implemented in our mobile assessment tool because of many reasons such as Android doesn't support Java applet etc. It is needed for such tools to support more interactivity in mobile learning and assessment which we aims to achieve in our

current and later work. The stimulation of transform behaviors is also limited compared with javax.xml.transform package.

The IMS Question and Test Interoperability specifications are a high priority to be used in e-assessment within UK assessment community. Implementing this specification in mobile learning to deliver questions improves flexibility with respect to location and timing and enables learners to get instant and effective assessment results. We developed a mobile tool on Android platform to render and execute QTI question items which supports more interactive e-assessment and off-line mobile learning.

### REFERENCES

- [1] Wills, G., Davis, H., Gilbert, L., Hare, J., Howard, Y., Jeyes, S., Millard, D. and Sherratt, R. (2009) Delivery of QTiv2 Question Types. *Assessment & Evaluation in Higher Education*, 34 (3). pp. 353-366. ISSN 1469-297X
- [2] <http://www.asdel.ecs.soton.ac.uk/>
- [3] Wills, G., Hare, J., Kajaba, J., Argles, D., Gilbert, L. and Millard, D. (2008) Assessment Delivery Engine for QTiv2 Tests. In: *International CAA Conference*, 8-9th July 2008, Loughborough UK.