

XML/XSLT-based technologies in SCORM-compliant e-learning for ODL learning support

URL	http://weko.wou.edu.my/?action=repository_uri&item_id=473
-----	---





XML/XSLT-BASED TECHNOLOGIES IN SCORM-COMPLIANT E-LEARNING FOR ODL LEARNING SUPPORT

Khor Ean-Teng (etkhor@wou.edu.my) and Chung Sheng Hung
(shchung@wou.edu.my)
School of Science and Technology
Wawasan Open University, Malaysia
Methodology and Technology

ABSTRACT

The paper discusses the implementation of dynamic techniques to generate media presentations in Learning Management System (LMS) using XML/XSLT-based Technologies. The architecture proposes to separate the appearance elements from content where contents are written in XML while the layouts for presentation are written in XSL. The proposed methods using XML and XSLT successfully separate the navigation and appearance (look and feel) elements from the underlying content for SCORM content object. The XML contents are converted into SCORM compliant output through XSL transformation process. The methods to dynamically generate media object presentations through XML and XSLT technologies are explained and the techniques were implemented in a prototype system.

INTRODUCTION

In all areas of e-era, personalization plays an important role, particularly in e-Learning (Esposito et al., 2004). In order to achieve maximal efficiency in a learning process, each student needs the own personalized treatment (Obitko et al., 2001). Hence, it is vital to overcome the one-size-fits-all approach and provide learners with individual learning experience or environment (Henze et al., 2004). In order to ensure the adaptability of each student to the course so that the student can learn better, there is a need to present the same learning contents through different media objects and generate e -Learning presentations dynamically. It is necessary to identify suitable presentation styles or formats for different students based on their preferences. Therefore, the dynamic generation of multimedia presentations is essential for any advanced distance learning system. The dynamic presentation generator is able to provide high-quality instruction anytime, anywhere, tailored to each learner's needs. It is aimed at catering multimedia information to the information needs of different learners by adapting the media and layout of presentation to each individual learner.

OVERVIEW OF SCORM

SCORM is one of the most significant and widely used specifications for learning content packaging. The SCORM Content Model describes the SCORM components used to build a learning experience from learning resources. The content model also defines how these lower level sharable, learning resources are aggregated into higher-level units of instruction. The SCORM Content Aggregation Model (ADL, 2004) contains the following components: Assets, Sharable Content Object (SCO) and Content Aggregations. Assets are an electronic representation of media objects, text, images, audio, Web pages or other data that can be presented in a Web client. A SCO represents a collection of one or more assets. To improve the reusability, a SCO should be independent of its learning context. A SCO can be reused in different learning experiences to fulfil different learning objectives. SCOs are meant to be small units, such that reusability in more learning objectives is feasible. A Content Aggregation is a map (content structure) that can be used to aggregate learning resources in a well integrated unit of education (for example course, chapter, lesson, and topic).

The SCORM standard is an XML-based industry formalization of the concept of single sourcing that enables the reuse of Web- based learning content across multiple environment and products (Rockley, 2002). SCORM is an XML-based method for representing course structures. SCORM enables the reuse of Web-based learning content across multiple environments and products. SCORM specification adopted the IMS Content Packaging specification to present in XML. It is a manifest which contains descriptions of course objectives, course structure, SCOs, and assets in a course which can run across different complying LMSs. SCORM released by ADL, describe a means of aggregating Sharable Content Objects (SCOs) using a scheme that combines Metadata, Content Packaging and Simple Sequencing. Other organizations are also contributing to SCORM by developing mechanisms to provide an adaptive sequencing of SCOs and navigation within SCOs. Parallel with these developments, some are considering the move to a Single-Source publishing paradigm for content that can provide greater interoperability for content across multiple presentation environments.

Although the SCORM uses content object technology that promotes reusability, a mechanism to allow for simple adaptation of content into these formats has not been described (SCORM, 2004). It is highly desirable to alter the appearance of a SCO without modifying the underlying content. Several organizations demonstrated the interoperability of learning content by importing SCORM and content into variety of Learning Management Systems. Content providers are often building client -specific SCOs in HTML that include a client's identity and appearance. This approach requires SCO providers to embed appearance into the SCO in such a way that it is difficult to alter after the fact. To reuse these SCOs for other clients, or in combination with SCOs from other providers, the content provider must rebuild and change the appearance of the SCO to suit new context. This problem becomes much more apparent as the number of sources for the SCOs increases.

If the appearance is defined separately from content, a SCO's appearance can be consistent with other SCOs without requiring a content provider's intervention and deliver a homogeneous learning experience. Thus, it is vital to define the separation of appearance from content (Carnegie Mellon University, 2002). The SCORM specification adopted the IMS Content Packaging specification to present in XML, a manifest which contains descriptions of course objective, course structure, SCOs and assets in a course which can run across different complying LMSs (CETIS, 2002). SCORM also provides an API and Data Model to allow the course 'object' and LMS to interact. Figure 1 explains the history of SCORM development.

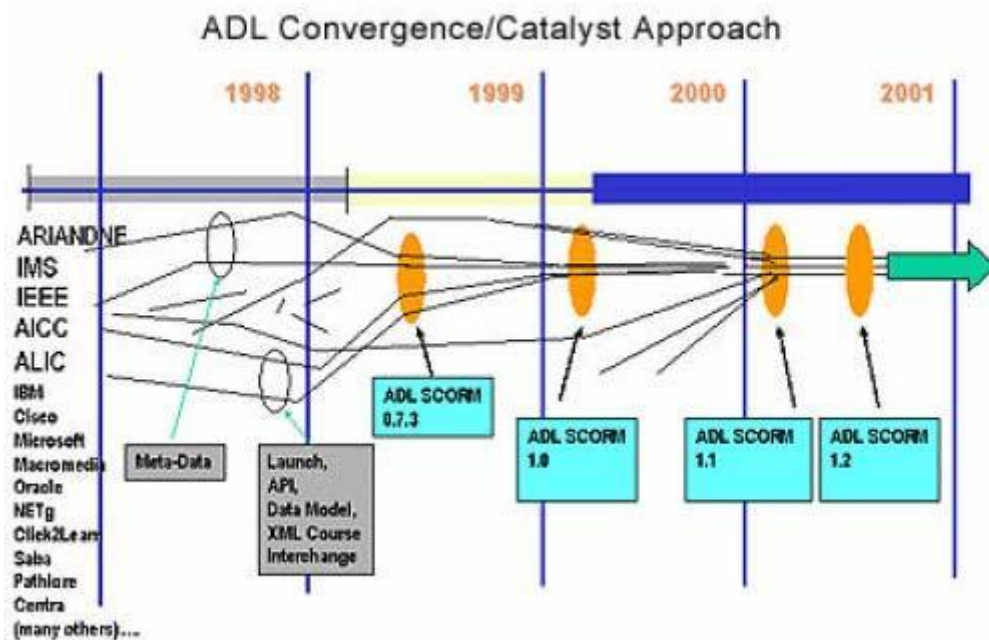


Figure 1: The Development of SCORM (ADL, 2001)

OVERVIEW OF LEARNING MANAGEMENT SYSTEM (LMS)

A Learning Management System (LMS) simplifies the process of administering education and training (Apostolopoulos et al., 2003). It is a complex system used by managers, administrators, instructors, and learners to schedule, register, bill, and track learner through courses and other learning events. It lets learners find and register progress through a course or program of learning. Finally, it helps administrators manage training programs and compile statistics and reports. LMSs help create and offer courses and curricula. Their primary function is to offer a collection of courses. They may also include capabilities for assembling individual courses into organized curricula or certificate programs (Lewis et al., 1998). At the course level, LMSs provide an ability to launch and track performance within courses.

Learning management in SCORM is handled by an LMS, whose key function is to manage and deliver content objects to learners through Web-based training (Sonwalkar, 2002). In many commercial implementations, an LMS for Web-based training often acts as the focal point for communication between the learning package and the learner, the coach, the Computer-Mediated Communication (CMC) tools and enterprise information systems. LMSs are based on a variety of development platforms, from Java EE based architectures to Microsoft .NET and usually employ the use of a robust database back-end (Beck et al., 2002). LMSs support portability and standards and should be able to personalize content and enable knowledge reuse.

XML/XSLT-BASED TECHNOLOGIES

XML/XSLT-based technologies are implemented to perform *dynamic function*. *Dynamic function* can be defined as process that creates different media object presentations at runtime to fulfill the learners' needs based on their preferences. In order to generate presentations dynamically, it is necessary to separate contents from presentations where content is written in XML and XSL is used to define presentation. Content can be referred as course data while presentation represents template for layout stylesheet. To be truly reusable, SCOs must be built on a strongly defined content structure. This structure uses XML, which is platform and software independent (Advanced Distributed Learning Initiative, 2003). Figure 2 summarizes the generation process, from XML input source file to HTML output source file through XSLT processor. XSLT processor is used to transform XML document to HTML document using XSL stylesheet.

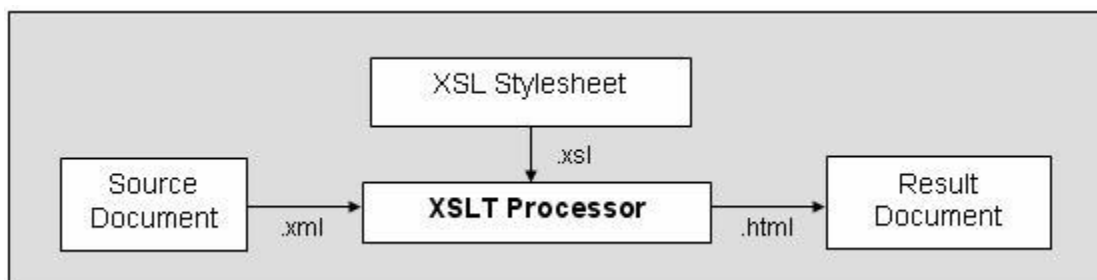


Figure 2: XML-to-HTML Transformation Process using XSLT

XSLT (W3C, 2002) is a process for transforming XML documents into another XML, HTML or WML document. The implementation of XSL transformation is to use different stylesheets to generate different presentations. In a transformation process, an XSLT processor reads both an XML document and an XSLT style sheet. Based on the instructions the processor finds in the XSLT style sheet, it outputs a new XML document or fragment thereof.

The implementation of XSL Transformation by using JavaScript is shown in Figure 3. The first block of code creates an instance of Microsoft XML parser (Microsoft Corp,

1999) which is known as XML Document Object Model (DOM) and loads the XML file into memory. The second block of code creates another instance of the parser and loads the XSL file into memory. The last line of code transforms the XML document using the XSL document and display in SCORM player. By setting .async to option false, it is saying to XML parser to show the data as soon as it begins to read it and the retrieving of XML data is speeded up.

```
function display()
{
    var xml =new ActiveXObject("Microsoft.XMLDOM")
    xml.async=false
    xml.load("SC02.xml")

    var xsl=new ActiveXObject("Microsoft.XMLDOM")
    xsl.async=false
    xsl.load("SC02(textAnimation).xsl")

    document.write(xml.transformNode(xsl))
}
```

Figure 3: XSLT Function

An XSLT document contains template rules (W3C, 2002). A template rule has a pattern specifying the nodes it matches and a template to be instantiated and output when the pattern is matched. When an XSLT processor transforms an XML document using an XSL style sheet, it walks the XML document tree, looking at each node in turn. As each node in the XML document is read, the processor compares it with the pattern of each template rule in the style sheet. When the processor finds a node that matches a template rule's pattern, it outputs the rule's template. This template generally includes some markup, some new data, and some data copied out of the source XML document. XSLT uses XML to describe these rules, templates, and patterns. The XSL Transformation makes it possible to output different versions and forms of information using the same input content.

MULTIMEDIA PRESENTATIONS

Rapid progress in technology for display, creation, storage and transfer of multimedia presentations gives the learner new possibilities to access and retrieve information of different kinds (Castells, 2001). Multimedia presentations are an effective way to present different kinds of information, since the presence of various media types give more expressive power and opportunities to catch learner attention.

It is evident that media played an important role in the content aspect of e-learning system. The benefit of e-learning system is the ability to complement the different media into a total content package that enhance learning. Every learner has their own learning preference and they learn from different sense. On order for a learning to be effective, students must choose the appropriate media.

Web sites offer a great variety of media object. However, the most popular media objects are *text*, *graphics*, *animation*, *audio* and *video*. The five types of media objects were taken into consideration for the development of the prototype systems. Students are allowed to choose any five media object or combinations of media objects. The media presentation will be then generated and display to students via LMS based on their selections. The implementation of XSL for each of the object type can be explained in Figure 4. The figure illustrates the example using video object type.

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" width="512"
height="352" title="Lesson 11">
  <param name="movie" value="elecCharge.mpeg" />
  <param name="quality" value="high" />
  <embed src="elecCharge.mpeg" quality="high"
pluginspage="http://www.macromedia.com/go/getflashplayer" type="video/mpeg"
width="512" height="352"></embed>
</object>
```

Figure 4: Implementation Function of XSL Media Object Type

ARCHITECTURE OF THE PROPOSED SYSTEM

The architecture of the proposed system as shown in Figure 5 consists of 3 main components. They are Presentation Generator (PG), Knowledge Base (KB) and SCORM player.

PG creates the multimedia presentations for students based on personalisation parameters from the SCORM player. The core of the PG is a piece of software called an XML parser. It is a component that reads an XML file and converts to another form for display in a Web browser. Microsoft's XML parser is a COM component that supports all the necessary functions to traverse the node tree, access the nodes and their attribute values, insert, delete nodes and convert the node tree back to XML.

KB forms a reusable storage of concepts to be learnt by students and to be prepared by lecturer while SCORM Player is used to deliver SCORM-compliant courses. API adapter and SCO functions are implemented to enable communication between SCO and LMS.

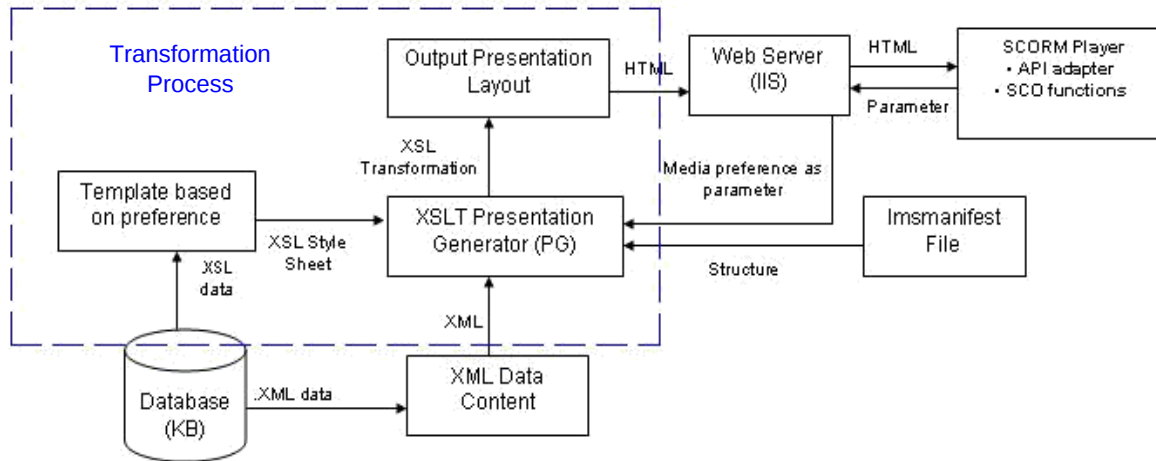


Figure 5: Architecture of the Proposed Prototype System for SCORM Conformant LMS

The LMS will send the media preference parameter to PG. PG retrieves the appropriate XSL template from database. From the input of *imsmanifest* file as structure, PG will retrieve the XML data content from database. The transformation process will be performed and the output presentation will be sent to SCORM player through IIS Web server.

OVERVIEW OF PROTOTYPE SCREENS

The prototype was developed using Microsoft .NET architecture and created a SCORM compliant LMS. The prototype is not intended to be full service LMS. Instead it is intended to be closely focused toward generation of media presentation dynamically through XML/XSLT Technologies. The XSL stylesheets development begins and the SCOs media layouts were selected for respective courses. The XSL stylesheets were determined and the XSLT SCOs pages development began. At the end, the XSLT dynamic media object presentations for respective SCOs lesson are included in manifest.

A learner can select a course from the course catalogue and register for that course if the learner has never registered before. Next, a learner may start to view or study the registered course. A learner chooses the preferred media at the beginning of the course. Presentation Generator (PG) will display the possible layouts based on the media chosen by the learner. PG will then generate the media presentation and display to students through SCORM player based on their selections.

At the beginning, students choose the course and the course will start from introduction of lesson one. For example, Figure 6 shows an introduction page of lesson one for Basic Electronics subject. After finished reading the introduction, a student will click on the

START button. The LMS will launch the media selection screen in SCORM player as shown in Figure 7. A student will choose the preferred media for SCO 1, (e.g. text and animation) and there will be a confirmation window being popped up as shown in Figure 8.

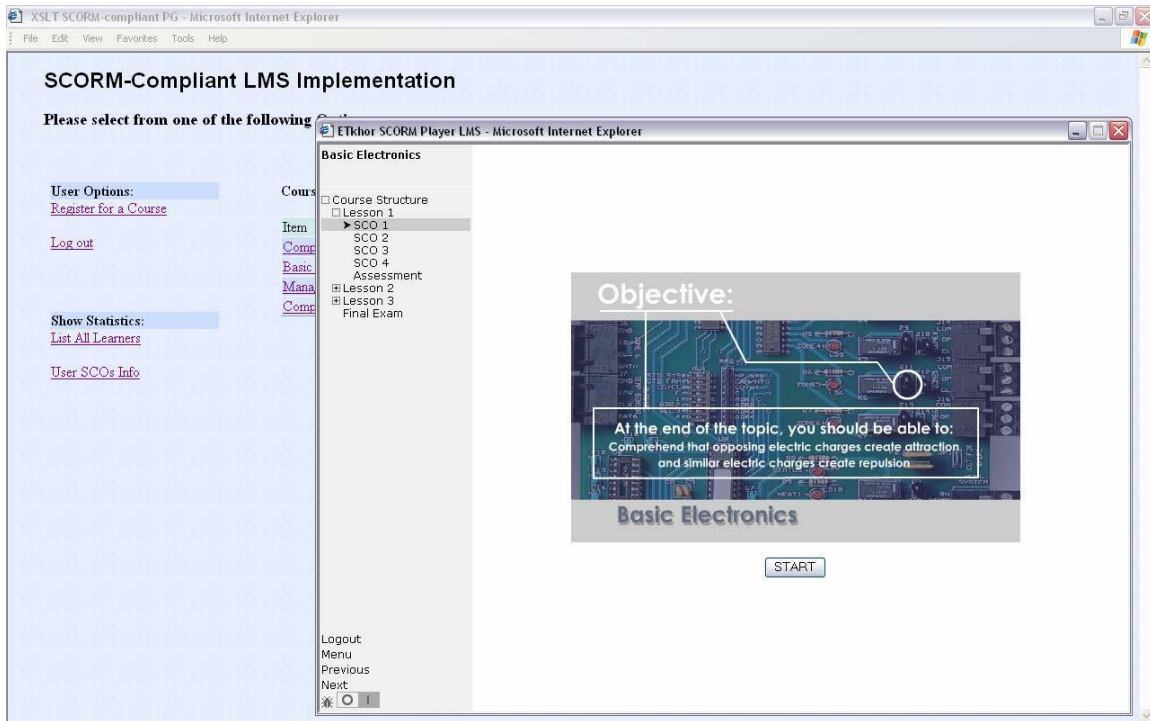


Figure 6: Lesson 1 Introduction

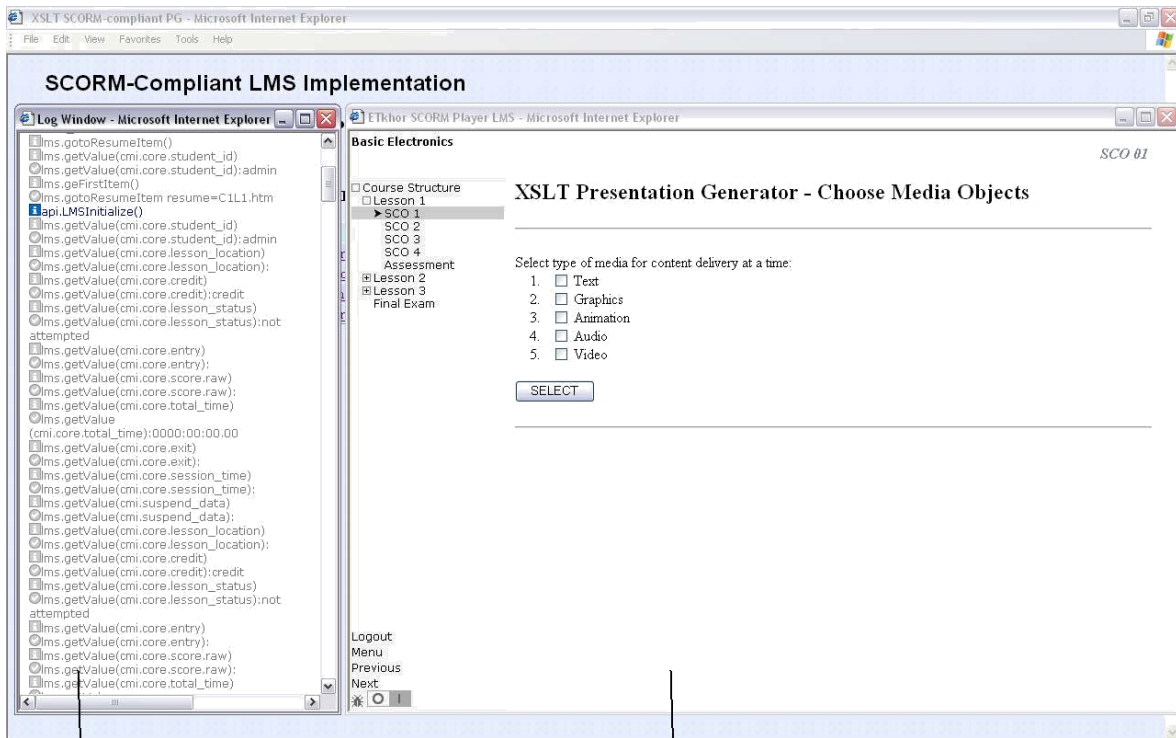


Figure 7: Media Selection Screen

Log Window

SCORM Player

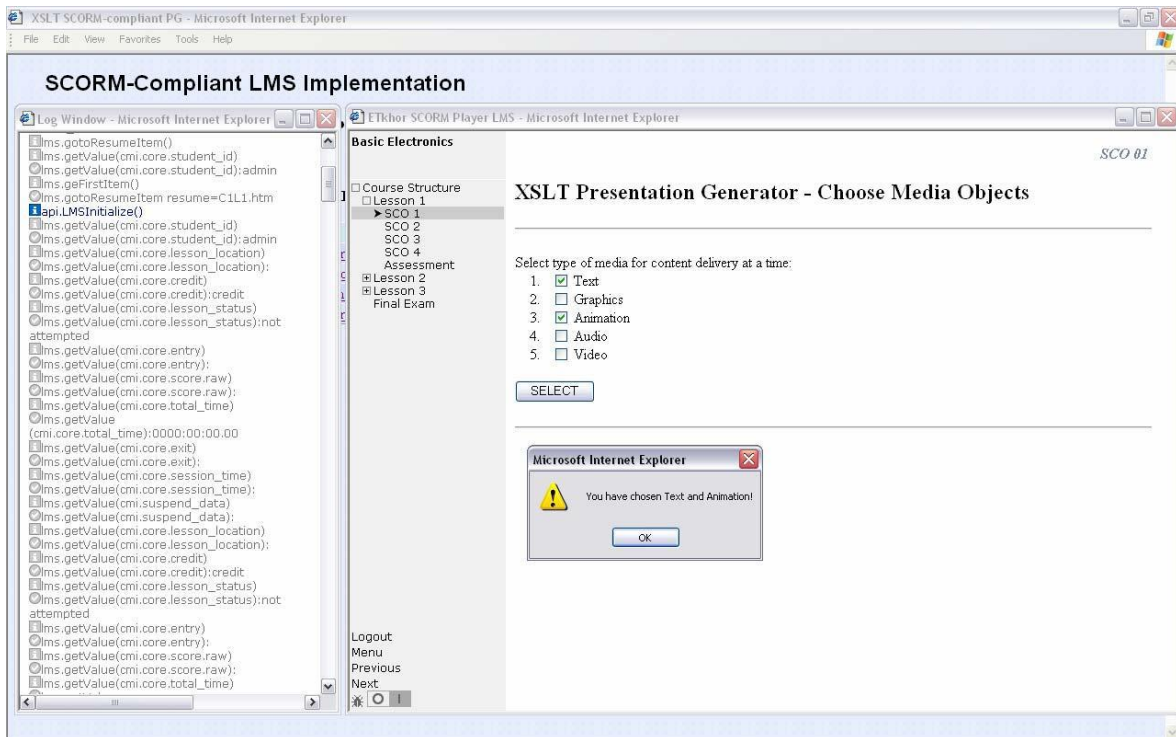


Figure 8: Media Selection Confirmation

After a student confirms the preferred media by checking the checkbox, PG will call the function and load the XSL stylesheet. Figure 9 illustrates the function for retrieving the XML data content and XSL stylesheet.

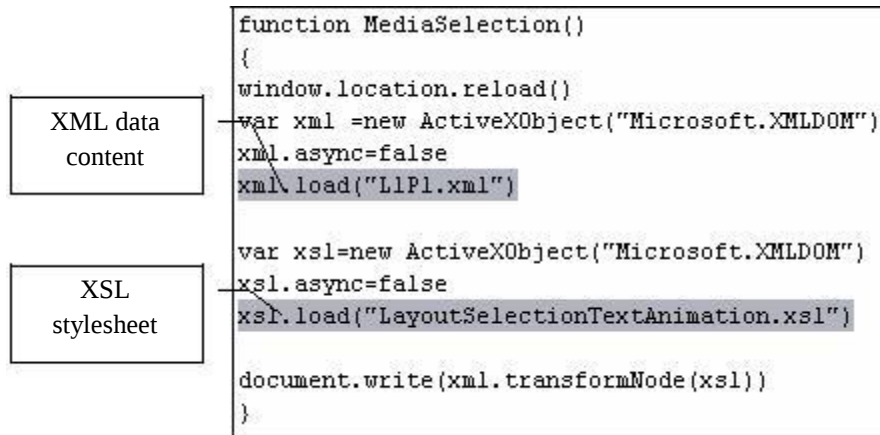


Figure 9: Media Selection for Generating of Presentation

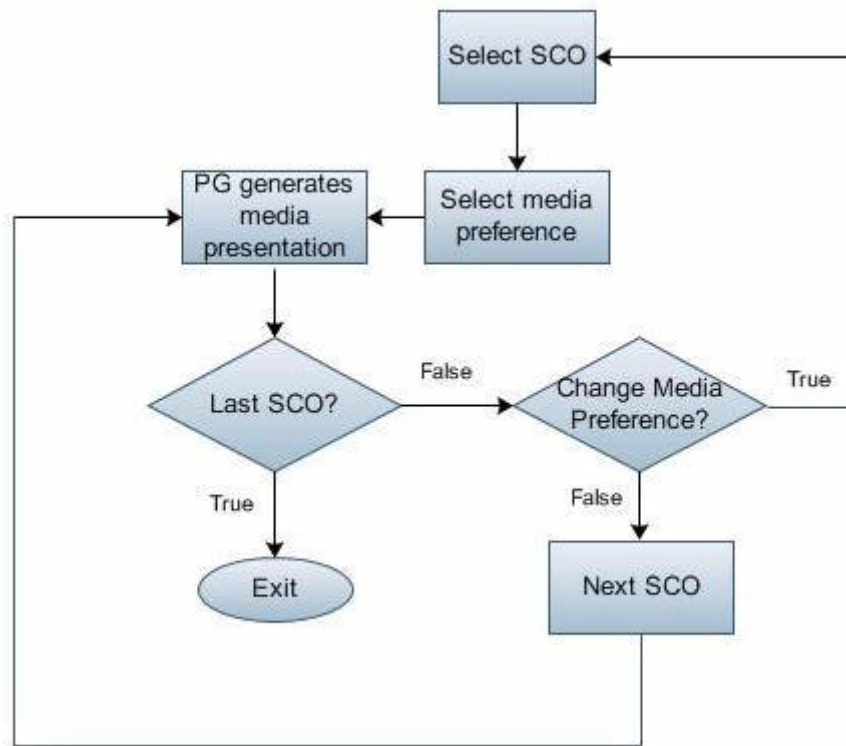


Figure 10: SCO 1 to SCO 2 Flow Chart

Each student chooses the preferred media presentation at the beginning of lesson, first SCO page. The same media presentation chosen by students will be applied throughout the all lessons of the course. For instance, Figure 11 and Figure 12 show the use of the same media presentation (text and animation) for SCO 1 and SCO 2 page.

If a student does not prefer the same media presentation to be applied throughout the lessons of course, the student may choose different media presentation for each SCO by clicking the particular SCO at side menu as illustrated in Figure 11. Figure 10 shows the presentation generation from one SCO page to another SCO page.

As observed from Figure 11 and Figure 12, the side menu explains the status of each SCO by indicating a tick (✓) next to SCO 1 when user completed the SCO. The indicator highlights the user current SCO's page (refer to Figure 12) and indicates the user has completed SCO 1 and currently reading SCO 2.

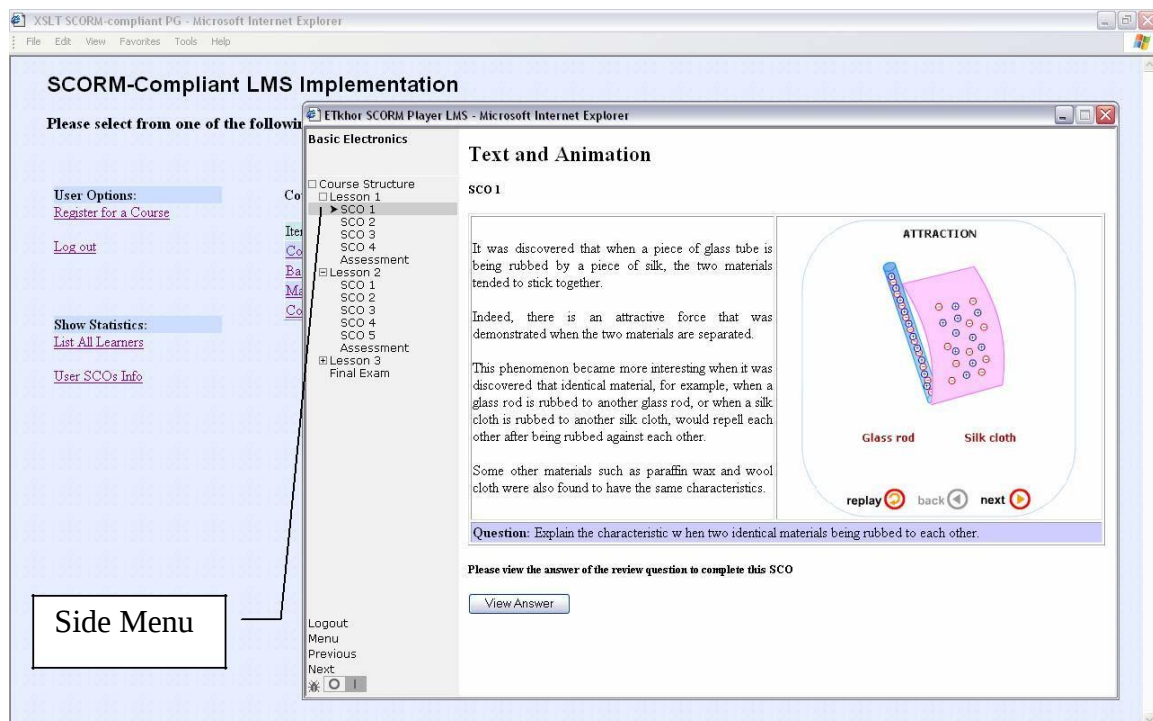


Figure 11: Content Presentation for SCO 1 using Text and Animation Media

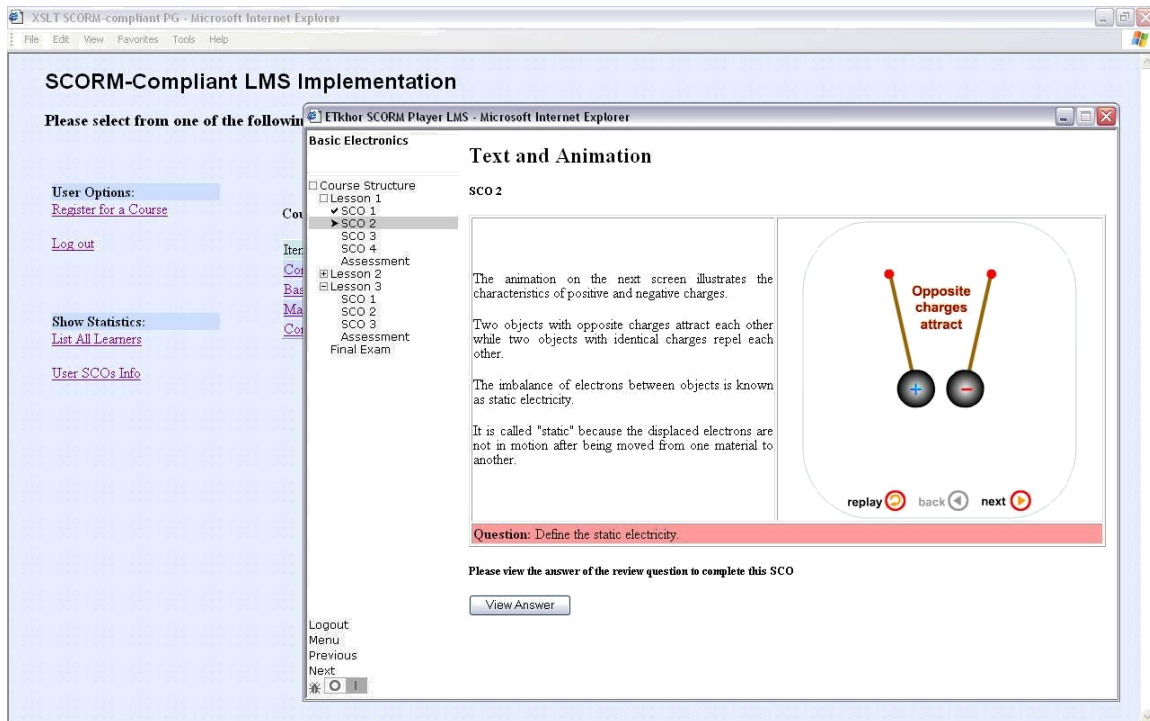


Figure 12: Content Presentation for SCO 1 using Text and Animation Media

CONCLUSION

A dynamic prototype system using XML/XSLT-based technologies is proposed and discussed. The prototype system consists of three main components: PG, KB and SCORM player. PG creates the multimedia presentations for students based on personalisation parameters from the SCORM player. KB forms a reusable storage of concepts to be learnt by students and to be prepared by lecturer. SCORM Player is used to deliver SCORM-compliant courses. The prototype demonstrated the use of XSLT and SCORM technologies to generate the presentations dynamically. Presentations with different media combinations and layouts were created using the same contents. The course contents are written in XML while the media presentation are defined using XSL. The outputs as presentations viewed by students via LMS are generated through XSL transformation process.

REFERENCES

Advanced Distributed Learning. (2004). The SCORM Content Aggregation Model-Introduction to the SCORM for Instructional Designer [Online]. Available: <http://www.adlnet.org/scorm/articles/3.cfm> [2005, October 10].

Advanced Distributed Learning Initiative. (2003). Sharable Courseware Object Reference Model (SCORM). Available <http://www.adlnet.org/Scorm/download.cfm> [2005, July 31].

Advanced Distributed Learning Org. (2001). The SCORM Content Aggregation Model. USA: ADL.

Apostolopoulos, T., and Kefala A. (2003). An E-Learning Service Management Architecture. Proceedings of the 3rd IEEE ICALT International Conference, 72(1), 56-62.

Beck, J., Baker, J., Radford, A., and Costigan, C. (2002). The Business of Open Learning: Open Learning Australia [Online]. Available: <http://www.educationau.edu.au/globalsummit/papers.htm> [2005, June 9].

Carnegie Mellon University. (2002). SCORM Best Practices Guide for Content Developers [Online]. Available: <http://www.lsal.cmu.edu/lsal/expertise/projects/develoersguide> [2006, January 11].

Castells, P. (2001). An Adaptive Hypermedia Presentation Modeling System for Custom Knowledge Representations. Proceedings of the World Conference on the WWW and Internet, 7(1), 372-377.

CETIS. (2002). Learning Technology Standards: An overview Centre for Educational Technology Interoperability Standards (UK) [Online]. Available: <http://www.cetis.ac.uk/static/standards.html> [2005, December 19].

Esposito, F., Licchelli, O., and Semeraro, G. (2004). Discovering Student Models in e-Learning Systems. Journal of Universal Computer Science, 10(1), 45-57.

Henze, N., Dolog, P., and Nejdl, W. (2004). Reasoning and Ontologies for Personalized e-Learning in the Semantic Web. Journal of Educational Technology and Society, 7(4), 82-97.

Lewis, B., Smith, R., Massey, C., McGreal, R., and Innes, J. (1998) Technology Mediated Learning: Current Initiatives and Implications for Higher Education [Online]. Available: <http://www.sfu.ca/tpi/cmec.pdf> [2005, July 8].

Microsoft Corp. (1999). Microsoft XML Parser [Online]. Available: <http://www.microsoft.com/xml/> [2005, September 19].

Obitko, M., Kurz, L., and Glucksmann, I. (2001). A Framework for Constructing Adaptive Web-Based Educational Systems. Proceedings of IEEE International Conference on Advanced Learning Technologies, 18(2) 437-438.

Rockley, A., and Manning, S. (2002). E-Learning, Single Sourcing and SCORM. Proceedings of the IEEE International Conference on Advanced Learning Technologies, 8(1), 270 -273.

SCORM. (2004). Sharable Content Object Reference Model [Online]. Available: <http://www.online-learning.com/papers/SCORMModel.pdf> [2005, July 27].

Sonwalkar, N. (2002). Demystifying Learning Technology Standards Part I: Development and Evolution [Online]. Available: <http://www.syllabus.com/article.asp?id=6134> [2005, June 26].

W3C. (2002). XSL Transformations [Online]. Available: <http://www.w3.org/TR/xslt/> [2005, December 11].