E-Learning Web Services

SU MOON TING, HOE HAN KER, YAM KIEN YEW
Faculty of Computer Science and Information Technology
University of Malaya
50603 Kuala Lumpur
MALAYSIA

Abstract: - More and more educational institutions are investing in e-Learning system to streamline the management of their educational business and to gain competitive edges. This gives rise to the proliferation of many such systems which are however non-customizable and provide very little avenue for reuse. One way to tackle these limitations is to identify the reusable portions of e-Learning system and build them as reusable components in the form of web services. Creating new e-Learning system will then involve assembling the needed services. This will tremendously reduce the time, effort and cost needed in building new e-Learning systems. The work presented here aims to construct the reusable e-Learning components as web services. Several web services are constructed (such as attendance taking, test/quiz, timetable generation and course management).

Key-Words: - E-Learning, service-oriented architecture, web services

1 Introduction

More and more educational institutions are investing in e-Learning systems to streamline the management of their educational business and to gain competitive edges. This gives rise to the proliferation of many such systems which are however non-customizable and provide very little avenue for reuse. Developing an e-Learning system requires a lot of resources in terms of time, effort and cost. When reusability is not possible, the carefully and painstakingly 'crafted' software artifacts are not 'transferable' to other similar systems and thus remain to be useful to one and only one system. Even when transfer is possible, the amount of customization needed to be done is too much to be deemed profitable.

Thus, the objective of this research is to investigate the reusable portions of e-Learning systems and to build them as reusable components in the form of web services. Creating new e-Learning system will then involve assembling the needed components which come in the form of services. This will tremendously reduce the time, effort and cost needed in building new e-Learning systems. Besides, as independent component by itself, a web service can be used on its own without being integrated into a complete system.

The outcome of the work will be a repository of sharable e-Learning service providers. These reusable e-Learning components exposed as web services, serve as service providers that offer services to e-Learning applications or end users.

2 Reusability and Interoperability in E-Learning

The developments of e-Learning systems are facing issues of reusability and interoperability. What is needed is to decouple the content from proprietary system and platform. Consequently, significant attention is being paid to the establishment of open standards for e-Learning. Some of these are described in this section.

In the aspect of learning system architecture, there is Learning Technology Standard Architecture (LTSA) [1]. It provides a generic architectural framework for building electronic-based learning, education and training systems which are assembled out of different components to address different needs of storage and delivery functionality [2]. The LTSA system components include processes (learner entity, evaluation, coach, delivery), stores (learner records, learning resources) and flows (learning preferences, behavior, assessment information, learner information, query, catalog info, locator, learning content, multimedia, interaction context).

Besides LTSA, Open Knowledge Initiative (OKI) aims at defining an open and extensible architecture for learning technology specifically targeted to the needs of the higher education community [3]. It also specifies the communication between components of a learning technology environment and with other campus systems. OKI stresses on interoperability. Components of a learning environment can be developed

independently and these components which are developed based on the OKI Application Programming Interface (API) can be plugged into any learning infrastructure that conforms to the OKI architecture. In addition, the COLIS (Collaborative Online Learning and Information Systems) project was also initiated to demonstrate the integration of e-learning components based on interoperable standards [4].

Apart from these, there is a joint development of standard by UK and Australia, which is known as e-learning technical framework. The parties involved are JISC (Joint Information Systems Committee), Centre for Educational Technology Interoperability Standards (CETIS) and Department of Education, Science and Training Australia (DEST) [5]. This e-Learning Framework is a service-oriented factoring of the core services required to support e-Learning applications, portals and other user agents [6]. For each of the service defined in the framework, it is envisaged that it be provided as a networked service within an organization [7]. Typically, these services will be implemented using either Web Services or a REST-style HTTP protocol. To achieve highest interoperability, the framework's ultimate goal is for each identified service to be able to be implemented using an open specification or standard [7]. It also aims to assist e-Learning system developers by providing open-source implementation toolkits. Further details on this framework and each of the services can be found at [8].

At the same time, considerable effort had been expended in reusing learning content. One of this is Sharable Content Object Reference Model (SCORM) produced by Advanced Distributed Learning Network. It is a set of specifications and standards to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility and reuse of web-based learning content [9]. It is built upon the work of AICC (Aviation Industry Computer-Based Training Committee), the IMS Global Learning Consortium, IEEE, ARIADNE (Alliance for Remote Instructional Authoring and Distribution Networks for Europe) and others to create a unified reference model for learning content. SCORM-conformed learning management system is able to deliver the right content to the learner based on his mastery of skill or competency.

Apart from SCORM, there is also Learning Object Metadata (LOM) standard that specifies the syntax and semantics of Learning Object Metadata, which is used to describe Learning Objects using attributes (in other words to support the annotation of Learning Objects) [10]. Learning objects are defined here as any digital or non-digital entity, which can be used, re-used or referenced during technology-supported learning. There are two aspects of the metadata descriptions of the learning objects [2]. The content description describes the educational aspects while the infrastructure description describes the technical aspects related to the assembly of the learning objects. These descriptions are held in searchable repositories which can be queried to formulate requirements in terms of the attributes.

3 Reusability via Web Services

Earlier work in the domain of learning had indicated that Web services are likely to have great impact on the future design of learning management systems (LMS) [11]. The technology of Web service has the ability to increase the much needed interoperability and reusability of educational components [12]. Prior to web service, many approaches had been used: educational task ontologies, design patterns, framework systems, wrapping legacy system and so on. However these approaches are limited in term of difficulty in the interpretation of the input and output of third-parties components which are improperly documented. Besides, in order for the components implemented in different languages and platforms to communicate, agreements needed to be arrived on the standards to pass data, the protocols and platforms.

Web service has the ability to solve these obstacles through its service description which contains the information that tells the service requestors on how to use it. Web services are based on open standards such as HTTP, XML, and SOAP [13]. These standards allow components of distributed applications and other services to be provided over the web, independently of the language, platform and vendor. With these capabilities, e-Learning components built as web services can be assembled to produce new e-Learning systems that meet the specific requirements of different educational institutions. Besides, since web services are loosely coupled, changing implementation of a service will not affect the connected partner. With such, a high level of reusability, interoperability and scalability can be achieved for e-Learning components.

4 Design Architecture

In this study, the identified e-Learning components are implemented as web services. The major benefit

of implementing e-Learning components as web services are that these services can be provided over the web easily. The provision of e-Learning services enables new e-Learning systems to be created by accessing the needed web services. Other than that this approach is not tied to any specific language of implementation, platform and vendor.

The e-Learning Technical Framework by JISC, DEST, CETIS is used as a guideline in identifying the functionalities that the web services need to provide though the implementation and certain interpretation do not follow the framework. Following that, a layered architecture is used in structuring the e-Learning services. As shown in Fig. 1, it consists of three layers: User Agent, Learning Domain Services and Infrastructure.

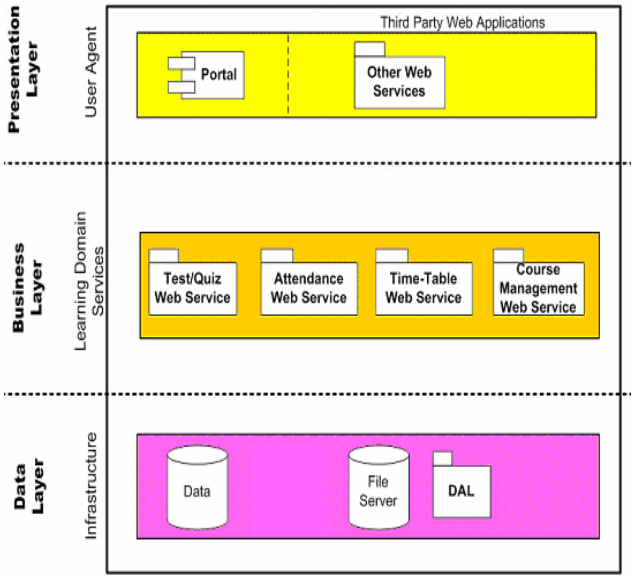

Fig. 1: Architecture for e-Learning web services

The User Agent layer is the presentation layer. It interacts and deals with the users directly. In this study, the portal and other third party applications are in this layer. From the portal, users will be able to access the Learning Domain Services. Currently, Learning Domain Services layer comprises of Test/Quiz, Attendance, Timetable Generation and Course Management web services. The Infrastructure layer consists of databases to store relevant data, file server and Data Access Layer (DAL) to access the databases. Adopting a layered architecture enables services to be segregated into their respective groups. Subsequent expansion of the set of learning services will involve the Learning Domain Services layer without affecting the other layers.

The first web service is the Test/Quiz web service which supports the administration of both online multiple-choice and true/false tests/quizzes. It employs assessment functionality to facilitate the delivery and scores of each test/quiz. It allows instructors to upload and store the questions and answers at the server. It also performs automatic marking of a particular instance of test/quiz taken by a learner as well as takes charge of the submission of grade for the test/quiz. Subsequently, it uses the reporting functionality to generate test/quiz reports. The reports vary from an overall performance analysis to an in-depth breakdown details.

The Timetable Generation web service provides the functionalities to generate the learner's individual timetable, the timetable for a particular course. The timetable is generated by retrieving the timetable details based on the identifier of the user and his course registration information.

The Attendance web service is used to record learners' attendance electronically. The attendance list is generated based on the course timetable details and the students registered for the subject. The generation of attendance report is also supported. The Course Management web service supports the management of courses, modules and other units of learning. It also allows the generation of emails to the instructor as a reminder to upload learning materials on time and to the learner as notification on newly uploaded learning materials.

The above web services make use of some common services such as Authentication, Authorization and Email Management. Besides that, the Role service allows the role and the access priority for each user to be determined when the user accesses the web services. There are three types of roles, namely administrator, lecturer and student.

5 Implementation

Following the design architecture, the implementation code is separated between the three different layers. Hypertext Markup Language (HTML) is used for structuring the interface in the form of web pages and user controls while Cascading Style Sheet (CSS) is used to define the style and format the display. C# language is used to program the code of the portal and the business logic layer of the architecture. Stored procedure written in the Structured Query Language (SQL) is used in the data access layer.

The Test/Quiz web service starts operational with the instructor first uploading the questions and answers of a test/quiz. The questions and answers are written according to a specific manner in an extensible markup language (XML) file. The main

rationale of doing so is to harness the power of XML file. An example of the file is as below:

```
<?xml version="1.0" encoding="UTF-8"?>
<test><objective>
<question>What does SQL stands for?</question>
<answer>Stand Query Language</answer>
<answer>Still Query Language</answer>
<answer>Small Query Language</answer>
<answer>Sure Query Language</answer>
<answer correct="yes">Structured Query Language
</answer>
</objective> </test>
```

In the XML file, a node is declared to contain the question itself while other nodes are used to hold all the options of answers. Subsequently, the node which corresponds to the correct answer will contain an attribute that differentiates it from the rest of the options. With that, code can be written to manipulate these XML nodes to support the processes of question loading and marking. The marking and grading functions are developed as web methods inside the Test/Quiz web service. The marking function will automatically return an indicator string upon comparing the correct answer with the learner's selected option. The totals of both correct and incorrect answers are then counted. The grading function will automatically return the appropriate grade from the predefined sets of grades and marks. Within seconds, learners are able to view their result. Also, the marks and grades are stored at the database simultaneously.

The Timetable web service generates the timetable in XML format. Below is an excerpt of the XML file:

```
<Eight_am/>
<Nine_am/>
<Ten_am/>
<Eleven_am/>
<Twelve_pm/>
<One_pm/>
<Two_pm/>
<Three_pm/>
<Four_pm/>
<Five_pm/>
<Six_pm/>
<Seven_pm/>
<Eight_pm/>
<Nine_pm/>
<Clashing/>
```

Each of the time nodes contains brief timetable information such as course code, venue and course type while the clashing node holds the clashing courses information. Finally, from the XML format, it will be transformed into a human readable text in a form of table.

In the implementation of the Attendance functionalities, the abstraction is decoupled from its implementation. It contains the Attendance class that implements the IAttendance interface. When the functionalities are exposed as web service, the web methods use the interface instead of the class directly. Thus, any changes made to the implementation will not affect the web service. Each method is designed to handle single transactional logical unit of work in order to promote reusability. In other words, methods are designed to be highly cohesive and loosely coupled. For example, a single call to the UpdateAttendanceList web method will record in a single transaction, the students who are present, absent or absent with reason for a particular course.

In term of the attendance reports, they are generated using Crystal Report by making use of XML data schema. The data schema provides the metadata and the definition for the data source. Basically, the data used in the Crystal Report are pulled from the database based on the schema defined. This promotes flexibility as the report data source is independent of the database structure due to the abstraction provided by the XML schema. The Course Management functionalities are implemented using the same approach as used in the Attendance.

Besides the web services, two window services are created. They have the ability to start automatically when the computer boots and can be manually paused, stopped or even restarted. The first window service namely Attendance window service triggers the automatic generation of notification email to related learners to inform them about their absenteeism. It supports the Attendance web service. On the other hand, Reminder window service generates email reminder to remind instructors to upload the learning materials on time. This window service supports the Course Management web service.

The presentation layer can be implemented using programming languages other than those used to implement the web services. In this study, and web services were implemented using C# with Microsoft .NET. The portal which provides access to the web services is implemented using C# as well. Nevertheless, these web services can be accessed by third party applications written in other

programming languages such as JAVA and PHP. The development model used in this project is iterative-and-incremental model [14].

6 Discussion of Results

Fig. 2 shows the portal main page whereas Fig. 3 to Fig. 6 show the snapshots of the developed e-Learning web services accessed through the portal. The core benefit from this implementation is reusability. System developers who wish to develop such system can reuse the desired parts by calling the web services instead of building the logics from scratch. The system logics, which are written in the web services, can be reused by other applications that conform to the web service specifications.

The Attendance web service generates a list of students based on the course registration details and enables the instructor to indicate whether the students are present or absent and to enter reasons for absenteeism. Storing the attendance records in the database allows attendance reports to be easily generated and this is particularly useful in tracking the students' commitment to formal classes. The course management web service enables instructor to upload learning materials for a particular course and registered students to download them.

The Timetable Generation web service will automate timetable generation processes according to each user and subject. For both instructors and learners, the generated timetables are certainly much more reliable as the related data is retrieved directly from the course database, reducing much human errors.

The Test/Quiz web service offers a cheap and effective way for institutions to provide online multiple-choice or true/false questions. It promotes tests/quizzes to be conducted online instead of offline, with both the marking and grading processes done automatically. This will definitely reduce instructors' workload of marking and grading each learner's test/quiz. By automating these processes, a better online test/quiz management can be ensured. The reporting analysis tools included in the Test/Quiz web service can be utilized to scrutinize learner's performance in each test/quiz.

The web services constructed were independent components. The modularity of these components is ensured where these web services can be modified or removed without affecting one another. New web services can also be added to the repository without much problem.

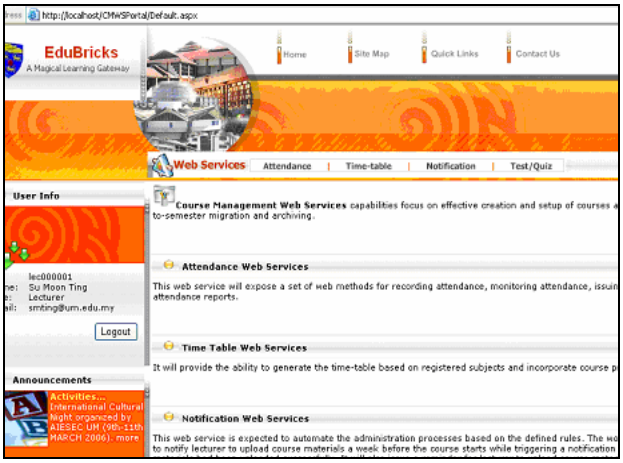

Fig. 2: The portal of e-Learning web services

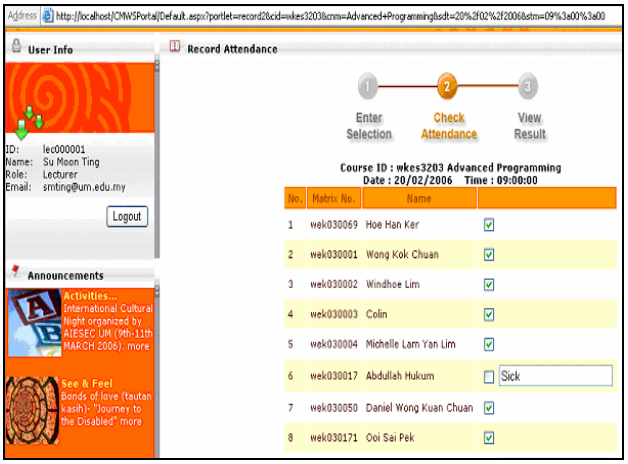

Fig. 3: Attendance web service

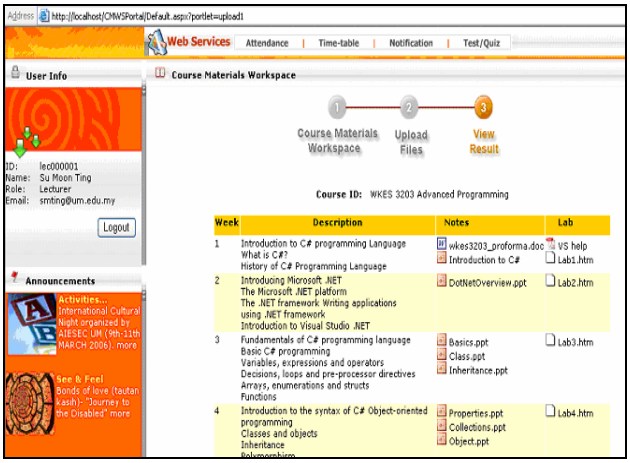

Fig. 4: Course management web service

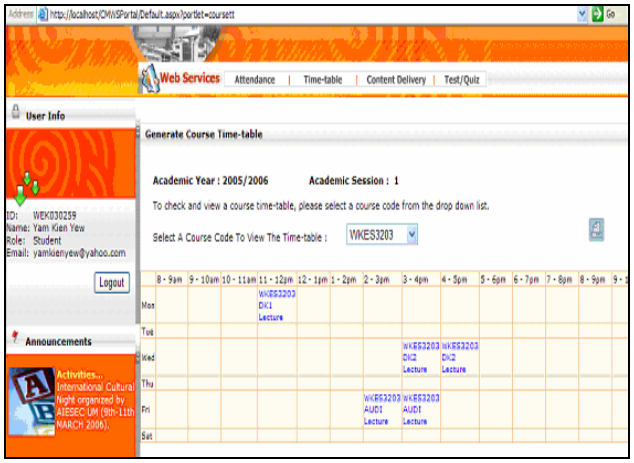

Fig. 5: Timetable generation web service

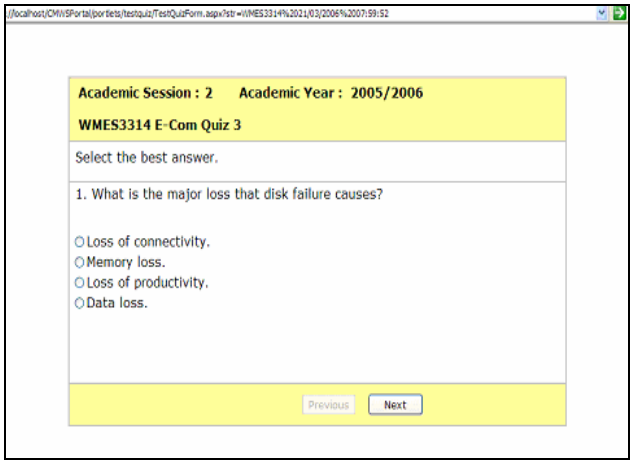

Fig. 6: Test/Quiz web service

4 Conclusion

This study produces a set of services that are useful for e-Learning systems. Further work will include adding new web services such as activity management, forum, search and so on as well as building learning content based on one of the standard specifications. By having a repository of e-Learning web services, a great deal of benefits can be gained in the development of e-Learning systems thus producing more robust systems at a faster rate.

References:

[1]IEEE, Learning Technology Standards Committee. IEEE P1484.1/D9, 2001-11-30 Draft Standard for Learning Technology - Learning Technology Systems Architecture (LTSA), 2001.http://ltsc.ieee.org/wg1/materials.html

[2] C. Pahl, R. Barrett, A web services architecture for learning object discovery and assembly, *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, WWW 2004, New York, USA, 17-22 May 2004, pp. 446-447.

[3] G. Collier, R. Robson, What is the Open Knowledge Initiative?, 2002. http://www.caret.cam.ac.uk/archive/pdfs_ppts/OKI_white_paper_092302.pdf.

[4]J. Dalziel, Reflections on the COLIS (Collaborative Online Learning and Information Systems) demonstrator project and the Learning Object Lifecycle, 2002. http://www.ascilite.org.au/conferences/auckland02/proceedings/papers/207.pdf

[5] JISC, DEST, CETIS, 2004. The E-Learning Framework Site. http://www.elframework.org/

[6] T. Roberts, JISC e-Learning Programme - Technical Framework and Tools Strand, 2005. http://www.jisc.ac.uk/elearning_framework.html

[7] S. Wilson, The case for a technical framework to support MLEs and e-Learning, *CETIS report*. Oct 2003. http://www.jisc.ac.uk/uploaded_documents/CASE-for-ELF.doc

[8] JISC, DEST and CETIS, The E-Learning Framework, 2004-2006. http://www.elframework.org/framework

[9] B. Spigarelli, Introduction to the SCORM for Instructional Designer, Advanced Distributed Learning, 2004. http://adlnet.org/search.cfm

[10]IEEE Learning Technology Standards Committee, WG12: Learning Object Metadata, 2005. http://ltsc.ieee.org/wg12/

[11] C. Mandal, V. L. Sinha, C. M. P. Reade, Web-based course management and web services, *Electronic Journal on E-learning*, February 2004, vol. 2, no. 1, pp. 135-144.

[12] W. Chen, Web Services -- What Do They Mean to Web-based Education?, *Proceedings of the International Conference on Computers in Education (ICCE'02)*, IEEE, 3-6 December 2002, Auckland, New Zealand, 2002, pp. 707-708.

[13]Microsoft, Microsoft MSDN, Web Services, 2000.hrtp://www.msdn.microsoft.com/webservices/

[14] Schach, S.R, *Object-Oriented and Classical Software Engineering*, 6th edition, McGraw-Hill, USA, 2004.