Intelligent Learning Objects (LOs) Through Web Services Architecture

Ahmad Luthfi[1]
Faculty of Computer Science
Bina Darma University
Palembang, Indonesia

***Abstract*— In recent years, e-learning has started to attract a lot of attention from researchers as well as practitioners. Many of the existing architectures for e-learning systems are based mainly on plain client-server or peer-to-peer architectures, and therefore suffer from drawbacks like poor scalability or the complicated interchange of content. In this paper we present a distributed, service-oriented architecture for e-learning systems based on web services, and describe the extensions to support software agents. Moreover, we show what advantages such an architecture may have to offer and propose the usage of intelligent software agents for the distributed retrieval of educational content. The implementation of these services enables a reuse of functionalities of an e-Learning platform. The present research has identified and created common services, which are essential for the creation and authoring stages of typical e-Learning system architecture by utilized Learning Objects (LOs). These services are Web Services based and will provide a common interface between various components leading to platform independence and interoperability between learning system.**

Keywords: e-Learning, Web Services, Learning Objects

1. INTRODUCTION

E-learning platforms and their functionalities today resemble one another to a large extent. Recent standardization efforts in e-learning concentrate on the reuse of learning material, but not on the reuse of application functionalities. Our *LearnServe* system builds on the assumption that a typical learning system is a collection of activities or processes that interact with learners and suitably chosen content, the latter in the form of learning objects. This enables us to subdivide the main functionality of an e-learning system into a number of stand-alone applications, which can then be realized individually or in groups as Web services.

The implementation of these services enables a reuse of functionalities of an e-learning platform. The *LearnServe* system is based on common standards, both in the area of e-learning and in the area of Web services. The realization in a distributed fashion leads to a number of challenges including the maintenance of content and services. However, on the other hand, it has potentials like direct integration of e-learning services into business applications or the access of learning services by different devices if there is an appropriate client for that device.

There are numerous supplementary factors to realize the target and concept of educational technology. For example, the rapid development in information technology has produced faster and better technologies in both hardware and software. As a result, by utilizing them humans beings can improve their skills and work more efficiently.

The Internet provides a distributed infrastructure for sharing information globally, with one estimation that the on-line population will reach 6,300 million users in 2004 [1]. This vast user market becomes a great motivation for the development of new technologies which enables one to build the next generation of the web based application. In particular, the size of the user base makes it attractive to develop collaborative applications that link the growing number of diverse clients with rich media web content.

Many institutions are currently offering courses for tertiary education. In order to pass a course, participants receive a checklist that describes the content of the course. Based on these descriptions, these learners can freely choose content from various providers. Traditional e-Learning platforms do not provide the flexibility that a learner needs in tertiary education. These platforms are usually centralized and offer courses with well-defined or fixed content instead of flexible checklists. Learners do not have the ability to choose from content offered by different authors and styles within a course. Moreover, the content is usually not selectable and adaptable to the specific needs of a learner [2].

A general consensus seems to exist regarding roles played by people in a learning environment as well as regarding the core functionality of modern e-Learning platforms. The main players in these systems are the learners and the authors; others include trainers and administrators. Authors (which can also be teachers or instructional designers) create content, which is stored under the control of a learning management system (LMS) and typically in a database [3, 4].

Existing content can be updated and also reused in other e-Learning systems. The administrator controls the learning management system (LMS). The LMS interacts with a run-time environment, which is addressed by learners, who in turn may be coached by a trainer. The interesting aspect of this

[1]Ahmad Luthfi is with Faculty of Computer Science, Bina Darma University, Jl. Jenderal Ahmad Yani No. 12 Palembang 30264, South Sumatera, Indonesia. Email: luthfie@mail.binadarma.ac.id.

An earlier version of this paper was presented at the ICA2009 International Conference in October 2009 in Bandung, Indonesia.

idea is the fact that these three components of an e-Learning system can be logically and physically distributed. That is, they can be installed on distinct machines and offered by different providers or content suppliers. In order to make such a distribution feasible, standards such as IMS and SCORM ensure plug-and-play compatibility to a large degree [5].

E-Learning systems are often not limited to only addressing the needs of a specific learner. They can be implemented in such a way that a customization of features and of the content appearance is adaptable to the needs of the individual learner. Learners vary significantly in pre-knowledge, abilities, goals for approaching a learning system, pace of learning, way of learning, and the time (and money) they are able to spend on learning. To fulfill the needs of such a flexible system, a learning platform has to meet a number of requirements, including the integration of a variety of materials, the potential deviation from predetermined sequences of actions, personalization and adaptation, and the verifiability of work and accomplishments [6].

We consider the approach to construct (standardized) wrappers around e-Learning content as being more promising. In particular, we follow a service-oriented approach that encapsulates educational content inside a Web Service in order to increase interoperability and re-usability. In addition, we also propose a general service-oriented architecture for e-Learning systems, in which the different components are implemented as Web Services [7, 8, 9]. The expected advantages are that system components and content can be distributed all over the Web and offered by different vendors.

Furthermore, the format in which content is stored will be less important, since the Web Services may provide functionality to extract and present it over the Web. For learners, e-Learning systems can be individually assembled by using the distributed components to provide the functionality they really need. We address the problem of content retrieval in such a distributed environment by using intelligent software agents, which take both the preferences of particular users as well as the data into consideration, which is stored inside services. The main focus of the paper is on a general service-oriented architecture for e-Learning systems based on Web Services.

In this paper, we present a mix between literature review and experimentation about e-Learning system, including Web Service Architecture, Learning Object Materials, and the LCMS concept. Moreover, we show the advantages that such a web service architecture may offer and propose the usage of e-Learning System for the distributed retrieval of educational content.

2. E-LEARNING STANDARDIZED

This section briefly introduces some fundamentals relevant to our work, which consists of e-Learning and the Web Services approach.

2.1 E-Learning System

E-Learning has attracted a lot of attention in recent years from researchers as well as practitioners.

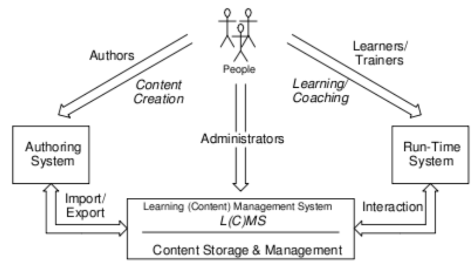

Figure 1. General View of e-Learning System

As depicted in Figure 1, a general agreement exists regarding roles played by people in a learning environment, as well as the functionality of e-Learning systems required in general [9]. In a typical learning environment, there are several groups of people involved: authors and learners, which are the main players, and administrators and trainers. Authors can be teachers or instructional designers who create e-Learning content by using an authoring system.

The core of an e-Learning system, which is under the control of an administrator, typically consists of a learning management system (LMS) or learning content management system (LCMS). An LMS provides functionality like managing learners and their profiles, tracking their progress, easing collaboration, or scheduling events. An LCMS is aimed at managing learning content which is typically stored in a database.

Content consumed by learners and created by authors are commonly handled, stored, and exchanged in units of learning objects (LOs). Basically, LOs are units of study, exercise, or practice that can be consumed in a single session, and they represent reusable granules that can be created independent of the type of delivery medium. The LOs can be accessed dynamically (e.g. over the Web). Ideally, LOs can be reused by different LMS and plugged together to build classes that are intended to serve a particular purpose or goal [10]. Accordingly, LOs need to be context-free, which means that they have to carry useful description information on the type and context in which they may be used. For example, a LO dealing with aching the basics of the SQL language can be used in classes on software engineering, database administration, and data modeling.

The concept of LOs represents one possible solution to the interoperability problem, since LOs are intended for use in many different e-Learning systems. In our view, Web Services represent a suitable technology for the implementation of LOs, since they address a similar problem domain and build on widely spread standards. A collection of Web Services can be employed to handle content and course offerings as well as other LCMS functionality. We briefly introduce the Web Service paradigm next.

2.2 Web Services

Web Service is a technology that has been developed to provide various types of services over a web connection. The main advantage of using Web Services technology is cross-platform communications. At the start of 2005 there were two major competing technologies in Web Services, namely from Microsoft and Sun Microsystems. As far as implementation is concerned both use common standards and protocols, such as *Simple Object Access Protocol* (SOAP), *Extensible Markup Language* (XML), *Web Service Description Language* (WSDL) and *Universal Discovery Description & Integration* (UDDI). SOAP is an XML-based message exchange protocol that is used to communicate between Web Services and their clients [12]. With the help of this lightweight protocol we can easily exchange structured information in a decentralized distributed environment.

WSDL provides description of a Web Service. Each Web Service has a WSDL file which is basically an XML file that describes a set of SOAP messages the Web Service uses and how the messages are exchanged between Web Services and clients [13]. UDDI is often called the Yellow Pages of Web Services [13]. A UDDI is a directory of Web Services having XML files describing a business and the services it offers. We will use UDDI in our architecture.

In essence, Web Services are independent software components that use the Internet as a communication and composition infrastructure. They abstract from the view of specific computers and provide a service-oriented view by using a standardized stack of protocols. In a typical invocation of a Web Service, a client may use a UDDI registry and the UDDI protocol to find a server that hosts a service. It then requests from the server a WSDL document written in the Web Services Description Language, which describes the operations supported by the service. For the invocation, the Simple Object Access Protocol (SOAP) protocol can be used, which builds upon HTTP to transport the data. More complex Web Services can be composed out of existing services using, for example, BPEL4WS. For more details on Web Services we refer the reader to [11].

2.3 Web Services Architecture

In a Web Service-based computing model, both clients and Web Service providers are unaware of implementation details. If the client wants to consume a Web Service, the client will need to go through four stages. These four stages are directory, discovery, description and data which is also called the wire format [13]. Figure 1 presents a Web Service infrastructure. At the first stage (directory), a client searches for a Web Service. Directories services such as UDDI provide a central place for storing published information about Web Services. The client searches a directory and finds a URL.

In the second stage a client sends a request for service description documents. The server returns the discovery document that enables the client to know about the presence of a Web Service and its location. In the third stage the client sends his request for a particular Web Service. The service description is sent by the server in XML which specifies the format of the messages that the Web Service can understand [14].

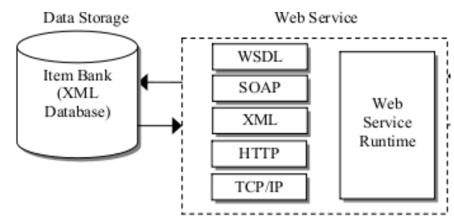

Figure 2. System Architecture

Figure 2 shows the overall architecture by applying a SOA concept. The architecture is separated into the two parts: data storage and Web Service. In data storage, the XML database stores items. An item bank data structure is described in an XML document. The item structure is a subject to a XML schema developed according some defined standards. This database structure is designed by using the basic data structure of the general testing parameters, such as question, multiple choice, and answer.

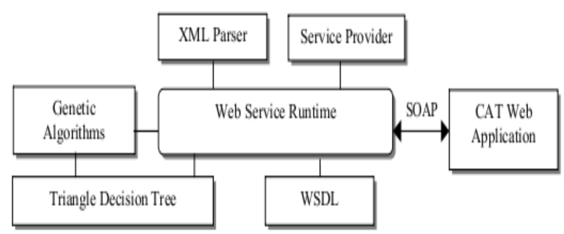

Figure 3. Web Service Architecture

In Web Service design, a set of standards for describing, publishing, discovering, and binding application interfaces is defined using WSDL, SOAP and UDDI. WSDL is a format for describing a Web Service interface [15]. It describes services and a way they should be bound to specific network addresses. The descriptions of services and messages are generally expressed in XML. SOAP provides the envelope for sending messages via the Internet (Figure 2).

UDDI defines a set of services supporting the description and discovery of service providers. The design of Web Service consists of four main modules – GAs, TDT, SOAP, and WSDL – as shown in Figure 3. The Genetic Algorithms (GA) are the item classification method to generate an optimal decision tree.

As shown in Figure 4, *LearnServe* is divided into two parts: a client software and Web services provided by several suppliers. The *LearnServe* client is the access point for users who can use the learning services. These services are implemented on distributed servers and in particular include authoring, content, exercise, tracking, a discovery services as well as communication services such as email and message boards. The exercise services are provided by our *xLx system* [21], that was enhanced to offer its functionality as a Web service and can thus already be used in external systems.

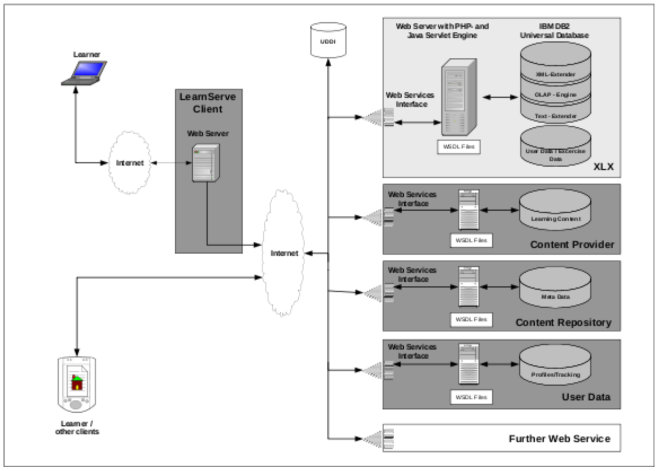

Figure 4. High Level Web Services Architecture

3. RESULTS

3.1 Web-Services-Based LCMS

In this section we present a service-oriented architecture of an LCMS based on Web Services, which is extended to support software agents. We assume that all the LCMS functionality including, the learning contents, are implemented as Web Services.

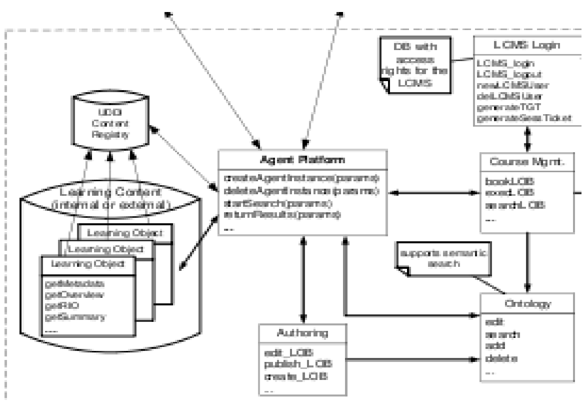

Figure 5. Learning Content Management System

In this Figure 5, Web Services are depicted as rectangles containing a name as well as the most important operations. Furthermore, the architecture is designed in such a way that a learner only needs an Internet browser to use the LCMS. We explain the architecture shown as well as most of the operations listed in this figure in more detail in the following subsections.

3.2 LCMS Architecture

The architecture of our LCMS is aimed at coordinating all learning-related activities and the management of learning materials. The PC computer of a learner interacts directly with the LCMS during a learning session [9]. All Web Services of the LCMS should also be accessible via Web pages, so that the learner only needs a Web browser to utilize the LCMS.

In a first step the learner has to authenticate in the LCMS, which is done by a LCMS login service. This service draws on a database with access rights and uses an authentication mechanism. When the learner is logged in and authenticated, he or she can access a Web page for course management, the functionality of which is implemented as a course management service. The learner can look for suitable courses with a *searchLOB* operation, which searches for learning objects with the help of the agent platform, also implemented as a Web Service. The *bookLOB* operation is called to enroll for a course that was found by an agent. A class can be attended by calling the *execLOB* operation on the remote LO.

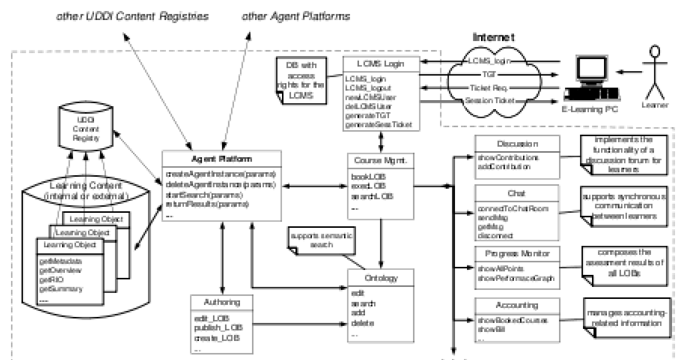

Figure 6. Components of a distributed, Web-Services-Based LCMS

The LCMS may also comprise of other services. In discussion boards or chat rooms learners can interact with instructors or other learners and ask questions. A progress monitor composes the assessment results from all lessons into a general overview; this can also be used to create certificates. An accounting service manages all processes which are related to financial aspects. It shows, for example, all booked courses or the bill that has to be paid by an individual. Finally, it should be mentioned that the functionality of the LCMS can be extended by other Web Services, which can either be provided internally (i.e. in a local network) or externally from other suppliers over the Web.

3.3 Learning Objects (LOs) and Metadata

Learning objects (i.e. educational content) is provided in form of Web Services. In general, LOs may have any desired structure. From a conceptual point of view, however, LOs typically contain parts like Metadata, Overview, Summary and one or more Reusable Information Objects (RIOs), which in turn contains a content part, a practice items part, and an assessment items part for the generation of online tests [3]. In addition to traditional LOs, Web-service-LOs come with operations to extract, manipulate, and present contents.

Therefore, the format in which content is stored inside a Web Service is less important. Operations might even be invoked to randomly generate online-tests from the assessment items.

The agent platform is an environment in which software agents can be executed to retrieve LOs, and which is wrapped by a Web Service. Agents are intended to assist learners with a focused search for LOs, according to the specifications they make. The search parameters of an agent, the start of a search, or the access to the list of retrieved LOs, for example, can be controlled by invoking appropriate Web Service operations which extract metadata from LOs.

Learning objects can be stored in a relational or an object-relational database and are typically a collection of attributes, some of which are mandatory, and some of which are optional. In a similar way, other information relevant to a learning system (e.g. learner personal data, learner profiles, course maps, LO sequencing or presentation information, general user data, etc.) can be mapped to common database structures. This makes interoperability feasible; moreover, it allows for a process support inside an e-Learning system that can interact with the underlying database appropriately [10, 17].

Metadata in reality is data describing data and it can be used to describe any digital resource. There are various metadata elements which describe different aspects of digital resources. For example the IEEE LOM specification [17] has metadata elements which enable the description of digital resources. Such descriptions include amongst other things the purpose of the resource, technical information about the resource, and the ownership of the resource. The ownership of the resource is described by digital rights metadata tags [18].

4. CONCLUSION AND DISCUSSION

Many custom e-Learning platforms can only present their material inside the platform, while on the other hand Internet-based Web Services are becoming ubiquitous, both at a professional and at a personal level. A service-oriented e-Learning system results from a perception of the various tasks and activities that are contained in such a system as processes or as work-flows; using appropriate encodings of objects and tasks in UDDI and WSDL forms and documents enable broad exchanges, flexible compositions, and highly customized adaptations possible.

We also identified the essential services in the functioning of a typical e-Learning based. These services (with real time Web Services technology) would provide a common interface between various components leading to platform independence and interoperability between learning systems.

REFERENCES

[1] Global Internet Statistic, http://www.glreach.com/globalstats/index.php3

[2] Westerkamp, P., *e-Learning as a Web Service*. University of Munster, 2005

[3] Husemann, B., J. Lechtenborger, G. Vossen, P. Westerkamp.“XLX - A Platform for Graduate-Level Exercises”, in *Proc. Int. Conf. on Computers in Education*, Auckland, New Zealand, December 2002, pp. 1262-1266.

[4] IEEE Standards Department (2002). *Draft Standard for Learning Object Metadata.* IEEE Publication P1484.12.1/D6.4, March 2002.

[5] Advanced Distributed Learning Initiative (2001). *Advanced Distributed Learning Sharable Content Object Reference Model*, the SCORM Overview, Version 1.2

[6] Casati, F., U. Dayal, eds. (2002). Special Issue on Web Services. *IEEE Bulletin of the Technical Committee on Data Engineering*, 25 (4), December 2002.

[7] Pankratius, V., Sandel, O., and Stucky, W. “Retrieving Content with Agents in Web Service E-Learning Systems”, *Proc. of the Symposium on Professional Practice in AI*, IFIP WG12.5, First IFIP Conference on Artificial Intelligence Applications and Innovations (AIAI), Toulouse, France, August 2004.

[8] Harrer A., “Software Engineering Methods for the Reuse of Existing Components in Educational Systems”, *Proc. International Symposium on Artificial Intelligence and Applications*, 2002, 430–435.

[9] Pankratius, V. *E-Learning Grids: Exploitation of Grid Computing in Electronic Learning*, Masters Thesis, Dept. of Information Systems, University of Muenster, Germany, 2003.

[10] Vossen, G., P. Jaeschke (2002). Towards a Uniform and Flexible Data Model for Learning Objects. In *Proc. 30th Annual Conf. of the Int. Bus. School Computing Assoc.* (IBSCA), Savannah, Georgia, July 2002, pp. 99-129.

[11] Booth, D., Haas, H., et. al (eds.), “Web Services Architecture”, W3C Working Draft 11 February 2004. http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

[12] Aaron S. (2003), *Understanding SOAP*, MSDN Library Articles, Microsoft Corporation. http://msdn.microsoft.com/webservices/default.asp

[13] Roger W. (2001), *XML Web Services Basics*, MSDN Library Articles, Microsoft Corporation http://msdn.microsoft.com/webservices/understanding/webservicebasics/default.asp

[14] Hussain, N. and Khan, M. K.,, *Enhancing E-Learning through Web Service and Intelligent Agents*, 2005.

[15] Phankokkruad, K. Woraratpanya, “Web Service Architecture for Computer-Adaptive Testing on e-Learning”, *Proceedings of World Academy of Science, Engineering and Technology*, Volume 36, December 2008, ISSN 2070-3740.

[16] Learning Management Systems and Learning Content Management Systems De-mystified, http://www.brandonhall.com/public/resources/lms lcms/

[17] LOM – Draft Standard for Learning Object Metadata, IEEE 1484.12.1-2002, 15 July 2002

[18] Mohan, P. and C. Brooks, “Learning Objects on the Semantic Web”, *Proceedings of International Conference on Advanced Learning Technologies*, Athens, Greece, 2003

[19] Manouselis, N. and D.Sampson, “Agent-based e-Learning Course Discovery and Recommendation: Matching Learner Characteristics with Content Attributes”,

International Journal of Computers and Applications (IJCA), Special Issue on Intelligence and Technology in Educational Applications, Volume 25, Issue 1, January 2003
[20] Mataric, M., “Interaction and Intelligent Behavior”, MIT, Cambridge, USA, 1994.
[21] Wasterkamp, Peter. “e-Learning as a Web Services”. University of Munster, Germany. 2007.

Ahmad Luthfi, S.Kom., M.Kom., born in Curup (Bengkulu) March 11th 1976, holding a Master Degree in Computer Science from Gadjahmada University in 2005 and also a Bachelor Degree in Information System from Bina Darma University in 1999. He is currently head of Information Technology Study Program for Bachelor and Master Degree program in Bina Darma University. He is also focus on several Field of research studies such as Semantic Web, Ontology concepts, and Intelligent Learning Management System, which published in several journal and proseding national and also international level including KNSI, SNEIE, SNATI, ICTS, ICA, and ICTS.