Research Article *ISSN: 2319-507X*
Sachi Khodke, IJPRET, 2013; Volume 1(8): 877-883 *IJPRET*

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

LEARNING OBJECTS MANAGEMENT SYSTEM FOR E-LEARNING

SHACHI KHODKE[1], PROF. A. V. DEORANKAR[2], DR. P. N. CHATUR[3]

1. M. Tech 2nd year, Department of Computer Science and Engineering, Government College of Engineering, Amravati (Maharashtra), India.
2. Asst. Professor, Department of Computer Science and Engineering, Government College of Engineering, Amravati (Maharashtra), India.
3. Head of Department, Department of Computer Science and Engineering, Government College of Engineering, Amravati (Maharashtra), India.

Accepted Date:

27/02/2013

Publish Date:

01/04/2013

Keywords

Learning Object,
Learning Object Metadata,
Loms,
Xml Binding Elearning,
Elearning Standards, Xml-Native Database.

Corresponding Author

Ms. Shachi Khodke

Abstract

E-learning is fast expanding into the role of independent educational entities that aspire not only to complementing traditional classroom teaching, but also allow open learning for distance and continued education. Many studies have defined a concept of "learning object" to address the issues and needs. The paper proposes a comprehensive learning object (LO) model, along with the associated system model, that will allow complete and flexible integration of content into the modern learning profile. The process will be user-centric (both for knowledge developers and learners) as well as metadata-centric. It is scalable and interoperable with legacy and existing content databases and display systems. This paper presents a LOMS (Learning Objects Management System) architecture based on the B/S architecture for the support to the reusability and interoperability of various learning objects. A dynamic approach in the use of the metadata and how this concept can dramatically improve the management of Learning Objects is illustrated. The work of user input during the metadata generation also has considerably been reduced by the template together with information gathered from the system. Our basic design ideas are to develop a user-friendly learning environment utilizing a simple but effective architecture, taking advantage of XML and making efficient use of available open source offering. Hence, we adopted native XML database technology instead of traditional RDBMS (Relational Database Management Systems), and Apache's open source offering, Xindice as tools to develop the LOMS. This paper covers how the LO model is integrated into the core of the learning content development, discovery, and delivery process. This results in terms of ease-of-use, flow-control, and feasibility of the model.

INTRODUCTION

As web-based Learning being widely used in educating and training, the amount of digital learning resources begin to accumulate. Reusability and interoperability of digital learning resources become one of the major challenges. The proposal to structure learning contents according to the model of the Learning Object (LO) has evolved out of this context. The characteristics of durability, interoperability and reusability and the related standardization process needed to achieve these goals, have played an important role in the diffusion of the Learning object model. Thus the cost of courseware development can be lowered through reusing of learning objects.

A learning object, as defined by the Institute of Electrical and Electronics Engineers (IEEE) Learning Technology Standards Committee, "is any entity, digital or non-digital that can be used, re-used or referenced during technology supported learning" (IEEE, 2002). This definition is intended to include any form of instructional material that can be used during "technology supported learning."

Learning objects should be stored in a database and organized by a management system for easy retrieval and search. For this purpose a small amount of additional information must be provided to describe each learning object. These information are called metadata. In common sense metadata is a label placed on any object, similar to the labels on cans of vegetables and library cards. In e-learning context metadata is the term to describe a package of information about an electronic resource, providing information such as author, title, subject matter, copyright information, and location.

The core of a learning object system is a central repository (database) containing hundreds to thousands of individual learning objects. The information stored in these repositories would be accessed by an array of applications and end users, including learners and the instructional designers. Attached to each learning object in the database is metadata. The metadata includes subject-specific information by conforming to the open standards.

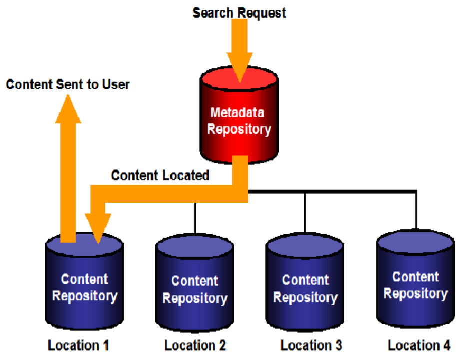


Figure1. Learning Object Repository Centralized Metadata

The basic LMS performs two major functions: it provides instructional designers with a means of locating learning objects, and it assembles them into standard compliant learning units. Although many types of LMSs are available, the enhanced LMS will contain four essential features: an authoring application similar to the computer assisted software environment , a collection of learning objects described above, a means of sending the completed course to a delivery system (called a delivery interface), and administration tools. The LMS systems have been enhanced to include additional features, such as intelligent tutoring or adaptive learning components for learners.

In particular the standardization process has focused on two main aspects: the description of LOs to provide efficient search and retrieve mechanisms; the model of LO to guarantee the reusability and interoperability of educational resources in the hundreds of learning platforms and learning systems available worldwide.

Standard organizations such as ACM (Association for Computing Machinery), DLF (Digital Library Federation) and IEEE (Institute of Electrical and Electronics Engineers), IMS (Global Learning Consortium, Inc.) and ADL (the Advance Distributed Learning Initiative, sponsored by the OSD (Office of the Secretary of Defense of USA)) have published several well known specifications for e-learning either independently or co-operatively

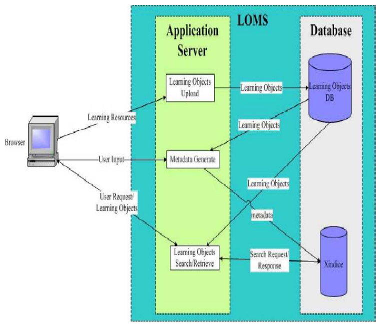
XML-based specifications supporting learning technologies have been developed by the SCORM initiative and distributed through the ADL (Advanced Distributed Learning) initiative's network [5]. There are two key mechanisms in SCORM, one is the content aggregation model and the other is run-time environment. SCORM provides specifications for building a unified content model and a run-time environment, enable courseware sharing and multi-platform (cross-platform) learning.

Our basic design goal is to develop a general-purpose, user-friendly digital reference room with a simple but effective architecture. So we adopted native XML database technology and open source offering Xindice after thorough investigation. This paper describes the architecture and implementation for Learning object management system using native xml database Xindice.[7]

SYSTEM ARCHITECTURE

The architecture is based on the B/S (Browser/Server) architecture. Contrast to other Client/Server architectures, B/S is convenient for user, for only a browser is required. In addition, the change of the system doesn't affect the user either. The LOMS consists of two sub-systems: application server and the database servers. The system is distributed on three layers: the presentation layer, the function/control layer and the data layer. The browser on the presentation layer is the entry to the system. Application server focuses on the implementing of the functions of the system, such as the metadata's generating. It is the function/control layer. All the data, including the metadata and the learning objects themselves are stored in different databases, which form the data layer. The metadata is stored in Xindice and the learning objects in file systems. Three components reside in the application server: Learning Objects Upload Module allows a user upload his/her learning resources to the system and stores them into the learning objects database. Metadata Generate Module generates the metadata for a learning object. Learning Objects Search/Retrieve Module accepts the user's request, finds the appropriate objects and then returns them.[5]

Fig 2.The overview of the LOMS



XML Binding for the Metadata

XML binding for a metadata defines the structure of the XML document, the elements appeared, including the elements' names, attributes and contents and the number of a specific element and order of some elements. For instance, the name of the root element of the XML document for LOM must be "lom". It also defines the value's ranges for some attributes and more for the organization of the XML document.

The Management of Metadata

The XML presentation of metadata should be stored for the management of learning objects. Most work of the LOMS is to generate metadata and store them for the adding of new learning objects, to search the appropriate learning object through searching the metadata. A sub-system called repository serves this purpose. Xindice is used as the repository in our system, which is a so-called XML-native database. XML-native means that when the XML documents are stored into the database the conversion of the structure of the document into the table-structure used in Relational database is not required and the retrieval of the documents don't need the reverse conversion. Xindice uses the XPath as the query language. It is quite suitable where just the XML binding is used for metadata.

Steps to manage the XML contents

Fig.3 shows a procedure for managing the XML contents. Considering the case when an instructor or an administrator wants to update/modify an available content in XML, for instance, logic puzzles, he or she can do in following steps:

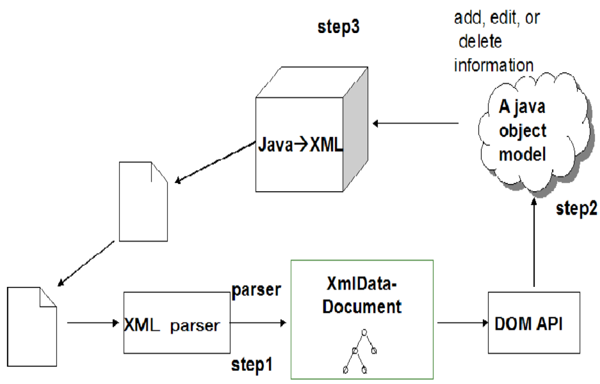


Figure 3. Steps to manage the XML contents

Convert the XML document to a Java object model using an XML parser and DOM

(Document Object Model). Using the DOM API, we will create our own a Java object model. An XML LRDB (logic reference database) document, for example, is converted to a Java object model for logic reference database. The DOM API is used to read in information from an XML document. DOM can also be used to change this information, and can be used to generate an XML document. There is an easier way of getting around using DOM for modifying and saving the XML data; working with the Java object model. The contents of the logic reference database are stored in the object model and this model is used to add, edit or delete information. Generate an XML document from a Java object model. The Java object model for the logic reference database is saved as an XML document [8].

CONCLUSION

This paper will serve as an e-learning portal in which new objects will be allowed to create and old objects to be reused. For the reusability and interoperability of learning objects, the development of a LOMS should be based on the concept of metadata and compatible to various metadata standards. The XML technology makes the multi-standards support feasible.

REFERENCES

1. Albert D. Ritzhaupt ," Learning Object Systems and Strategy: A Description and Discussion", Interdisciplinary Journal of E-Learning and Learning Objects Volume 6, 2010.

2. Y. Vovides , S. Sanchez-Alonso," The use of e-learning course management systems to support learning strategies and to improve self-regulated learning" ,Science Direct Educational Research Review 2 ,64–74 ,2007.

3. Alex Koohang, "Creating Learning Objects in Collaborative E-learning Settings", Issues in Information Systems, Volume V, No 2,2004.

4. Wang Xuan, Zheng Li, and Yang Fang," An Implementation of Learning Objects Management System", Springer-Verlag Berlin Heidelberg 2004

5. Olivier Motelet, Nelson A. Baloian ,José A. Pino," Hybrid System for Generating

Learning Object Metadata", JOURNAL OF COMPUTERS, VOL. 2, NO. 3, MAY 2007

6. Apache Software Foundation, The Apache Xindice Project. http://xml.apache.org/xindice/