

Learning Design Implementation in SCORM E-Learning Environment

Elena Shoikova*, Malinka Ivanova**

*Associate Professor, Technical University of Sofia, Department of Electronics, Bl.1, Rm.1360,
8 "Kliment Ohridski", Sofia-1000, Bulgaria
Phone: +(359 2) 962 40 42; 965 21 40; Fax: +(359 2) 962 40 49; Email: shoikova@tu-sofia.bg.

**PhD Student, Email: m_ivanova@tu-sofia.bg

Abstract: This paper describes a project started at the Technical University of Sofia Research & Development Laboratory "E-Learning Technologies" and examines implementation of Learning Design in Shareable Content Object Reference Model (SCORM), open source based e-learning environment ATutor. For increasing the quality of the e-learning courses and to improve their efficiency, effectiveness and attractiveness in higher education Learning Design Module (LDM) has been originally designed. The design methodology includes two aspects: (1) conceptual modeling based on Educational Modeling Language (EML) and IMS Learning Design (LD) Specification and (2) software development and integration comprises Unified Modeling Language (UML), Unified Development Process (UDP), an object-oriented approach (forward and reverse engineering), Use Case Analysis, as well as a client-server architecture solution. The implementation of the LDM is done by applying the following technologies: object-oriented PHP language, HTML, JavaScript, XML and relation Database (DB).

Keywords: Learning Design Module, Implementation, EML, IMS LD Specification, SCORM E-Learning Environment

Introduction

To be successful, e-learning must offer effective and attractive courses and programmes to learners, while at the same time providing a pleasant and effective work environment for authors/instructors/learner designers who have the task of developing course materials, planning the learning processes, providing tutoring, and assessing performance. Learning Design as systematic process of translating general principles of learning and instruction into plans for instructional materials provides the product of the design process: a description of a method enabling learners to attain certain learning objectives by performing certain learning activities in a certain order in the context of a certain learning environment [1]. The using of LD requires a designer to think about pedagogy - to consider which objectives a learning process is designed to achieve and which prerequisites are involved, to identify the activities learners and staff should undertake and which tools should be used in the process. In engineering education is important for authors of e-learning content to follow pedagogical framework of LD which guides them to reflect on learning activities and the achievement of learning objectives. LD

can be used by designers working in different teaching areas. But it is a unique assistant that supports authors-engineers to model effective educational process, personalized and flexible learning, to prepare competency profiles and assessment models following formats and rules for the design of a didactic approach. The present paper describes a project started at the Technical University of Sofia R&D Laboratory "E-Learning Technologies" and aimed implementation of a LDM in a SCORM-compliant, open source based e-learning environment ATutor. LDM is implemented using object-oriented PHP language, HTML and JavaScript. The module is web-based with client/server architecture and as part of e-learning environment works with MySQL Database.

Needs Analysis and Requirements Definition

The LDM requirement specification is created after analyzing the existing LD Tools. The made analysis of LD Tools points out the importance of working with exact tool in the process of creating the learning scenarios. This analysis summarizes the best practice in LD Tools and gives possibility to defining the requirements to LDM in the situation of

engineering education. Some popular at this moment LD Tools used by authors of Units of Learning (UOLs) at design time are grouped as Tree-based Editors: RELOAD [2], aLFanet LD Editor [3], Komposer [4] and High level Editors: MOT [5], LAMS [6].

Now days the main strategy in designing the support applications is focused on these that hide complexity from the user with interoperability between different stages in learning design process. Therefore the LDM is designed as a High Level Specific Purpose Tool, and Distance from IMS LD Specification. The LDM's requirement specification defined at different level of details in the context of needs is the following:

- Functional: template-based LD Editor with user-friendly GUI, that guides authors in designing process of UOLs, with constrained pedagogy, distant from Specification, allows the sequencing of the activities and binding with resources, points to services;
- Operational: web-based LD Editor with client/server architecture, with end user installation and maintenance free, easy integration in e-learning environment, standard protocols and technologies used, based on EML and LD Specification, SCORM e-learning content used;
- Management/organizational: services of e-learning environment are used.
- Technological: for easy integration in the e-learning environment ATutor, LDM has to be developed in object-oriented PHP language with JavaScript, with MySQL Database communications; output is LD XML file, http/ftp protocols used.
- Standards used: IMS LD, EML, SCORM e-learning content.

Methodology of the Design and Development Process

The design methodology includes two aspects: (1) conceptual modeling based on pedagogy concepts of Educational Modeling Language (EML) and IMS Learning Design Specification and (2) software development and integration comprises an object-oriented approach (forward and reverse engineering), Unified Development Process (UDP), Use Case Analysis, Unified Modeling Language (UML) as well as a client-server architecture solution.

Conceptual Modeling

The LDM is constructed upon pedagogy concepts of EML and technical requirements of IMS LD Specification. EML is a notational system intended to provide a way of describing teaching and learning interactions at a level of abstraction above the specific instance of the context in which it was created. At the heart of EML is the idea that *Learners* perform *Activities* in an *Environment* with *Resources* and *Services* in order to achieve *Learning objectives* according to defined *Method*. The EML is basis of the IMS LD specification [7]. The aim of this specification is to provide a digital format for encoding, transporting and playing learning designs. The core components of LD are based around the conceptual entity of a *UoL*. This is the smallest unit that satisfies one or more learning objectives. The specification implements the theory of LD into a machine-readable technical standard. It is a highly complex specification, involving three levels - A, B, C of implementation.

Software Implementation and Integration

The UDP is used as the main development strategy, relies on the incremental process model [8]. In the project, UDP is chosen because: it is planned and managed; it is predictable; it accommodates changes to requirements with less disruption; it is based on evolving executable prototypes; it is risk driven. The UDP consists of four phases: Inception, Elaboration, Construction, and Transition. The main stages, which are iterated through each increment, are: iteration planning, requirements capturing, analysis&design, implementation, test and prepare release.

Use Case Analysis

The Use Case analysis is the most efficient method for requirements capturing and software system functional specification. It is used to define the features of software application. The Use Case analysis also helps to layout the actors or users and their role in running the system.

The main actors that interact with the LDM are as follows: Author-human actor and non-human actors-Learning Management System (LMS) and SCORM e-learning Content Editor.

Conceptual strategy for building UOL is depicted in Figure 1 by six Use Cases: *Describe General Information for UOL*, *Define Roles*, *Describe Activities and Activity Structures*, *Describe Environments*, *Define Method* and *Use Performance Support Tool* that are detailed examined below. The General Information for UOL is used to form instructional content and metadata about UOL. The descriptive element that metadata contains may be used for searching in digital repository. Prerequisites express the necessary requirements for starting a UOL, or an activity within a UOL. One or more Learning Objectives can be defined for each UOL and for each activity within the UOL. The Roles are used to specify the actors that play different roles in a UOL. To invoke a learning process one or more activities are assigned to the certain Role. Different sequences can be defined within Activity Structures. The Learning Environments specify which objects and services are needed for the entire set of activities. Method is used to describe how activities to be performed in a particular environment by different Roles. The Author uses Performance Support Tool to help and guide himself in time of working with LDM. SCORM e-learning Content Editor is powerful tool in process of SCORM content creation. LMS has to give and support LDM with services.

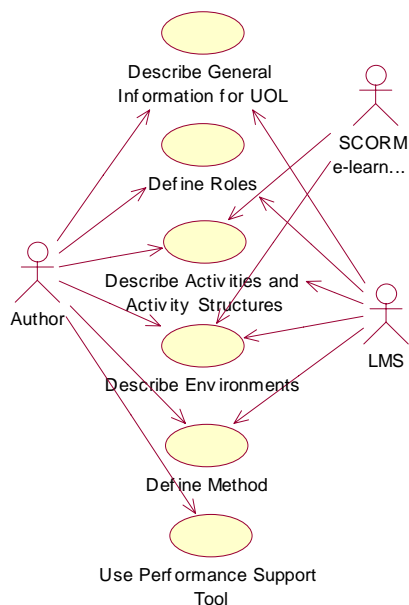


Figure 1 Use Case Model of LDM

Architecture

The LDM is integrated part of SCORM-compliant e-learning environment as is

depicted in Figure 2. E-learning environment includes LMS, SCORM-compliant Learning Content Management System (LCMS), Digital Repositories (MySQL DB, SCORM content, XML LD files), and LD Run Time Environment (RTE) and SCORM RTE.

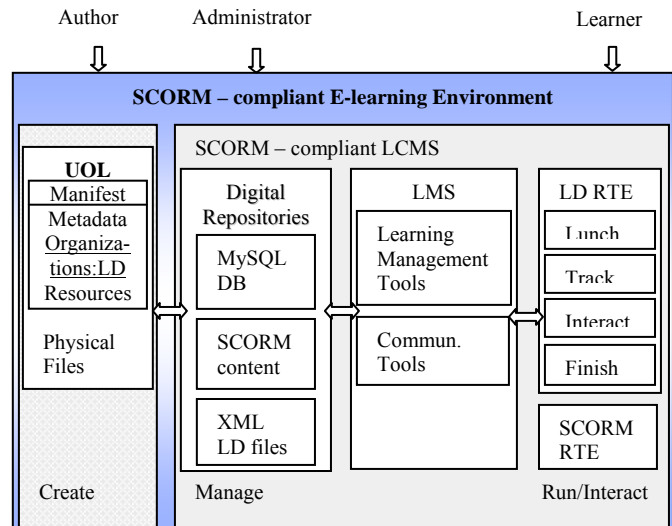


Figure 2 Architecture of SCORM-compliant E-learning Environment

Implementation

Implementation is concerned with building, testing, and deploying the new LD software application. Specifically, this activity includes detailed design of each LDM component (user interface and programmes), construction, and integration of the code and user interface of LDM in SCORM-compliant e-learning environment. Implementation is largely dependent on the architectural choices middleware and on used web e-learning environment. For realization of LDM has been chosen object-oriented PHP language and JavaScript in order it to be modular part of existing SCORM-compliant e-learning environment. It is developed as a distributed application with client/server architecture approach. The LDM prototype is presented in Figure 3. The software module is placed on server side. The Author works only with web browser and populates the Form' fields in order to design UOL on client side. He uses existing DB data and creates new one. When the Author creates UOL he needs some information for learners and staff which is placed on MySQL DB. Then he makes request to the web server that extracts the needed code and submits it to PHP Engine and PHP application, which process this code and

executes the request to DB. MySQL Server receives request, processes it and sends back result. Author temporary saves new data in DB tables, in order to use them in different Form' fields in the designing process of UOL. For this purpose have been designed some new DB tables which to temporary store the objective, the specific objectives, prerequisites, etc. The first flow reflects on data movements between

web browser and MySQL DB. Some LD components as activities and environments need to be bundled with resources and the second flow presents the inserting of SCORM content in LD. In order to illustrate the realization of LDM and its integration with ATutor several examples are presented in Figure 4.

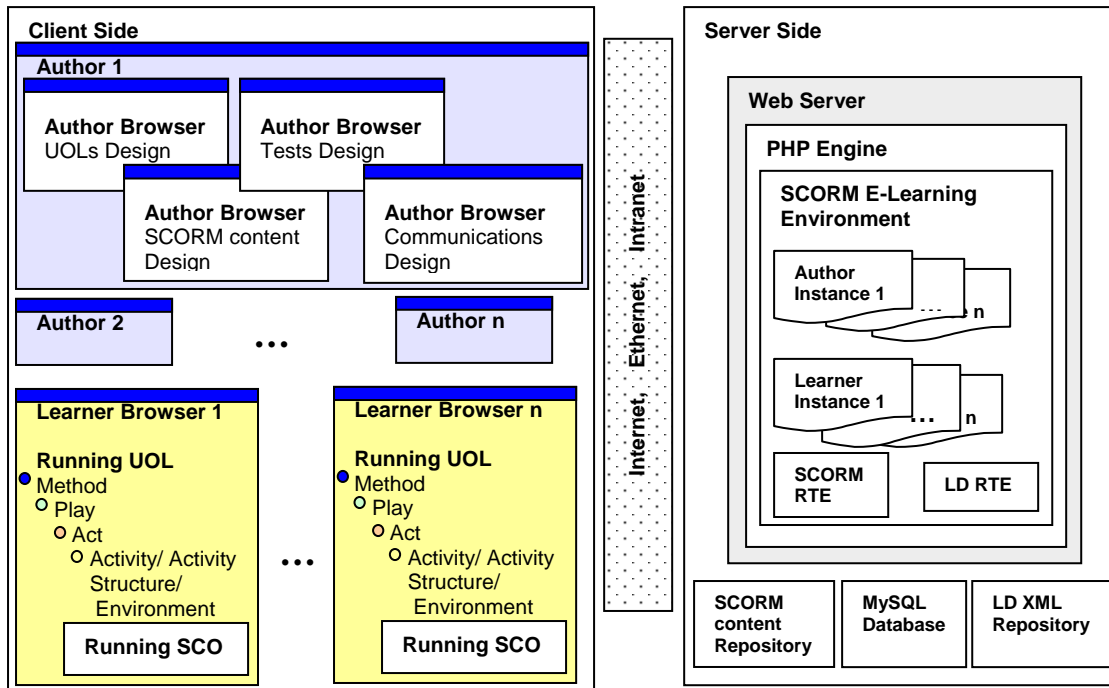


Figure 3 Prototype Architecture

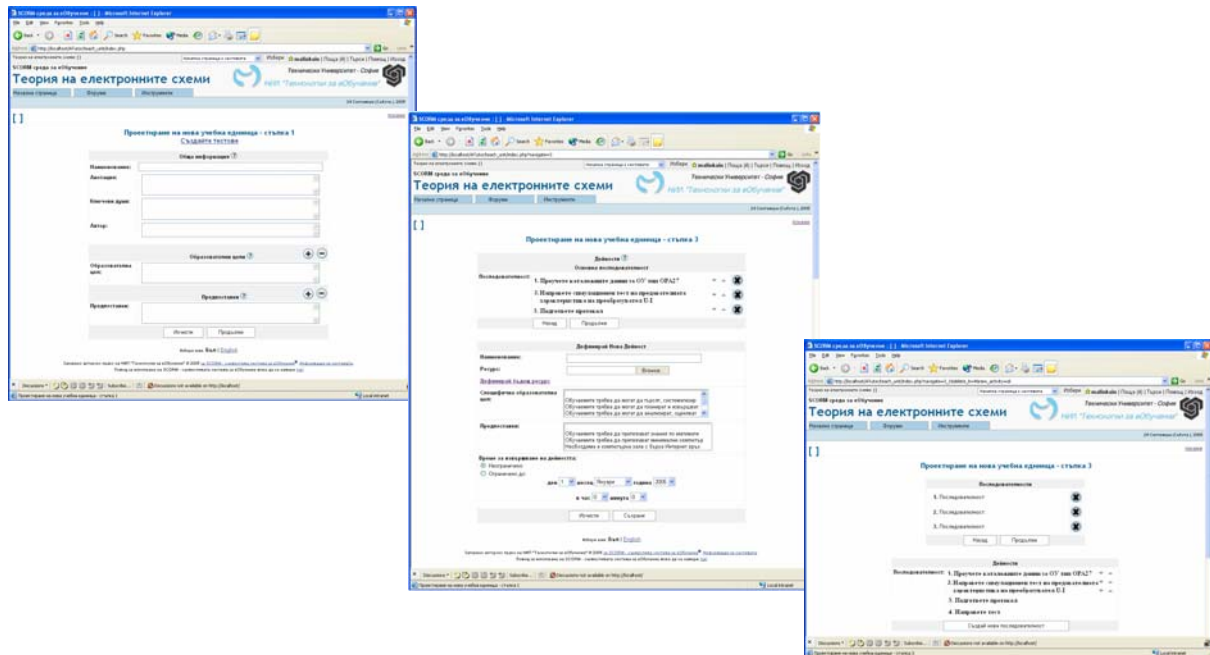


Figure 4 LDM Realization

Pilot Test

The pilot test started in December 2005 with experts from Regional Educational Departments and with students MSc Degree on E-Learning from Sofia University. During the test information collection procedures related to the following main evaluation questions are being performed and they pertain to:

- the pedagogic effectiveness of the LDM
- the usability of the LDM as integrated part of ATutor
- the quality and reusability of learning designs.

Conclusion

A significant step has been taken for implementing new LDM as a part of SCORM-compliant e-learning environment for the purposes of higher engineering education and training in Technical University of Sofia. In paper has been defined the requirements to LDM. The main picture of LDM' features and software system functional specification is defined through Use Case analysis. For integrating the module in existing e-learning environment is proposed Architecture of SCORM-compliant E-learning Environment. The developed LDM Prototype has a client/server architecture. GUI is constructed and software application is coded in PHP language, HTML and JavaScript.

References

- [1] Koper, R., Modeling units of study from a pedagogical perspective: The pedagogical meta-model, 2001
- [2]RELOAD Project, 2004, <http://www.reload.ac.uk>
- [3]Carrion J., C. Fuentes, General System Architecture and Design. AL-Fanet, SAGE, Madrid, 2004, <http://hdi.handle.net/1820/258>
- [4] Koper R., C. Tattersall, Learning Design, Springer, 2005
- [5]Paquette G., Educational Modelling Languages from an Instructional Engineering Perspective, 2002
- [6] Dalziel, J. R., Implementing Learning Design: The Learning Activity Management System (LAMS), Proceedings of the 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education. Adelaide, 7-10 December 2003
- [7] IMS Global Learning Consortium, Inc., IMS Learning Design Information Model, XML Binding Document, Best Practice Guide, 2003
- [8] Kratchen F., Rational Unified Process, Addison-Wesley, 2002
- [9] CETIS, Learning Design inspiration, 2003, <http://www.cetis.ac.uk/content2/20031105152011>
- [10] Wilson S., Europe focuses on EML, CETIS staff, 2002
- [11] Advanced Distributed Learning, SCORM Overview, 1.3.2, July 22, 2004