

A Tool for Managing Domain Metadata in a Web-based Intelligent Tutoring System

John Garofalakis^{1,2}, Ioannis Hatzilygeroudis¹, Panayiotis Kyriakou^{1,2}

1. Department of Computer Engineering & Informatics, University of Patras,
26500 Patras, Greece

2. Research Academic Computer Technology Institute, N. Kazantazaki str.,
26504 Patras, Greece

e-mail: {garofala, ihatz, kiriakou} @ceid.upatras.gr

Abstract

A basic unit of an Intelligent Tutoring System (ITS) is its domain knowledge. In our ITS, domain knowledge consists of learning objects, i.e. the teaching material, and a meta-description concerning the concepts related to the learning objects. Concepts consist the knowledge of a field and are organized through a concept network, represented in the meta-description. In this paper, we present a tool for creating and editing the concept network and corresponding meta-description. The user can organize concepts in a hierarchy, which can edit whenever necessary. Moreover, the user can add meta-descriptions, which are compatible with the IEEE LOM standard, through user friendly forms. Another feature of our tool is that the user can see the metadata, which have been saved in xml form. We used the tool to implement the domain knowledge for an ITS teaching "Evaluation Techniques for Computing Systems".

Keywords: IEEE LOM, XML, Intelligent Tutoring System, managing tool, concept, metadata, learning objects, domain knowledge

1. Introduction

In our fast-forward, technologically advancing society, the invasion of internet in all aspects of our everyday routine, is undeniable. Education could not be an exception. Web-based intelligent tutoring systems(ITS) help or even replace the traditional place/way/year of teaching. With the term "intelligent" we mean the systems that are adapted to the user, depending on his/her possibilities, his/her learning rhythm and preferences. An ITS consists of the following components: the domain knowledge, containing the structure of the domain and the educational content, the user modeling component, which records information concerning the user, the pedagogical model, which encompasses knowledge regarding the various pedagogical decisions and the supervisor unit [Prentzas et. Al. (2002)].

The teaching material must be structured in such a way that can be recognized by this kind of systems, in order to be adapted to the user's needs. Thus, teaching material is split in small pieces of information that correspond to concepts of the

learning object. These pieces of information, which are called ‘course units’, should be accompanied by a description, in order to be manageable. This description is called meta-description of the teaching material. Meta-description describes various characteristics of the teaching material according to a metadata description standard, in our case IEEE LOM (Learning Object Metadata) [Holzinger et. Al. (2001)] . On the other hand, a learning object (LO) is determined by a group of concepts which constitute it. Those concepts are related between them and with the teaching material. The teaching material, the concepts and the relations between them are reported as domain knowledge of an intelligent tutoring system.

In this paper, we deal with the representation of the domain knowledge of a specific course, namely "Evaluation Techniques for Computing Systems", as well as with the construction of a domain knowledge unit related to that course.

Concepts which constitute the teaching material, have bonds between them that show the pre-required concepts that should be comprehended before them and also some other relations between them. These bonds create a network of concepts. This is the network of concepts of the domain knowledge of field of the course "Evaluation Techniques of Computing Systems". The application that has been created allows the administrator (teacher/content engineer) to add, modify and remove teaching material from the system. Still, the administrator has the possibility of adding metadata in each concept that he/she imports in the system and of creating xml files, which he/she can browse and open.

The paper is organized as follows. In Section 2, basic aspects of the IEEE LOM standard are briefly presented. Section 3 deals with the domain knowledge of the specific domain, while Section 4 presents the created tool. In Section 5, related work is outlined and, finally, Section 6 concludes the paper.

2. Learning Object Metadata standard

The IEEE Learning Technology Standards Committee working group IEEE P1484.12 (Learning Object Metadata Working Group, Chair: Wayne Hodgins) has developed a standard conceptual model . The IEEE conceptual model for meta-data definitions is a hierarchy. At the top of this hierarchy is the so called "root" element. This root element contains many sub-elements. If a sub-element itself contains additional sub-elements it is called a "branch." Sub-elements that do not contain any sub-elements are called "leaves." This entire hierarchical model is called the "tree structure" of a document.

2.1. Overview of the Metadata Structure

Data elements describe a learning object and are grouped into categories such as the following:

- The **General** category groups the general information that describes the learning object as a whole.
- The **Lifecycle** category groups the features related to the history and current state of this learning object and those who have affected this learning object during its evolution.
- The **Meta-Metadata** category groups information about the metadata instance itself (rather than the learning object that the metadata instance describes).
- The **Technical** category groups the technical requirements and technical characteristics of the learning object.
- The **Educational** category groups the educational and pedagogic characteristics of the learning object.
- The **Rights** category groups the intellectual property rights and conditions of use for the learning object.
- The **Relation** category groups features that define the relationship between the learning object and other related learning objects.
- The **Annotation** category provides comments on the educational use of the learning object and provides information on when and by whom the comments were created.
- The **Classification** category describes this learning object in relation to a particular classification system.

Collectively, the above nine categories form the LOMv1.0 Base Schema. The Classification category may be used to provide certain types of extensions to the LOMv1.0 Base Schema, as any classification system can be referenced.

2.2 Data elements

Categories group data elements. The LOM data model is a hierarchy of data elements, including aggregate data elements and simple data elements (leaf nodes of the hierarchy). In the LOMv1.0 Base Schema, only leaf nodes have individual values defined through their associated value space and datatype. Aggregates in the LOMv1.0 Base Schema do not have individual values. Consequently, they have no value space or datatype. For each data element, the LOMv1.0 Base Schema defines:

- **name:** the name by which the data element is referenced;
- **explanation:** the definition of the data element;
- **size:** the number of values allowed;

- **order**: whether the order of the values is significant (only applicable for data elements with list values)
- **example**: an illustrative example.

For simple data elements, the LOMv1.0 Base Schema also defines:

- **value space**: the set of allowed values for the data element – typically in the form of a vocabulary or a reference to another standard ;
- **datatype**: indicates whether the values are LangString, DateTime, Duration, Vocabulary, CharacterString or Undefined.

Both the size and datatype information may include smallest permitted maximum values. Extensions to the LOMv1.0 Base Schema retain the value space and datatype of data elements from the LOMv1.0 Base Schema. Extensions do not define datatypes or value spaces for aggregate data elements in the LOMv1.0 Base Schema. The numbering schema of the data elements represents an aggregation hierarchy of data elements and their components. As an example, the aggregate data element 7.2:Relation.Resource has two components, 7.2.1:Relation.Resource.Identifier and 7.2.2:Relation.Resource.Description. The former data element is in turn an aggregate, as it contains components: 7.2.1.1: Relation.Resource.Identifier.Catalog and 7.2.1.2: Relation.Resource.Identifier.Entry.

All data elements are optional: this means that a conforming LOM instance may include values for any data element defined in LOMv1.0 Base Schema. As the LOMv1.0 Base Schema imposes an aggregation relationship, components can by definition only be present in a LOM instance as a component of the aggregate element to which they belong. As an example, 7.2.1:Relation.Resource.Identifier appears by definition as a component of 7.2:Relation.Resource. In that sense, the presence of the component implies automatically the presence of the aggregate element to which the component belongs.

3. Domain Knowledge

The teaching subject of our system is “Evaluation Techniques of Computing Systems”. The teaching material was split in knowledge concepts, which are elementary pieces of knowledge of the specific domain. Every concept has a number of general attributes such as name, level of difficulty, level of detail, lowest acceptable knowledge level [Prentzas et. Al. (2002)]. Furthermore, it can have links to other concepts. These links denote its prerequisite concepts. In this way, one or more *concept networks* are formed representing the pedagogical structure of the domain being taught.

The course units constitute the teaching material presented to the system users as Web pages. The user is required to know this concept’s prerequisite concepts in order to grasp the knowledge contained in the specific course unit. The course units present

theory, examples or exercises. In the Figure 1, part of the concept group of the knowledge domain is presented. The ‘scissors’ symbol is used to show where a hidden sub-graph exists.

In Figure 1, the general idea of a concept group is shown. You can see the subject, sections, sub-sections and concepts of the concept network.

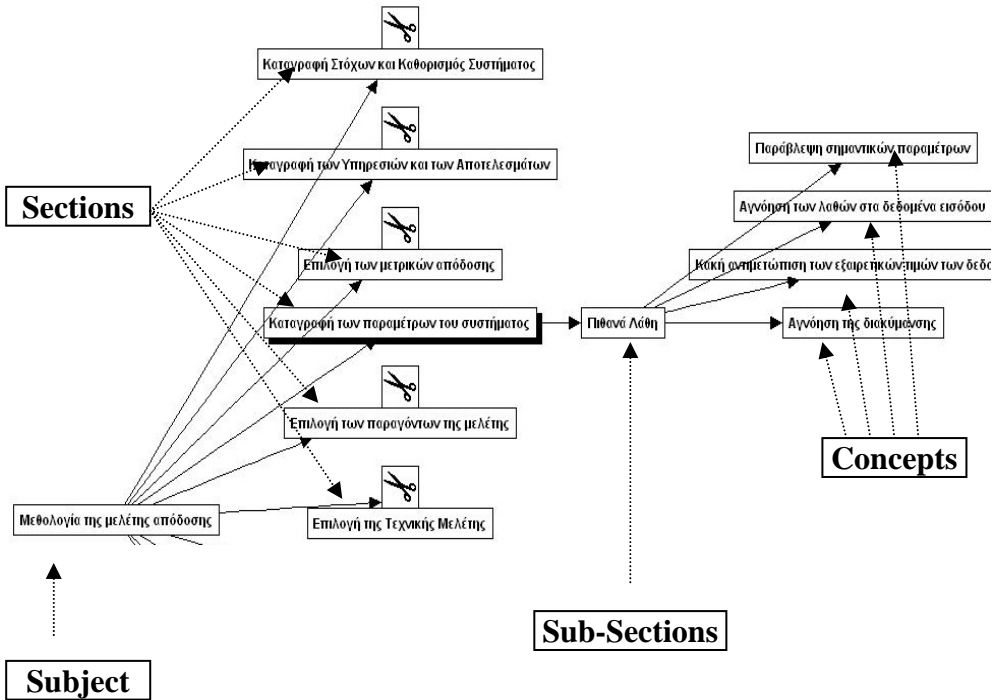


Figure 1. Part of the concept network

4. Creating a tool for managing metadata

After the construction of the concept network of the domain knowledge and the course units (html files) for every concept, the next step is to add the metadata for each concept based on the IEEE LOM standard [IEEE 1484.12.1 (2002)]. For this purpose, a database (in MySQL) is created, which is the storage of the concepts, their metadata and the relations between them.

The first step in the process of creating this application was to create the database according to the standard used. In Figure 3, the image shows the entities created and the relation between them. Through MySQL and PHP, we have created an application where the user (a teacher or a knowledge engineer) has the capability to introduce concepts in a hierarchy, which can be edited or expanded anytime. The process of adding metadata to the concepts is very functional and easy to use, because it is achieved through filling in fields in structured forms that have been created for this purpose (see for example Figure 2). Another feature of this application is the capability to create, store, view and edit the metadata in XML [IEEE P1484.12.3].

Metadata TECHNICAL addition to the concept

1 Technical

File Αναζήτηση

Requirement

OrComposite

Operating System none

Minimum Version

Maximum Version

OrComposite

Browser any

Minimum Version

Maximum Version

Installation remarks

Other platform requirements

Duration null

Εγγραφή

Figure 2. An example of the forms used

4.1. Saving the Concept Network

In Figure 3, we can see the root and the nodes in the first level of the saved hierarchy of the concept network. In each concept (node), the user can add a child-concept, metadata, view and edit the metadata in html forms. The user can also delete a concept from the concept tree, with the only restriction that it has no children. Figures 3 and 4 show how a new node (concept) can be added to the concept hierarchy.

4.2 XML

Following the XML link, we can view the concept tree with two links on each node. The first link (source), at the left, directs us to the actual teaching material, the course unit of the concept. The link at the right (XML), directs us to a dynamically built XML file which is related with each concept (see Figure 5). This file is created using the metadata that have been already saved in the database.

Figure 3. Subject and Sections

Figure 4. Node Addition

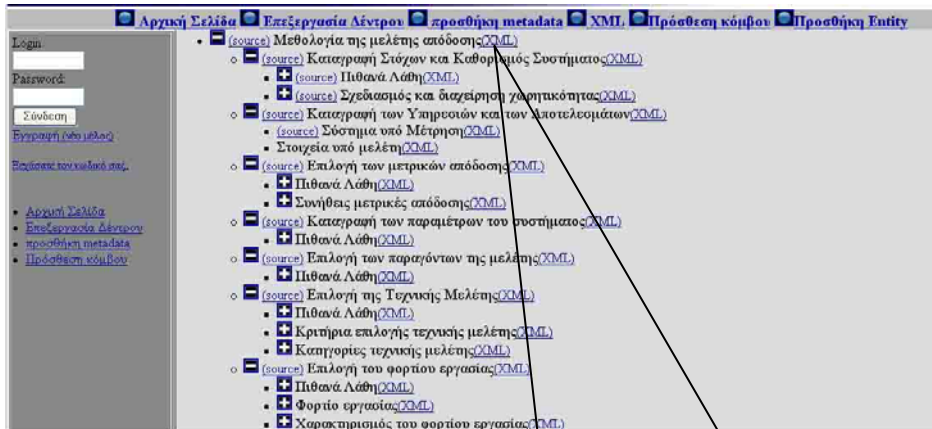


Figure 5. XML tree

```

<?xml version="1.0" encoding="iso-8859-7" ?>
<!DOCTYPE http://www.ieee.org/xml4.dtd [
  <!-- general -->
  <!-- entities -->
  <!-- character sets -->
  <!-- namespaces -->
  <!-- key words -->
  <!-- user agent -->
  <!-- aggregation -->
  <!-- version -->
  <!-- status -->
  <!-- technical implementation -->
  <!-- ENFORCE -->
  <!-- ENFORCE -->
  <!-- technical -->
  <!-- format -->
  <!-- location -->
  <!-- application -->
  <!-- processing -->
  <!-- namespace -->
]
<!-- general -->
<!-- entities -->
<!-- character sets -->
<!-- namespaces -->
<!-- key words -->
<!-- user agent -->
<!-- aggregation -->
<!-- version -->
<!-- status -->
<!-- technical implementation -->
<!-- ENFORCE -->
<!-- ENFORCE -->
<!-- technical -->
<!-- format -->
<!-- location -->
<!-- application -->
<!-- processing -->
<!-- namespace -->

```

Figure 6. XML file

5. Related Work

There are a number of tools, which are used for the management of metadata in learning objects. In spite of their many features, none of them was suitable for our task.

“LOM Editor” [LOM-Editor, (2001)] developed by Darmstadt University of Technology -Germany, and written in Java, is a standalone desktop application which

includes superior abilities for editing metadata, such as: Tabular presentation of metadata categories; vocabulary management; multiple-language values management; metadata template generation to avoid the necessity to repeatedly enter the same data in multiple fields. Some of the drawbacks of this metadata authoring tool include: No help or documentation; omission of specific details of LOM Model (e.g. multiple-language values support for metadata elements); some standards like vCard (vCard, 1996) and ISO 8601 Date/Time standard (ISO8601, 1998) are not supported in the representation of the metadata elements; storage of the metadata record is done in a database and there is no export option for the XML document.

“Reggie Metadata Editor” [Reggie (1998)] developed by Distributed Systems Technology Centre, a joint venture supported by the Australian Government’s Cooperative Research Centres Program, enables easy creation of various forms of metadata with the one flexible program providing means of choosing an already defined schema from a list or importing a new XML schema. If the resource to be described has a web page, Reggie Metadata Editor allows entering the URL of the page. It will extract the metadata tags from the page and attempt to add them to the most appropriate fields for the chosen schema. It also has support for multi-language values and on-line help. Some of the downsides of this editor include: No metadata template facility which would speed metadata record creation; in trying to achieve generality supporting multiple schemas, it loses some specific aspects of the LOM model such as standards and metadata vocabulary support; metadata elements are not organized in their corresponding categories or any other structure, thus creating a poor form layout.

“ALOHA II” [Aloha 2 (2004)] is a Java-based tool that is used for indexing, aggregating, sharing, multi-purposing, and re-purposing learning objects. It is created to meet the needs of indexers, educators and learners and includes versatile and powerful indexing tools and flexible searching of multiple educational object repositories. The software is based on the educational standards of IMS and SCORM. ALOHA II is not web-based and not based on the IEEE LOM standard.

“Curriculum Online Tagging Tool” [Curriculum Online Tagging Tool], which is designed to make the process of creating metadata and outputting as easy and intuitive as possible. A version of this tool is being developed to support UK LOM Core and aspects of CanCore. It enables creation and storage of details about the learning resources. It also allows adding those details to the Curriculum Online portal, so that teachers can find out about the learning resources. Once the details about a resource are added to the portal, the tagging tool can be used to update them at any time, or even remove them completely. This tool cannot create a network of resources, but just an unstructured repository of them.

“eRIB Metatagging Tool” [eRIB Metatagging Tool] is concordant with the CanCore Guidelines and permits users to create Metadata records for learning objects stored on a personal computer and thus constituting a personal repository (Repository-in-a-Box, RIB). The eRIB provides all the basic tools to add a new node.

It consists of a database (eXist) with a built-in IEEE LOM data structure and a set of tools to create, manage and find metadata records in the eduSource Network, a teachers' network. Again, eRib cannot create a network of concepts suitable for an ITS.

“Explor@-2” [Explor@-2] is a software environment for the delivery of courses or distance learning events on the Internet. It allows creating a virtual training centre that delivers a set of courses on the Internet according to a variety of models and using a LO repository facilitating information access, production, follow-up and coaching of learners as well as training management. It is fully compatible with the IEEE LOM, Cancore and Normetic. Explor@-2 is more an e-learning environment, not web-based, rather than a domain knowledge management tool.

“LomPad” [LomPad (2005)] is a free, open source tool to be used for educational purposes. It is actually a LO metadata editor that allows users to tag objects according to several major application profiles, namely LOM/IEEE, Normetic, CanCore and SCORM. The LomPad interface is bilingual, French and English. It seems not to be able to impose a structure on concepts related to the LOs.

6. Conclusion

In this paper, we present a tool for learning objects (teaching material) metadata management in the context of an ITS. The tool allows adding, viewing, editing and deleting the teaching material. In addition to this, the user has the capability to add metadata to each concept added to the system and create XML files which he/she can view and edit in the process.

The tool satisfies the following requirements set by our needs: Greek language support, web based, creating and editing a concept network, adding metadata and storage in a web-based database, compatible with IEEE LOMv1.0, uploading files, creation and storage of XML files. The pre-existing tools could not satisfy our need of managing the Domain Metadata because of the lack of specialization, in most of the cases.

Using the tool, we created the domain knowledge component of an ITS that teaches the subject of “Evaluation Techniques for Computing Systems”. Since this is part of the whole ITS, we aim at developing the rest of it.

Our tool provides the capability of specifying only the “prerequisite” relation between concepts. A future effort will be to the direction of providing other types of relations, like e.g. the “part-of” one, too.

References

- ALOHA 2, 2004: <http://aloha.netera.ca/about.php>
- Andreas Holzinger, Thomas Kleinberger, Paul Müller, *Multimedia Learning Systems based on IEEE Learning Object Metadata(LOM)*, Proceedings of ED-MEDIA 2001 World Conference on Educational Multimedia, Hypermedia & Telecommunications, Tampere, Finland, June 25-30, 2001.
- Curriculum Online Tagging Tool (v3x):
<http://www.curriculumonline.gov.uk/SupplierCentre/taggingtool.htm>
- eRIB Metatagging Tool: <http://demo.licef.teluq.quebec.ca/eRIB/>
- EXPLORA-II LCMS and Learning Object Repository / EXPLORA-II:
<http://explora2.licef.teluq.quebec.ca/demo/>
- IEEE 1484.12.1–2002, *Standard for Learning Object Metadata*
- IEEE P1484.12.3, *Draft Standard for Learning Technology—Extensible Markup Language (XML) Schema Binding for Learning Object Metadata*.
- Jim Prentzas, Ioannis Hatzilygeroudis, John Garofalakis, *A Web-Based Intelligent Tutoring System Using Hybrid Rules as its Representational Basis*, Proceedings of the International Conference on Intelligent Tutoring Systems (ITS-2002), June 2002, as S. A. Cerri, G. Gouarderes and F. Paraguacu (Eds), *Intelligent Tutoring Systems*, LNCS 2363, Springer, 2002, 119-128.
- LOM-Editor. (2001). *LOM-Editor Version 1.0*. Darmstadt University of Technology.
<http://www.multibook.de/lom/en/index.html>
- LOMPad: <http://demo.licef.teluq.quebec.ca/LomPad/>
- Reggie. (1998). *Reggie Metadata Editor*. Distributed Systems Technology Centre, Australia. <http://metadata.net/dstc/>