Journal of Universal Computer Science, vol. 16, no. 19 (2010), 2841-2861
submitted: 1/3/10, accepted: 29/9/10, appeared: 1/10/10 © *J.UCS*

A Tool for Managing Domain Knowledge and Helping Tutors in Intelligent Tutoring Systems

Panayiotis Kyriakou, Ioannis Hatzilygeroudis, John Garofalakis
(Department of Computer Engineering & Informatics, University of Patras, Patras, Greece
pkiriakou@gmail.com, ihatz@ceid.upatras.gr, garofala@ceid.upatras.gr)

Abstract: Intelligent Tutoring Systems (ITSs) constitute a popular type of intelligent educational systems. Domain Knowledge (DK) is a basic part of an ITS and usually includes information about the concepts the ITS is dealing with and the teaching material itself. The teaching material consists of a set of learning objects (LOs). A LO is described by a data set called its metadata. Concepts are usually organized in a network, called a concept network (or map). Each concept is associated with a number of LOs. Existing tools for managing DK mainly deal with either LOs or concepts, but not with connecting them. In this paper, we present a tool for managing both types of information in DK: creating and editing a concept network and LO metadata as well as connecting them. Additionally, the tool can produce corresponding XML descriptions for each LO metadata. Finally, it provides facilities for helping tutors in organizing and composing their lessons. A small scale evaluation has shown more than satisfactory acceptability of the tool.

Keywords: Intelligent Tutoring Systems, Domain Knowledge management, Learning Objects management, Lesson organization
Categories: L.1.0, L.1.2, L.2.0, L.3.0, K.3.1

1 Introduction

Intelligent Tutoring Systems (ITSs) constitute a popular type of intelligent educational systems [Urretavizcaya-Loinaz, 02] [Hatzilygeroudis, 06] [EXPLOR@-2]. They are a relatively new generation of computer-based educational systems, encompassing intelligence in order to increase their effectiveness by offering personalized learning. ITSs take into account the user's knowledge level and skills and adapt presentation of the teaching material to the needs and abilities of individual users. This is achieved by using mainly Artificial Intelligence techniques to represent pedagogical decisions as well as domain knowledge and information regarding each student. ITSs were usually developed as stand-alone systems. However, the emergence of the WWW gave rise to Web-based ITSs [Brusilovski, 99] [Hatzilygeroudis, 04].

The structure of an ITS is illustrated in Figure 1. An ITS typically consists of three main modules [Polson, 88] [Nwana, 90] [Claude, 90]: (a) the domain knowledge, which contains the teaching content and meta-information about the subject to be taught, (b) the user model, which records information concerning the user, and (c) the pedagogical model, which encompasses knowledge regarding various pedagogical decisions.

In the domain knowledge, the teaching material must be structured in such a way that can be easily recognized and used by the pedagogical unit, in order to adapt teaching to user's needs. A quite helpful way is to distinguish between the teaching material itself and its meta-information, typically called its metadata [Prentzas, 02]. The teaching material itself usually consists of learning objects (LOs), which are autonomous, self-contained digital entities (e.g. web pages) used to support learning [LOM-Wiki]. To be able to manage domain knowledge in an ITS, we need a tool that will be able to manage LOs metadata and real teaching material too. However, apart from those, domain knowledge also contains information about the concepts the system is concerned with.

Although there are a number of tools dealing with management of metadata for LOs, most of them are not suitable for dealing with concepts. For example, tools like Reggie Metadata Editor [Reggie, 98], Curriculum Online Tagging Tool [COTT, 05] and eRIB Metatagging Tool [eRib], although they can create and manage metadata, they cannot organize them in a structure and relate them with the concepts used in ITSs. Also, they cannot create and manage a concept structure. On the other hand, although there are tools dealing with concepts, they are not able to relate them with LOs. For example, tools like Inspiration [Inspiration], IHMC CmapTools Kit [Cañas, 03a] and SMART Ideas [SMART] can create and manage a concept network, but they cannot connect actual teaching materials with them and manage them. Furthermore, none of the above tools offer facilities for helping tutors in organizing and composing lessons.

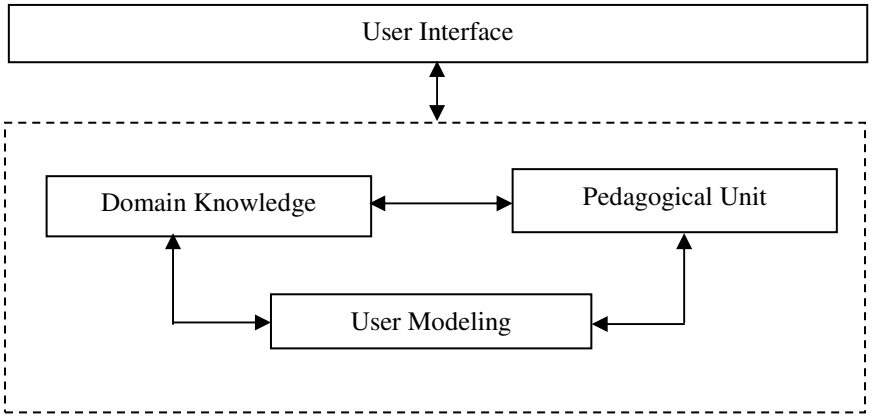


Figure 1: The Basic Structure of an Intelligent Tutoring System

In this paper, which is an extension of [Kyriakou, 08], we present a tool that is able to deal with both LOs and their metadata description as well as with concepts and their relations. Also, it helps tutors in preparing lessons. Description of LOs metadata is based on the IEEE LOM standard [Holzinger, 01] [LTSC, 01]. The tool has its own delivery system, which delivers teaching content and author content to the associated ITS. Its delivery system is not presented in this paper.

The paper is structured as follows. Section 2 presents the structure of domain knowledge in an ITS and its requirements of managing it. Section 3 presents related work and shows the inadequacies of existing tools. Section 4 presents the basics of the IEEE LOM standard for modelling learning objects metadata, whereas Section 5 deals with the introduced tool itself. Section 6 describes the evaluation process of the tool and presents its results. Finally, Section 7 concludes the paper.

2 Domain Knowledge

The motivation for creating the tool introduced in this paper derived from the need for managing the domain knowledge of a certain ITS, presented in [Prentzas, 02]. Domain Knowledge is an essential component of an ITS since it contains the teaching material and its structure. The domain knowledge of that web-based ITS has the structure of Figure 2. It consists of three components: knowledge concepts, course units and meta-description.

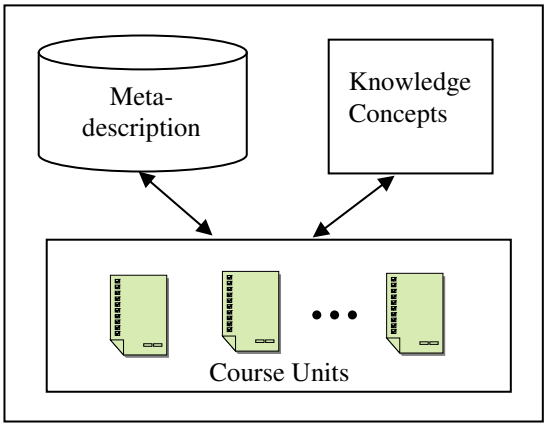


Figure 2: The Structure of Domain Knowledge

The *knowledge concepts* are elementary pieces of information of the specific domain. Every concept has a number of general characteristics such as the name, the difficulty level, the detail level, the prerequisite knowledge, etc. Moreover, a concept has relations with the other concepts which mainly show the prerequisite concepts that contain the prerequisite knowledge for that concept. For example, in teaching logic in an Artificial Intelligence course, 'logic syntax' and 'logic semantics' could be two concepts. 'logic syntax' could have as prerequisites the concepts 'constant', 'variable', function' etc. However, there may be other types of relations, like "generalizes", "specializes", "part-of" etc. For example, 'left-hand expression' may be "part-of" the 'implies expression'. The concepts and their links form a network (see Figure 3), which is a semantic network that represents the pedagogical structure of the teaching subject.

The teaching material consists of two parts: (a) the *course units* and (b) the *meta-description*. The course units mainly are in the form of web pages and are equivalent to LOs. Each course unit may contain a variety of data types (e.g. text, images, animations etc).

Course units are used in composing lessons. Each course unit is related to a concept and through the concept network, that has been created, the system chooses the next course unit (web page) to be presented to the user. A course unit can be of a theory, an example or an exercise type. Examples help the student to understand the theory. Exercises are based on the examples and are used to evaluate the knowledge level of the user. This information is used to update the model of the user.

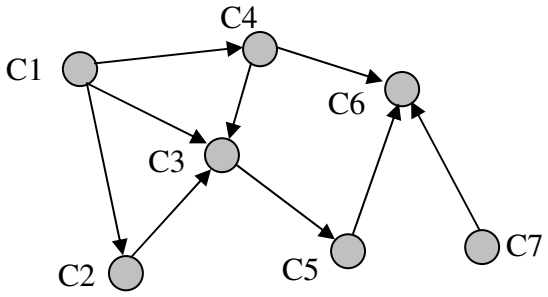


Figure 3: A Concept Network

The domain knowledge also contains meta-descriptions of the course units and their main attributes. Such attributes are mainly its difficulty level, its pedagogical type (theory, example, exercise), its representation type (text, image, animation), its detail level, etc. Those meta-descriptions of the course units are based on the IEEE LOM standard schema, which is presented in the Section 4.

3 Related Work

We distinguish two categories of tools that could be used for management of the domain knowledge of an ITS. The first category includes tools that deal with management of the metadata of LOs. Such tools are the following.

LOM Editor [LOM-E, 01], developed by Darmstadt University of Technology in 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 and ISO 8601 DateTime standards 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, 98], 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 looses 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] is a Java-based tool that is used for indexing, aggregating, sharing, multi-purposing, and re-purposing LOs. It has been 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. ALOMA II is not web-based and not based on the IEEE LOM standard.

Curriculum Online Tagging Tool [COTT, 05], 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] is concordant with the CanCore Guidelines and permits users to create Metadata records for LOs 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, non web-based, rather than a domain knowledge management tool.

LomPad [LOMPad] 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.

The second category concerns tools that mainly deal with creating concepts and concept maps. Inspiration [Inspiration] uses a web as a basic graph structure, and repositions the initially entered concept in the middle of the screen. This tool does not enforce any particular graph structure and the representation does not require linking phrases. The software allows the user to switch from graph to outline view and back.

SMART Ideas [SMART] allows users to create multi-level Concept Maps to organize ideas, to link Concept Maps to files and Web sites, to switch between diagram and outline views, and to publish Concept Maps on the Web.

LifeMap [LifeMap] is designed for free educational use. Group packages with support are available. LifeMap provides the capability to make Vee diagrams as described in [Novak, 84].

IHMC CmapTools software kit [Cañas, 03a] is to enable users to collaborate during Concept Map construction and to easily share and publish the resulting knowledge models. The software is based on a client-server architecture [Cañas, 03b] that allows users to share and browse Concept Maps stored in CmapServers distributed throughout a network that covers the whole world.

Luckie Concept Connector is a software suite currently in development at Michigan State University. This system allows students to build Concept Maps online and to receive immediate feedback about their maps based on automatic scoring systems that are derived from scoring methods detailed in [Novak, 84]. The Concept Mapping system is based upon a pre-defined set of concepts and linking phrases. The system is currently being used for online homework assignments.

Finally, the Grapple Authoring Tool (GAT) has three main components: a Domain Model authoring tool (DM), for creating a conceptual representation of an application domain or "course", a Concept Relationship Type authoring tool (CRT), for defining types of pedagogical relationships between concepts and their associated adaptation, and a Conceptual Adaptation Model (CAM) authoring tool for defining the pedagogical structure of a course, [GRAPPLE]. GAT is a more general tool than ours, i.e. it covers more aspects of adaptive course authoring , whereas our tool refers to a part of them, i.e. mainly to managing domain knowledge. Although the DM authoring tool of GAT is similar to our tool, it has not been designed for managing LOs and their metadata. So, LOs cannot be directly used in creating lessons. Also, it does not provide any help to tutors/educators in designing a lesson. However, in GRAPPLE all this is adaptively done by the system itself, given that it has a different philosophy. The main objective of the GRAPPLE project is not to manage the actual teaching material, but to deliver the content to the learners according to their needs. The main objective of our tool is first the management of LOs and then their delivery to the learners through an ITS. So, our tool does not deal with authoring adaptation aspects, which is left (at the moment) to the associated ITS. On the other hand, GRAPPLE provides a separate tool for defining pedagogical relationships between concepts, namely the CRT authoring tool. This separation is another main difference from our tool, which mixes domain model relationships with pedagogical ones in the same concept map. This could be considered as a shortcoming of our tool, although the prerequisite concepts highlighting facility (see Section 5.3) allows for a partial separation.

The tools of the first category can help in creating and managing LOs, but they cannot help in creating and managing a concept network. The tools of the second category can create and manage a concept network, but they cannot associate LOs with them and manage them. Furthermore, none of them offer any help to tutors for creating lessons. The tool presented in this paper bridges the gap between these two categories, by offering facilities for creating, managing and connecting both LOs and concept networks. Also, it offers facilities for helping tutors in preparing lessons.

4 Learning Object Metadata

Learning Object Metadata (LOM) is a data model used to describe a learning object and similar digital resources used to support learning. The conceptual model of LOM is presented in Figure 4 (taken from [LOM-Wiki]). The purpose of LOM is to support reusability of LOs, to aid discoverability, and to facilitate their interoperability, usually in the context of online learning management systems [LOM-Wiki].

The IEEE 1484.12.1 Standard [14] is an internationally-recognized open standard for the description of LOs according to LOM model. The model includes attributes for the description of a LO, like type of object, author, owner, terms of distribution, format, and also pedagogical attributes, such as teaching or interaction style.

4.1 Overview of the IEEE LOM

According to LOM conceptual model, data elements that describe a LO are grouped into categories such as the following (taken from [LTSC, 02]):

- 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.

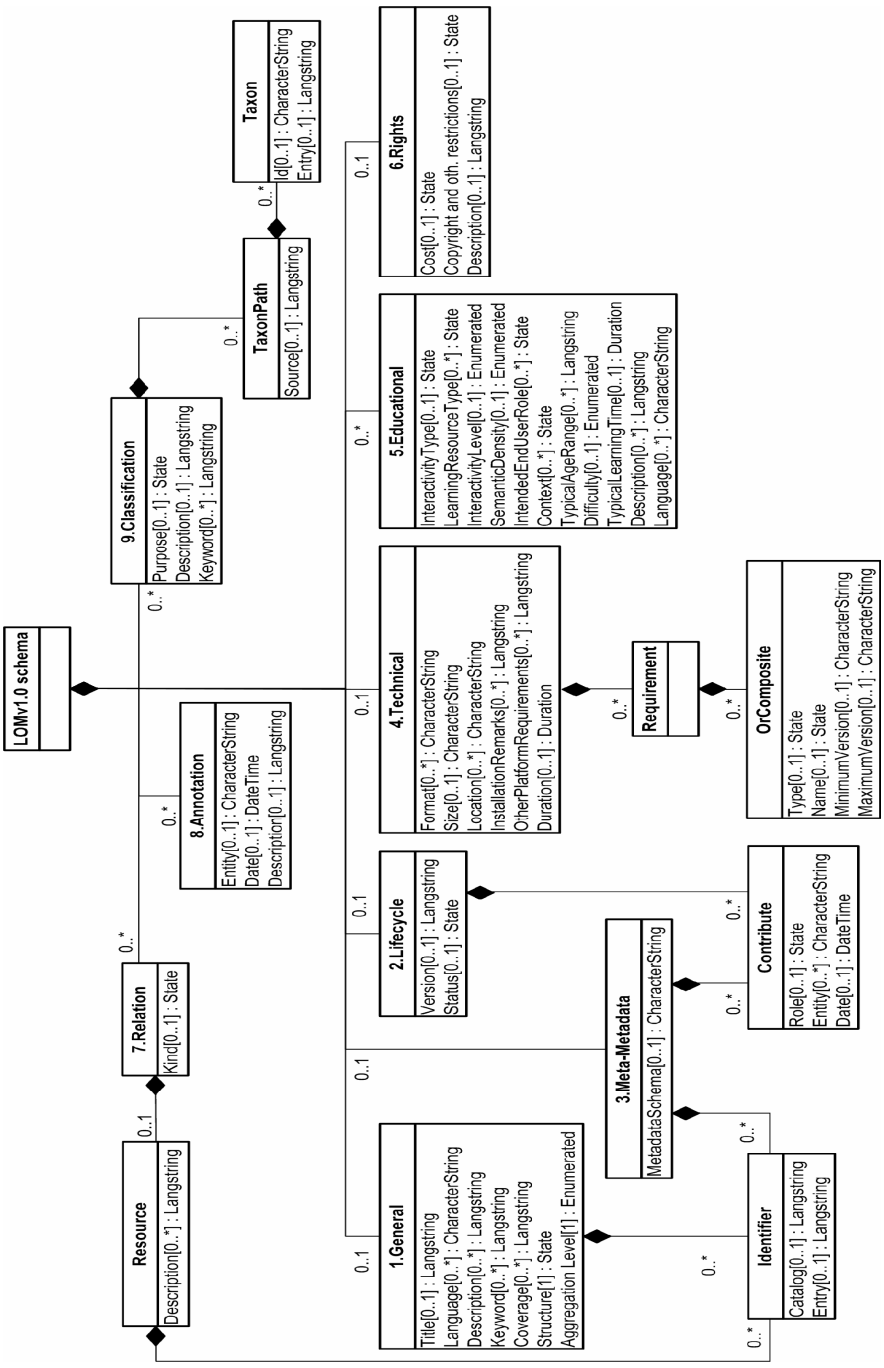


Figure 4: The Hierarchy of the IEEE Conceptual Model for Metadata Definition

- The *Classification* category describes this learning object in relation to a particular classification system.

Collectively, the above nine categories form the LOM v1.0 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.

4.2 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 [LTSC, 02]:

- *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.

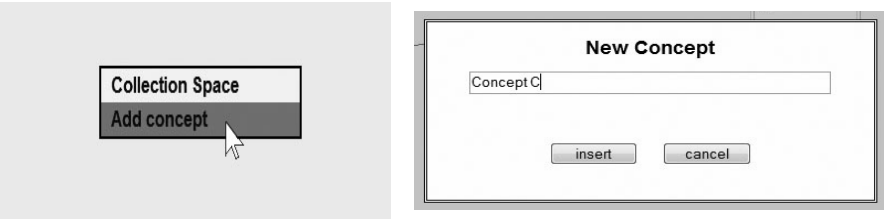
5 Creating a tool for managing domain knowledge

We created a tool for managing the domain knowledge of an ITS based on the requirements of the Domain Knowledge of an ITS, as presented in Section 2, and the LOM data model. So, using the tool one can create the concept network of a domain, using a variety of relation links, which him/herself can define.

Also, can create, store, view and edit the metadata for the LOs (i.e. course units) in XML format. Furthermore, one can create a lesson by organizing a number of LOs.

5.1. Concept Map

The tool provides facilities for easily creating a map (network) of the concepts involved in the domain. To create a new concept, one can click on the tool workspace and when the menu of Fig. 5a appears choose "Add concept". Then the interactive form of Fig. 5b appears where he/she can define the name of the new concept.



(a) *(b)*

Figure 5: New Concept Creation

To create a relation between two concepts, one should click on the icon, laying at the right-hand side of a concept (see Fig. 7) and then on the related concept. Then the interactive form of Figure 6 appears, where he/she can choose a relation type from a list of existing ones or create a new type of relation.

Figure 6: Form for adding a relation

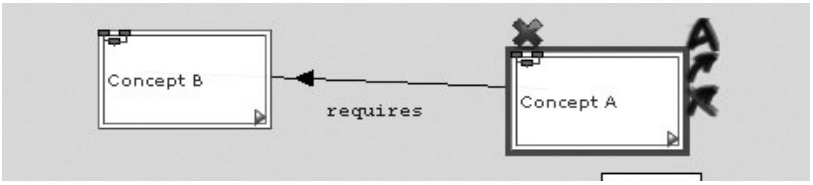


Figure 7: Concept Map Creation

Figure 7 shows two concepts connected with the "requires" relation. At each concept one can see, at the first glance, the name of the concept and its relations with other concepts. Also, when the mouse is over a concept one can see more icons that perform certain functions. The icon calls the renaming function; the icon calls the function that deletes a relation; and the icon deletes the selected concept.

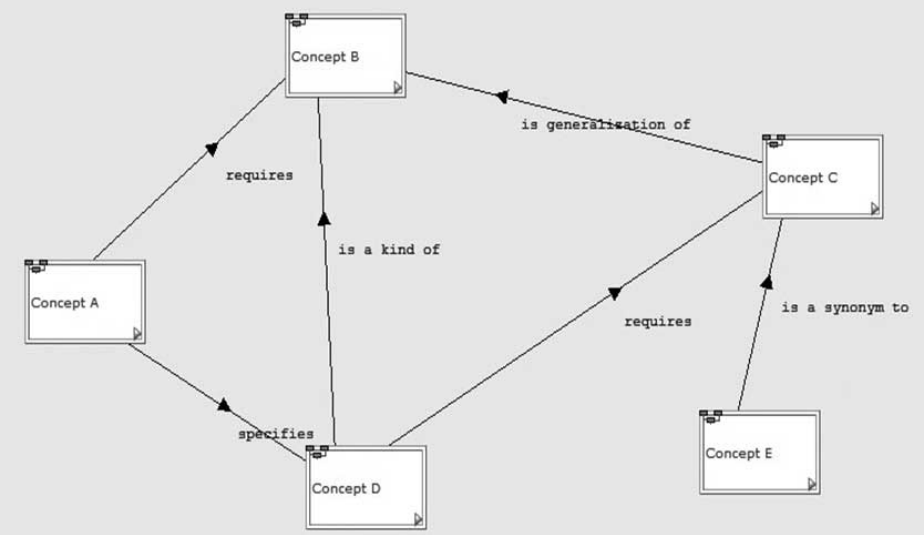


Figure 8: A Concept Map (Network)

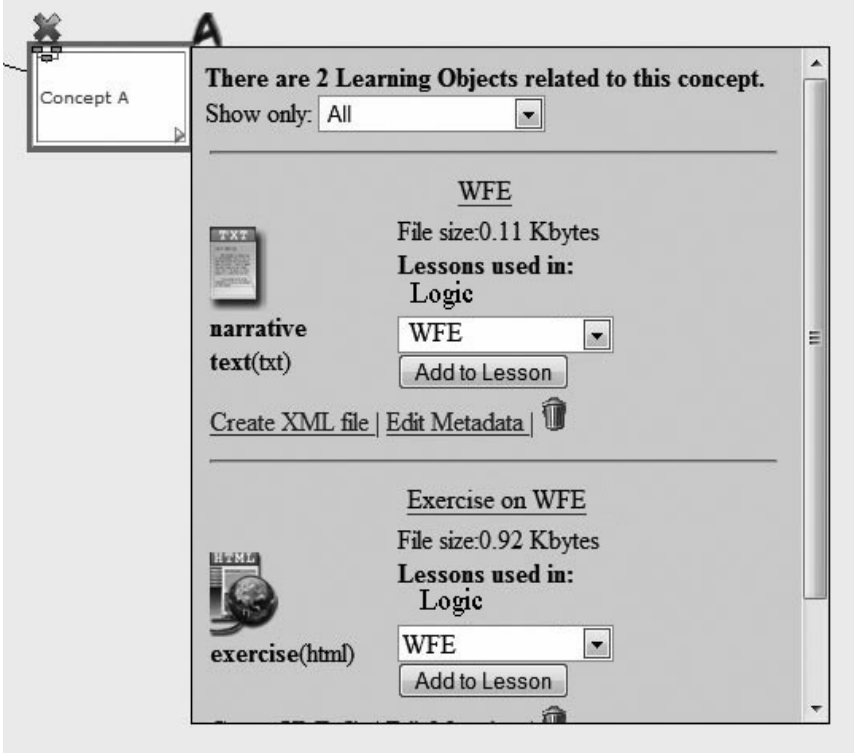


Figure 9: Displaying Learning Objects

The user can insert concepts at any position of the concept map, drag the concepts across the map in order to organize them, create new relations and connect them. In Figure 8, a concept network, where concepts are connected with various types of relations, is presented.

For the convenience of the users, there is a search bar, which we can give in any keyword related to concepts or to the metadata of the LOs. The result is to center the concept map to the selected concept and highlight it.

5.2 Managing Learning Objects

The LOs (course units) attached (i.e. related) to each concept can be displayed by double clicking on the corresponding concept (see Fig. 9). We can add/create a LO, by clicking on the "Add Learning object" link or delete any of the existing ones (not shown in Fig. 9).

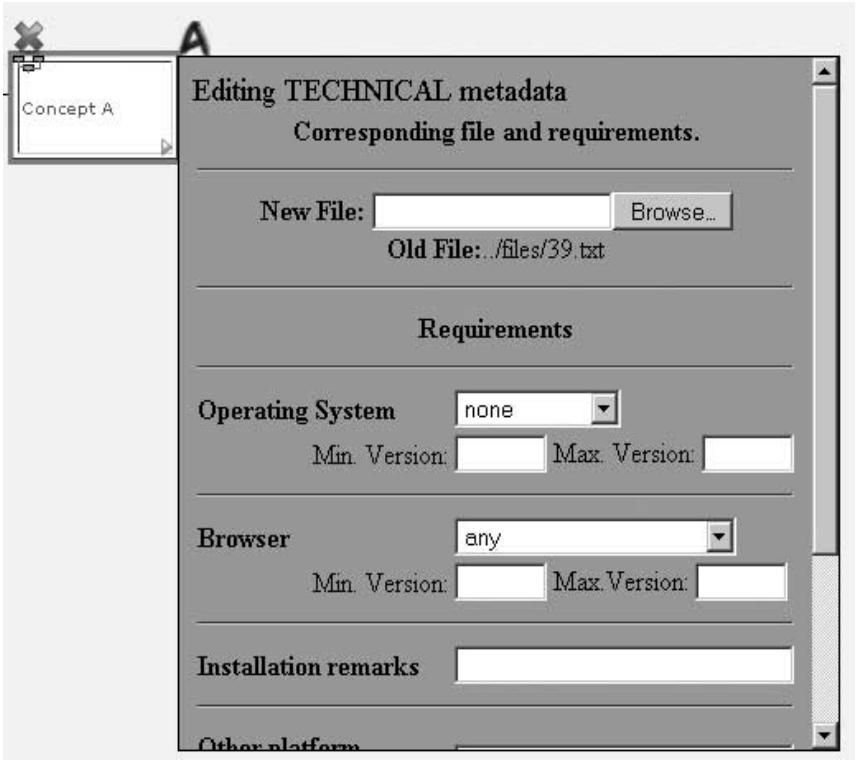


Figure 10: Displaying and Editing

Having displayed the LOs, we can do two other things:

(a) Display and edit its metadata, by clicking on the "Edit Metadata" link of a LO (see Figure 9). Its metadata appears in a structured way (see Figure 10) so that can be easily read and edited (i.e. modified or deleted), using the

corresponding link in each metadata category. The structure of metadata follows the IEEE LOM standard. On this set of metadata there is the 'Relation' category, which however is not the same as the 'relation' between concepts. It specifies relationships between the LOs (course units).

(b) Create an XML file representing the object's metadata. The information that we have specified in the concept metadata is stored in our database as well as in XML files for future use. These XML files can be accessed by other elements of any other system that has access to our file system, which through that can decide on the selection of the concepts for use in a tutoring session. Therefore, we have included access to these files through links of our system. To see the XML file of a LO we should just click on the "Create XML file" link of the LO (Fig. 9). The XML file is then dynamically created (Fig. 11).

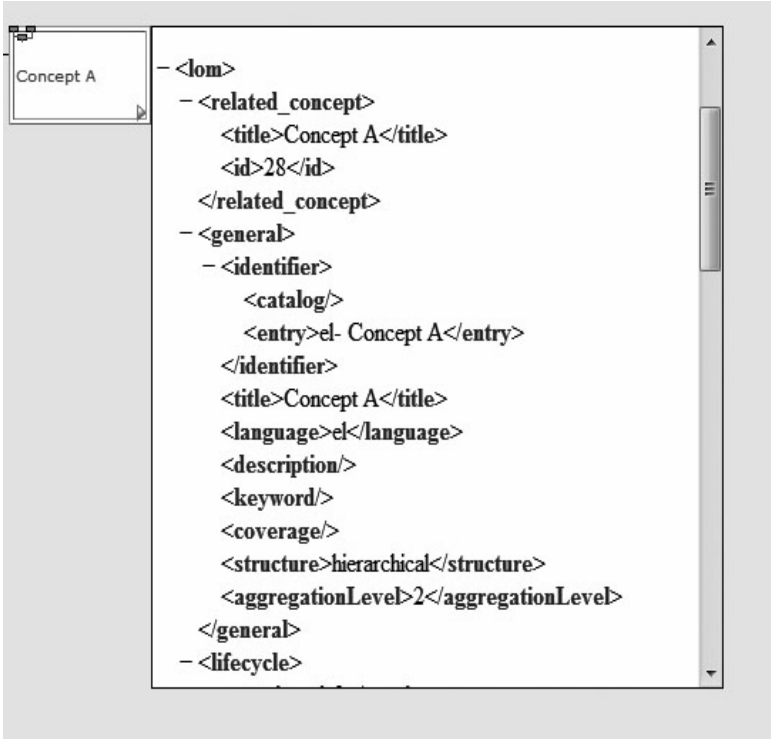


Figure 11: The XML file of a Learning Object

5.3 Organizing and Creating Lessons

The tool offers also facilities for helping tutors in preparing their lessons. There are two basic such facilities. The first is of high level, in the sense that it helps a tutor to organize the lessons on a subject. The second is of low level, in the sense that it helps a tutor to compose specific lessons.

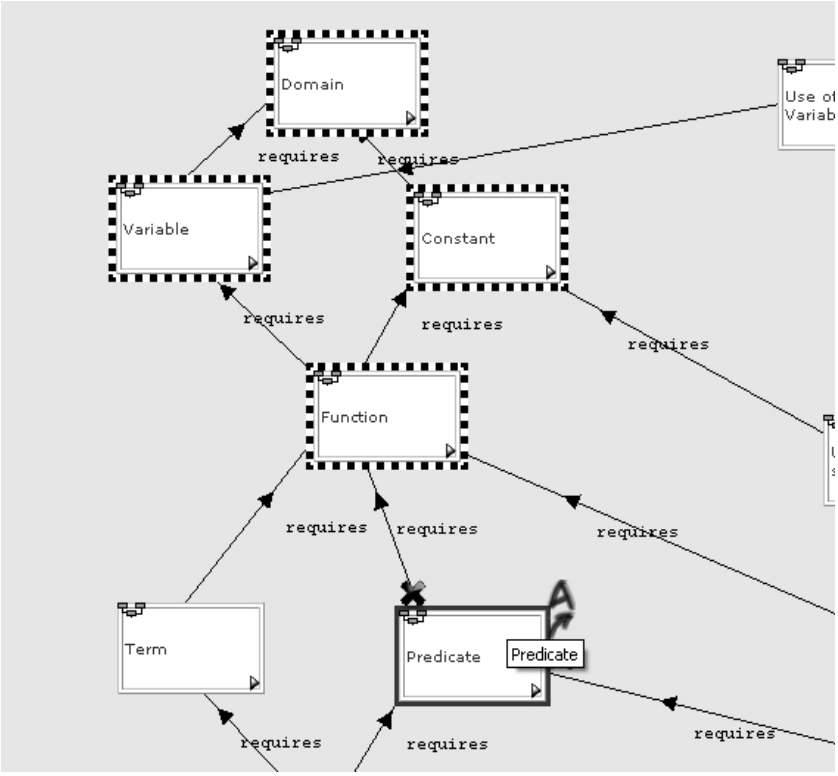


Figure 12: Highlighting prerequisite concepts

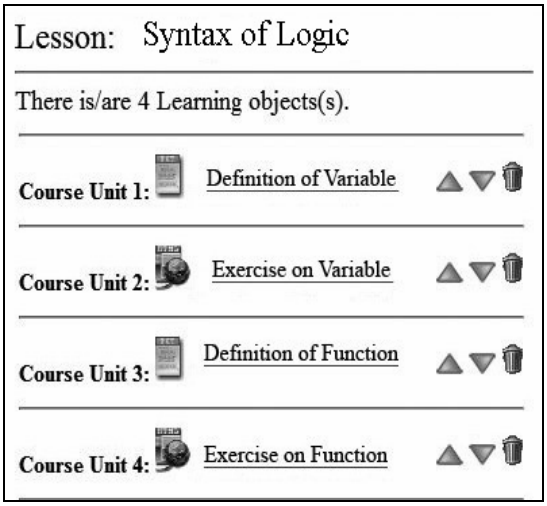


Figure 13: Lesson Editing Form

The high level facility that the tool offers to a tutor for organizing his tutoring is the one that highlights, by clicking click on the icon of a concept, all the prerequisite concepts of that concept. For example, in Figure 12, all the prerequisites of the concept "predicate" are highlighted. This helps the tutor to specify and organize a number of lessons in the right order.

The low level facility offered by the tool is the ability to compose a lesson by choosing the desired LOs. The tutor can create and display a lesson and add or delete any of the available LOs (course units). In Figure 13, a lesson with the LOs that it is composed of is displayed. LOs are of various types (theory, exercise etc.). An initial specification of the LOs for a certain lesson can be done via the pedagogical unit of the ITS. However, through the above facility, the tutor can modify it.

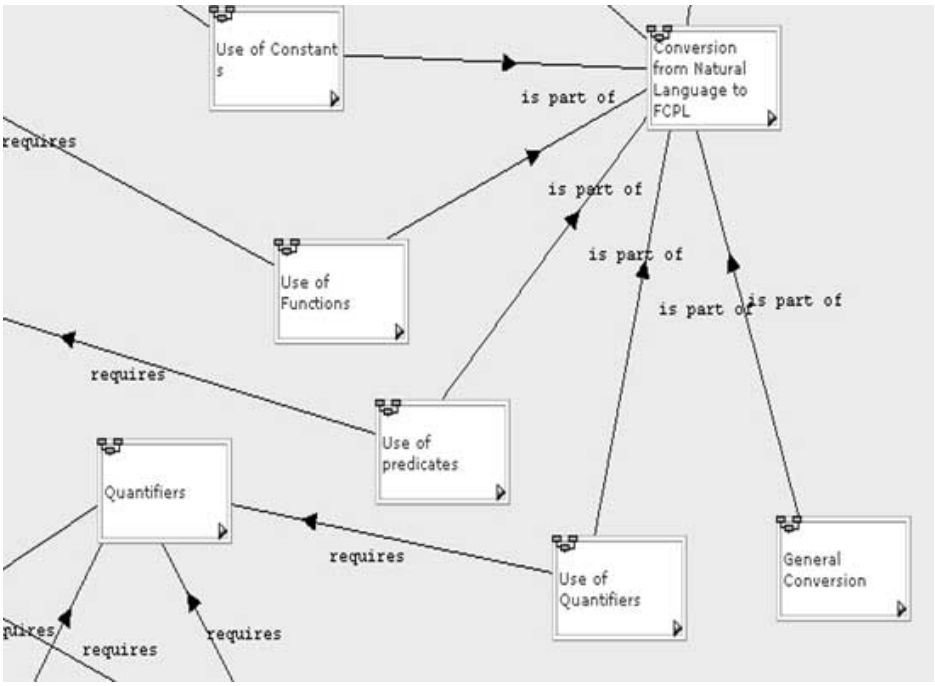


Figure 14: Part of the Concept Map for the Domain Knowledge of 'Logic as a Knowledge Representation Language'

5.4 An Example

We have used our tool to create the concept map for the subject 'logic as a knowledge representation language'.

In Figure 14, part of the concept map of that subject is presented, where two basic relations are used, namely 'requires' and 'is part of'. After the creation of the concept map, in the way described by Figures 5-8, a tutor assigned corresponding LOs to the concepts of the map to be able to create lessons. A snapshot of an instance of this process is illustrated in Figure 15. Then, for each LO its metadata, which follows the IEEE LOM standard, is edited, as illustrated in Figure 10.

Although existing tools like Inspiration [Inspiration], IHMC Cmap Tools Kit [Cañas, 03a] and SMART Ideas [SMART] can easily create a concept map (like that in Figure 14), they do not offer facilities for associating LOs with concepts and managing their metadata. On the other hand, tools like ALOHA II [ALOHA], LomEditor [LOM-E, 01], the Reggie Metadata Editor [Reggie, 98] and eRIB Metatagging Tool [eRib], although are great for managing metadata for LOs, they lack the fundamental functionality of our tool, the connection of the concept network with the actual LOs and their metadata.

After assigning LOs to concepts, a tutor can create lessons for his/her own students. Helpful in this case is the ability of the tool to highlight the prerequisite concepts of a certain concept. For example, in Figure 12, the prerequisite concepts of 'Predicate', i.e. 'Function', 'Variable', 'Constant' and 'Domain' are highlighted (since they are connected with the "require" relation). So, the tutor should take into account that information in preparing a lesson for 'Function'. In creating the lesson, the tutor can assign, in a similar to above way, specific LOs, which he/she thinks are appropriate for a specific class or group of students, as illustrated in Figure 13.

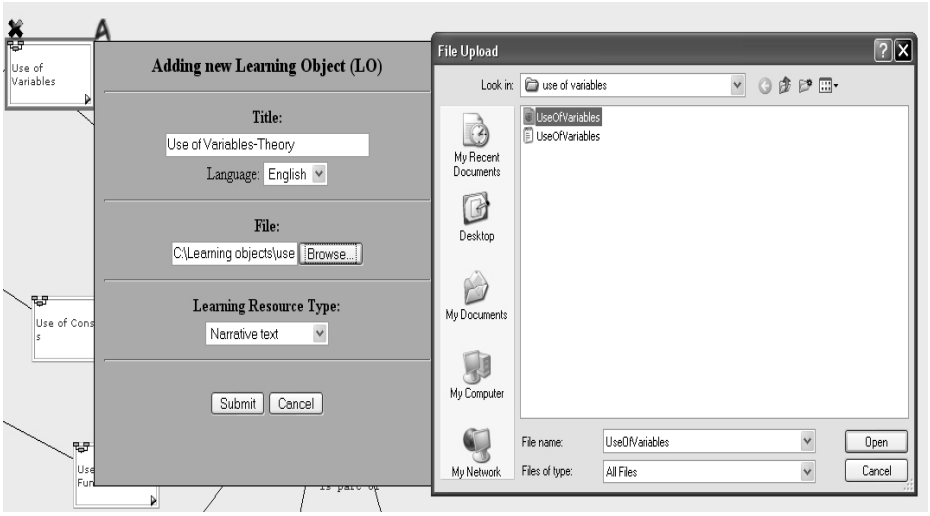


Figure 15: Assigning a LO to the concept 'Use of Variables'

5.5 Implementation Aspects

The heart of the system is its relational database, which is designed to store all vital information concerning the concept description and connections, the LOs metadata as well user related data. The database is organized in 20 tables, 1 related to 'learning object' (which is the central table), 11 for metadata management, 5 for the concept map creation, 2 for lesson handling and 1 for user identification.

WAMP5 version 1.6.1, containing Apache 1.3., PHP5 and MySQL4, has been used for the implementation of the system. MySQL has been used for the database, whereas PHP for the rest of the system, like e.g. the creation of XML files. For the concept map drawing, javascript has been used from the Open-JACOB Draw2D

library and the script.aculo.us library. The queries to MySQL are made through AJAX technology that connects the index.html file with the php files.

6 Evaluation

To accomplish a small-scale evaluation of the tool, we implemented the following procedure. We asked five tutors from different areas to test our tool. After a 10 minute training on the system use, we asked them to perform some tasks. The first task was to create a concept map consisting of a number of (at least 8) concepts and their relations, related to a subject of their choice. The next task was to link some LOs to the concepts they had created and add any essential metadata to the LOs. Next, they had to decide on a topic that they wanted to teach, create an empty lesson in the system and select from the concept map the concepts that they would use in their lesson. For this task, we asked them to use the "prerequisites" button that highlights the prerequisite concept(s) of a specific concept. Then by parsing each concept they had to select the LOs form each concept and add them to the lesson they had created. We also asked them to put in the right sequence the LOs in the lesson plan and delete any unnecessary LOs. Finally, we asked them to complete a questionnaire to record their views for the tool.

The questionnaire included 16 questions. Questions Q4, Q5, Q7-Q9 and Q15 were based on a 4-options Likert scale (1: strongly disagree, 2: disagree, 3: agree, 4: strongly agree). Questions Q1, Q2 and Q13 were multiple choice questions. Q1 was about the type of tutoring they are doing, whereas Q2 about their experience in tutoring. Q13 was about previous experience in using authoring tools. The rest questions were open text questions. Q3 was about how many concepts added, Q6 was about how many LOs added, Q10 about the total time spent on the tool, Q11 and Q12 were about the features of the tool found useful or difficult respectively and Q14 was about how they compare it with other tools, if the answer to Q13 was "yes".

No	QUESTION	ANSWERS			
		1	2	3	4
Q4	I thought it was easy to create concepts.	0	0	3	2
Q5	I found it easy to create relations between the concepts.	0	0	4	1
Q7	I felt confident adding Learning Objects to the Concepts.	0	0	2	3
Q8	It was easy to access and edit the metadata of each Learning Object.	0	0	3	2
Q9	It was easy to create a lesson with learning objects.	0	0	2	3
Q15	In overall this tool was easy to use and effective.	0	0	2	3

Table 1: Likert scale questions and results

All tutors, but one, have more than five years experience in tutoring (answers to Q2). Table 1 shows the questions that use a Likert scale and the results of the questionnaire regarding those questions. The results show that with a minimum training the users were able to use the tool quite easily and were quite satisfied. One further remark here is that users found creating relations between concepts (Q5) relatively more difficult than the other functions (Q4, Q7, Q8, Q9). This may be due to the fact that domain and pedagogical relationships are mixed in our tool. Notice that four of them had no previous experience with authoring tools (answers to Q13). The answer of the one who had previous experience is subsumed by the phrase "simpler to use, but more specific in use". It is true that our tool is not a general e-learning authoring tool, but covers part of the e-learning authoring aspects.

In question Q11, most of the users declared the concept map creation facility as most useful, whereas in question Q12 some of them declared that found some difficulties in creating relations and in using the LOs metadata editing tool. In Q16, the users made a variety of suggestions like, adding a help file, give an example of creating and editing LOs metadata, adding the functionality of exporting a lesson to a SCORM package. Finally, the average number of concepts added by the users was 10.40 (answers to Q3), the average number of LOs added to the concepts 4.60 (answers to Q6) and the average time users spent with the tool was 38 minutes (from answers to Q10).

7 Conclusions

In this paper, we present a tool that is suitable for managing the domain knowledge in an ITS and also capable of helping a tutor to organize and compose lessons. More specifically, the tool allows for creating, viewing, editing and deleting knowledge concepts, which are organized in a network (map). Concepts can be connected between each other by a variety of relations (e.g. "requires", "isa" , "generalizes" etc), which can be user defined. Also, it allows associating concepts with corresponding LOs, i.e. real teaching material. In addition, the user has the capability of adding to or modifying metadata (which complies with IEEE LOM data model) of each LO associated with a concept and create XML files, which he/she can view and edit in the process.

The contributions of our tool are: (a) It offers facilities for creating concept maps and managing both LOs and concept maps in the same framework. (b) As a consequence, it offers the capability of associating LOs with concepts in a concept network. (c) The tool can help tutors in organizing and composing their lessons. It offers a facility for specifying and displaying the prerequisite concepts of a concept and thus giving help in putting lessons and LOs in the right order. A tutor can view and edit the list of LOs associated to a lesson. (d) The tool also can provide XML descriptions for LOs metadata. These are not offered in such an integrated way by existing tools.

A possible alternative to our tool, as far as creation of a concept map and LOs descriptions is concerned, would be to use a LOM editor in conjunction with a concept map editor, but there it stops. The rest (i.e. associating LOs with concepts and furthermore creating lessons in an intelligent way, which is the target objective of our

work) should be made manually by the tutor, given that possibly there is no way to connect the results of the two tools in an automatic or semi-automatic way.

In our tool, concept structural relationships are mixed with pedagogical ones in the same concept map. So, it gives some difficulty to the user in handling them separately, as required in different authoring functions. Although our tool offers the facility of highlighting prerequisite concepts (i.e. concepts connected with "requires", a type of pedagogical relationship), a more clear way of handling them separately is needed.

Our tool is designed to be part of an ITS, which will "read" the concept map and deliver the course material to the learners automatically. So, it has its own delivery tool. One possible extension would be to export a lesson to a SCORM package. This functionality would be very useful to tutors that are using an educational platform that support SCORM packages like Moodle [Moodle]. This functionality will be one of our priorities in the next version of the tool.

References

[ALOHA] ALOHA II: http://aloha.netera.ca/about.php. Accessed 1st June 2010.

[Brusilovski, 99] Brusilovski, P.: "Adaptive and Intelligent Technologies for Web-based Education"; In: Rollinger, C. and Peylo, C. (Eds), Kustliche Intelligenz, Special Issue on Intelligent Systems and Teleteaching, 4 (1999), 19-25.

[Cañas, 03a] Cañas, A. J., Hill, G., Carff, R. and Suri, N.: "CmapTools: A knowledge modeling and sharing toolkit"; Technical Report IHMC CmapTools 93-01, Institute for Human and Machine Cognition (2003).

[Cañas, 03b] Cañas, A. J., Hill, G., Granados, A., Pérez, J. D. and Pérez, C.: "The network architecture of CmapTools"; Technical Report IHMC CmapTools 93-02, Institute for Human and Machine Cognition (2003).

[COTT, 05] Curriculum Online Tagging Tool (v3.1.2-2005): http://developer.k-int.com/cott3/. Accessed 1st June 2010.

[eRIB] eRIB Metatagging Tool: http://demo.licef.teluq.uquebec.ca/eRIB/. Accessed 1st June 2010.

[EXPLOR@-2] EXPLOR@-2: http://explora.teluq.uquebec.ca/2.0/en/index.htm. Accessed 1st June 2010.

[GRAPPLE] The GRAPPLE project: http://www.grapple-project.org/. Accessed 1st June 2010.

[Contreras, 07] Contreras, F. W., Galindo, G. E., Caballero M., E., Caballero, M. G.: "An Intelligent Tutoring System for Education by Web"; Procs of the IASTED International Conference on Web-Based Education (WBE-2007), March. 14-16, Chamonix, France (2007).

[Claude, 90] Frasson, C., Gauthier, G., (Eds): "Intelligent Tutoring Systems, At the Crossroads of Artificial Intelligence and Education", Intellect Books (1990).

[Hatzilygeroudis, 04] Hatzilygeroudis, I., Guest Editor: Special Issue on AI Techniques in Web-Based Educational Systems; International Journal on AI Tools (IJAIT), 13(2) (2004).

[Hatzilygeroudis, 06] Hatzilygeroudis, I., Prentzas, J.: "Knowledge Representation in Intelligent Educational Systems"; Z. Ma (Ed), "Web-Based Intelligent e-Learning Systems: Technologies and Applications", Idea Group Inc. (2006), 175-192.

[Holzinger, 01] Holzinger, A., Kleinberger, Th., Müller, P.: "Multimedia Learning Systems based on IEEE Learning Object Metadata"; Proceedings of ED-MEDIA 2001 World Conference on Educational Multimedia, Hypermedia & Telecommunications, Tampere, Finland, June 25-30 (2001).

[LTSC, 02] IEEE LTSC. Draft Standard for Learning Objects Metadata 2002. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf . Last retreival, 1st June 2010.

[Inspiration] Inspiration: http://www.inspiration.com/. Accessed 1st June 2010.

[Kyriakou, 08] Kyriakou, P., Hatzilygeroudis, I., Garofalakis, J.: "A Tool for Managing Domain Metadata in a Web-based Intelligent Tutoring System"; Proceedings of the 3rd International Conference on Software and Data Technologies (ICSOFT 2008), Porto, Portugal, 5-8 July, Vol.ISDM/ABF (2008), 5-11.

[LifeMap] LifeMap: http://www.robertabrams.net/conceptmap/lifemaphome.html. Accessed 1st June 2010.

[LOM-E, 01] LOM-Editor. Darmstadt University of Technology, 2001: http://www.multibook.de/lom/en/index.html. Accessed 1st June 2010.

[LOMPad] LOMPad: http://demo.licef.teluq.uquebec.ca/LomPad/. Accessed 1st June 2010.

[Moodle] Moodle, Open Source Course Management System (CMS): http://moodle.org. Accessed 1st June 2010.

[Nwana, 90] Nwana, H. S.: "Intelligent Tutoring Systems: an overview"; Artificial Intelligence Review 4 (1990), 251-277.

[Novak, 84] Novak, J. D., Gowin, D. B.: "Learning how to learn". New York: Cambridge University Press (1984).

[Polson, 88] Polson, M. C., Jeffrey, R. J., Soloway E.: "Foundations of Intelligent Tutoring Systems", Lawrence Erlbaum Associates (1988).

[Prentzas, 02] Prentzas, J., Hatzilygeroudis, I., Garofalakis, J.: "A Web-Based Intelligent Tutoring System Using Hybrid Rules as its Representational Basis"; Proceedings of the International Conference on Intelligent Tutoring Systems (ITS-2002); as Cerri, S. A., Gouarderes, G., Paraguacu, F. (Eds): "Intelligent Tutoring Systems", LNCS 2363, Springer (2002), 119-128.

[Reggie, 98] Reggie- The Metadata Editor. Distributed Systems Technology Centre, Australia, 1998: http://metadata.net/dstc/. Accessed 1st June 2010.

[SMART] SMART Ideas: http://www.smarttech.com. Accessed 1st June 2010.

[Urretavizcaya-Loinaz, 02] Urretavizcaya-Loinaz, M., Fernandez de Castro I.: "Artificial Intelligence and Education: an Overview", UPGRADE, The European Online Magazine for the IT Professional, 3(5) (2002), 53-58.

[LOM-Wiki] Wikipedia: http://en.wikipedia.org/wiki/Learning_object_metadata. Accessed 1st June 2010.