AIED 05 WORKSHOP 3

Amsterdam, 18-22 July, 2005

SW-EL'05: Applications of Semantic Web Technologies for E-Learning

AIED 05 WORKSHOP

12th International Conference on Artificial Intelligence in Education, Amsterdam, the Netherlands

AIED'05 WORKSHOP

SW-EL'05: Applications of Semantic Web Technologies for E-Learning

in conjunction with
12th International Conference on Artificial Intelligence in Education
(AIED'05)

July 18th, 2005
Amsterdam, The Netherlands

Preface

The AIED'05 session of the SW-EL'05 workshop focuses on Semantic Web-based knowledge representation and engineering approaches and methods for the needs of intelligent learning systems and discusses issues related to their use for content and knowledge components specification, effective intelligent courseware construction and modelling the learner. The following topics are addressed in the attempt to achieve the Educational Semantic Web:

- Using ontologies and/or Semantic Web technologies for:
    - knowledge representation in intelligent educational systems
    - modularised and standardized architectures (e.g. separation of domain models and application models)
    - achieving interoperability between intelligent learning applications, sharable user models and knowledge components
    - supporting authoring of intelligent educational systems
- Semantic Web technologies for Adaptive Learning Systems
    - Personalization and adaptation in educational systems (flexible user models)
    - Ontology-based reasoning for personalising the educational Semantic Web
    - Techniques and methods to capture and employ learner semantics
- Semantic Web technologies for Educational Information Systems
    - SW-based indexing/annotation of educational content (incl. individual and community based)
    - Ontology-based information browsing and retrieval
    - Semantic Web/ontology based recommender systems
- SW-based Methods, Techniques, and Tools for:
    - Building and sharing educational content, models of users, and personalisation components.
    - Services in the context of intelligent educational systems (i.e. authoring service, user modelling service, etc.)
    - Ontology evolution, versioning and consistency
- Empirical research for Intelligent Educational Systems:
    - Real-world systems and case studies
    - Community and individual support with Semantic Web-technologies and Ontologies

To answer the open questions posed by this workshop, researchers from Computer Science, Education, Information Systems, Communication studies, Sociology, AI and HCI, come together to share their experience by working with divergent communities of teachers and learners and, possibly, involving educational web industries.

Two special sessions are organized within the context of this workshop:

Special Session on Semantic Web for Adaptive Learning Environments
Session co-chairs:
Vania Dimitrova (University of Leeds, UK)
Judy Kay (University of Sydney, Australia)

Special Session on Semantic Web-based Educational Information Systems

(KALEIDOSCOPE network of excellence)
Session co-chairs:
Niels Pinkwart, University of Duisburg-Essen, Germany
Danièle Hèrin, University of Montpellier II, France
Felisa Verdejo, National Distance Learning University, Spain
Cyrille Desmoulins, University of Grenoble, France

The 2005 edition of the SW-EL workshop is organized in three sessions held at three different conferences in order to discuss the current problems in e-learning from different perspectives, including those of advanced Web-based educational applications, Artificial Intelligence in education and the implications of applying Semantic Web standards and technologies for solving them:

- SW-EL'05 Session at ICALT'05, 5th July, 2005
- SW-EL'05 Session at AIED'05, 18th July, 2005
- SW-EL'05 Session at K-CAP'05, 2nd October, 2005

Researchers with interest in various aspects of Web-based educational systems get together at SW-EL'05 in order to discuss important issues in this field and present their latest results. We hope that the workshop will continue to contribute towards the realization of the vision of the Educational Semantic Web.

July, 2005
Lora Aroyo and Darina Dicheva

Workshop Committee

Co-chairs:
Lora Aroyo, Eindhoven University of Technology
Darina Dicheva, Winston-Salem State University

International Program Committee:
Grigoris Antoniou, FORTH, University of Crete, Greece
Cristina Baroglio, Università di Torino, Italy
Jacqueline Bourdeau, University of Québec, Canada
Joost Breuker, University of Amsterdam, The Netherlands
Peter Brusilovsky, University of Pitttsburgh, USA
Weiqin Chen, University of Bergen, Norway
Hugh Davis, University of Southampton, UK
Vladan Devedzic, University of Belgrade, Serbia and Montenegro
Peter Dolog, University of Hanover, Germany
Erik Duval (Katholieke Universiteit Leuven)
Jim Greer, Univeristy of Saskatchewan, Canada
Andreas Harrer, University of Duisburg-Essen, Germany
Nicola Henze, University of Hanover, Germany
Geert-Jan Houben, Eindhoven University of Technology, The Netherlands
Mitsuru Ikeda, JAIST, Japan
Kinshuk, Massey Univeristy, New Zealand
Rob Koper, Open University, The Netherlands
Antonio Kruger, University of Saarbruecken, Germany
Frank Linton, MITRE, USA
Gordon McCalla, University of Saskatchewan, CA
Tanja Mitrovic, University of Cantebury, NZ
Riichiro Mizoguchi, University of Osaka, Japan
Enrico Motta, Open University, UK
Ambjörn Naeve, Royal Institute of Technology, Sweden
Gilbert Paquette, LICEF, Canada
Alun Preece, University of Aberdeen, UK
Dave Reynolds, Hewlett-Packard Laboratories Bristol, UK
Enver Sangineto, University of Rome 3, Italy
Amy Soller, Institute for Defense Analyses, USA
Josie Taylor, Open University, UK
Julita Vassileva, University of Saskatchewan, Canada

Table of content

Posters

Flexible Learning Object Metadata

Christopher BROOKS, Gordon MCCALLA, Mike WINTER
Advanced Research in Intelligent Educational Systems Laboratory
Department of Computer Science, University of Saskatchewan
Saskatoon, SK, Canada
cab938@mail.usask.ca, mccalla@cs.usask.ca, mfw127@mail.usask.ca

Abstract. By far the most popular specification for learning objects is the IEEE Learning Object Metadata (LOM) standard. In it are outlined 76 different elements that correspond to pedagogical, technical, and administrative aspects of learning objects. This standard, however, has proven to be ineffective for creating computer adapted dynamic courseware.

This paper outlines some initial research we are doing in acquiring, describing, and using learning object metadata. Instead of the IEEE LOM, we argue for a more flexible approach to both defining and associating metadata with learning objects. By creating domain, educational, and learner characteristic ontologies, content can be dynamically linked to those competencies that are observed in a running e-learning system. This provides for a set of evolutionary metadata, where software agents can inspect multiple metadata instances for a given learning object and reason over them for a particular goal. As more metadata instances are added to the system, agents are expected to be able to provide more accurate reasoning, eventually leading to the dynamic delivery of personalized course content.

Introduction

Perhaps the most widely used and accepted learning object specification is the IEEE Learning Object Metadata (LOM) standard [14]. This standard identifies 76 different aspects by which a learning object can be annotated, and is supported in some way by all major learning object repositories and e-learning platforms. One would think then, that learning objects should be rich with metadata markup, and that the interrogation of such metadata by a content management system could be used to dynamically assemble a course. This, however, is not the case – content management systems are increasingly static, with even relatively simple rules-based sequencing specifications seeing little to no adoption [6].

The ability to dynamically assemble a course from learning objects is an important goal within the educational technology community. Nonetheless, current e-learning standards and specifications are both too restrictive in the variety of metadata they capture, and too lax in how they express the structure of such metadata. Many learning object repositories support only a few of the fields available, and most do not support an external query format (e.g. [17]) which could be used by computer agents to retrieve objects from the repository. The result is that nearly all learning object based courses are created directly by instructional designers, who align content they have explicitly hand crafted for a given educational purpose. This purpose generally includes both an educational outcome (e.g. "understand relational operators in ECMAScript at a level such that the student can apply them to new situations") as well as an educational instruction style (e.g. a particular language, background, or learning style that a student is assumed to have). This makes dynamic delivery difficult, as an instructional designer must create many different versions of a course for the different kinds of purposes he or she hopes to achieve.

Instead, we argue that a more flexible method of associating metadata with learning objects will help in realising the on-demand assembly of courses for different educational purposes. A larger set of well-defined ontologies sufficient for particular purposes should be used instead of a single highly constrained taxonomy of values like the LOM. Further, the ontologies should be marked up in an unambiguous syntax such that they are able to be understood by software agents. This syntax must take into account the kinds of data types that agents are able to manipulate, and must appropriately codify metadata instances to conform to these. Finally, repository and content management software must be able to associate multiple metadata instances with a given learning object, and allow for agents to pick and choose those instances that fit their needs.

This paper is organized as follows; Section 1 outlines specific issues both we and others have had when trying to use the IEEE LOM with software agents. Section 2 outlines in general terms our approach, dubbed the *ecological approach*. Section 3 indicates how we are using semantic web techniques help to enable this approach in real e-learning systems. Finally, section 4 concludes the work by identifying some potential related areas for exploration.

1. Issues with the Learning Object Metadata Standard

The IEEE Learning Object Metadata (LOM) standard [14] provides a format for representing technical, administrative, and pedagogical metadata about a learning resource. While adopted by e-learning vendors and used in many other e-learning specifications (e.g. [18] [13] [10]), its use in actual deployed learning systems is sparse at best. The specification suffers from three main issues. Firstly, to create a conforming LOM document requires only a few of the many available fields be filled in. Friesen provides a compelling example of this in a study of 250 learning object metadata records (chosen from five different projects evenly) where only 36% of the elements were used more than half of the time, with many elements never used at all [11]. Further, the elements used often referred to custom or local vocabularies, a practice that effectively eliminates semantic interoperability. While it has been suggested that automatic metadata generation tools are a potential solution for this, work to date has been less convincing. For instance, in [8] a set of tools were developed to try and automatically generate LOM data by directly data mining the learning object content and the context in which it is being delivered. This worked only for a small set of fields (less than 25%) and proved to be an error prone process where many of the automatically generated fields disagree with the values set by content experts.

In addition to a lack of instance data, many learning object based products released have poor support for the full LOM. Expressed in the Learning Object Metadata Best Practices guide, "Many vendors expressed little or no interest in developing products that were required to support a set of meta-data with over 80 elements…[and the] burden to support 80+ meta-data elements on the first iteration of a product is too great for most vendors to choose to bear". [3] The end result is partial implementation, where many of the more complex fields (which happen to be the most useful for dynamic courseware generation agents, as they relate a learning object directly to domain vocabularies) are discarded, and only the simple fields (such as title, or description) are kept.

While the goal of the LOM was to make data available to both human interpreters (generally teachers and instruction designers) and computer agents, the standard provides several examples of poor data typing, leading to potentially ambiguous situations. Some of our previous work has identified that the version and lifecycle elements of the LOM are generally stored as arbitrary human readable text, and are thus unreliable for automatic processing [5]. Further, Friesen indicates that even those elements that are required by the standard to be strictly data typed are often not, as was the case of vCard contact information, where none of the documents in a 3,000 instance test set were found to conform [11].

In addition to internal issues with applying the metadata standard, there are external issues. In particular, most learning object repositories allow for only a single metadata instance to be associated with a learning object. Anecdotal evidence observed in assigning metadata to a set of computer science learning objects [9] suggested that inter-rater reliability is often quite low, especially when only a few fields are chosen by authors. This appears to be a general trend in educational metadata, whether the resource being described is a tutorial, discussion thread, or other digital artefact. By restricting learning objects to single instances of metadata, repositories are significantly limiting both the quality and quantity of information that can be expressed about a given resource.

2. The Ecological Approach

We are working on implementing an alternative theory of metadata, called the *ecological approach*, to overcome the defficiencies present in the standards based approach. In the ecological approach the e-learning system keeps a learner model for each learner, tracking characteristics of the learner and information about the learner's interactions with the learning objects they encounter. After a learner has interacted with a learning object, the learning object is associated with an instance of the learner model. The information in such a learner model instance can include

- information about the learner, including cognitive, affective, and social characteristics and their goal(s) in accessing the content;
- information about the learner's perspectives on the content itself, including the learner's feedback on the content, the learner's knowledge of the content (as determined, for example, by a test administered during the learner's interactions with the learning object);
- information about how the learner interacted with the content, including observed metrics such as dwell time, number of learner keystrokes, patterns of access, etc.;
- information about the technical context of use, including characteristics of the learner's software and hardware environment;
- information about the social context of use, including links to the learner model instances attached to learning objects previously encountered by the learner.

Over time, each learning object thus slowly accumulates learner model instances that collectively form a record of the experiences of all sorts of learners as they have interacted with the learning object. The collected learner model instances can then be inspected for patterns about how learners interacted with the learning object, for example that learners whose knowledge has been evaluated as weak did not have long dwell times, or that learners with certain cognitive characteristics did well. The sequence of learner model instances for a particular learner forms a "learning trail" through the learning object repository, and this trail can also reveal interesting patterns of success and failure for the learner.

There are an enormous number of patterns that can be found when inspecting actual learner behaviour. The key to finding meaningful patterns is the *purpose* (in the sense of [20]) for which the patterns are sought. Each such purpose places its own particular constraints on what patterns are meaningful, how to look for these patterns, and how to use what these patterns reveal in order to achieve the purpose. Thus, determining whether to recommend a specific learning object to a particular learner may require comparing this learner to other learners on important characteristics and then looking at how similar learners have evaluated (or been evaluated on) the content (and, moreover, the characteristics considered to be important are themselves determined by the learner's own goals). On the other hand, determining whether a learning object is now obsolete may require an examination of all learners' evaluations of the content, trying to extract temporal patterns in the evaluations that show how recent learners like or dislike the content. The key point is that it is the purpose that determines what information to use and how it is to be used. An ideal goal for a real time e-learning system is that this determination be made *actively* (in the sense of [15]) at the time the purpose is invoked, so that no *a priori* interpretation needs to be given to the information; however, time constraints on executing the data mining algorithms may mitigate against such real time computation in many circumstances.

In sum, then the ecological approach promotes the notion that information gradually accumulates about learning objects, the information is about the use of the learning object by real learners, and this information is interpreted only in the context of end use. The approach is ecological because over time the system is populated with more and more information, and algorithms emulating natural selection based on purposes can determine what information is useful and what is not.

There are many possible applications for the ecological approach in e-learning. The approach could underlie the design of

- a study aid, for example to retrieve for a learner relevant papers from a cache of such papers for a graduate student trying to learn about an area of research (e.g. [19]);
- a recommender system, to recommend some content to a learner that is relevant to his or her current task (e.g. [16]);
- an instructional planner, to plan out a sequence of content pages of relevance to a learner, sort of an individualized curriculum of study;
- a group formation tool, to suggest to the learner a group of other learners relevant to solving a particular task or learning about a particular subject (e.g. [22]);
- a help seeker, to find another learner who can help the learner solve a problem he or she has encountered (e.g. the I-Help system [12]);
- a reminder system, to keep a learner updated with new relevant information, say from the web, that is relevant to the learner's goals;
- an evaluation tool, to allow learners' interactions with educational content to be studied by instructional and cognitive scientists, in particular to look at the experiences of all learners or particular types of learners with some educational content;
- an end-use tagging system, to automatically derive educational content tags from pre-established ontologies based on the experiences of the actual users of the content, and that can be parameterized by end use variables such as type of learner, success/failure of the educational content for each type of learner, etc. A variant of this is possibility is the ability to refine, modify, or change pre-assigned metadata based on inferences from end use;
- an "intelligent" garbage collection system, to determine the on-going relevance of educational content and, if necessary, to suggest modifications or even that it be deleted as no longer being useful to learners (e.g. as discussed in [4]).

3. A Semantic Web Approach to Supporting the Ecological Approach

3.1 Introduction

As described more fully in [6], the Department of Computer Science at the University of Saskatchewan has developed a set of e-learning applications which includes both a discussion forum system (asynchronous and synchronous), as well as a learning object-based content management system. Each of these systems is connected to the Massive User Modelling System (MUMS) [7] – a piece of semantic web based middleware which allows these systems to create small packages of learner modelling information called events. These events are marked up using the Resource Description Framework (RDF), and correspond to one or more RDF schemas[1]. Events are then forwarded from e-learning applications to higher level applications, in particular software agents, where they can be analysed and acted upon.

Our initial work in applying the ecological approach to learning object metadata is based primarily around our content management system, the iHelp LCMS. This system delivers standard IMS Content Package [18] formatted learning objects to learners, and typically includes text, video, and interactive exercises. In addition to reading the content, learners must complete a short quiz both before and after the learning object is delivered. This quiz contains multiple choice questions which are related to the content that is being taught. Each question/answer pair in a quiz is mapped to a particular domain concept expressed in our domain ontology, as well as an entry in an educational objectives ontology. Our domain ontology is a large (1,000+ node) RDF graph that represents the relationships between concepts covering basic computer science for non-majors, focusing on web technologies (HTML, ECMAScript, etc.) and the history of computer science. Our educational objectives ontology is based on the work done by Anderson et al. [2], which itself is built off of work done by Bloom et al. [1], and indicates the depth of cognition a student has demonstrated in a given topic.

Consider the following example taken from a lesson on operators:

> Question: What is the result of the operation ((2<9) && (3>2))?
> i) true
> ii) false

If the learner answers true, it shows they can understand procedural knowledge in both of the topics "RealtionalOperators" and "LogicalOperators". If the learner answers false, they have not demonstrated any knowledge or ability in particular. Using the case of the former as an example, the results can then be expressed in RDF (shown graphically in figure one). It is worth noting that the content management system itself knows only about the user and the question/answers he or she has submitted (the left hand side of the figure) – semantic web rules can be used to value-add the RDF with a derived understanding of the competencies a student has gained after the fact.

---

[1] Examples of these schemas are available online at http://ai.usask.ca/mums/best_practices

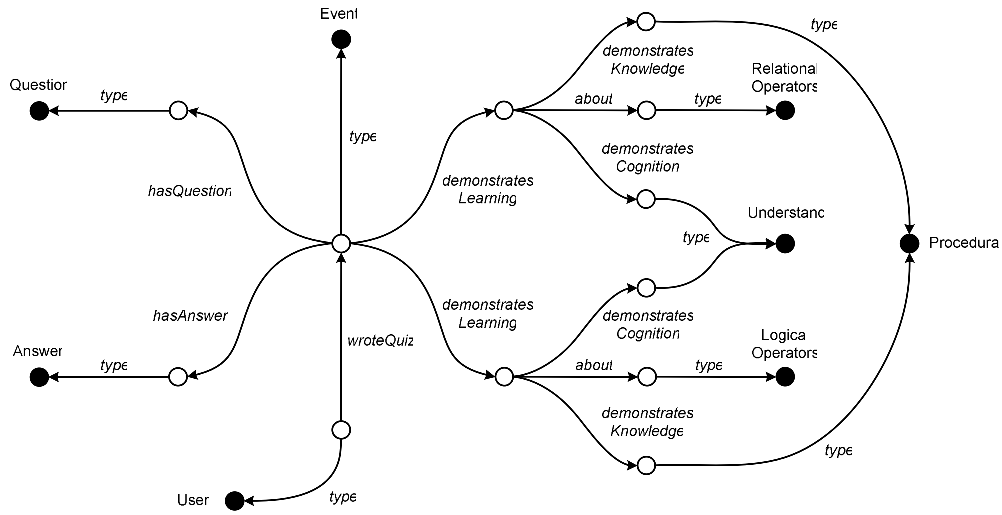


Figure 1: Graphical representation of RDF Model. Empty circles are instances while filled in circles are classes. Namespace prefixes and instance values have been omitted to aid in readability.

This form of collecting student competencies is relatively agile – there is no need to indicate "correct" and "incorrect" answers, instead, the instructional designer can indicate explicitly which answers demonstrate which competencies. This is especially useful in multiple choice tests where there is often one best answer, but several other options still demonstrate some smaller set of that knowledge. Further, the open world model of RDF allows for an arbitrary number and type of statements to be made about a students' interaction with a quiz. This is useful in making multiple assertions about student knowledge (as in above). We anticipate that associating known misconceptions (a form of bug libraries) with given answers will allow us to further value-add the learner model.

Once assessment data has been collected, it can be attached to the learning object. Instead of just associating the raw data with the object, the data can be summarized by subtracting all of those competencies demonstrated in the pre-test from those demonstrated in the post-test. This then shows the net gain in knowledge the student achieved by interacting with the learning object.

3.2 Issues with this Approach

A significant challenge within the educational technology community is in making e-learning artefacts interoperable. This challenge is the primary reason standards such as the IEEE LOM exist. The problem is that once an implementation deviates from a standard (as in our approach), interoperability begins to get severely hampered. To combat this, we anticipate that agents native to our metadata repository will be needed to reason over and summarize metadata to convert it to a more standardized form for export. Unfortunately, this is a lossy process, as in many instances summarized data cannot contain multiple records, and the new semantics we would like to introduce (through deeper user modelling) are meaningless to external repositories.

During the 2004-2005 regular session we began implementing this approach and collected assessment data from approximately 50 students using our online course, as well as feedback from the instructor. During this time it became apparent that students were frustrated with the pre-tests as they often lacked sufficient knowledge to understand the question being asked. We are addressing this by changing the pre-test from actual assessment to a declaration of self knowledge, where students would indicate what they felt their level of knowledge in each topic is using a likert scale. Values on this scale would then be mapped to approximate entries in the educational outcome taxonomy being used. In addition, there is a delicate balance between asking too many questions, risking that some of them may be off topic, and not asking enough, thus missing potential useful metadata entries. This is a trade off we are trying to mitigate by working closely with the instructor and instructional designers for the course.

Finally, it should be noted that while this technique allows for the collection of metadata about a given learning object, it does not in and of itself predict if that learning object is going to be useful for a given future learner. It is likely that we will investigate the use of probabilistic models, such as those presented in [23], when actually constructing an automated instructional planner.

4. Conclusions

Metadata specifications, in particular the nearly ubiquitous IEEE LOM standard, have yet to be proven effective at capturing enough metadata at an appropriate level to be used by automated instructional planners. In an attempt to bring automatic instructional planning to our online courseware, we have proposed a more lightweight metadata collection method, dubbed the *ecological approach*. In this approach arbitrary metadata statements are made about learners and attached to them in the form of a learner model. As a learner interacts within the learning environment, their model is associated with the learning artefacts that exist. In this way metadata is descriptive (based on actual observed interactions), as opposed to prescriptive (assigned by a content expert).

To being to concretize this approach we have started to collect competency lists for each learner both before and after they interact with a learning object, by way of pre and post quizzes. The difference between these lists results in the topics that the given learning object has taught to that particular learner. By associating these lists with the learning object, we are able to form a corpus of evidence that can be used by an instructional planner when sequencing objects together for other learners.

It should be noted that this is just one way in which we are applying the ecological approach. A larger research agenda built around our learner modelling middleware, the Massive User Modelling System (MUMS) [7] is also being pursued. In this we are capturing student interaction with both synchronous and asynchronous discussion forums, such as postings read, time dwelt on a posting, chats participated in, and general availability online. While an immediate end-goal for capturing this is to augment our peer help system (as described more fully in [21]), we also intend to associate this semantic data with learner models, which will then be attached to various artefacts in our systems. We anticipate that this, as well as the extra user interaction information that we are capturing from our content management system (e.g. which learning objects were read, how long they were read, what order they were read in, etc.) will prove useful when trying to adapt learning resources (from peer helpers to traditional learning objects) for personalized instruction.

References

[1] B.S. Bloom ed., *Taxonomy of Educational Objectives, The classification of educational goals, Handbook I, Cognitive Domain*, David McKay Company Inc., 1956.

[2] L. Anderson and D. Krathwohleds., *A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives*, Addison Wesley Longman Inc., 2001.

[3] *IMS Learning Resource Meta-data Best Practices and Implementation Guide, Version 1.1*, IMS Global Learning Consortium Inc., 2003

[4] Bannan-Ritland, B. , Dabbagh, N., and Murphy, K., "Learning Object Systems as Constructivist Learning Environments: Related Assumptions, Theories and Applications.," In The Instructional Use of Learning Objects (on line version). AIT/AECT, 2000,

[5] C. Brooks, *Versioning of Learning Objects*, master's thesis, Saskatoon, SK, Canada, University of Saskatchewan, 2005.

[6] C. Brooks, L. Kettel, and C. Hansen, "Building a Learning Object Content Management System" *World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education* (E-Learn 2005), Association for the Advancement of Computing in Education (AACE), 2005. In submission.

[7] C. Brooks et al., "The Massive User Modelling System" *7th International Conference on Intelligent Tutoring Systems* (ITS04), Springer-Verlag, 2004, pp. 635-645.

[8] K. Cardinaels, M. Meire, and E. Duval, "Automating Metadata Generation: the Simple Indexing Interface" *The 14th International World Wide Web Conference 2005* (WWW 2005), International World Wide Web Conference Committee (IW3C2), 2005.

[9] J. Cooke et al., "Computer Science Tutorial Page"; http://www.cs.usask.ca/resources/tutorials/csconcepts/index.html.

[10] *Shareable Content Object Reference Model (SCORM) Content Aggregation Model (CAM) Version 1.3.1*, Advanced Distributed Learning, 2004

[11] N. Friesen, Final Report on the "International LOM Survey", tech. report Document 36C087, Canadian Avisory Commiittee for ISO/IEC JTC1/SC36, 2004.

[12] J. Greer et al., "The Intelligent Helpdesk: Supporting Peer-Help in a University Course" *Fourth International Conference on Intelligent Tutoring Systems* (ITS 1998), Dpringer-Verlag, 1998, pp. 494-503 .

[13] *IMS Learning Design Information Model*, IMS Global Learning Consortium Inc., 2003

[14] *IEEE P1484.12.1-2002, Draft Standard for Learning Object Metadata*, IEEE, Inc., 2002

[15] G. McCalla et al., "Active Learner Modelling" *Intelligent Tutoring Systems 2000* (ITS2000), 2000.

[16] M. Recker and D. Wiley, "A Non-Authoritative Educational Metadata Ontology for Filtering and Recommending Learning Objects." *Interactive Learning Environments Journal: Special Issue on Metadata*, 2001, pp. 1-17.

[17] *Simple Query Interface (SQI) for Learning Repositories, Version 0.8*, 2004

[18] *IMS Content Packaging Information Model, Version 1.1.4*, IMS Global Learning Consortium Inc., 2004

[19] T. Tang and G. McCalla, "Smart Recommendation for an Evolving E-Learning System" *Workshop on Technologies for Electronic Documents for Supporting Learning, International Conference on Artificial Intelligence in Education* (AIED 2003), 2003.

[20] J. Vassileva, G. McCalla, and J. Greer, "Multi-Agent Multi-User Modeling in I-Help" *User Modeling and User-Adapted Interaction: Special Issue on User Modelling and Intelligent Agents*, vol. 13, no. 1, 2002, pp. 1-31.

[21] M. Winter, B. Daniel, and C. Brooks, "Towards Automatic Discovery of Peer Helpers from an Large Message Board System" *Workshop on Usage Analysis in Learning Systems, the 12th International Conference on Artificial Intelligence in Education* (AIED 2005), 2005. In submission.

[22] M. Winter and G. McCalla, "An Analysis of Group Performance in Terms of the Functional Knowledge and Teamwork Skills of Group Members" *Workshop on User and Group Models for Web-based Collaborative Environments, 9th International Conference on User Modeling* (UM 2003), pp. 35-45.

[23] J.-D. Zapata-Rivera and J. Greer, "Inspectable Bayesian student modelling servers in multi-agent tutoring systems" *International Journal of Human-Computer Studies*, vol. 61, no. 4, 2004, pp. 535-563.

A Semantic Web System for Helping Teachers Plan Lessons Using Ontology Alignment

Toshinobu KASAI [†] Haruhisa YAMAGUCHI [†]
[†] *Faculty of Education,*
Okayama University, Japan
Kazuo NAGANO [††] Riichiro MIZOGUCHI [‡]
[††] *Department of Education,* [‡] *The Institute of Scientific and Industrial Research,*
University of the Sacred heart, Japan *Osaka University, Japan*

Abstract. In Japan, there are very few specialist teachers of IT education, though interest in IT education has continued to grow. And, it is difficult for them to gain the necessary knowledge and skills, since the educational goals and techniques of IT instruction are not yet clearly defined. So, we built ontology of the goal of IT education and its applications based on Semantic Web technology. This application was based on the alignment of ontologies to reuse the result of other research, and we have already described the outline of this application in the different paper. In this paper, we focus on technical details based on Semantic Web technology to realize this application.

Introduction

As a result of the widespread use of the Internet and the development of numerous large information systems, the necessity and importance of IT education have increased. However, there are very few specialist teachers who have the specific skills for teaching IT. Further, it is difficult for them to gain the necessary knowledge and skills, since the educational goals and techniques of IT instruction are not yet clearly defined. For example, most of the teachers who are not specialists mistakenly believe that the use of the technology itself is the main goal of IT education, though the ability to use information systems is a more complex and indispensable aspect of IT education.

Many instructors and researchers have published their opinions on various concepts of IT education and the relationships between these concepts [4]. However, it is difficult for teachers who are not specialists to understand and apply them. And, many organizations provide web pages that show teachers of IT education various useful resources--e.g., digital content, lesson plans, and Q & A [2], [3]. However, it is very difficult to collect the necessary resources for teachers because the relevant web pages are too numerous, and their formats and viewpoints are not unified even when the resources have the same purpose.

One of the causes of these problems is that various concepts related to IT education and practical skills are not yet clearly defined. Because most of the guidelines and commentaries about IT education present the concepts in a disorganized fashion, we believe that these concepts are not conveyed to teachers effectively. So, we built ontology which defines the concepts related to the goal of IT education. This ontology consists solely of concepts of the goal of IT education, and ensures that no confusion of various concepts occurs. However, because the ontology is quite abstract, we think that it is not effective to directly provide teachers with it. To solve this problem, we proposed a framework that can reuse results of other research [4]. That research clarified concepts of goals of IT education and provided teachers with them. The characteristic of this classification is that it is easy for teachers to understand, though from an ontological engineering viewpoint that may need modification. So, by aligning our ontology and this classification, it is possible to provide teachers with the integrated benefits of both classifications. Moreover, we built a Semantic Web application based on this framework to provide teachers of IT education with useful resources from the various viewpoints they require.

We have already described our ontology and its application in other papers [1], [6]. However, in these papers, we focused on only the contents and functions of this application and did not focus on technical viewpoints. So, in this paper, we focused on technical details based on Semantic Web technology for realization of our application.

1. An Outline of Our Approach

In this section, we describe the outline of our system developed using Semantic Web technology.

1.1 Outline of the system architecture

The purpose of our project is to provide teachers of IT education with useful resources in accordance with the various viewpoints that they might have. The architecture of our system is shown in Figure 1. This framework includes two metadata: one is based on our ontologies, which are described in another paper in detail [1]. We authored metadata of various resources about IT education in RDF using the ontology of the goal of IT education and the ontology of the fundamental academic ability as the tag. The other is based on the Goal List of IT education [4], which was taken from other research which has been conducted by one of the authors for years.

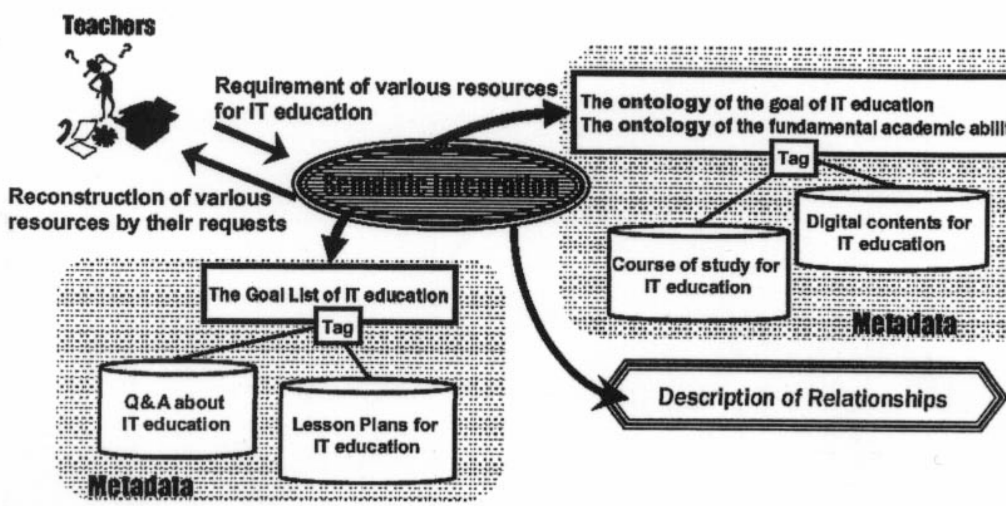


Figure 1. The outline of our approach that is compliant with the *openness* of the Semantic Web

The purpose of this Goal List is to provide teachers with viewpoints from which to evaluate the learner's activity during instruction in IT education. Because this Goal List was not generated based on the ontology theory, its quality is not as high as that of an ontology [1]. However, this Goal List already has been so widely used with the same purpose as an ontology that many information resources that support teachers for IT education in Japan are annotated using it. Therefore, in this paper, we regard this Goal List as an ontology.

In this study, we realize semantic integration between the metadata based on separate ontologies by describing relations between our ontologies and the Goal List clearly. For example, in this framework, the system can reconstruct lesson plans tagged based on the Goal List from the viewpoint of our ontologies and provide with them. In addition, the system can integrate lesson plans based on the Goal List with digital contents based on our ontologies which are able to be used in each step in them. With this framework, it becomes possible for teachers to use many useful resources more effectively for a wider range of purposes.

In this paper, we describe the alignment of our ontologies and the Goal List which is the base of this framework and an example of application services based on it from the technical viewpoint.

1.2 The Layered Structure of Our Semantic Web Application

In this section, we describe the outline of the alignment of ontologies of our Semantic Web Application. The layered structure of this part is shown in Figure 2.

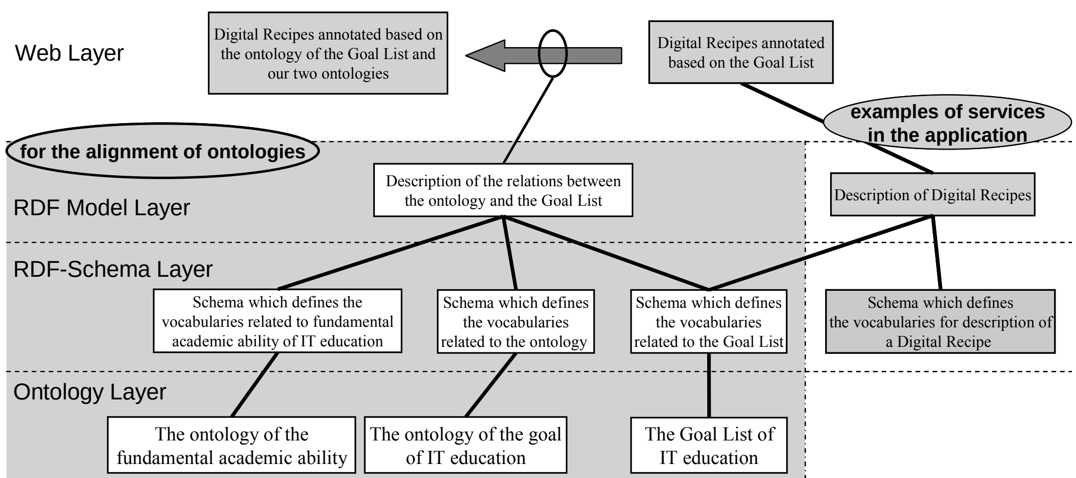


Figure 2. The Layered structure of part concerned with the alignment of ontologies of the Application

This application is constructed in four layers. The bottom layer is the ontology layer in which we define all of the concepts related to our two ontologies and the Goal List of IT education. The second layer is the RDF-Schema Layer in which the vocabularies of classes and properties used in the third layer, the RDF Model Layer, are defined. In the part for the alignment of ontologies, there are three schemata in this layer. As these schemata, the vocabularies of classes and properties related to our ontologies and the Goal List defined in the bottom layer are defined. The third layer is the RDF Model Layer in which, for the alignment of ontologies, we authored metadata of a resource that shows the relations between our two ontologies and the Goal List of IT education by using the vocabularies defined in the RDF-Schema Layer.

Thanks to this framework, we can reuse resources which were annotated based on the Goal List in our application. And, the application can provide teachers with the integrated benefits of both ontologies. As an example of resources, in this paper, we take up simple lesson plans on the Web (called Digital Recipes) provided by Okayama Prefecture Information Education Center [2]. These Digital Recipes are open to the public as resources related to concepts of the Goal List. In this example, by this application, for each step in a flow of the instruction, the viewpoints of evaluation are provided with expression that is easy for teachers to understand these meaning as the benefit of the Goal List, and the goals of IT education, which is easy to be hidden in its shadow as the benefit of the ontology, are provided.

2. Details of Each Layer for Realization of Our Semantic Web Application

In this section, we describe the detail of each layer which was shown in the above section for realization of the alignment of ontologies.

2.1 Three Ontologies in the Ontology Layer

In this section, we describe the outline of these ontologies which are used in our Semantic Web application briefly, because we have already described them in other contributions [1], [6]. And, we show the advantages of our ontologies by comparison with the other classification from the viewpoint of the ontology theory.

A portion of the *is-a* hierarchy as the ontology of the goal of IT education is shown in Figure 3. This ontology was built on the editor "Hozo" [7], which is an environment for building ontologies. The ontology of the goal of IT education consists solely of the concepts of the goal of IT education. Stratification based on *is-a* relation is the essential property of these concepts, and ensures that no confusion of various concepts occurs; such confusion can obstruct teachers' understanding of concepts of IT education. To build this ontology, we first extracted three concepts that can be the goal of IT education. These are "Knowledge about information", "Skills to utilize information in the information society", and "Independent attitude toward participating in the information society". This classification is compliant with Bloom's taxonomy of instructional objectives [5]. Furthermore, we classified these three concepts into more specific ones.

We extracted and classified the goal of the "Period of Integrated Study" as the ontology of the fundamental academic ability [6]. The "Period of Integrated Study" was created to cultivate ways of learning and thinking and an attitude of trying to solve or pursue problems independently and creatively. The purpose of describing this ability is to show more clearly the essence of the concepts of the goal of IT education by identifying its differences from the academic ability, which is attained in other subjects. For this ontology, we classified three concepts; "Knowledge to live in the society", "Skills to utilize information in the society" and "Independent attitude toward participating in the society", as well as the goal of IT education.

The Goal List has three top-level categories, "Practical Ability of using information", "Scientific understanding of information", and "Awareness toward participating in the information society", which the Ministry of Education prepares in more detail in the same way as our *part-of* hierarchy. For this purpose, examples of more concrete learning activities that are easy for teachers to understand are provided with a level that shows when learners should attain this goal. We think that it is more suitable for teachers' understanding to provide them with information on activities related to concepts of learning. We think that it is easier for teachers to grasp each description when concepts of learning activities are included in the information provided. Further, it is difficult to set a level of difficulty for a goal of IT education without presenting concepts of learning activities. Consequently, the Goal List has many advantages as information that is provided to teachers directly.

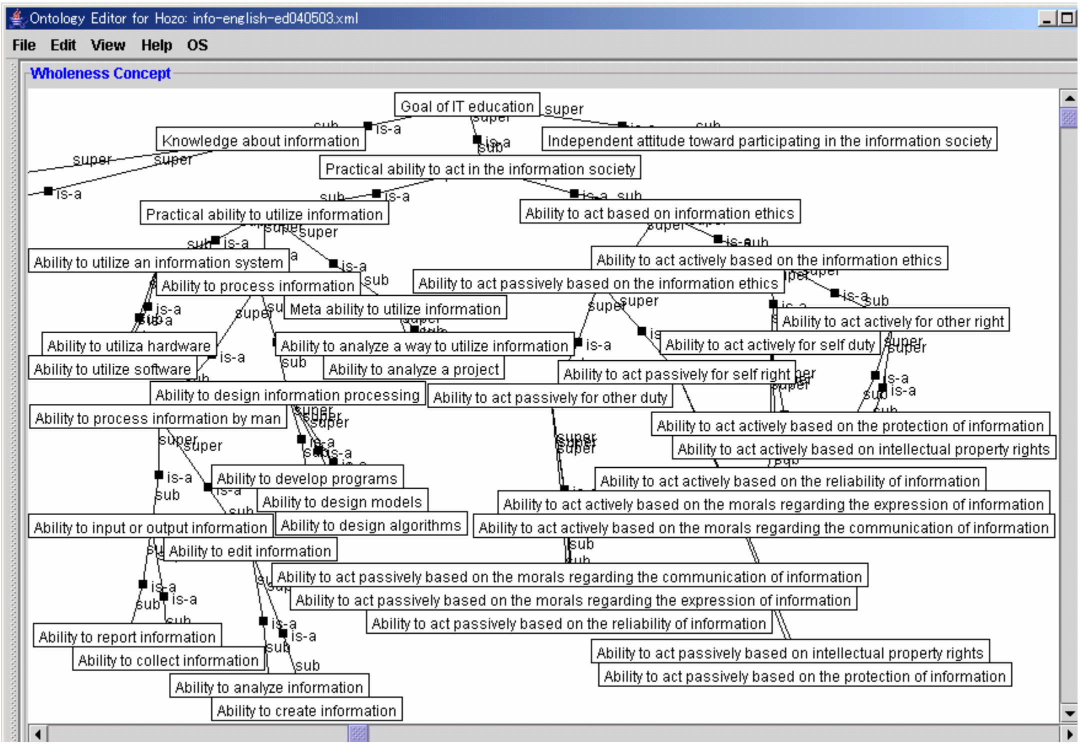


Figure 3. A part of the ontology of the goal of IT education (*is-a* hierarchy)

However, the Goal List has some faults from the viewpoint of classification of the goal of IT education. Although, essentially, the classification of educational goals should be performed by extracting the intrinsic goals that should be attained in education and systematizing them, in many of the current classifications, we find concepts other than goals; for example, learning activity and learning environment related to goals are incorrectly mixed up. Moreover, systematization like that in this example, in which other concepts are mixed, sometimes causes another problem: the extracted concepts are not completely independent of each other.

Given these considerations, to the best of our knowledge, there is no goal classification that properly captures the intrinsic educational goals of IT education without any confusion regarding learning activity, standard of evaluation for education, etc. It is difficult to separate various concepts related to IT education, because most goals of IT education are meta-abilities that are attained in the process of problem solving. Considering the fact that the purpose of the classification of the goal of IT education is to give teachers a clear understanding of the educational goals, our goal ontology is more suitable, based on the fact that it reveals the inherent conceptual structure of educational goals and thereby facilitates a teacher's understanding of those goals.

2.2 Definition of Vocabularies of Classes and Properties in RDF-Schema Layer

In this section, we describe the vocabularies defined in the RDF-Schema Layer for tags that are used in the RDF Model Layer. In this study, we define the vocabularies used to realize the alignment of ontologies in three namespaces.

The vocabularies defined in these three namespaces are shown in Table 1. The first schema which is referred to by the prefix "it_goal" in this study, defines the vocabularies of classes and properties related to the goal of IT education. The second schema which is referred to by the prefix "pre_goal" in this study, defines the vocabularies of classes and properties related to the fundamental academic ability. The third schema which is referred to by the prefix "goal_list" in this study, defines the vocabularies of classes and properties related to the Goal List of IT education. In Table 1, we omit a description of the classes defined in these namespaces, because all classes show concepts of three ontolgies which we explained in the above section.

Table 1. Vocabularies defined in the RDF-Schema Layer

prefix	type	vocabulary	explanation
it_goal	Property	target_it_goal	Target goal as IT education
pre_goal	Property	target_pre_goal	Target fundamental academic ability as goal
goal_list	Property	target_example	Target learning activity that includes same goal
goal_list	Property	example_activity	Example learning activity in the category of the Goal List
goal_list	Property	level_of_difficulty	Level of difficulty
sch_ed	Class	Subject	Subject in school
sch_ed	Class	Unit	Unit for learning content of education
sch_ed	Class	Activity	Learning activity by students
sch_ed	Class	Grade	Grade in school
sch_ed	Property	realization	Educational goal of an ideal to expect realization
sch_ed	Property	goal_Description	Description about target educational goal
sch_ed	Property	grade	Target grade of Subject or Sub-subject in school

In the RDF-Schema Layer, we also define other vocabularies to use in RDF Model Layer for description of other metadata which are used in our application. Here, in Table 1, we show parts of vocabularies which are used for examples of services which are shown in Figure 2. These vocabularies are defined in a namespace which are referred by the prefix "she_ed". This schema defines the vocabularies of classes and properties related to the school education.

2.3 Description of relationships between our ontologies and the Goal List of IT education in the RDF Model Layer

In this section, we explain the RDF Model for the description of resources which show the relationships between our ontologies and the Goal List of IT education. In this study, we describe the relationships of these by giving meaning to concrete learning activities which are provided in the classification of the Goal List based on our ontologies.

In the Goal List of IT education, for this purpose, examples of concrete learning activities that are easy for teachers to understand are provided with a level that shows when learners should attain this goal. Each example of these learning activities is practical activity and contains educational goals that the concepts of our two ontologies explain. We authored metadata of these learning activities in RDF, which belong to the respective concepts of the Goal List, by using the vocabularies defined in the RDF-Schema about the concepts of the ontology of the goal of IT education and the fundamental academic ability. A part of the RDF Model for description of the metadata is shown in Figure 4.

As shown in this model, all learning activities which are provided in the classification of the Goal List are tagged based on the ontology of the goal of IT education and the ontology of the fundamental academic ability. So, the relationships between our two ontologies and the Goal List can be described around these learning activities in this model. Thanks to description of this model, the alignment of our ontologies and the Goal List can be realized, because the metadata which have already been open to the public are authored based on these learning activities in the Goal List by teachers.

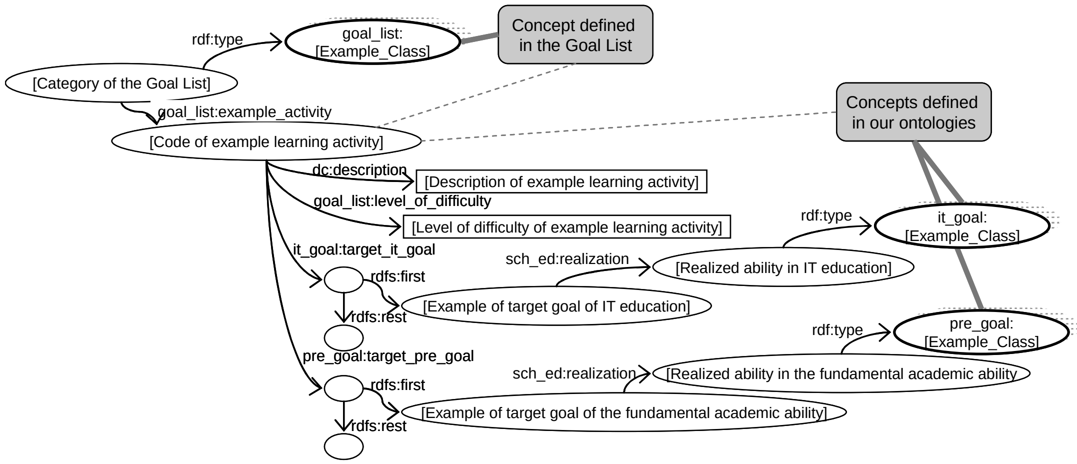


Figure 4. A part of an RDF Model for the description of the relationships between ontologies

3. Details of an Example of the Services of the Application by the alignment of ontologies

In this section, we describe details of an example of the services of the Semantic Web application by the alignment of ontologies. This example is a function which uses the Digital Recipes explained in Section 1. These Digital Recipes show the lesson plans of several classes for a unit. Each plan includes not only the basic information (such as target grade, target subject, aim in the whole classes and so on) but also the flow of concrete learning activities. In the metadata of the Digital Recipes, each learning activity is tagged based on the Goal List. We show an example of web pages of Digital Recipes which are tagged with the Goal List on the left in Figure 5. The services of the application explained in detail in this section is a system that converts the contents of the screen shot on the left in Figure 5 to the contents of the screen shot on the right. Next, we explain the details of processing to realize this service.

A part of description of the metadata of Digital Recipes based on the Goal List which are authored by teachers is shown on the left at the bottom in Figure 5. These metadata are output in RDF automatically from the contents which teachers input on the spread sheet software in the form specified by the macro function. The system analyzes the metadata of a Digital Recipe and extracts concepts of the Goal List tagged in this resource, then the system extracts the concepts of the ontology of the goal of IT education related to those concepts of the Goal List from the other resource (Description of the relations between the ontology and the Goal List) in the RDF Model Layer. The system integrates the original resources with extracted concepts of the ontology and outputs it as an HTML file.

This system can provide teachers with the integrated benefits of both ontologies. In this example, for each step in a flow of the instruction, the viewpoints of evaluation are provided with expression that is easy for teachers to understand these meaning as the benefit of the Goal List, and the goals of IT education, which is easy to be hidden in shadow of it as the benefit of the ontology, are provided.

4. Related Work

Many organizations and researchers have been trying to enhance shareability and reusability of various educational resources. Here, we introduce some of these efforts that are related to our approach briefly.

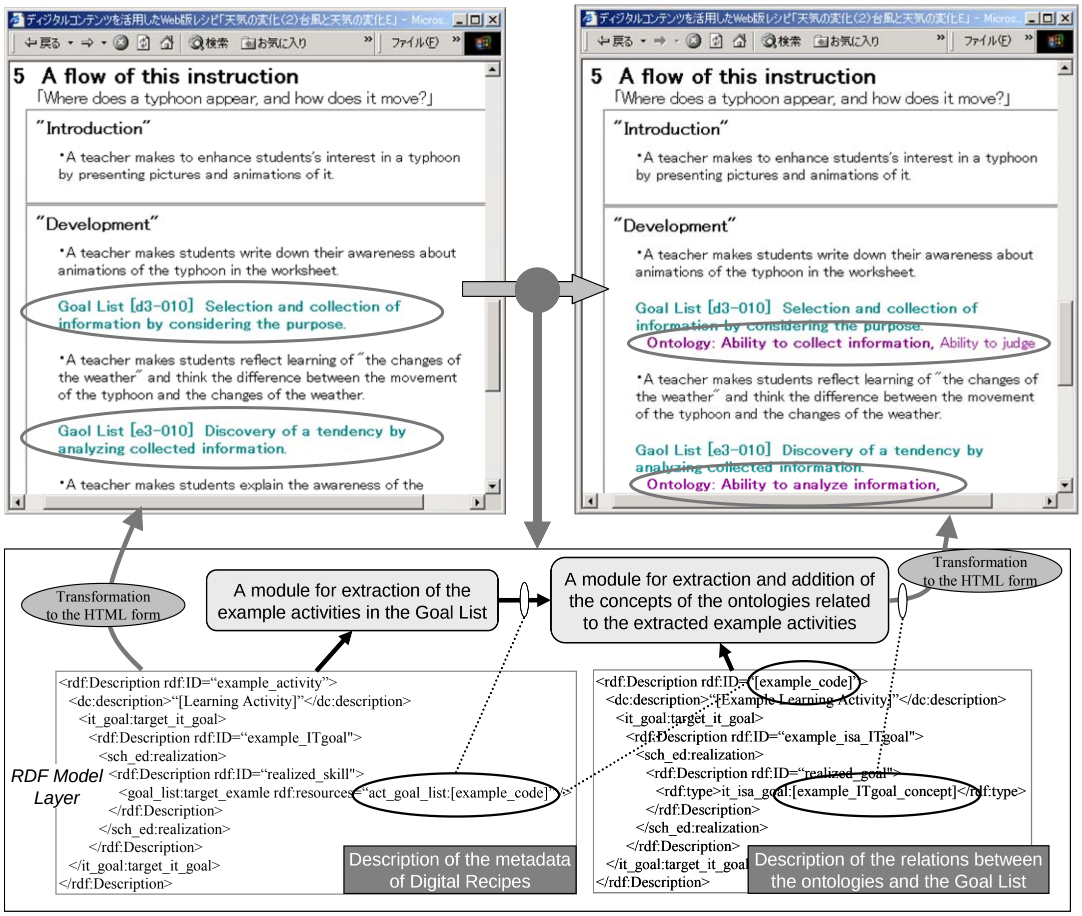


Figure 5. Details of an example of the services of the application by the alignment of ontologies

The Learning Object Metadata (LOM) was provided by The IEEE Learning Technology Standards Committee (LTSC) [8]. The LOM specifies the syntax and semantics of Learning Object Metadata, defined as the attributes required to full/adequate description of a learning object. We cannot describe the contents of the Learning Objects in compliance with the LOM standards because they focus on the minimal set of attributes to allow these LOs to be managed, located, and evaluated in total independence of their contents. Our approach aims at describing the detailed contents by limiting objects concretely such as lesson plans.

There is the IMS Learning Design project which aims at making the standard to describe the instruction/learning activities, the learning environment, and the learning objectives that can be expressed in lesson plan [9]. In compliance with this standard, we can express the contents of lesson plan in detail. However, we think that this expression is too complex for teachers who do not understand the contents and goal of education enough yet. Our approach aims at expressing them with solely educational goal for the teachers who do not understand them.

There are some researches based on these standards and various ontologies [10], [11]. The goal of [10] is to specify an evolutional perspective on the Intelligent Educational Systems (IES) authoring and in this context to define the authoring framework EASE: powerful in its functionality, generic in its support of instructional strategies and user-friendly in its interaction with authors. And, the study [11] proposes a theory-aware ITS authoring system based on the domain and task ontologies of instructional design.

And, there is a research (TM4L) [12] which describes the metadata of digital learning resources based on ontologies as well as us. TM4L describes the metadata of digital course libraries based on the ISO XTM standard (XML Topic Maps) to organize and retrieve information in a more efficient and meaningful way. The difference between our approach and TM4L's approach is that we realize the alignment with other ontology.

We intend to build a support system for designing an instructional system for cultivating practical skills to solve various problems based on the framework which is proposed in this paper with referring to the results of these related works.

5. Summary

We described the Semantic Web system which provides teachers of IT education with useful web resources. This system is based on the alignment of our ontologies and other ontology to reuse the results of other research. In this paper, we focused on the technical details based on Semantic Web technology to realize this system.

In future work, we intend to build the system which supports teachers of IT education dynamically to plan a lesson by describing relations between the concepts of our ontologes systematically.

References

[1] T. Kasai., H. Yamaguchi, K. Nagano, R. Mizoguchi (2004), Building of an Ontology of the Goal of IT Education and Its Applications, In Proceedings of the 3rd International Workshops of Applications of Semantic Web Technologies for E-Learning (SW-EL'04), pp.55-65.

[2] Okayama Prefectural Information Center (2002), Okayama Prefectural Information Education Center, Digital Contents Recipes and Worksheets, http://www2.jyose.pref.okayama.jp/cec/webresipi/index.htm.

[3] NICER (2003), National Information Center for Educational Resources HomePage, http://www.nicer.go.jp.

[4] The Meeting of Tuesday (2001), The Meeting of Tuesday, The Goal List of Information Education, Mail-Magazine of the Meeting of Tuesday, http://kayoo.org/home /project/list.html.

[5] B.S. Bloom, J. T. Hastings, G. F. Madaus (1971), Handbook on formative and summative evaluation of student learning, McGraw-Hill.

[6] T. Kasai., H. Yamaguchi, K. Nagano, R. Mizoguchi (2005), Goal Transition Model and Its Application for Supporting Teachers based on Ontologies, In Proceedings of the Artificial Intelligence in Education (AIED2005), in press.

[7] K. Kozaki, Y. Kitamura, M. Ikeda, R. Mizoguchi (2000), Development of an Environment for Building Ontologies which is based on a Fundamental Consideration of "Relationship" and "Role", The Sixth Pacific Knowledge Acquisition Workshop (PKAW2000), pp.205-221.

[8] IEEE LTSC (2002), IEEE Standard for Learning Object Metadata, http://ltsc.ieee.org/wg12/.

[9] IMS (2002), IMS Learning Design Specification ver.1.0, http://www.imsglobal.org/learningdesign/.

[10] L. Aroyo, A. Inaba, L. Soldatova, R. Mizoguchi (2004), EASE: Evolutional Authoring Support Environment, In Proceedings of the seventh International Conference on Intelligent Tutoring Systems (ITS2004).

[11] J. Bourdeau, R. Mizoguchi (2000), Collaborative Ontological Engineering of Instructional Design Knowledge for an ITS Authoring Environment, In Proceedings of the 6th International Conference on Intelligent Tutoring Systems (ITS2002), pp.399-409.

[12] D. Dicheva, C. Dichev (2004), Educational Topic Maps, In Poster Abstracts of the 3rd International Semantic Web Conference (ISWC2004), pp.19-20.

Ontological Web Portal for Educational Ontologies

Darina Dicheva[1], Sergey Sosnovsky[2], Tatiana Gavrilova[3], Peter Brusilovsky[2]
[1] *Winston-Salem State University, Department of Computer Science*
601 Martin Luther King, Jr. Drive, Winston Salem, N.C. 27110, USA
dichevad@wssu.edu
[2] *University of Pittsburgh, School of Information Sciences*
135, North Bellefield Ave., Pittsburgh, PA, 15260, USA
{sas15, peterb} @pitt.edu
[3] *Saint-Petersburg State Polytechnic University, Intelligent Computer Technologies Dept.*
195251, Politechnicheskaya, 29, St. Petersburg, Russia
tgavrilova@gmail.com

Abstract. In spite of the fact that the field of applying ontological research in education is fairly young it is already quite broad and fuzzy. The set of technologies used and developed there have roots in a variety of diverse areas of information and pedagogical sciences. To facilitate the process of scientific and scholastic search the domain needs to be structured.

This paper presents two main results: an ontological overview of the *Ontologies for Education* field and an initial report on the development of an ontology-driven web portal providing a single network place, where researchers, students, and practitioners can find information about available research projects and successful practices in this field.

Introduction

Among the variety of modern trends in educational technology development the application of ontological research is probably one of the most fashioned and rapidly evolving. After the first dedicated workshop in 1999 [1] seven more workshops [2, 3, 4, 5, 6, 7, 8] and three special journal issues [9, 10, 11] have been brought out. This does not include numerous papers in related conferences, journals, and books. In spite of the fact that the filed is fairly young it is already quite broad and fuzzy. The set of technologies applied and developed there have roots in a variety of diverse areas of information and pedagogical sciences. To facilitate the process of scientific and scholastic search and the research in this area we decided to collect and classify the available information in the field and to use it for building the *Ontologies for Education (O4E)* Web Portal.

A number of papers have been devoted to the analysis of the field providing overviews of its different aspects. Mizoguchi and Bourdeau in their seminal work [12] enlisted a number of challenges that have not been met yet by the AI-ED technologies and proposed a roadmap of how the application of ontological engineering could assist in dealing with those challenges. Similar work is done in [13, 14, 15], as well as in [16] for the more specific domain of Web-based Intelligent Systems. Several overviews of existing tools or created domain ontologies have been also performed. Examples of the former are the overview and comparison of ontology engineering environments [17], and the analysis of semantic annotation tools for learning material made in [18]. An example from the latter group is the overview of ontologies in the domain of engineering design [19].

This paper however intends neither to give a broader overview of ontology-based solutions to existing problems in e-Learning nor to evaluate the developed artifacts. It rather attempts to provide a general *snapshot* of the state-of-the-art research in the filed of applying and developing ontologies for education. It also proposes building of a single web place, where relevant research projects and successful practices are classified and annotated. We consider that such an effort is well-timed, since there is already a critical but not yet excessive mass of information, so it is still possible to handle it. Pretty soon the situation will be much more unfavourable; note that in 2004 alone three special journal issues related to the Educational Semantic Web have been published! [9, 10, 11]. We believe that building of even modest ontology and using it to index the more important available resources (publications, events, projects, groups, etc.) would be of great benefit to the *O4E* community. We further believe that the *O4E* Portal will make it possible for the interested part of the *O4E* community to collaborate on extending and refining the ontology and keeping the resources current. The reported here work is the first step in this direction.

The following discussion is divided into three parts focused on the development process of the *O4E* ontology, the ontology itself, and the *O4E* Web Portal.

1. The *O4E* Ontology Development

Our first goal was to collect and structure the available information related to the use of ontologies in the field of education. When the question for classifying and representing the collected information arose, it was natural to think about building of an ontology.

It is often the case in computer science that the technological developments appear before the theoretical ones. *Ontology engineering* is not an exception of this pattern. Several related standards, languages, and powerful tools have been already developed [20]. However, the (e-learning) ontology designers and practitioners still face major problems and lack of support to develop even simple ontologies. We mean not *syntactic* but *semantic* problems. What ontology designers currently miss are some theory-backed *practical* techniques for manual ontology development. Recent findings in knowledge engineering along with the first results in the educational ontological research [12] can help in tackling this “bottleneck” problem. We tried to use those findings when eliciting, structuring and designing the *O4E* ontology.

In our project, as in many other ontology-based applications, we have to deal with two types of knowledge - *subject domain* and *structure*, which leads to two types of ontologies. A *domain ontology* represents the basic concepts of the domain under consideration along with their interrelations and basic properties. A *structure ontology* defines the logical structure of the content. It is generally subjective and depends greatly on the goals of the ontology application. It typically represents hierarchical and navigational relationships. While a domain ontology can be used as a mechanism for establishing a shared understanding of a specific domain, a structure ontology enforces a disciplined approach to authoring, which is especially important in collaborative and distributed authoring.

The process, named in all methodologies just “create ontology”, is a time- and mind-consuming iterative procedure of *categorization* or *laddering,* together with *disintegratio*n or *detailing*. It is a totally informal analytical design, and output structures are rather subjective and sometimes awkward. The collective work gave us the opportunity to merge personal viewpoints into a common framework.

One of the guidelines with regard to the *structure ontology* relates to the clarity and *mapability* of the structure. Taking into account that our ontology is to be used not only as a knowledge component of an information system but also as a *mind tool* for manual information search and navigation, we tried to follow the principle of good shape (or beauty), which is not new in basic scientific abstraction and modeling (e.g. physics, chemistry, etc.). The most substantial impulse to it was given by the German psychologist Max Wertheimer. His criteria of good *Gestalt* (image or pattern) [21] we partially transfer to ontological design:

- Law of Pragnanz (the law of good shape): the organization of any structure in the nature or cognition will be as good as the prevailing conditions allow. ‘Good’ here means regular, complete, balanced, and/or symmetrical.
- Law of Parsimony: the simplest example is the best (the Ockham’s razor principle); entities should not be multiplied unnecessarily.

In case of building ontological hierarchies we have to keep in mind that a well-balanced hierarchy is much more comprehensible. We enlist below some tips that we consider useful in formulating the idea of “harmony” in a hierarchical structure:

- Concepts of one level should be linked to their parent concept by one type of relationships, for example, “is-a”, “has-part”, etc. This means that concepts of one layer should have similar nature and level of granularity.
- The ontological tree should be balanced, that is, the depth of the paths in the tree should be more or less equal (±2 nodes). This will insure that the general layout is symmetrical. Asymmetry might indicate that the shorter branches are less investigated or the longer ones are too detailed.
- Cross-links should be avoided as much as possible.

When building a structure ontology to be used for information visualization and browsing, it is important to pay special attention to clarity. Minimizing the number of displayed concepts is the best tip according to the Law of Parsimony. The maximal number of branches and the number of levels may follow Miller’s “magical number” (7±2), which is related to the human capacity for processing information [22].

The “beautification” bias is a methodological approach that can help finding points (nodes) of “growth”, “weak” branches, inconsistencies, and excessiveness. But, in fact, specific domain knowledge features may be of higher priority than structure ontology design principles.

2. The *O4E* Ontology

Figure 1 shows the result of our efforts in developing the *O4E* domain ontology. In fact we started with the development of a *partonomy*, using only the “part-whole” relationship type. We plan to consider later the use of other suitable relationships.

The top-level meta-concepts of the domain ontology divide the whole field according to the role ontologies play in the research. When an ontology is considered as an object (the result of an activity) the research is focused on the theoretical and/or practical issues of the ontological engineering that are specific to the educational context. Ontologies might also serve as a technology, facilitating the solution of some educational problems, such as interoperability of knowledge-based systems and components, or assessment of structural knowledge. In this section we summarize the important issues in those two fields. Note that we haven’t included references here; they have been included in the developed *O4E*-based topic map (see Section 3.).

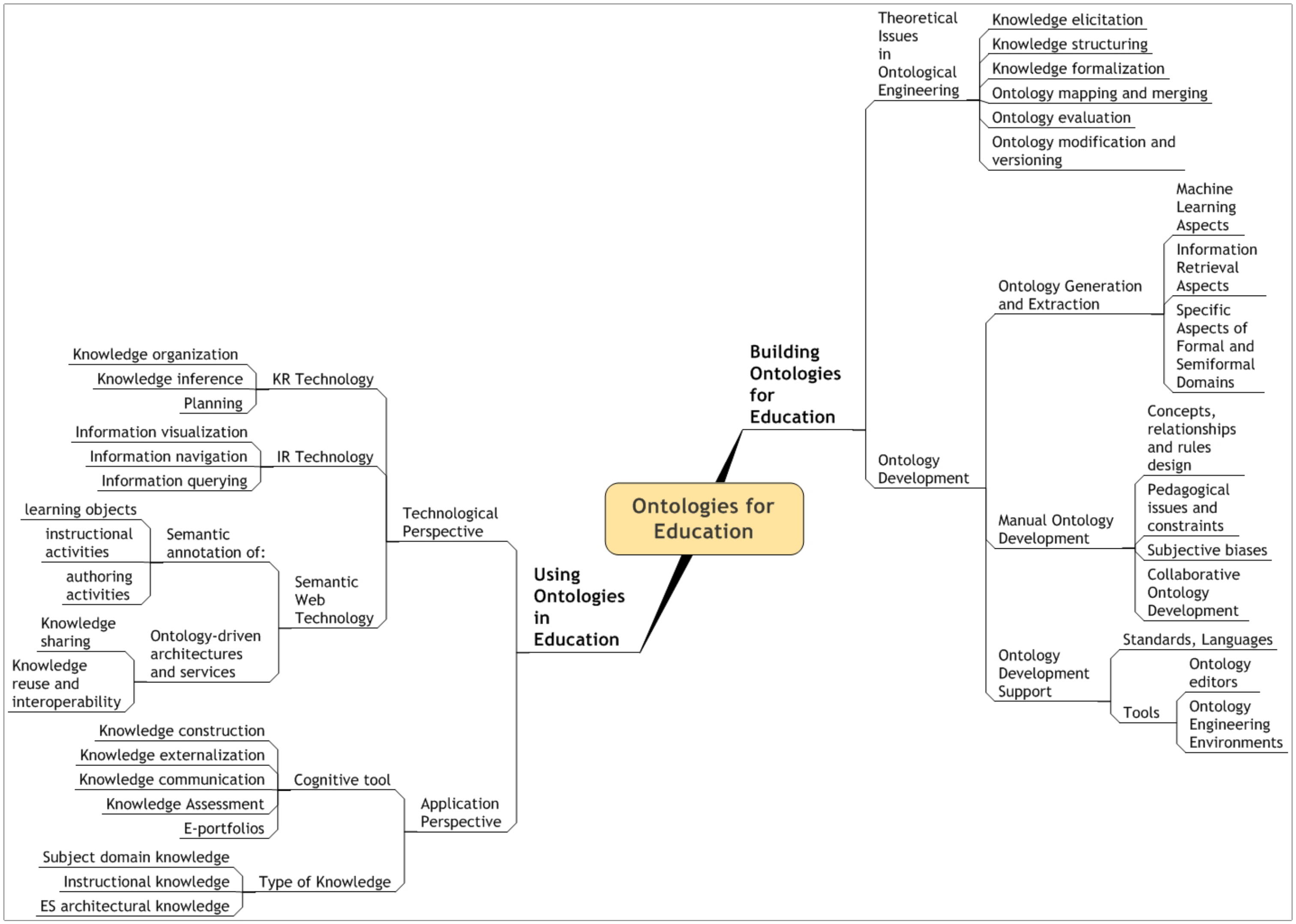


Figure 1. State-of-the-Art Ontology of Ontological Technologies for Education

2.1 Building Ontologies for Education

When analyzing resources focused on different tasks of educational ontology development we identified two naturally separated areas of research. While some papers mostly study the theoretical issues of ontology engineering, another large set of resources is about practical aspects of ontology development. Three large groups could be identified inside the latter part:

- Automatic and semi-automatic ontology generation and extraction using different kinds of sources and different technologies.
- Manual ontology development, where the research is focused on problems either related to the ontology engineering process or specific to educational technology.
- Research on using different standards and languages for ontology implementation, including attempts of binding Semantic Web and educational (e.g. LOM or SCORM) standards or reporting case studies on implementing general-purpose ontological formalisms in educational settings.

2.2 Using Ontologies in Education

This field combines diverse research on different kinds of educational applications of ontologies. We tried to look on this branch from two perspectives depending on what kind of technology is implemented and what role an ontology plays within a project. Speaking about the technological perspective, we defined three main areas, two of which (knowledge representation and information retrieval) are kind of "technological donors" for the ontological research, while the third one (Semantic Web) benefits from it the most.

As to the application perspective, ontologies have been considered for a long time only as a technical artifact acting as a knowledge base component. The field of education is one of the first, where understanding of an ontology as a cognitive tool came around. In many respects it was due to the wide spread of the constructivist paradigm of learning and the broad use of such knowledge technologies as concept maps, mind maps and others for learning purposes.

3. The *O4E* Web Portal

Our second goal was to create a web place not only for publishing the created ontology but also to serve as a point of access to the relevant online information.

The initial idea was to design a website containing a graphical representation of the developed taxonomy along with an index page linking all resource web pages we have found. Figure 2 shows a screenshot of the website.

After creating the first version of the website we felt that we should further represent the ontology in an *exchangeable*, *shareable* and *interoperable* format, so that it can be easily moved, updated, merged, etc., which is *a must* for its further development and survival. We chose to represent it as a Topic Map (TM) [23], since this Semantic Web technology is very appropriate for formalization of lightweight ontologies and for structuring and representing ontology-based web information. For its development we used TM4L, which is briefly described in the following section.

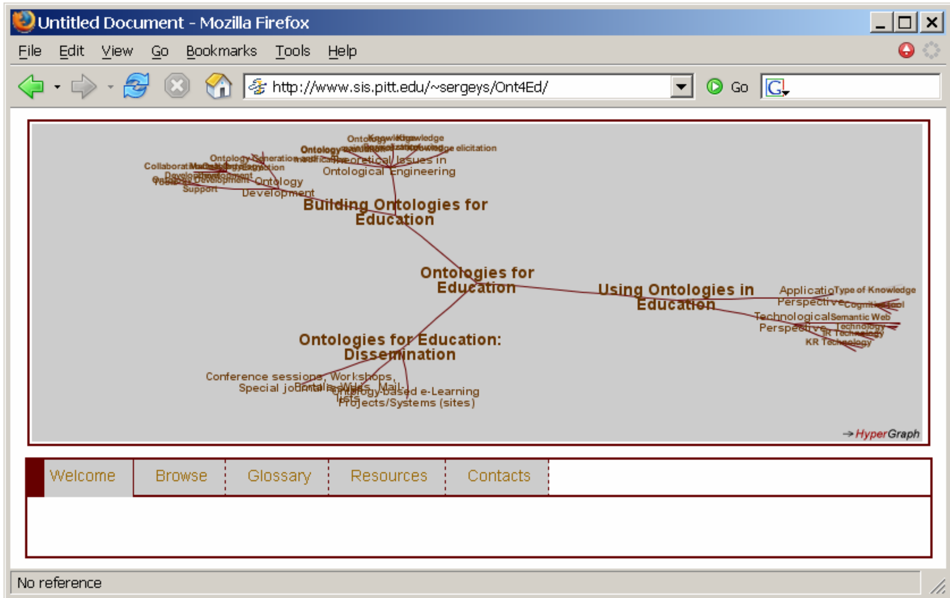


Figure 2. Graphical representation of the developed ontology.

3.1 Creating the Ontology with TM4L

TM4L (Topic Maps for e-Learning) is an environment providing support for creating and using ontology-aware topic maps-based repositories of online (learning) materials [24]. It consists of two tools: a TM Editor and a TM Viewer, both standalone applications. In addition, a client-server topic map navigator (the TM4L Browser) is currently under development. The topic maps created with the TM4L Editor (see Fig. 3) are fully compliant with the XML Topic Maps (XTM) standard and thus interchangeable and interoperable with any standard TM tools (can be browsed with any standard TM browser/viewer). The Editor supports also the TM fundamental feature of easy and effective merging and interrelating of existing information resources while maintaining their meaningful structure. This allows for flexibility and expediency in re-using and extending existing repositories.

The main objects that the TM4L Editor manipulates are topics, representing subjects (e.g. domain concepts), relationships, resources, and scopes (views) [26]. We have currently chosen to use the following types of resources: Research Papers, Special Journal Issues, Books, Projects, Software, Conferences, Workshops, Mail Lists/Wikis, Research Labs and Working Groups, and People. We presume that each resource has a unique URI (e.g. the online paper's URI, the project, conference or person's webpage, etc.) and use these as *subject identifiers* for the resources. This allows merging deferent versions of the *O4E* topic map correctly.

An additional advantage of using the TM4L Editor and Viewer is that they are free standalone personal tools for creating and browsing educational topic maps, so any interested person can download them [25], along with the current version of the XTM representation of the *O4E* ontology for their own use. This would allow personal (individual) collection of information resources and augmentation of the topic map. If a person doesn't change the ontology (i.e. only adds resources and uses the already defined relationship types) and uses the agreed subject indicators, he or she can later contribute the result of their work to the *O4E* community. For that purpose they need to send their XTM file to the *O4E* Portal administrator for merging with the main *O4E* topic map. As we already mentioned, the TM4L Editor supports topic maps merging.

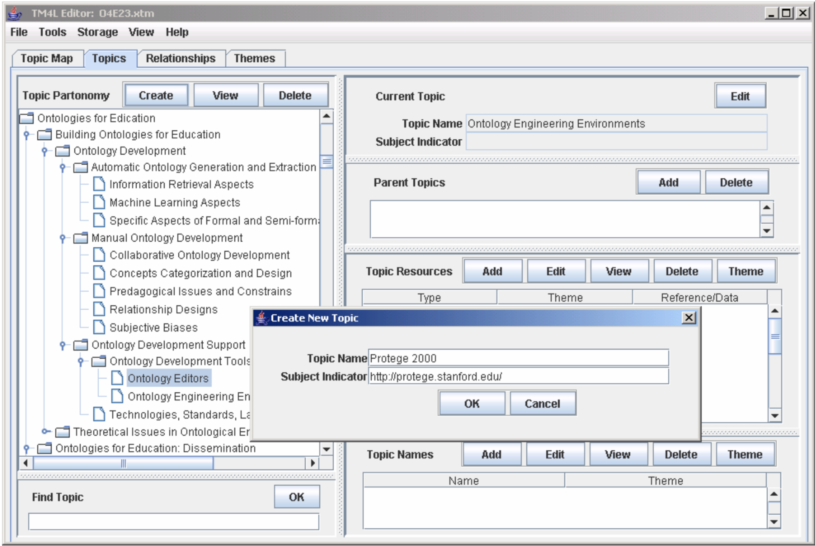


Figure 3. Creating the *O4E* ontology with the TM4L Editor.

3.2. Ontology Navigation

The TM4L Viewer offers three views: *Graph View*, *Text View* and *Tree View* (see Fig. 4). The user can browse the ontology by selecting an object (topic, relationship, relationship type or role) related to the currently displayed one. When navigating, the user can choose in which panel the information about the selected topic to be displayed. This allows browsing of different objects related to the current one without loosing the focus.

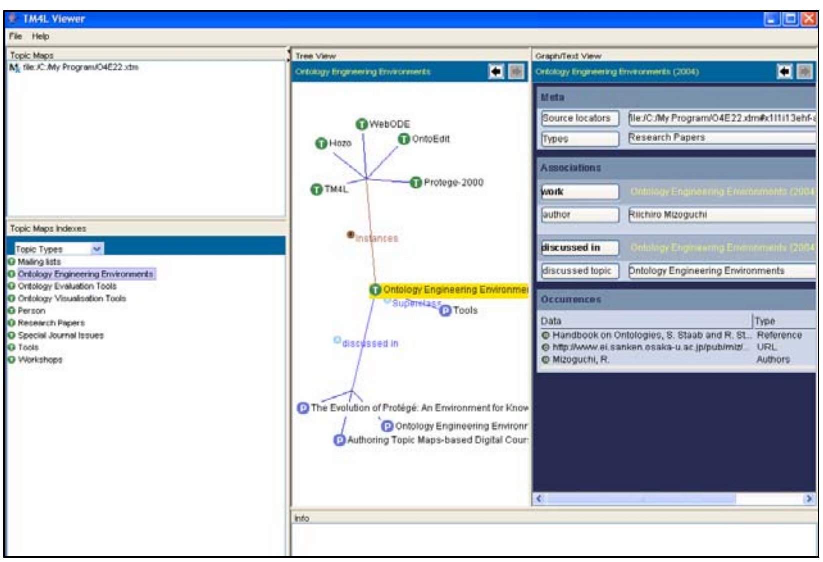


Figure 4. Browsing the *O4E* ontology in the TM4L Viewer.

When the user selects a particular resource type all instances of that type, for example, all workshops, special journal issues, etc. will be listed in the Tree view. By further selecting a particular resource, the user can see the topics (concepts) it is related to, as well as its content, or can follow the link in case it is an external resource.

4. Future Work and Discussion

We are convinced in the usefulness of the *O4E* Web Portal for the *O4E* community and plan to continue the project in several directions.

First, we need to refine further the designed partonomy. The preliminary results of web resource indexing show that some leaf concepts participate in a large number of recourses (that is, incorporate large amount of research), while other nodes could hardly be used for indexing of a single resource. Consequently, such "outlying" concepts need to be redesigned. Too 'rich' concepts (like *subject domain knowledge*) should be further divided into smaller ones, while some concepts lacking relevant projects might be combined with their neighbors.

We also plan to develop further the *O4E* Web Portal, focussing on the possibility of remote online editing of the ontology and resources.

Adding new resources to the *O4E* Web Portal is another direction of future work. Several more related workshops and conferences are being held this year and presumably more special journal issues are coming (we currently know about one [27]). As the number of resources grows, the problem of navigation throughout the portal will arise. To facilitate resource navigation the resources have to be carefully organised and structured. For example, a paper is a part of a workshop, and a person is a participant of some research group, etc. However, when the number of resources gets too large, only structuring would not help to avoid a navigational burden; then adaptive hypermedia technologies (like social navigation) might be applied.

We believe that the SW-EL@AE-ED'05 workshop participants will contribute to the development of the *O4E* Web Portal by providing feedback on the structure of the proposed ontology and by proposing the inclusion of new resources. We hope that the portal will take the role of a platform for collaborative community building of ontology-based repository of *O4E* resources.

References

[1] Workshop on Ontologies for Intelligent Educational Systems, in conjunction with AI-ED'99, Le Mans, France, July 18-19, 1999 (available at: http://aied.inf.ed.ac.uk/abstracts/Vol_10/ontologies.html).

[2] Workshop on Concepts and Ontologies in Web-based Educational Systems, in conjunction with ICCE'02, Auckland, New Zealand, Dec. 3-6, 2002 (available at: http://icce2002.massey.ac.nz/workshop_4.html).

[3] Workshop on Semantic Web for Web-based Learning, in conjunction with CAISE'03, Klagenfurt/Velden, Austria, June 2003 (available at: http://www.sw-wl03.bessag.net/).

[4] Workshop on Applications of Semantic Web Technologies for Educational Adaptive Hypermedia, in conjunction with AH'04, Eindhoven, The Netherlands, August 23-26, 2004 (available at http://www.win.tue.nl/~laroyo/swel/2004/swel-ah.html).

[5] Workshop on Applications of Semantic Web Technologies for Web-based ITS, in conjunction with ITS'04, 30 August – 03 September 2004, Maceió-Alagoas, Brazil (available at: http://www.win.tue.nl/~laroyo/swel/2004/swel-its.html).

[6] Workshop on Applications of Semantic Web Technologies for E-learning, in conjunction with ISWC'04, Hiroshima, Japan, November 7-11, 2004 (available at: http://www.win.tue.nl/~laroyo/swel/2004/swel-iswc.html).

[7] Workshop on Learning Design and Topic Maps, January 26-27, 2005, Oslo, Norway (http://www.estandard.no/ld/)

[8] Workshop on Applications of Semantic Web Technologies for E-learning, in conjunction with ICALT'05, Taiwan, July, 2005 (http://www.win.tue.nl/SW-EL/2005/swel05-icalt05/swel05-icalt05.html).

[9] Special Issue on "Ontologies and the Semantic Web for E-learning" of International Journal on Edication Technology and Society, Vol. 7, Issue 4, 2004, Sampson, D., Lytras, M., Wagner, G. and Diaz, P. (Editors) (available at: http://www.ifets.info/others/issues.php?id=25).

[10] Special Issue on Concept and Ontologies in Web-based Educational Systems of International Journal of Continuous Engineering Education and Life-long Learning, 14 (3), 2004, Dicheva, D. and Aroyo, L. (Editors).

[11] Special Issue on the Educational Semantic Web: Visioning and Practicing the Future of Education: Journal of Interactive Media in Education, 2004 (1), Anderson, T. and Whitelock, D. (Editors).

[12] R. Mizoguchi and J. Bourdeau. Using Ontological Engineering to Overcome Common AI-ED Problems. *International Journal of Artificial Intelligence in Education*, Vol.11, No.2, pp.107-121, 2000.

[13] Devedzic V. (2001). The Semantic Web – Implications for teaching and Learning. *Proceedings of ICCE* 2001, Seoul, Korea, 26-28.

[14] Dicheva D., Aroyo L. Concepts and Ontologies in Web-based Educational Systems. *Int. J. Cont. Engineering Education and Life-long Learning,* 14 (3), 187-190, 2004.

[15] Aroyo, L., Dicheva, D. (2004). The New Challenges for E-learning: The Educational Semantic Web. *J. Educational Technology & Society*, 7 (4), 59-69.

[16] Devedžic, V. Web Intelligence and Artificial Intelligence in Education. Educational Technology & Society, 7 (4), 29-39, 2004.

[17] Mizoguchi, R. Ontology Engineering Environments. *Handbook on Ontologies*, S. Staab and R. Studer (editors), 275-295, Springer, 2003.

[18] Faiçal Azouaou, Cyrille Desmoulins, Weiqin Chen, Semantic Annotation Tools for Learning Material, *Workshop on Applications of Semantic Web Technologies for Educational Adaptive Hypermedia*, Eindhoven, The Netherlands, August 23-26, 2004.

[19] Kitamura, Y., Mizoguchi, R. Ontology-based systematization of functional knowledge. *Journal of Engineering Design*, Taylor & Francis, 15 (4), 327-351, August, 2004.

[20] Gómez-Pérez, A., Fernández-López, M., Corcho, O. (2004) *Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, London: Springer-Verlag.

[21] Werthheimer, M. (1945) *Productive Thinking*, New York: Harper Collins.

[22] Miller, G. (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review*, vol. 63, pp. 81-97.

[23] Biezunski, M., Bryan, M., Newcomb, S.: ISO/IEC13250:2000 Topic Maps: Information Technology, www.y12.doe.gov/sgml/sc34/document/0129.pdf.

[24] Dicheva D., Dichev C., Sun, Y., Nao, S.: Authoring Topic Maps-based Digital Course Libraries, *Workshop on Applications of Semantic Web Technologies for Educational Adaptive Hypermedia, SW-EL@AH'04*, Eindhoven, The Netherlands (2004), 331-337.

[25] The TM4L Project, http://www.wssu.edu/iis/NSDL/index.html.

[26] Dicheva D., Dichev C. (2004). Educational Topic Maps, *3rd International Semantic Web Conference* (ISWC'2004), November 7-11, 2004, Hiroshima, Japan, 19-20.

[27] Special Issue on Advances of Semantic Web for E-learning: Expanding learning frontiers. *British Journal of Educational Technology*. Naeve, A., Lytras, M., Nejdl, W., Harding, J., Balacheff, B (Eds).

Adaptive E-Learning Content Generation based on Semantic Web Technology

Edmond Holohan*, Mark Melia, Declan McMullen, Claus Pahl
Dublin City University, School of Computing,
Dublin 9, Ireland.
eholohan@computing.dcu.ie

Abstract. The efficient authoring of learning content is a central problem of courseware engineering. Courseware authors will appreciate the benefits of tools which automate various authoring tasks. We describe a system, *OntAWare,* which provides an environment comprising a set of software tools that support learning content authoring, management and delivery. This system exploits an opportunity provided by the emerging technologies of the Semantic Web movement, most notably knowledge-representation standards and knowledge-processing techniques. The system represents a combination of these newer developments with earlier work in areas such as artificial intelligence (AI) and intelligent tutoring systems (ITS).

A key feature of the authoring environment is the semi-automatic generation of standard e-learning and other courseware elements (learning objects). Widely available standardised knowledge representations (ontologies) and ontology-structured content are used as source material. Standard courseware elements are produced by the application of graph transformations to these ontologies. The resulting products can be hosted by standards-compliant delivery environments.

Adaptivity is an important characteristic of the system as a whole. Authors can select and customise new or existing subject ontologies and employ an appropriate teaching/learning strategy in the generation of learning objects. Instructors can configure the delivery environment either to offer strictly sequenced presentations to students, or to allow also varying degrees of free student navigation, based on the the runtime incorporation of domain ontologies. Students in turn can take the generated courses in the preconfigured delivery environment, and this delivery is dynamically customised to the individual student's preferences and constantly monitored learning track.

The combination of the semi-automatic generation of learning objects with an adaptive delivery environment is a central feature of this new system.
Keywords: courseware generation, Semantic Web, ontology, learning object, adaptivity.

1. Introduction

The key goal in most forms of instruction is the acquisition by the learner of certain *problem-solving* skills. Invariably, this presupposes the lesser goal of *conceptual knowledge* acquisition. Much published instructional training and test materials, including e-learning content, concentrates on this lesser, easier goal [1]. While, in the longer term, we intend to support both categories of learning, in this paper we concentrate mainly on conceptual learning.

The Semantic Web has triggered some new developments in knowledge engineering and machine learning, most notably the standardisation of knowledge specifications. Standardised *ontologies* already serve as shared conceptual-knowledge skeletons, and declarative *knowledge-processing* specifications may soon be used to model tasks requiring common problem-solving skills such as *procedure selection* and *application*. The time is long since ripe for exploiting the above developments for e-learning *courseware engineering* [2, 3]. This project represents some steps in this direction. Specifically, we propose to semi-automate the generation of courseware learning objects by applying graph transformations to the concept graphs represented by the ontologies of the Semantic Web.

The overall structure of this paper reflects the functional architecture of our system. Firstly, we position our work in the broader context of current *of Artificial Intelligence in Education* (AIED) research. Then we describe the basic functionality of our system, *the generation and export of static courseware*. By this we mean the production and export of stand-alone standard learning objects such as slide sequences or tests. Next, we discuss extensions to this functionality in *the generation of courseware for flexible delivery*. By *flexible delivery* here we mean run-time support for combinations of the various forms of *navigation* and *adaptivity* discussed in Section 4. Both Sections 3 and 4 address the production of learning objects for the imparting of conceptual knowledge content. In conclusion, we evaluate our experiences to date and we briefly discuss the possibility of semi-automating also the generation of learning objects that can exercise a learner's problem-solving skills.

2. Background and Related Work

Our work addresses some of the concerns recently expressed by others such as Devedzic [4]. He notes that there are several challenges in improving Web-based education, such as providing for more adaptivity and intelligence, and that a key enabler of this improvement will be the provision of ontological support. The creation of educational Web content with ontological annotation should be supported by ontology-driven authoring tools based on a number of underlying ontologies, e.g., those describing the domain itself, as well as various theories of learning and the instructional design process. Devedzic refers to the the most notable work in the AIED community related to the development of educational ontologies, e.g., that of Mizoguchi and Bourdeau [1], who have outlined a road map towards an ontology-aware authoring system.

Others have taken approaches to concept-based courseware authoring similar to ours, e.g., the *AIMS* system [5, 6] which provides the author with assistance in creating content through domain and instructional models and in configuring these for delivery. A special feature of *AIMS* is the support of a generic set of authoring tasks within the system. A user model captures the learner profile, and provides the information on which adaptivity is based. See also the *Courseware Watchdog* [7] which supports lecturers in the preparation of new courses. It is based on the Edutella peer-to-peer network and employs focused crawling and conceptual browsing to provide personalized access to learning material that is available somewhere on the Web. Another relevant system is *AdaptWeb* [8] which offers adaptive content presentation of Web-based courses, according to selected programs and student's profile. Course contents are customized according to complexity, sequencing, the use of examples and supplementary materials.

3. The Generation and Export of Static Courseware

3.1 The Modularisation of Knowledge and Courseware

Concept graphs, semantic nets or *ontologies* are the natural organising skeletal structure for courseware content. They are similar to popular mind-maps and may be likened to the mental cognitive structures of various pedagogical models [9]. The AI and ITS communities have employed such graph structures for some time and their processing is well understood. But now, thanks to the impetus of the Semantic Web, some globally accepted, interoperable, standard knowledge representations have emerged, e.g., *XML Topic Maps* (XTM), the simpler *Resource Description Framework/Schema* (RDF/S) and the RDF/S-based *Web Ontology Language* (OWL). In addition there is now adequate ontology-processing tool support, e.g., the *Jena RDF Toolkit* from Hewlett-Packard and the *Protégé* GUI ontology editor from Stanford University, together with a widening range of freely available standardised upper and domain ontologies, e.g., the *Standard Upper Ontology* [10].

3.2 The Generation of Learning Objects

The *OntAWare* system contains a tool that accepts as input an RDF/S-based (OWL) ontology file and generates as output a variety of static courseware files, consisting of both interactive and non-interactive learning objects (LOs). The user is required to select one subject ontology from a list of same. For simplicity, let us suppose that the goal of the generated LOs will be to teach the ontology's knowledge content and its class-subclass and class-instance relationships in particular. For illustration, we employ a simple top-down deductive instructional strategy. At the start the author is presented with a menu of target learning concepts (actually the classes in the selected ontology) and s/he can select one of these as the main target learning goal. Then on demand, s/he may automatically generate certain LOs constructed by a standard concept sequencing algorithm, *recursive pre-order depth-first graph traversal* (See [11], [12]), applied to the subtree formed by the selected concept and its related subconcepts. The concept and its description are listed, followed by direct example instances of the concept and their property values. Then, if the concept has direct subconcepts, these are given the same treatment, and so on for their subconcepts too in a recursive fashion.

The non-interactive LOs produced include a lesson plan/outline and a corresponding (e.g., PPT, HTML or other de-facto standard) slideshow sequence, consisting of bullet-pointed slides showing relevant concepts and examples taken directly from the ontology. The above generative graph-traversal algorithm has been modified to produce also an interactive LO in the form of an objective multiple-choice test. For example, this test may include questions such as, *"Which of the following items is (or is not) an example of the concept, X?"* These questions and their answer options are randomly generated, dynamically, and based solely on the interface to the underlying ontology. The objective tests generated by the system conform to the IMS/QTI [13] standards and can be exported to compliant external courseware delivery platforms. The products exported by this component of *OntAWare* would be useful to the student for self-learning or to the courseware author as an aid in the authoring process. Usually, the generated slides would need to be fleshed out further with standard textual or graphic content, and the inter-LO sequencing would need to be determined. The tool provides a demonstration of the technical feasibility of converting standard Semantic-Web ontologies into useful standard learning objects.

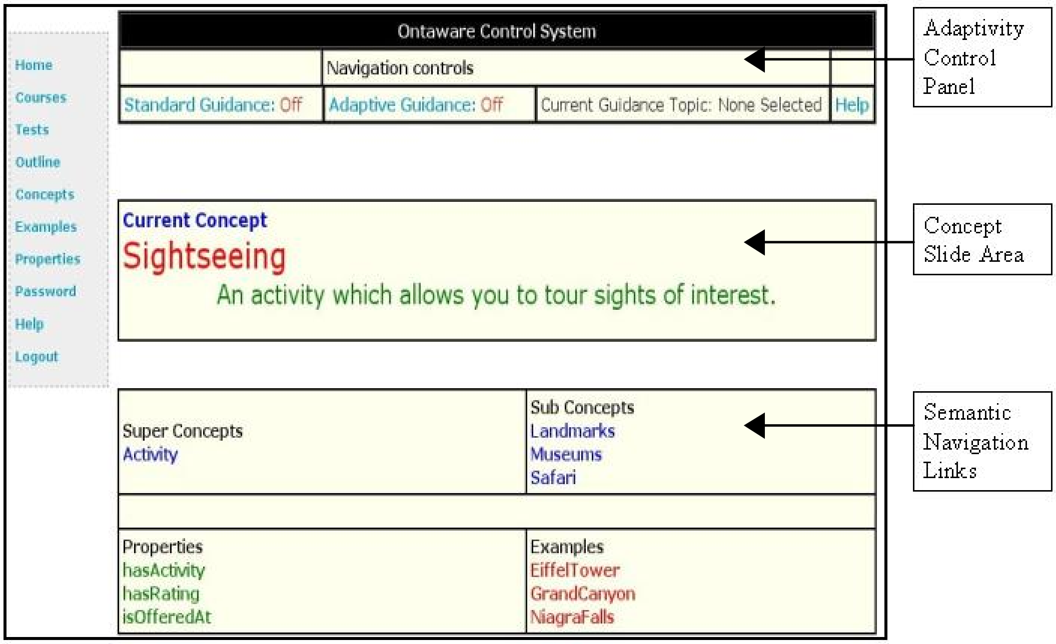


Figure 1: *OntAWare* Active Slide /Concept Navigation Screen

4. The Generation of Courseware for Flexible Delivery

4.1 Constrained, Sequenced E-Learning versus Free Navigation

Semantic-Web ontologies are intended primarily to support free navigation of semantically linked content. In addition to the generation of static courseware, the *OntAWare* system also has its own delivery environment which offers the best of both worlds by supporting totally free ontology-based navigation and allowing the user to specify increasing degrees of constrained navigation based on the generated courseware lesson plans (concept sequences).

4.2 Free Ontology-based Navigation

In addition to the generative functionality described above (3.2), the *OntAWare* system contains a knowledge and courseware delivery environment. In its basic mode of operation this environment offers free ontology-based navigation to the user. At the front end this consists of three types of browser screen layout:

- *Concept Screen* (Figure 1) showing the name and description of the current concept, and links to:
  - Superconcepts of the current concept
  - Subconcepts of the current concept
  - Properties defined for the current concept
  - Individual instances/examples of the current concept
- *Property Screen* showing the name of the current property, and links to:
  - Domain concepts of the current property
  - Range concepts of the current property
- *Individual Screen* showing the name, description and attribute values of the current individual, and links to:

- Concept classes of the current individual
- Property value instances of the current individual

The above screens are dynamically constructed by delivery-tool servlets (Figure 2). These screens, particularly the *concept screen* and the *individual screen*, play a double role in the system. On the one hand they serve as ontology-based content navigation screens, but on the other, they also play the role of *active slides* in the delivery environment. The *slide* metaphor is important educationally, since each slide/screen is intended to focus the user's attention on only one concept at a time and on information directly related to it. The use of these *active slides* requires at least a runtime version of the *OntAWare* system, rather than merely a third-party viewer.

4.3 Option of Guided Navigation

If so desired by the instructor/student, a courseware lesson plan, in the form of guided navigation through the above ontology-based screens, may be activated. In order to generate these lesson plans (i.e., concept sequences), the system currently employs the same ontology graph traversal algorithm described in Subsection 3.2 above. Normally, the system is configured to support two separate categories of user: the author/instructor, who generates the lesson plan, and the student, who follows it. At any stage when following the guidance of a given lesson plan, the student has the option of freely wandering using any of the on-screen links described above. The system constantly monitors the student's movements, so s/he can move forward, backtrack, or return to the generated guided lesson plan at any time.

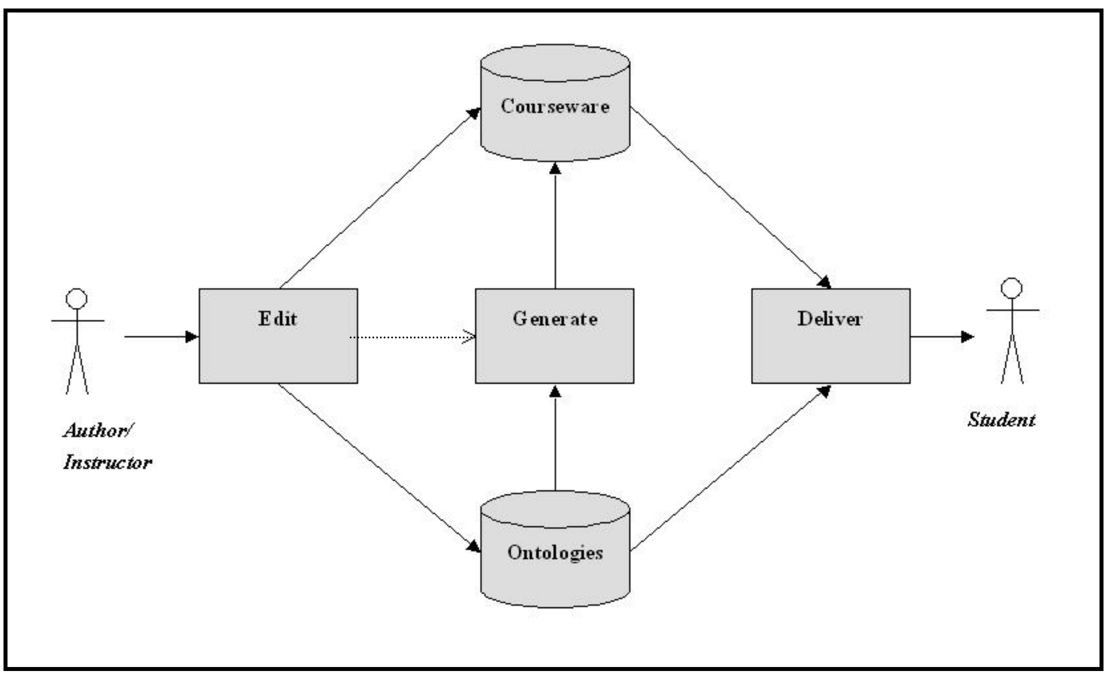


Figure 2: Basic *OntAWare* Functional Architecture.

4.4 Option of Adaptive Guided Navigation

Based on the fact that the student's learning track is now being monitored at lesson delivery time, it is a natural step to introduce some real-time adaptivity into the guidance provided. By default, if *adaptive guidance* is switched on (see Figure 1), the system makes the simplifying assumption, that, if a student has visited a particular concept screen, s/he *knows* that concept and does not need to revisit it. This minimal level of adaptivity can be left under the student's control, or alternatively, the degree of free navigation or of guidance adaptivity allowed to the student may be preset by the instructor. This functionality can be further enhanced by basing adaptivity on student knowledge evaluation using the automatically generated pre- and post-tests mentioned in Subsection 3.2 above.

4.5 Combining Flexibile Delivery with Interoperability

As already noted, by *flexibility* here we mean the options of *free navigation*, *guided navigation* and *adaptive guided navigation* discussed above. Crucially this flexibility is based on the availability of the ontology information at lesson-delivery (i.e., guided-navigation) time. The system also has an advanced feature which allows it to incorporate such flexibility in certain standardised LOs it exports to third-party delivery platforms. We can export whole courses consisting of standardised packages of LOs, e.g., lessons, tests, etc., to third-party delivery platforms. At the same time we preserve the delivery *flexibility* already demonstrated within these LOs. The system uses the SCORM Content Aggregation Model [14] for the packaging of these LOs, together with Simple Sequencing standard for inter-LO sequencing. We do not expose the internals of our generated LOs to third-party environments. Rather, we maintain complete control over intra-LO sequencing/navigation. This implies that a runtime version of *OntAWare*, together with the source ontologies, must be made available within the context of the delivery platform. For greater flexibility learning grid services may be combined with the SCORM runtime environment in the future [15].

4.6 Further Types of Flexibility

At the back end of the system there is support for ontology database maintenance. Courseware learning objects may be generated either from ontologies in the database or from externally sourced ontology files. If so desired, these ontology files can be imported into the *OntAWare* ontology database, and extended by the courseware author with his/her own concepts, examples, etc. Little if any ontology expertise is required of the courseware author, since this is accomplished via user-friendly screens similar to that in Figure 1. This feature provides the basis for further types of flexibility in the system.

For example, this ontology base can act as a master index into a wide range of traditional courseware, web and other content repositories. The courseware author/teacher is allowed to insert his/her own favourite web-page links on the generated screens and these are stored along with the underlying ontology. Tool support to aid link discovery is provided by employing the concept-based metadata searching techniques of the Semantic Web. At lesson presentation time these links may be displayed automatically on the relevant lesson screens. The ability to write to the ontology base also facilitates the implementation of persistent user models, which are overlaid on the underlying ontology, e.g., a simple percentage value can indicate how well the student knows a given concept. Each user's state of domain knowledge can be modelled in this way and updated automatically by the delivery process. This can serve to extend the adaptivity described in 4.4 above.

5. Conclusion

The advent of the *Semantic Web* shows great promise for education and knowledge management generally. However, in addition to the benefits acccruing from the normal uses of this new environment, we believe that the underlying technologies currently being developed to realise the Semantic Web vision can already be exploited in support of courseware engineering.

While others have adopted more theory-driven architectural approaches to the design of similar systems [16, 17], we have consciously taken an incremental practical approach. This strategy has proved to be a fruitful one. First, we verified that we could automatically generate and export useful learning objects from the domain ontologies of the Semantic Web. Second, we developed an ontology-based user-friendly free content navigation system – this then served as the basis for further developments. Third, by combining the first two steps, we introduced ontology-based navigation guidance for the student user of the system. Fourth, we added real-time user monitoring and corresponding adaptive navigation guidance. Fifth, we incorporated convenient transparent ontology modification for the teacher. So far, we have used a simple concept-sequencing algorithm as a representative instructional strategy. However, in the currently ongoing steps we are generalising on this by allowing the teacher to specify in advance the sequencing algorithms and broader instructional strategies to be employed.

The ontology graph transformations employed so far in the *OntAWare* system have been hardcoded (in Java) and so necessarily incorporate an implicit instructional strategy, e.g., the traditional, top-down, deductive, from-abstract-concepts-to-concrete-examples approach. We hope to replace such hard-coded algorithms with declarative ontology-transformation specifications. These will be selected and perhaps customised by the courseware author and used to capture a variety of instructional strategies.

The LOs that can readily be generated from ontology sources necessarily focus on the imparting of knowledge to the student. But the really interesting kinds of LO are those which constructively train the student in some skill requiring knowledge application, e.g., *problem solving* (See: [9], [18], [19]). Future work will examine whether Semantic Web technology can be used to semi-automate the production of such advanced dynamic LOs. Semantic-Web technology consists, not merely of the static source ontologies themselves, but also of the tools used to reason with and transform these ontologies. If we begin by limiting the problem domain to well-defined categories, and describe a given problem scenario stepwise, where each step is defined in terms of a set of preconditions and its successful solution as a set of postconditions, then we may be able to generate LOs that teach or test the skill of solving such a problem. Languages, e.g., OWL-S [20], currently being developed for the standardisation of web service specifications may prove useful here.

So, we expect that ontology graph transformation representations will be doubly useful in the future. Not only will they will provide a method for capturing generalised instructional strategies for knowledge acquisition, but also these graph transformations themselves may well act as sources for the semi-automatic generation of the dynamic learning objects required for skills acquisition.

References

[1] Mizoguchi, R., & Bourdeau, J. (2000). Using Ontological Engineering to Overcome Common AI-ED Problems. In *International Journal of Artificial Intelligence in Education (2000), 11, 107-121.*

[2] Pahl, C., & Holohan, E. (2004). Ontology Technology for the Development and Deployment of Learning Technology Systems – a Survey. *International Conference on Educational Hyper- and Multimedia Edmedia'04. AACE., 2004.*

[3] Holohan, E. (2003). *Automating the Generation of Courseware*, MSc. Dissertation, School of Computing, Dublin City University, 2003.

[4] Devedzic, V. (2004). Education and the Semantic Web. In *International Journal of Artificial Intelligence in Education 14 (2004) 39-65.* IOS Press.

[5] Aroyo, L., Dicheva D. (2004). The New Challenges for E-learning: The Educational Semantic Web. In *Educational Technology & Society, 7(4), pp. 59-69, 2004.*

[6] Aroyo, L., Mizoguchi, R., & Tzolov, C. (2003). OntoAIMS: Ontological Approach to Courseware Authoring. *International Conference on Computers in Education (ICCE'03).*

[7] Staab, S., Studer, R., Stumme, G., Tane, J., & Stumme, G. (2002). Accessing Distributed Learning Repositories through a Courseware Watchdog. *World Conference on E-Learning in Corp., Govt., Health., & Higher Ed. 2002(1), 909-915.* Available: http://dl.aace.org/9460

[8] de Freitas, V., Marçal, V.P., Gasparini, I., Amaral, M.A., Proença Jr., M.L., Brunetto, M.A.C., Pimenta, M.S., Ribeiro, C.H.F.P., de Lima, J.V., & de Oliveira, J.P.M. (2002). AdaptWeb: an Adaptive Web-based Courseware. *ICTE - International Conference On Information And Communication Technologies In Education, 2002. p. 131-134.*

[9] Wiley, D. A. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The Instructional Use of Learning Objects: Online Version*. Retrieved Jan 21, 2005, from the World Wide Web: http://reusability.org/read/chapters/wiley.doc

[10] Niles, I., & Pease, A. (2001). Towards a Standard Upper Ontology. In C. Welty & B. Smith (Eds.), *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems*.

[11] Fischer, S. (2001). Course and exercise sequencing using metadata in adaptive hypermedia learning systems. In *Journal on Educational Resources in Computing (JERIC) 1(1), 2001.*

[12] Karampiperis, P., & Sampson, D. (2004). Adaptive Instructional Planning using Ontologies. In *Proc. Of the 4th IEEE International Conference on Advanced Learning Technologies ICALT 2004.*

[13] IMS (2002). *IMS Question & Test Interoperability. ASI Best Practice & Implementation Guide -Final Specification Version 1.2.* IMS Global Learning Consortium, Inc.

[14] SCORM (2004). *Sharable Content Object Reference Model*. http://www.adlnet.org (last visited on 11-Feb-2005).

[15] Neumann, F., & Geys, R. (2004). SCORM and the Learning Grid. *4th International LeGE-WG Workshop - Towards a European Learning Grid Infrastructure: Progressing with a European Learning Grid*. Retrieved Feb 8, 2005, from the World Wide Web: http://ewic.bcs.org/conferences/2004/4thlege/session3/paper2.htm

[16] Inaba, A. and Mizoguchi, R. (2004). Learning Design Palette: An Ontology-aware Authoring System for Learning Design. In *Proceedings of ICCE2004*.

[17] Kasai, T., Yamaguchi, H., Nagano, K., and Mizoguchi, R. (2004). Development of a System that Provides Teachers with Useful Resources from Various Viewpoints Based on Ontology. In *Proceedings of ED-MEDIA2004*.

[18] Tecuci1, G., Boicu1, M., & Marcu1, D. (2003). Toward a Disciple-based Mixed-Initiative Cognitive Assistant. In *Proceedings of the IJCAI-03 Workshop on Mixed-Initiative Intelligent Systems, 2003.*

[19] Gašević, D., Djurić, D., Devedžić, V., Damjanović, V. (2004). Approaching OWL and MDA Through Technological Spaces. In *Proceedings of the 3rd Workshop in Software Model Engineering - WiSME2004 at 7th International Conference on the Unified Modeling Language: Modeling Languages and Applications*.

[20] The OWL Services Coalition. (2005). *OWL-S: Semantic Markup for Web Services*. Retrieved Feb 8, 2005, from the World Wide Web: http://www.daml.org/services/owl-s/1.0/owl-s.pdf

Semantic Description of the IMS Learning Design Specification

Manuel Lama, Eduardo Sánchez, Ricardo R. Amorim, and Xosé A. Vila
Departament of Electronics and Computer Science, Faculty of Physics.
Campus Univesitario Sur s/n, University of Santiago de Compostela.
15782 Santiago de Compostela, A Coruña, Spain.
lama@dec.usc.es, eduardos@usc.es, rramorin@usc.es, vila@dec.usc.es

Abstract. In this paper we present a learning design ontology that is based on the IMS Learning Design (IMS LD) specification. The IMS LD is a metadata standard that describes the elements of the design of any teaching-learning process on the basis of a well-founded conceptual model. However, this specification has been modelled and represented using the XML-Schema language, which is not expressive enough to describe the semantics of all the elements of such conceptual model. To solve these limitations, we have developed an ontology using Protégé at the knowledge level, and then translated into OWL, to represent it in the standard language of the Semantics Web, and first order logic, to formalize the axioms defined in the ontology.

Keywords. IMS Learning Design; Ontologies; Semantic Description; Formal Axioms.

1. Introduction

In the last years, the growing of the Internet have opened the door to new ways of learning and education methodologies. Furthermore, the appearance of different tools and applications has increased the need for interoperable as well as reusable learning contents, teaching resources and educational tools [1]. Driven by this new environment, several metadata specifications describing learning resources, such as IEEE LOM [2] or Dublin Core [3], and learning design processes [4] have appeared. In this context, the term *learning design* is used to describe the method that enables learners to achieve learning objectives after a set of activities are carried out using the resources of an environment. From the proposed specifications, the IMS Learning Design (IMS LD) [5] has emerged as the *de facto* standard that facilitates the representation of any learning design that can be based on a wide range of pedagogical techniques.

The metadata specifications are useful solutions to describe educational resources in order to favour the interoperability and reuse between learning software platforms. However, the majority of the metadata standards are just focused on determining the vocabulary to represent the different aspects of the learning process, while the meaning of the metadata elements is usually described in *natural language*. Although this description is easy to understand for the learning participants, it is not appropriate for software programs designed to process the metadata. To solve this issue, ontologies [6] could be used to describe *formally* and *explicitly* the structure and meaning of the metadata elements; that is, an ontology would semantically describe the metadata concepts. Furthermore, both metadata and ontologies emphasize that its description must be shared (or standardized) for a given community.

In the educational domain, authors have developed ontologies to: (1) describe the learning contents of technical documents [7]; (2) to model the elements required for the design, analysis, and evaluation of the interaction between learners in computer supported cooperative learning [8]; (3) to specify the knowledge needed to define new collaborative learning scenarios [9]; or (4) to formalize the semantics of learning objects that are based on metadata standards (like LOM) [10]. The focus of that research is either on the development of a taxonomy of concepts on the basis of a established theory or specification [7,8,9], or on the formal definition of the metadata using an ontology language [10]. However, none of them deal with the formal description of the meaning of the concepts, and they do not address the ontological modelling of any specification for learning design.

In this paper, we present a *learning design ontology* based on the IMS LD Level A specification. In this ontology, the IMS LD elements are modelled in a *concept taxonomy* in which the relations between the concepts are explicitly represented. Furthermore, a set of *axioms* constraining the semantics of the concepts has been formulated from the restrictions (available in natural language) identified in the analysis of the IMD LD specification.

The paper is structured as follows: in section 2, the need for a learning design ontology is justified; in section 3, the concept taxonomy and the ontology axioms are described; then, in section 4, an example that illustrates how the ontology could be used; finally, the contributions of the paper are summarized.

2. The Need for a Learning Design Ontology

The IMS Learning Design specification is a metadata standard that describes all the elements of the design of a teaching-learning process [5]. This specification is based on: (1) a well-founded *conceptual model* that defines the vocabulary and the functional relations between the concepts of the LD; (2) an *information model* that describes in an informal (natural language) way the semantics of every concept and relation introduced in the conceptual model; and (3) a *behavioural model* that specifies the constraints imposed to the software system when a given LD is executed in runtime. In other words, the behavioural model defines the semantics of the IMS LD specification during the execution phase.

To facilitate the interoperability between software systems, the IMS LD specification has been formally modelled through the XML-Schema language [11,12]. However, the knowledge model of this language is not expressive enough to describe the semantics (or meaning) associated to the elements of the IMS LD. Thus, the main limitations of the XML-Schema language are [13]:

– Hierarchical (is-a) relations between two or more concepts cannot be *explicitly* defined. Therefore, there are no inheritance mechanisms facilitating the representation of concept taxonomies. For example, in the IMS LD specification, the `Learner` and `Staff` elements do not inherit the attributes and relations of the `Role` element: they are just included as XML sub-elements of the `Role` element.

– General and formal constraints (or *axioms*) between concepts, attributes, and relations cannot be specified. These axioms describe more precisely the semantics of the concepts, because they constrain how the instances of the concepts could be created. For instance, the axiom "*if an `Act` is executed in the context of a `Play`, and both have a given value for the `time limit` attribute, the value of this attribute for the `Play` should be greater or equal than the value for the `Act`*" could not be represented in the XML-Schema language.

To solve these limitations of the XML-Schema language, a software system processing any given LD should automatically manage the semantics of the specification, guaranteeing that both restrictions and concept hierarchy are verified. Therefore, the modelling of the IMS LD specification needs to be enriched in order to describe *explicitly and formally* the semantics of its elements. To achieve this goal, we have developed an ontology [6], as it facilitates the semantic description of the conceptual model as well as the definition of formal axioms related to both information and behavioural models. This ontology is based on a knowledge model that includes complex taxonomic relations (like both hierarchical and ad-hoc relations, disjoint and exhaustive partitions, etc.) as well as formal axioms.

3. The Learning Design Ontology

To develop the Learning Design ontology we have created a *concept taxonomy*, which describes the elements of the IMS LD conceptual model and the IMS LD information model, and a *set of axioms*, which formally constraint the semantics of the concept taxonomy on the basis of the explanations formulated in natural language in both information and behavioural models.

3.1. Description of the Concept Taxonomy

The upper node of the LD ontology is the `Unit of Learning` concept (Figure 1) that defines a general module of an educational process, like a course or a lesson. Following the IMS LD specification, a unit of learning is modelled as a content package [5] that integrates the description of both the LD and the set of resources related to it. The `Resource` concept allows to represent various entities, like physical resources (Web pages, files, etc.), and concepts whose attribute description is domain-dependent (learning objectives, prerequisites, etc.). To model the different kinds of resources, we have extended the IMS LD specification with a new hierarchy of concepts (grey boxes in Figure 1). In this way, when a LD concept refers to any of the resource properties, it establishes a relation with the `Item` concept, which in turn, has a set of subclasses that replicate the hierarchical structure of the resources (following a one-to-one correspondence). These two hierarchies have been introduced to decouple the references to the resources (`Item` hierarchy) from their modelling (`Resource` hierarchy). Thus, if two applications use the same LD to model a course, but define the resources in a different way (for example, if the learning objectives are specified either as textual description or through their corresponding attributes), the LD does not need to be changed because the links to the resources are indirectly established through the `Item` hierarchy.

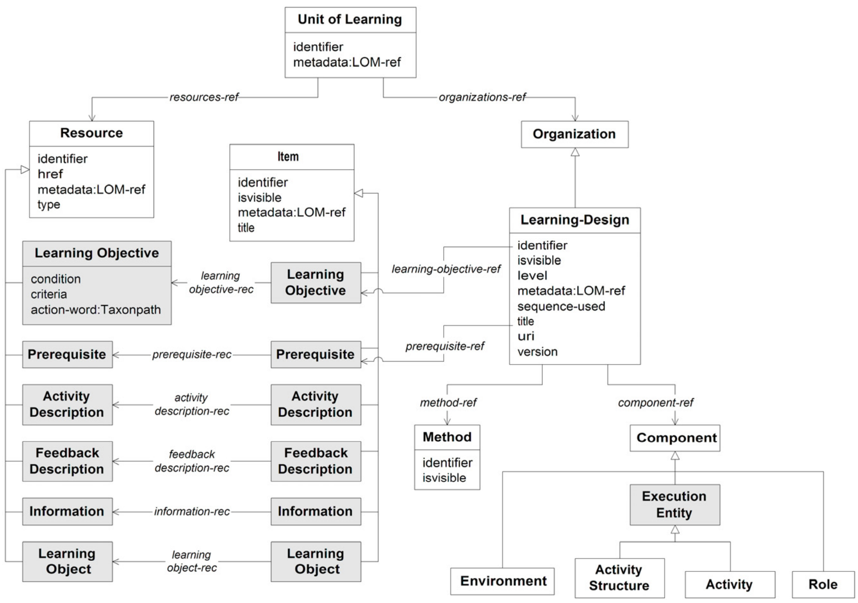


Figure 1. Upper concepts of the Learning Design ontology.

Learning Design Description

The `Learning Design` concept is related to the `Learning Objective` and `Prerequisite` concepts, which define the intended outcomes when the unit of learning is carried out, and the previous knowledge needed to participate in it, respectively. Both concepts are subclasses of the `Item` concept, and therefore they will be mapped onto the `Learning objective` and `Prerequisite` concepts of the `Resource` hierarchy.

The `Learning Design` concept has a number of `Components` used to describe the learning process: the `Execution Entities` to be carried out, which can be `Activities` or `Activity Structures` (groups of activities that will be executed in sequence); the `Roles` that participate in the execution of those activities as instances of the `Learner` and `Staff` concepts; and the `Environments` that describe the educational resources to be used in the activities. These concepts constitute an exhaustive and disjoint partition, because an instance of a `Component` must necessarily be an instance of one of its subclasses.

The `Learning Design` concept is also related to the `Method` concept, which describes the dynamics of the learning process (Figure 2): a method is composed of a number of instances of the `Play` concept that could be interpreted as the *runscript* for the execution of the unit of learning. All the play instances have to be executed in parallel, and each one consists of `Act` instances, which could be understood as a *stage* of a course or module. The Act instances must be executed in sequence (according to the values of the `execution order` attribute), and they are composed by a number of `Role Part` instances that will be executed concurrently. A `Role Part` associates a `Role(s)` with an `Execution Entity` to be carried out in the context of the act. Finally, every `Execution Entity` requires an `Environment`, which manages `Learning Objects` as resources. In summary, the execution of an act consists on the simultaneous participation of roles in an activity or group of activities, and once the activities are completed, the associated roles could participate in the execution of any other activity through different role part instances.

The `Activity` concept has two subclasses: the `Learning Activity` concept and the `Support Activity` concept. A `Learning Activity` models an educational activity that establishes a relation with the `Prerequisite` and the `Learning Objective` concepts. The `Support Activity`, however, is introduced to facilitate the execution of a learning activity, but it does not cover any learning objective. These two classes

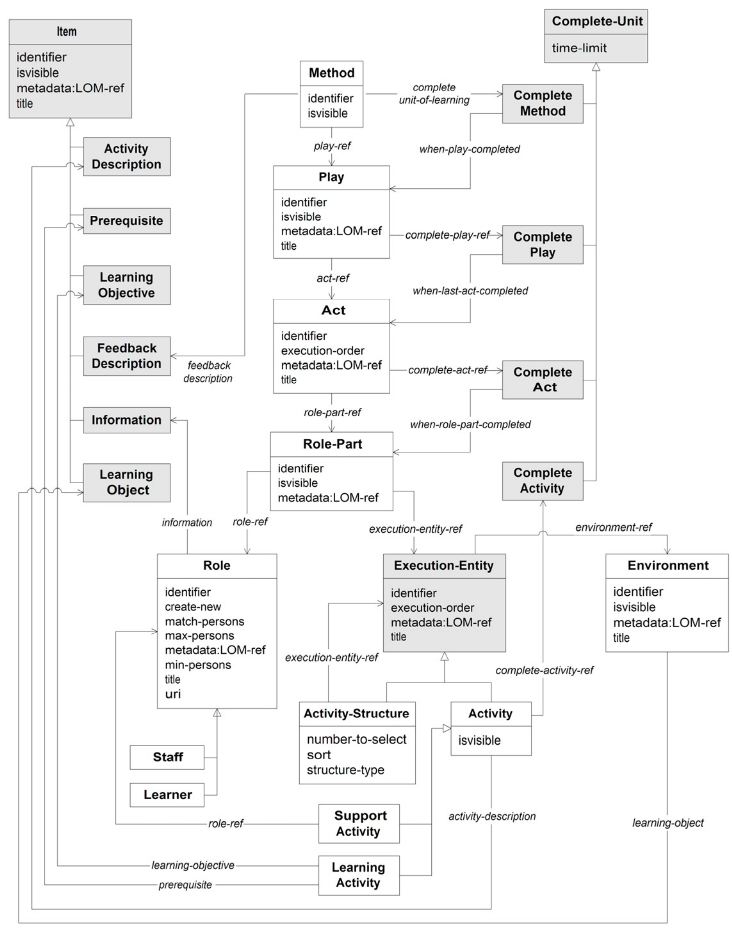


Figure 2. Concept taxonomy that describes the dynamics of a learning design.

constitute a disjoint and exhaustive partition, because an instance of the `Activity` concept should be either a learning or a support activity.

Every concept involved in the dynamics of the learning process (`Method`, `Play`, `Act`, and `Activity`) establishes a relation with one of the subclasses of the `Complete Unit` concept, which indicates when an execution is finished. In the IMS LD Level A, this condition can be specified through the `time limit` attribute, which define the temporal duration of the execution, or referred to an instance of the entity of which is composed by. For example, an act would be completed when the instance of the `Role Part` pointed out by the relation `when-role-part-completed` has finished. Furthermore, in both Level B and C of IMS LD, the modelling of these subclasses will be extended to enable the specification of more complex completion conditions.

3.2. Description of the Learning Design Ontology Axioms

The three models of the IMS LD specification contain a natural language description of the semantics of all the taxonomy concepts, including the *constraints* that should verify their instances when they are created and managed by a software system. To incorporate these restrictions to the LD ontology we have applied the following procedure: first, the description of the constraints is identified in the text of IMS LD; then, if necessary, this description is reformulated considering the elements of the LD concept taxonomy (concepts, relations, and attributes); and, finally, the restrictions are represented in a declarative, formal, and language-independent way as *axioms* in first order logic. In Table 1, three axioms obtained following this procedure are presented.

Depending on the stage where the axioms are applied, we distinguish between two different kinds of axioms:

- *Design axioms*, which determine how the instances of the taxonomy concepts will be created when a given learning design has been specified. For example, the first axiom of the Table 1 will not allow to create a play with a value for the time limit attribute less than the time limit of any of its acts. Following the procedure previously described, we have extracted and formally defined 28 axioms, most of them from the IMS LD concept and information models.
- *Runtime axioms*, which are associated with the management and monitoring of the execution of the learning design created during the design phase. For example, one of the axioms of this category must guarantee that the plays of the method are executed in parallel. However, to specify many of these axioms it is necessary to *extend* the LD ontology for including a runtime model (not defined in the IMS LD specification) that would represent the different states of execution of the learning design. Currently, these axioms have been extracted from the behavioural model.

From a modelling point of view, the formal definition of the semantic constraints of the LD concepts is the main advantadge of the learning design ontology when icompared with the IMS LD XML-Schema specification. On one hand, the semantics of the concepts is *completely* included in the ontology (not only the taxonomic structure), and, on the other hand, the programmers will not need to interpret the descriptions of the IMS LD models in order to translate its meaning into software systems desinged towards to manage learning design elements.

Table 1. Formal definition of some ontology axioms identified in the analysis of the IMS models. For the IMS LD Level A, 28 axioms have been identified.

Design Axiom 1	IMS LD Specification (*natural language*)	*Page 38 (item 0.2.2)*: "The time limit specifies that it is completed when a certain amount of time has passed, relative to the start of the run of the current unit of learning. The time is always counted relative to the time when the run of the unit-of-learning has been started. Authors have to take care that the time limits set on role-parts, acts and plays are logical."
	Explanation	The value of the attribute `time limit` of a `Method` must be greater than the value of the `time limit` of any `Play`. That is, the `Play(s)` cannot finish after the `Method`.
	Formal Description	$\forall \text{ m, p, cm, cp} \mid \text{m} \in \text{Method} \wedge \text{p} \in \text{Play} \wedge \text{cm} \in \text{Complete-Method} \wedge \text{cp} \in \text{Complete-Play} \wedge \text{play-ref(p, m)} \wedge \text{complete-unit-of-learning-ref(cm, m)} \wedge \text{complete-play-ref(cp, p)} \rightarrow \text{time-limit(cm)} \geq \text{time-limit(cp)}$
Design Axiom 2	IMS LD Specification (*natural language*)	*Page 90*: "The same role can be associated with different activities or environments in different role-parts, and the same activity or environment can be associated with different roles in different role-parts. However, the same role may only be referenced once in the same act."
	Explanation	For the same `Act`, the `Roles` involved in the execution of the `Act` are disjoint.
	Formal Description	$\forall \text{ a, r, rp} \mid \text{a} \in \text{Act} \wedge \text{r} \in \text{Role} \wedge \text{rp} \in \text{Role-Part} \wedge \text{role-part-ref(rp, a)} \wedge \text{role-ref(r, rp)} \rightarrow \neg \exists \text{ rp1} \mid \text{rp1} \in \text{Role-Part} \wedge \text{rp1} \neq \text{rp} \wedge \text{role-part-ref(rp1, a)} \wedge \text{role-ref(r, rp1)}$
Runtime Axiom 1	IMS LD Specification (*natural language*)	*Page 25 (item 0.2.1)*: "The create-new attribute indicates whether multiple occurrences of this role may be created during runtime. When the attribute has the value "not-allowed" then there is always one and only one role instance."
	Explanation	If the value of the attribute `create-new` is "not-allowed", it can have an only instance of the `Role` at which it is applied.
	Formal Description	$\forall \text{ r} \mid \text{r} \in \text{Role} \wedge \text{create-new(r)} = \text{"not-allowed"} \rightarrow \neg \exists \text{ r1} \mid \text{r1} \in \text{r}$

4. Sample Scenario

The scenario presents a learning design that models a unit of learning in the environmental education. This learning design has been obtained from the Educator's Reference Desk site (http://www.eduref.org), a well-known site of resources for the educational community that is commonly used as a reference for the verification of the pedagogical expressiveness of the EML UONL [14]. The unit of learning, entitled *environmental determinism*, is applied to two groups of students during the same period of time: one group for the in-classroom modality and the other group for the e-learning modality. The proposed activities in the learning design are: (1) a presentation of introductory topics related to the study domain; (2) a number of sessions of practice using the local region like an environmental laboratory; (3) development of activities like collaboration in classroom, data collection in field, preparations of reports, etc.; and (4) evaluation of the learning process through examination tests.

As Figure 3 shows, the method of the unit of learning consists of two different plays: `Env-Det-Pres-Play` and `Env-Det-Dist-Play`, one for presential learning, and the other for distant learning. The acts of both plays have the *same activities*, and the difference between them is the schedule considered for these activities: the e-learning group will execute the activities asynchrously. Thus, the method completes its execution after one month and 20 days, the `Env-Det-Dist-Play` play will finish after one month and 10 days, and

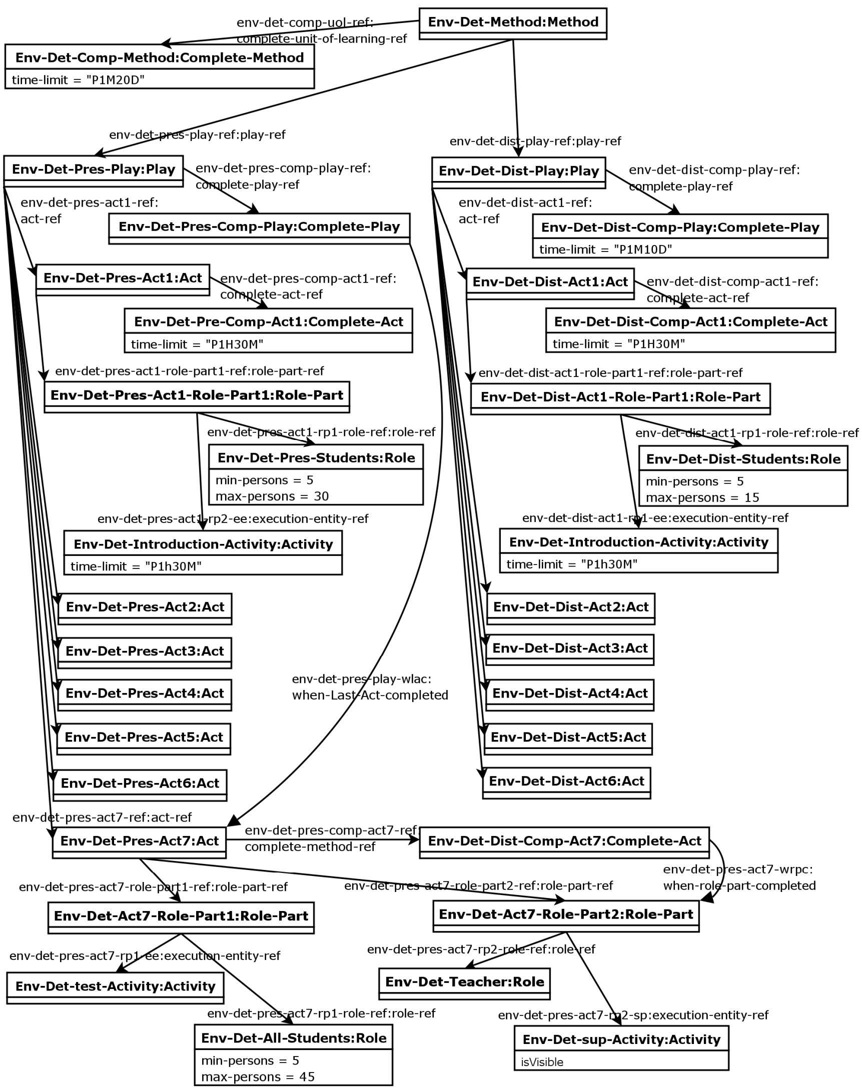


Figure 3. LD ontology instances for the learning design of the unit of learning of the sample scenario.

the Env-Det-Pres-Play play is completed when the seventh act finishes the execution (when-last-act-completed = Env-Det-Pres-Act7), that is, when the role part Env-Det-Act7-Role-Part2 has been completed. To guarantee the consistence between the time limit values for the method and the two plays, the first axiom of the Table 1 is verified. However, this axiom cannot be applied to the Env-Det-Pres-Play play, because it does not define a value for the time limit attribute.

According to Figure 3, for the role Env-Det-Pres-Sudents, the values for the attributes min-persons and max-persons are 5 and 30, indicating the minimum number of students that justify economically the course and the maximum capacity of the classroom respectively. In order to guarantee the consistence of the values of these attributes, the following axiom is verified:

$$\forall\ r \mid r \in \text{Role} \rightarrow \text{max-persons}(r) \geq \text{min-persons}(r)$$

In the two plays of the method, the first act will be completed in one hour and 30 minutes, and the activity Env-Det-Introduction-Activity will be finished in the same period of time. In this case, the specification of the act and its roles is consistent, because the value of the time limit of the act equals the time limit of the activity related with the role part that is executed in the context of the act. The general axiom related to this condition could be expressed as:

$$\forall\ a, ca, actv, cacty, as, rp \mid (a \in \text{Act} \wedge ca \in \text{Complete-Act} \wedge \text{complete-act-ref}(ca, a)) \wedge (rp \in \text{Role-Part} \wedge \text{role-part-ref}(a, rp)) \wedge (actv \in \text{Activity} \wedge cactv \in \text{Complete-Activity} \wedge \text{complete-activity-ref}(cactv, actv)) \rightarrow \text{time-limit}(ca) \geq \text{time-limit}(cactv)$$

In the seventh act all the students must carry out an activity consisting on an examination test (Env-Det-test-Activity), while the teacher has to explain how the test should be answered by the students (Env-Det-sup-Activity). Therefore, as established by the second axiom of Table 1, the roles that participate in the role parts of a same act have to be different. In this case, the roles (students and teacher) are involved in different role parts (or in other words, in different activities).

5. Conclusions and Future Work

The IMS LD specification is expressive enough from the point of view of the learning process designers. Nevertheless, the informal specification of the IMS information and behavioural models increases the complexity of the IMS LD to be understood by programmers, as they are not usually educational specialists. With the development of the learning design ontology the semantics of the concepts is precisely defined, and, in consequence, there should be no misinterpretations or errors when the instances of the concepts are created or manipulated in runtime. Furthermore, the expressiveness of the IMS LD specification is preserved.

The learning design ontology has been developed at the knowledge level using the Protégé tool. Then, the ontology was directly translated into the OWL [15] language (it can be downloaded from *http://www.eume.net/ontology/imsld_a.owl*), which is the W3C recommendation for the Semantic Web. The ontology is also available in first order logic to enable the reasoning with the formal axioms defined in the ontology construction. Currently, the ontology is used to exchange knowledge between the software agents developed in the context of the EUME project [16], whose aim is to enable teachers and students the ubiquitous access to hardware devices and services available in the classroom.

As future work we have planned to translate the ontology axioms into SWRL [17], which is the language currently proposed to express restrictions in OWL. On the other hand, we are working on the extension of the ontology to include the concepts and axioms of the levels B and C of the IMS LD specification.

Acknowledgments

The authors would like to thank the Xunta de Galicia for their financial support in carrying out this work under the project PGIDT02TIC20601PR.

References

[1] D. Wiley. *Learning Object Design and Sequencing Theory*. Department of Instructional Psychology and Technology, Brigham Young University, Doctoral Thesis, 2000.

[2] IEEE Learning Technology Standard Committee (LTSC). *Draft Standard for Learning Object Metadata (LOM)*, 2002. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf.

[3] Dublin Core Metadata Initiative. *Dublin Core Metadata Element Set, Version 1.1. Reference Description*, 2004. http://dublincore.org/documents/dces.

[4] A. Rawlings, P. Rosmalen, R. Koper, M. Rodríguez-Artacho, and P. Lefrere. Survey of Educational Modelling Languages, In *CEN/ISSS WS/LT Learning Techonologies Workshop*, 2002.

[5] IMS Global Learning Consortium. IMS Learning Design Information Model. Version 1.0 Final Specification, 2003. http://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html.

[6] A. Gómez-Pérez, M. Fernández-López, and O. Corcho. *Ontological Engineering*, Springer Verlag, 2004.

[7] S. Kabel, B. Wielinga., and R. de Hoog. Ontologies for indexing Technical Manuals for Instruction. In *Proceedings of the AIED-Workshop on Ontologies for Intelligent Educational Systems*, pages 44-53, LeMans, France, 1999.

[8] A. Inaba, T. Tamura, R. Ohkubo, M. Ikeda, R. Mizoguchi, and J. Toyoda. Design and Analysis of Learners Interaction based on Collaborative Learning Ontology. In *Proceedings of the Second European Conference on Computer-Supported Collaborative Learning (Euro-CSCL'2001)*, pages 308-315, 2001.

[9] B. Barros, F. Verdejo, T. Read, and R. Mizoguchi. Applications of a Collaborative Learning Ontology. In *Proceedings of the Second Mexican International Conference on Artificial Intelligence (MICAI 2002)*, pages 301-310, Yucatan, Mexico, 2002.

[10] J. Brase, and W. Nejdl. Ontologies and Metadata for eLearning. *Handbook on Ontologies*, Springer-Verlag, 2004.

[11] H. Thompson, D. Beech, M. Maloney, and N. Mendelsohn. XML-Schema Part 1: Structures Second Edition, 2004. http://www.w3.org/TR/xmlschema-1.

[12] IMS Global Learning Consortium. IMS Learning Design Binding. Version 1.0 Final Specification, 2003. http://www.imsglobal.org/learningdesign/ldv1p0/imsld_bindv1p0.html.

[13] Y. Gil and V. Ratnakar. A Comparison of (Semantic) Markup Languages. In *Proceedings of the Fifteenth International FLAIRS Conference*, pages 413-418, Pensacola Beach, Florida, 2002.

[14] R. Koper. Modelling units of study from a pedagogical perspective the pedagogical meta-model behind EML. Open University of Netherlands, 2001. http://dspace.learningnetworks.org/retrieve/33/ped-metamodel.pdf

[15] M. Dean, and G. Schreiber (editors). OWL – Web Ontology Language Reference. W3C Recommendation. 2004. http://www.w3.org/TR/owl-ref.

[16] A. Riera, E. Sánchez, M. Lama, R. Amorim, X. Vila, and S. Barro. Study of Communication in a Multi-Agent System for Collaborative Learning Scenarios. In *Proceedings of the Twelfth Euromicro Conference on Parallel, Distributed and Network based Processing (PDP2004)*, pages 233-240, A Coruña, Spain, 2004.

[17] I. Horrocks, P. Pater-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Submission, 2004. http://www.w3.org/Submission/SWRL.

Semantic annotation for the teacher: models for a computerized memory tool

Faiçal AZOUAOU, Cyrille DESMOULINS
CLIPS + MTAH & Grenoble University
BP 53, 38041 Grenoble cedex 9, France

Abstract. This article aims to propose a model of semantic annotation dedicated to the teacher. This model must adapt to the teacher's activity specificity, who needs to master both a pedagogical and domain expertise. In this paper, we analyze the particularity of the teacher's annotation language that enables the teacher to express his own expertise. First, we identify the concepts of this annotation language used by the community of teachers. Then we propose a conceptual model of this language based on ontologies. We use these ontologies to propose an annotation model (MemoNote) in order to enable the teachers annotating using their own language. In order to check the validity of this model (ontologies and annotation language) in representing the teachers' semantics, we describe the results of an investigation done with chemistry teachers. Finally we specify the external and internal representations of the annotation tool for the implantation.

Introduction

The teacher during his activities handles various teaching documents (designing, reading, reviewing... etc). At the same time, he needs to memorize ideas and corrections or to plan actions to be made. This memorization is often materialized by annotations that the teacher puts on these documents.

Whereas the teacher nowadays uses more and more software tools to handle teaching documents in a digital format, annotation is still often made on paper, which requires the teacher to first print his documents, and implies an extra effort. For example, the result of the IMAT european project [1] pointed out the need for teachers to be able to annotate directly on the digital document and to manage a software memory of his activity. Thus, it appears necessary to propose a software annotation tool enabling teachers to express and clarify their feedbacks directly on the digital teaching documents.

As in any community of practice [2]; [3], the teachers' community uses a specific shared language to annotate. This language is a set of common forms and concepts. Consequently, the teacher needs a dedicated tool of annotation that integrates this language.

The object of this article is indeed to identify the teacher's annotation language to be used as a basis for defining a tool of semantic annotation dedicated to him. The assumption of our research is that the memory resulting from this tool could enable the teacher to improve the effectiveness of his teaching and support its activity trough the remembrance it provides.

The article is organized as follows. In the first section, we specify the meaning of a semantic annotation and we explain why this semantic is important. In order to provide semantics to teacher's annotation, we characterize in section two the teacher's annotation language. Starting from this characterization, we define in the third section the basic concepts of this teacher's annotations language, represented with dedicated ontologies. We then use these ontologies and a generic annotation model to propose a conceptual model for a dedicated teacher annotation language. We explain in section five, how we can use this ontology based model to improve the annotation's retrieval. In order to check the validity of this conceptual model (ontologies and annotation model) in representing the teachers' practices, we present in section six, the results of an investigation done with chemistry teachers. In the last section, we describe the implementation of this conceptual model in a teacher dedicated annotation software tool in terms of external and internal representations.

1. What is a semantic annotation?

Before studying the annotation language, we need to specify precisely what an annotation is in general and what a semantic annotation is in particular. Some authors provide informal definition of an annotation, varying upon the research field Human Computer Interaction (HCI[4], the cognitive science [5], and the digital libraries and document retrieval field [6]. To sum up, all these informal definitions agree that an annotation is both an object added to a document and the activity that produces this object. This twofold view on annotation is also reflected in the formal definition we present here.

Euzenat [7] formalized semantic annotation in the context of the Semantic Web. From two sets of objects, documents and formal representations, two functions can exist: a function from document to formal representations, called *annotation* and a function from formal representations to documents called *index*. Usually, these two functions are created at the same time during an activity called annotation or indexing. The Semantic Web aims to provide annotations web documents with an explicit semantics for the computer, and not only for the human that created it or handle it. A Semantic Web will create an extension of the current World Wide Web, in which information is given well-defined meaning, so machines become much better able to process and "understand" the data that they merely display at present [8].

Marshall [15] makes a distinction between explicit and implicit annotations. An explicit annotation is an annotation that other readers can understand and interpret. At the opposite, an implicit annotation is “telegraphic, incomplete and tacit”. These annotations “pose interpretive difficulties for anyone other than the original annotator”. It is the case of highlighted text, of a cryptic asterisk without comment, etc. Thus, if every annotation has a semantic, it remains mostly implicit. Annotation semantics is implicitly carried, for a given annotator, by the shape used. For example, a given annotator used to underline in red the parts which he considers as important. The lack of explicit annotation semantics makes it difficult for other people to interpret them. A useful annotation tool should enable an annotator to explicit the semantic of his annotations.

2. Teacher's annotation language for authoring pedagogical documents

As in any community of practice [2]; [3] teachers use a specific shared community language to annotate. This language is a set of common annotation forms and objectives. The annotation's objective is considered as a central point by most of the authors, because it defines the semantics of the annotation.

Whereas we did not found any result about teacher's annotations objectives, some authors studied the various objectives an annotation could carry in general. Marshall studied university students' annotations and extracted the following objectives [2]: procedural signalling for future attention, place marking and aiding memory, problem-working, interpretation, tracing progress through difficult narrative and incidental reflection about the material circumstances of reading. This study also points out the fact that an annotation can serve several objectives at the same time.

Veron [9] and Huart [10] took up the objectives of Virbel [11] on active reading at the BNF (Bibliothèque Nationale de France). They identified four families of annotation goals: classifying (organising into a hierarchy, contextualising); adding information (reformulating, commenting, documenting); planning (scheduling, indirect annotating); correlating.

Finally, Mille [12] studied the didactic annotation of a text exam. She identified several annotation goals, grouped in two main categories: understanding the document and finding information.

A dedicated teacher's annotation tool should enable teachers to express their own expertise using their own annotation language as they are used to do on paper. To identify this language (teacher's annotation objectives), we first study the nature of his expertise. We consider, as [13] that the teacher's annotation is a language that references his self-expertise: while annotating the teacher is in fact transforming his implicit knowledge into an explicit form.

Teaching expertise has many facets, according to [14], he uses during his activity. The teacher organizes the **subject to be taught** (domain) in several lessons, and each lesson may include several topics that are combined into learning objectives. For each topic, the teacher defines appropriate **pedagogical presentations and activities** in order that the learners reach their learning objectives. They use **different teaching documents**. The teacher, then, should have two kinds of expertise:

1. **Pedagogical expertise**: knowledge about organizing the lesson, evaluating learners, designing pedagogical activities, asking good questions, etc.
2. **Domain expertise**: declarative or procedural knowledge of the domain to be taught.

Some annotations concern only the document itself: the teacher annotates to memorize elements **about the design, the structure of the document; f**or instance; the teacher annotates to correct a syntax error, to move a paragraph, or to add a picture.

Consequently, the teacher's annotations express his objectives relating to three different levels: pedagogy, domain and document.

- **The pedagogy level**: The teacher organizes the content to be taught in several lessons; he adapts the content to the learners' context. For each lesson, he designs different activities that help learners building their own knowledge. The teacher also decides to ask appropriate questions to learners and adapts the different lessons to the feedbacks and questions of the learners [14]. All these teacher's activities mean a high level of pedagogy expertise, that the teacher can memorize using annotation.
- **The domain level**: The domain level covers the knowledge specified in the content of the lessons. This knowledge can be rather general knowledge like "including/understanding the theory of relativity" or more precise one like "knowing the capitals of the countries of Europe". It can also be declarative knowledge such as "knowing the great cities of the world", or procedural one such as: "to know how to carry out an experiment of oxidation in chemistry's lab". We situate in this level the teaching activities specified in the program or by the teacher himself.
- **The document level**: The document level concerns what is related to the document itself; in particular it concerns two of its structure:
  - **Physical structure**: the document presentation, its typographical characteristics: font, colour, size, grease…
  - **Logical structure**: the role and the nature of each segment in a document: title, subtitle, paragraph, etc

The teacher augments his own memory using annotations about these three levels. The result of the memorisation is the objective of the annotation. For instance, during the design of the pedagogical document, the teacher annotates that he should review in the future some exercises' results (domain expertise), or add some definitions for the learners (pedagogical expertise).

3. Ontology based conceptual model for annotation semantics

In the previous section, we have classified the teacher annotation's semantics into three levels: pedagogy, domain to be taught and document. To model these objectives, we choose to use ontologies [15]. They formalise the concepts shared by a community and their relationship (hierarchy, metonymy, etc.) by providing a precise and explicit semantic. They define the scope of the set of "concepts" handled by the annotator and they also enable the annotations to be shared among people using the same ontologies.

To design these ontologies, for the field of chemistry (1st year of university), we have used the literature (for pedagogy and document levels); teaching and learning documents (for domain level).

3.1 Pedagogy annotation objectives ontology

The pedagogy ontology concerns the annotations that the teacher makes to memorize elements of the pedagogy level.

Table 1 Ontology of annotation's objectives on the pedagogy level

To memorize ...	To memorize...
1. Non significant objective.	4. learning objectives not ambitious enough
1.1. To ignore.	4.1. Compared to the students
1.2. To work if there remains time.	4.2. Concept already comparable by the students
1.3. To reduce	4.3. Not enough of concepts
2. Significant objective	4.4. Too low constraints on the situation
2.1. To deepen	4.5. Too much time in the meeting
2.2. To illustrate	4.6. Too low material constraints.
2.3. To evaluate	5. Non relevant learning objective
3. Badly elaborated objective	5.1. Not part of the program
3.1. Measurable objective not assessed	5.2. Already represented in another objective
3.2. undefined situation of training	6. Bad content of the text (spot of reading learning).
3.3. Non objective evaluation	
3.4. Non operable objective	

3.2 Domain annotation objectives ontology

This second ontology concerns teacher's annotations relating to the domain level. The teacher is more an expert within teaching and pedagogy than the domain he teaches, thus, he needs to annotate elements of this domain to not forget them. So, a novice teacher of database can annotate the SQL's syntax of a specific data base management system (DBMS), especially if this system is different from that he taught the previous year. This domain ontology depends on the topic to be taught. The domain of our study is the chemistry program (1st year of university).

Table 2 Ontology of annotation's objectives on the domain level

To memorize...

1. a lab result
2. the detail
   - 2.1. of an object
   - 2.2. of a domain's procedure
3. complements
   - 3.1. of a domain's object
   - 3.2. of a domain's procedure
4. references
5. possible errors
   - 5.1. Handling.
   - 5.2. Calculation
6. precaution
   - 6.1. Quality
   - 6.2. Safety.
7. to plan changes
   - 7.1. material problem
   - 7.2. time problem
8. irrelevant passages

To memorize ...

9. Bad structuring of tasks
10. bad composition of tasks
11. missing task
12. tasks too many
13. Bad distribution of the tasks' values
    - 13.1. tasks under-valued
    - 13.2. tasks over valued
14. Bad order of the spots
15. Bad content of learning task
    - 15.1. Error
      - 15.1.1. on the procedure
      - 15.1.2. formulate
      - 15.1.3. definition
      - 15.1.4. chemical equation
      - 15.1.5. On the resources' availability
    - 15.2. Difficulty not adapted to learners.
    - 15.3. risk
      - 15.3.1. bad safety
      - 15.3.2. Ambiguous data
16. mediocre text's content

3.3 Document annotation objectives ontology

Finally, the last ontology concerns annotations' objectives relating to the logical and physical structures of the document: titles, paragraphs, font, colours, etc. The teacher annotates to memorize different improvements and changes to do on the document. These annotations have an effect on the reading of the document (increased comfort of reading, better structured document, corrected errors...).

Table 3 Ontology of annotation's objectives on the document level

1. To restructure
   - 1.1. To give a title
   - 1.2. To treat on a hierarchical basis
   - 1.3. To synthesize
   - 1.4. To reformulate.
2. To add a personal remark
   - 2.1. To criticize
   - 2.2. To express a related idea
   - 2.3. To develop
   - 2.4. To express its own comprehension
   - 2.5. To add an example
     - 2.5.1. To solve a problem
     - 2.5.2. To explain textually
   - 2.6. To refer to another document
3. To categorize
   - 3.1. By importance's value
   - 3.2. By predetermined type
     - 3.2.1. Theorem
     - 3.2.2. Definition
   - 3.3. By personal type
   - 3.4. By content's similarity
4. To create a relation between two passages
   - 4.1. Relation presentation /detail
   - 4.2. Relation presentation / explanation
   - 4.3. Relation definition / explanation
5. To review
   - 5.1. textual error
   - 5.2. incomplete illustration / table
   - 5.3. missing illustration / table
   - 5.4. missing index/glossary
   - 5.5. incorrect assertion
   - 5.6. ambiguous content
   - 5.7. an indefinite abbreviation
   - 5.8. document's structure
   - 5.9. To remove a passage
   - 5.10. To reformulate a passage
   - 5.11. To add a passage
   - 5.12. To plan an action
   - 5.13. To support the attention
6. to spot

3.4 The relation between the three types of objectives

When the teacher annotates with an objective relating to the knowledge level, this annotation has often an effect on the two other levels (domain and document). Indeed, there is dependency between the three levels. Each learning topic of the domain is adapted to learners using pedagogy and then transcribed on teaching and learning documents.

We illustrate this dependency using an example: a teacher prepares his chemistry lab; he decides to plan an assessment (pedagogical element) during the lab. In order not to forget to do this assessment, he annotates his document with a comment. Before to go to the chemistry lab, he reminds with the help of this annotation, to add an assessment to the activity described by the document. Consequently, the teacher modifies the document (document level) by adding the assessment questions (pedagogy level), which relate to a specific knowledge objective (domain level).

4. Ontology based annotation model

In order to define a teacher dedicated annotation model, we use the ontologies specified in the previous section and a generic annotation model.

In [16], we presented a generic pedagogical annotation model (MemoNote) composed of three parts:

1 **The tangible part**: represents the visible part of the annotation (the form, the anchor...). It is constituted of the following attributes: physical anchor (URL + location in the document), visual form and the syntactic anchor (annotated content).

2 **The episodical part**: describes the context of the annotation (author, date, location...). It is made up of the following attributes: Author, Date, location, Activity and context of memorizing

3 **The semantic part**: express the meaning the author gives to the annotation. This semantics is represented primarily by an objective attribute. It is made up of the following attributes: objective, content, importance, confidence, recipient, activity and remembrance's context.

Among these three annotation parts, the semantic one is the most significant. Indeed the author of the annotation annotates for a given objective which is often implicit in the annotation form. The annotation is required to understand and re-use this annotation. Consequently, the loss of this implicit semantics makes the annotation useless.

In order to propose a dedicated teacher annotation model, we modify the generic model by adding elements relying to the teacher activity. In particular, we change the episodical and the semantic parts:

1 **The tangible part**: same as previous.

2 **The episodical part**: the teacher annotation model should specify the teacher annotation context: the type of the domain to be taught (chemistry, mathematics…), the type of activity (exercise, lab, course…), the phase of the teaching (before the course, during the course (with the learners). These data will be used by the teacher to retrieve annotations he has done in a particular context (for instance: last month during the lab).

The semantic part: we divide the semantic part into three categories which correspond to the three levels: pedagogy, domain and document. The annotation's objective (the main attribute of the semantic part) takes its value in one of the three ontologies specified in the previous section.

5. Semantic indexing using ontologies

As specified by Euzenat [7], each annotation action define an inverse function which is indexing. Concretely speaking, while teacher is annotating a document using our ontology-based model, he/she is simultaneously indexing it with concepts from the three ontologies of objectives. The result of the annotation process is a semantically annotated document that is indexed with ontological concepts instead of simple keywords. But unlike the semantic web indexing which aims to describe the objective content of resources, our annotation model enables the teacher to index documents using his own viewpoint using subjective annotations.

Ontology indexing provides several advantages while retrieving documents or annotations[17]. It helps users to select queries criteria thanks to the ontology guidance. For instance, in order to review and correct the domain-level errors on all his pedagogical documents, the teacher will define a search criteria by choosing in the document's ontology the concept "*to memorise possible errors*", which will displays him all the documents with annotations "*to memorise possible errors*". Consequently, the teacher will no longer use key words but only select a concept (or several) in one ontology (or more) and be guided by the ontologies hierarchy. In this way, the user interface's usability is improved.

6. Investigation

In order to check the validity of this model (ontologies and annotation language) to represent the teachers' practices, we describe in this section, the results of an investigation done with chemistry teachers.

We realized six interviews with chemistry teachers. We wanted to extract the semantics of the annotations they add on their teaching documents for chemistry's lab. First, during a six-month period, each teacher annotated his chemistry's lab document. We organise individual interview with each teacher. They were asked to bring their documents on which the colours of annotations were deferred. Then on the basis of these documents we ask the teachers to explain the semantics of each annotation using our ontologies.

This exploratory investigation provided several results. First, it partially validated our model based on three levels: i.e. we could verify that the teachers (chemistry in our investigation) really annotate their teaching documents according to objectives relating to the three levels. Then, this investigation provides us elements to make up the three ontologies by adding new concepts that were lacking to the teachers during the interview. For instance:

- Document related ontology.
    - to spot
- Domain related ontology.
    - to plan changes
    - material problem
    - time problem
    - irrelevant passages
- Pedagogy related ontology.
    - Mediocre text's content (useless pedagogically).

This investigation confirmed our assumption (described in section 3.4) about the existence of dependencies between the three ontologies. Indeed, to express the objective of a given annotation, teachers often indicated an objective taken in several ontologies. These dependencies are in single-directed: pedagogy to domain, domain to document. For example if the teacher annotates to ignore a given objective, this will have an effect on the pedagogical activity defined for this objective, which does not have any more reason to exist on the document and consequently will be removed from it.

On the other hand this dependency does not exist in the other direction: if the teacher annotates at the document level (to correct a misspelling for example), this will not have an effect on the knowledge or pedagogy levels.

7. Implantation's details

To implement our teacher's annotation model, we re-use our generic annotation tool called "MemoNote". MemoNote enables users to manage the note of events and knowledge they want to memorize during their pedagogical activities (teaching or learning) and to retrieve them in the future. The MemoNote project aims at formalizing and implementing computerized external memories made of notes added directly and voluntary on the training material by its user. It covers memorization and remembering tools, for individual and groups, mainly for teachers and learners. The MemoNote annotation tool represents the memorization part of the project. It is currently dedicated to personal annotation but not to teacher. The first mock-up of the tool has been implemented on TabletPC computers for pencils based annotations, extending the MobiPocket reader software[18]. A quite similar web based mock-up has been developed too, mainly to provide retrieval and synchronisation functionalities.

7.1 MemoNote background

MemoNote enables the user to annotate pedagogical documents. It is not dedicated to a special field of teaching or a specific type of activity. For a specific teaching activity, MemoNote can adapt the user's context by selecting a set of ontologies. This set of ontologies describes the users, the teaching domain, the pedagogical activities (content, location, time) and the annotation's objectives.

This ability to change its context with a set of ontologies makes MemoNote both a generic tool, which can be used in every context, and a specific one, once the context is fixed by ontologies. This formalizes the results of ecological studies on annotation of teaching material [19] demonstrating that the annotation process is rather generic whereas the annotation content (forms, objectives, etc.) depends upon the learning/teaching context.

7.2 External representations of the conceptual model.

While reading a document in its pedagogical activity, the MemoNote user annotates this document by:

- Defining the source of the annotation (tangible part): anchor: where it is located on the document, Visual form: the shape and color the annotation takes on the document.
- Defining the target of the annotation (semantic part): addressee, **objective (at document, domain and pedagogical levels),** content.
- defining the annotation link itself (episodic part): annotator, date, location, teaching context,

Fig 1 MemoNote interface

The user interface in both cases is the same (figure 1). It has three main parts. The first part is a reader (reading software) embedding MemoNote annotation tools. In the first mock-up, this reader is MobiPocket [18]. It provides reading facilities quite similar to paper ones. In this reading interface, the user can choose an annotation tool (for example red underlining) and put it on the document surface (on the touch screen). The second part is the annotation interface where the user can define (or not) each semantic fields (addressee, objective, content, importance and confidence) shown on the top the TabletPC on figure 3. The third part is the ontology browsing interface. For each attribute the user want to define, this interface pops up until the ontological value of the field is fixed.

For some entirely automatic patterns, the interface for annotation and ontology browsing does not open and fields are filled in automatically. The main pattern type is a pattern where there is only one ontology to fix and a subpart of the ontology is selected by the pattern; and a pattern where some fields remains to be defined from scratch.

7.3 Internal representations for the conceptual model

Representing the semantics of annotation with ontologies is the same idea that in the semantic Web [8] approach (the main difference is that we use it for subjective annotation). We can then use the same languages as the semantic web to represent annotations and ontologies.

First, to represent and store the annotations we use RDF (Resource Description Framework)[20]. RDF is an infrastructure that enables the encoding, exchange and reuse of structured metadata. RDF is an application of XML that imposes needed structural constraints to provide unambiguous methods of expressing semantics. RDF additionally provides a means for publishing both human-readable and machine-processable vocabularies designed to encourage the reuse and extension of metadata semantics among disparate information communities [21].

Then, for representation of the different ontologies, we use OWL [22], otherwise known as the Web Ontology Language. OWL provides a language for defining structured ontologies that provide rich integration and interoperability of data. It uses both the URIs for naming and the description framework provided by RDF. OWL builds on RDF and RDF schema, adding more vocabulary for describing properties and classes as well as relationships between classes [23].

The use of these two semantic web standards guarantees the capacity to share annotations between different teachers, even if these annotations are made using different tools, because the two languages represent a unified data exchange format. These two languages also offer us the possibility to publish the annotated documents on the web; thus they can be indexed using the annotations by the new semantic web search engines like SWOOGLE [24] or they can also be processed by different web agents.

8. Conclusion

In this paper, we have demonstrated that the teaching activity is particular, because the teacher has both an expertise in the domain that he teaches and an expertise in pedagogy. When annotating, teacher expresses an annotation's objective which is related to the concepts of these two expertises. In addition, the teacher can annotate the physical and logical structure of the document itself. Thus, this teacher's annotation language enables the teacher to explicit objectives that belong to three different levels: pedagogy, domain and document. These objectives represent the annotation's semantics that remains mostly implicit. The lack of explicit annotation semantics makes it difficult to reuse the annotations.

We model this shared language using ontologies. We propose then an ontology based annotation model dedicated to the teacher. This model enables the teacher to explicit his annotation's objective using the three levels of semantics (pedagogy, domain and document). Using this model the teacher can retrieve his annotations easily guided by the ontologies.

We carried out a first validation by making an exploratory study with chemistry's teachers which enabled us to confirm our assumption and to supplement and correct our three ontologies of objectives. Lastly, we presented an implementation of this model using languages borrowed from the semantic web researches.

Our exploratory study relates to the particular case of chemistry. We need to check out in what extent our annotation model is generalizable to the other teaching's disciplines and the way of rapidly extracting annotation ontologies for a given domain.

References

1. de Hoog, R., Wielinga, B., Kabel, S., Anjewierden, A., Verster, F., Barnard, Y., DeLuca, P., Desmoulins, C., Riemersma, J. *Re-using technical manuals for instruction: document analysis in the IMAT project*. in *Integrating Technical and Training Documentation Workshop, ITS'02*. 2002. Biarritz.
2. Marshall, C. *Annotation: From Paper Books to the Digital Library*. in *ACM International Conference on Digital Libraries (DL 97).* 1997.
3. Bourdieu, P., *Outline of a Theory of Practice*. Cambridge Studies in Social and Cultural Anthropology. 1977, Cambridge: Cambridge University Press.
4. Baldonado, M., et al. *Notable: At the Intersection of Annotations and Handheld Technologies*. in *HUC*. 2000. Bristol.
5. Adler, M.J. and C.v. Doren, *How to Read a Book*. 1972, New York: Simon and Schuster.
6. Neilsen, J. *Online documentation and reader annotation*. in *Work with Display Units*. 1986. Sweden.
7. Euzenat, J., *Eight Questions about Semantic Web Annotations.* IEEE INTELLIGENT SYSTEMS, 2002. **17**(2): p. 55-62.
8. Berners-Lee, T., J. Hendler, and O. Lassila, *The Semantic Web.* Scientific American, 2001(May).
9. Veron, M., *Modélisation de la composante annotative dans les documents électroniques*. 1997: DEA, ENIB - ENSEEIHT - IRIT, Toulouse,. p. 53.
10. Huart, P., *Définition d'un poste de lecture active de documents électroniques*. 1996, ENSEEIHT Informatique: Toulouse.
11. Evrard F., V.J., *Projet de réalisation d'un prototype de station de lecture active et utilisation en milieu professionnel, Final report. French gouvernment grant #9300571*. 1996: Toulouse.
12. Mille, D., *Vers une mémoire individuelle de formation : annotation didactique de ressources pédagogiques. Internal report, Joseph Fourier University. Grenoble*. 2001.
13. Marwick, A.D., *Knowledg mangement technology.* IBM Systems Journal, 2001. **40**(4): p. 16.
14. Grandbastien, M. *Teaching Expertise is at The Core of ITS Research*. in *International Journal of Artificial Intelligence in Education*. 1999.
15. Guarino, N. *Understanding, Building, and Using Ontologies : A Commentary to 'Using Explicit Ontologies in KBS Development'*. in *IJHCS*. 1997.
16. Azouaou, F., Desmoulins, C., Mille, D. *Formalisms for an Annotation-based Training Memory: Connecting Implicit and Explicit Semantics*. *10th Artificial and Education conference*. 2003., Australia.
17. Hripcsak, G., et al., *Access to Data: Comparing AccessMed with Query by Review.* The American Medical Informatics Association, 1996. **3**(4).
18. Mobipocket.com, *Mobipocket reader software*. 2004: Paris, FRANCE.
19. Marshall, C. *Toward an ecology of hypertext annotation*. in *ACM Hypertext*. 1998. Pittsburgh, PA.
20. Staab, S., A. Maedche, and S. Handschuh. *An Annotation Framework for the Semantic Web*. in *Workshop on Multimedia Annotation*. 2001. Tokyo, Japan.
21. Miller, E., *An Introduction to the Resource Description Framework.* D-Lib Magazine, 1998.
22. Horrocks, I., P.F. Patel-Schneider, and F.v. Harmelen, *From SHIQ and RDF to OWL: The Making of a Web Ontology Language.* Journal of Web Semantics, 2003.
23. IBM, *Ontology-based Web Services for Business Integration.* 2005.
24. Ding, L., et al. *Swoogle: A Search and Metadata Engine for the Semantic Web*. in *Thirteenth ACM Conference on Information and Knowledge Management*. 2004. Washington D.C., U.S.A.

Community Support Based on Thematic Objects and Similarity Search

Niels Pinkwart, Nils Malzahn, Daniel Westheide and H. Ulrich Hoppe
University of Duisburg-Essen, Germany

Abstract. This paper describes an approach for community support based on similarity of learning objects: the current document a user is working on is used as a search template, which is matched against a learning object repository. The paper presents a simple similarity measurement, discusses potential enhancements, and shortly describes the results of a first usage study.

1. Introduction

Current educational practice shows a wide variety of computational tools being used by learners and learning groups in highly heterogeneous settings. The particular role of digital tools in these scenarios differs considerably. A common point for most of the networked applications is the use of digital media as a means for sharing and exchanging resources. This can be a very fruitful support for educational communities, since jointly used resources can play a key role for knowledge sharing and discovery. In addition, sharing resources offers potential for building and supporting communities of interest - groups of learners that have a joint interest in certain topics.

The construction of applications for these purposes of community support is a challenging task. Quite a number of applications and developments already exist in this field. Some approaches rely on user models to suggest communities and propose documents. These *recommender systems* typically have a weak point in that at least initial user models and/or document ratings have to be provided manually. Some techniques [4] try to address this problem with underlying *ontologies* - yet, still a manual rating of documents is necessary here. Finally, techniques like [1], which are able to dynamically recommend peers as interaction partners, usually need a detailed domain model for their calculations.

The approach presented in this paper relies on an alternative and simple conceptual model: it uses the learning objects created by the users as primary source of information. A repository service is able to propose *similar* learning objects - recommendations for artifacts which match the current context of the learner. These recommended documents can then either be accessed directly (anonymous object centered exchange), or can serve as a base for stimulating interaction among the users that created the “similar” objects.

2. Approach for a Similarity Search on Learning Objects

A critical point of the approach outlined in the introduction is that the similarity calculation needs semantically rich data in order to produce meaningful results. Standards like LOM, Dublin Core or IMS-LD are important contributions to syntactic and semantic interoperability, but they do not address three problems: First, the time-consuming *creation* of metadata is a necessity which most users try to avoid. Second, a *navigation* through document databases using traditional *retrieval* mechanisms and user interfaces is often based on complex electronic forms. In addition, free text input fields for specific metadata slots are of little help for retrieval of semantically similar documents. Third, a restriction to "standard" metadata is not likely to lead to fruitful retrieval, since the standards (have to) stay on a rather generic level.

To address these problems, the proposed approach relies on a partially automatic generation of metadata that exceeds current standards [5]. Using this generated data, the archive is queried for similar documents. This associative lookup enables users to find “interesting” documents without specifying exactly what they look for. Furthermore by applying ontologies potential collaborators sharing the same or topic-related interests can be pointed out to the user [2].

The results of these searches can then serve as a base for further navigation in the archive, which practically eliminates the need for manual input of search terms.

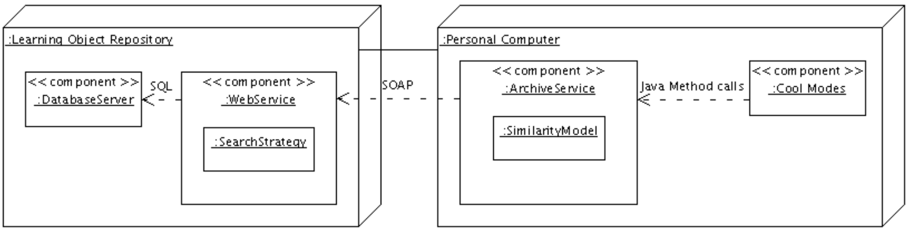


Figure 1. A deployment diagram of the system architecture

Figure 1 shows the system architecture . The flexible four-tier design, which allows each of the components to be exchanged provided that the technical interfaces are retained, includes two server-side components: the Learning Object Repository (LOR) [7], a central database where learning objects can be stored together with semantically rich metadata, and a web service which serves as an interface to transparently communicate with the repository. Two components are located on the client side: the concrete application used by the learner, and an archive service whose primary function is to access the web service. Details about the employed tools and the XML-based communication between them are described in [5].

Apart from the flexibility resulting from the multi-tier architecture design, also the core function of similarity search is customizable in two ways: on the client side, an exchangeable *similarity model* defines a measure for similarity of documents based on their metadata. This makes it possible to not only define which metadata of the source document are important, but more importantly to use any kind of analysis mechanism, from simple exact text matches connected with boolean logics to more sophisticated mechanisms. Metadata is preferred to full-text search because it abstracts over the concrete data format. So different sources of data can be compared. Similarly, an exchangeable *search strategy* component on the server side can be used to implement different retrieval methods reducing network traffic.

Our first implementation includes simple prototypes of similarity models and search strategies: the latter makes use of boolean retrieval in the sense that for each metadata slot that is considered important, a query with only this metadata slot is sent to the LOR. For each document, the number of (exact) matches is counted. Only documents trespassing a certain threshold are considered relevant. There, the prototype similarity model consists of a simple ranking by the number of "hits" (i.e., metadata slots that match).

3. Illustrative Example

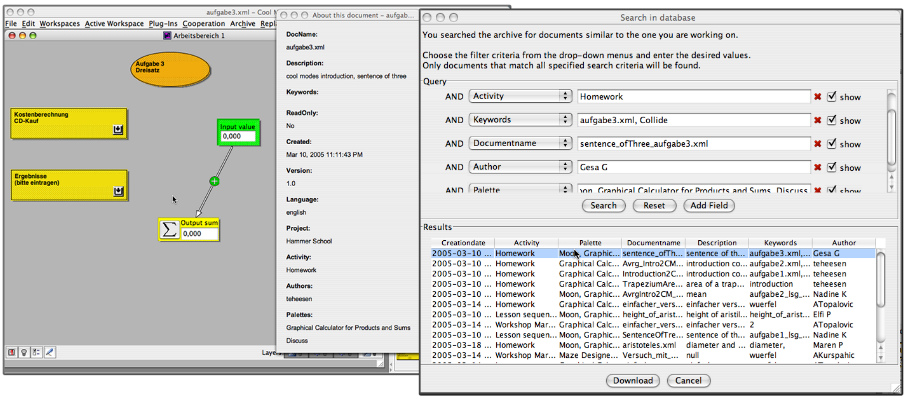


Figure 2. Similarity search example: a source document, its metadata and the search results.

To evaluate the presented architecture, the system was used in the maths lessons in a class of 20 students from a nearby higher education school. One of the students' tasks was to solve a word problem which involved applying the rule of three (cf. fig. 2). If students had problems they were allowed to use the document repository. Students looking for help made use of the "similarity search". Based on the semi-automatically generated metadata of the source document containing the task, the retrieval mechanism presented other students' suggestions for a solution to the task (cf. fig. 2). Thus, the search for similar documents provides valuable results to students because they can consult others' solutions to mathematical tasks in order to get a better understanding of the matter. In-depth evaluation studies are subject of subsequent research.

The similarity search also proved to be successful in more complex situations. For instance, when working on a document for calculating a diet based on human weight and energy needs, a similarity search finds documents from related domains of health, e.g. a system dynamics model for calculating people's blood sugar.

4. Conclusion and Future Work

This paper presented a flexible architecture enabling users of the Cool Modes system to search for similar documents in a given repository. Speaking in abstract terms, we allow the users to define queries to a repository containing learning objects by defining it in graphic notation using exactly the same elements they expect to find in the resulting documents.

Despite its simplicity, the results of the boolean retrieval mechanism currently used are promising. The next steps will be to implement more elaborated retrieval mechanisms.

The first step to a better retrieval method will be to weight the different attributes of the meta data currently used for retrieval purposes. For example, the author entry may be more decisive than the creation date. To get proper weightings, three approaches will be followed: (1) user defined weights, (2) TF-IDF-values[6] to improve the influence of characteristic entries, and (3) categorizing the documents into an ontology giving higher rankings to documents which belong to the same ontology node.

To prepare enhanced retrieval mechanisms, a measure must be defined. One approach is to define the measure per plug-in. The idea is that each particular plug-in has got certain semantics influencing the definition of similarity. For example, the exact position of the places and transitions does not matter when comparing two petri nets. In contrast to that, for a concept map the exact positions may be very important to decide if two different documents are similar.

While the proposed approach of defining distance measures on the basis of plug-ins is easily applied if only one kind of plug-in is used, some questions arise when using more than one plug-in. Since the use of multiple plug-ins is intended by our applications it must not be restricted. So we will establish a second level of weightings. The results of each plugin will be calculated and afterwards combined to get an overall result. This kind of approach has produced decent results on web documents [3] and seems promising in our case. The weights in later formula may then be adjusted based on implicit or explicit user feedback strategies.

References

[1] Mitsuru Ikeda, Shogo Go, and Riichiro Mizoguchi. Opportunistic group formation. In *Proceedings of the Conference on Artificial Intelligence in Education (AI-ED)*, pages 167–174, Amsterdam, 1997. IOS Press.

[2] Nils Malzahn, Sam Zeini, and Andreas Harrer. Ontology facilitated community navigation – who is interesting for what i am interested in? In A. Dey et al., editor, *CONTEXT 2005*, Lecture Notes in Artificial Intelligence 3554, pages 292–303, Heidelberg, 2005. Springer-Verlag.

[3] André Masloch. Ein intelligenter URL-Checker. Master's thesis, Fachbereich Informatik, Universität Dortmund, 2003.

[4] Stuart E. Middleton, Harith Alani, and David C. de Roure. Exploiting synergy between ontologies and recommender systems. In *Proceedings of the Semantic Web Workshop at the WWW Conference*, Honolulu, HI, 2002.

[5] Niels Pinkwart, Marc Jansen, Maria Oelinger, Lena Korchounova, and H. Ulrich Hoppe. Partial generation of contextualized metadata in a collaborative modeling environment. In *Workshop proceedings of the Conference on Adaptive Hypermedia (AH)*, pages 372–376. Technische Universiteit Eindhoven, 2004.

[6] Gerard Salton. *Automatic text processing — the transformation, analysis, and retrieval of information by computer*. Addison-Wesley series in computer science. Addison-Wesley, 1989.

[7] M. Felisa Verdejo, Beatriz Barros, Jose I. Mayorga, and Tim Read. Designing a semantic portal for collaborative learning communities. In R. Conejo, editor, *Lecture Notes in Artificial Intelligence 3040*, pages 251–259, Berlin, 2004. Springer.

Semantic Annotation in e-Learning Systems based on Evolving Ontologies

Delia ROGOZAN, Gilbert PAQUETTE
Télé-université of Québec
4750, avenue Henri-Julien, Montreal (QC), H2T 3E4, CANADA
drogozan, gpaquett@licef.teluq.uquebec.ca

Abstract. In this paper we discuss an approach for annotating e-Learning resources based on skill/performance and learning-domain ontologies and we propose a framework for managing ontology changes and their effects on the semantic annotation of resources.

Introduction

The research presented in this paper follows the initial ideas developed in ([**1**]) regarding the annotation of resources in e-Learning systems, and the ideas developed in ([**2**]) regarding the ontology evolution. In Section 1 we discuss an approach that uses skills/performance and learning-domain ontologies to annotate resources in a standard manner. In Section 2 we propose a framework for managing ontology changes in an appropriate manner; that is preserving the integrity of ontology-based annotation of resources after ontology evolution.

1. Enriching the Knowledge-based Annotation with a Competency Level

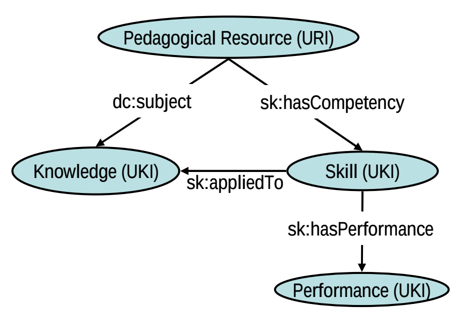
According to ([**3**]), a competency is the statement of a relationship among a knowledge, a skill and a performance degree: e.g. "Instantiate (*skill*) the agricultural practices (*knowledge*) in an expert manner (*performance*)". In order to annotate resources according to their competencies, we propose an RDF-based technique that uses two ontologies: a *learning-domain ontology* specifying a consensual view of a subject-matter ([**4**]) and a *skill/performance ontology* specifying the generic mastery levels that may be applied to any knowledge element from a learning domain ([**3**]).

The skill/performance ontology has two root classes: `Skill` and `Performance`. The `Skill` class contains four principal classes (`Receive`, `Reproduce`, `Create` and `Self_Manage`), each of them having more subclasses (e.g. `Instantiate`, `Apply`). The `Performance` class refers to the `Performance` degree (e.g. `Familiarized`, `Autonomous` or `Expert`) of a skill when it is applied on a knowledge. In order to describe the relations between these classes, three properties have been defined: (1) the `appliedTo` property, which asserts that any `Skill` subclass may be applied to a class from a learning-domain ontology; (2) the `hasCompetency` property, which states that anything may have a competency whose value is a `Skill` applied to a knowledge; (3) the `hasPerformance` property, which links a `Skill` to a `Performance` degree.

1.1 Standard Model for Binding Knowledge and Competencies to Resources

Figure 1 (a) shows the RDF-based model we propose for annotating resources. The `PedagogicalResource` identifies a specific document, tool, operation or actor. According to Dublin Core best practices, we link the `dc:subject` property to the entry in a learning-domain ontology, namely one of its `Knowledge` element.

The `sk:hasComptency` property connects a `PedagogicalResource` to one of its competencies defined by a `Skill` that (1) is applied to a `Knowledge` and (2) has a specific `Performance`. Therefore, the value of the `hasCompetency` property is an element from the `Skill` class-hierarchy, the value of the `appliedTo` property is an element from a learning-domain ontology and the value of the `hasPerformance` property specifies an element from the `Performance` class-hierarchy of the skills/performance ontology.




```
<rdf:RDF
<!-- Namespaces declaration -->
xmlns:dc= "http://purl.org/dc/elements/1.1/
xmlns:sk="http://example.org/skill_ontology/1.0/">

<rdf:Description about="Pedagogical Resource_URI">
 <dc:subject rdf:resource="Knowledge_UKI"/>
 <sk:hasCompetency>
   <rdf:Description about="Skill_UKI">
     <sk:appliedTo rdf:resource="Knowledge_UKI"/>
     <sk:hasPerformance rdf:resource="Perform_UKI"/>
   </rdf:Description>
 </sk:hasCompetency>
</rdf:Description>

</rdf:RDF>
```

Fig. 1(a) RDF model of an annotated resource; **Fig. 1(b)** Formalization of the related RDF model

Figure 1(b) shows the formalization of the RDF-based model using the RDF syntax ([**5**]). The namespace declaration (i.e. `xmlns`) provides a means to locate from where the elements with the related prefix are. The `rdf:Description` statement declares a resource, its properties and their respective values.

URI (*Uniform Resource Identifier*) ([**6**]) provides a standard mechanism for resource identification. In order to clearly distinguish between ontology terms and other type of resources (e.g. documents, services), we introduced the expression UKI (*Uniform Knowledge Identifier*). A UKI is as a compact sequence of characters, which explicitly and completely specifies a link to a unique element, in a specific ontology. An example is shown in Figure 2.

```
http://example.org/skill_ontology/1.0#Instantiate
scheme  authority        path           fragment
```

Fig. 2. UKI Example. The path includes the ontology name (i.e. `skill_ontology`) and version (i.e. `1.0`). The fragment specifies a unique ontology entity (i.e. `Instantiate`)

In this section, we have described an RDF-based technique to annotate resources (i.e. documents, tools or human resources) in a modular approach, according to their competencies (i.e. knowledge + skill/performance). Seeing that we can therefore search from resources according to their knowledge (e.g. documents describing the `AgriculturalPractices`) and/or competencies (e.g. actors being able to `Instantiate` the `AgriculturalPractices` in an `Expert` manner), this modular annotation will enable search-agents to suggest resources adapted to users' knowledge and competency.

2. Ontology Evolution in the Educational Semantic Web

Ontologies often evolve over time ([**2, 7**]). Changes in ontology domain, adaptations to different learning tasks, or changes in conceptualization require the modification of learning-domain ontologies. We define the *ontology evolution* as the process through which a former ontology version (V_N) is changed into a new version (V_{N+1}), while preserving the integrity of the ontology-based annotation of resources. This is an essential issue since changes such as the removal, merge or splitting of classes and properties may hinder the access to resources that were annotated with terms from corresponding ontology ([**8, 7**]). Hence, we propose a framework that supports the consistent ontology evolution in the Semantic Web context.

2.1 Framework for Managing Ontology Changes

The conceptual framework we propose to manage ontology changes is depicted in Figure 3. The **Ontology Workbench** allows the elementary (e.g. Add, Delete) and complex (e.g. Merge, Split) change editing.

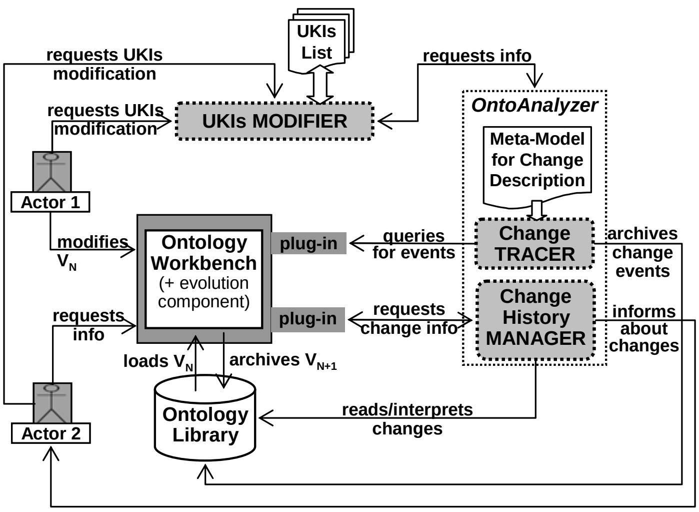


Fig. 3. Ontology Change Management Framework

The main functionalities of the **Change TRACER** are: (1) to track changes during ontology evolution and (2) to archive them to allow later retrieval.

At any time, the **Change History MANAGER** can provide users and software agents with information about: (1) the changes applied to V_N to obtain V_{N+1} and (2) the effects of these changes on the semantic annotation of resources.

The Change Tracer and the Change History Manager form together the **OntoAnalyzer System** we described in ([**9**]).

The goal of the **UKIsModifier** is to preserve the access to semantically annotated resources after the ontology evolution. The next section gives an account of this function.

2.2 Aim of Preserving the Integrity of Semantic Annotation based on Evolving Ontologies

Nowadays, a major issue emerges in the ontology research: “the interlinkage between objects and evolving ontologies need to be managed” ([**10**]). Consequently, we propose the **UKIsModifier** whose major function is to maintain the integrity of the ontology-based annotation of resources after the ontology evolution. To illustrate our proposal, let us consider a simple example of an ontology evolution (see Figure 4).

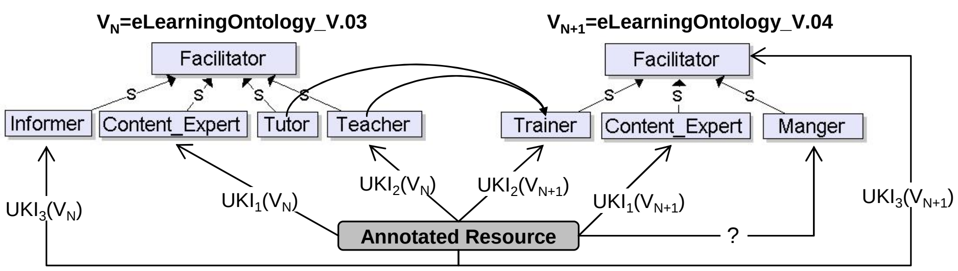


Fig. 4. Ontology changes: (1) the class `Informer` was deleted, (2) the classes `Tutor` and `Teacher` were merged into `Trainer` in the fourth version and (3) the class `Manager` was added in the fourth version.

In order to preserve the access to and the interpretation of annotated resources, the UKIsModifier modifies the UKIs associated to resources according to the changes applied to V_N to obtain V_{N+1} (see Table 1 and Table 2). These changes are previously identified by OntoAnalyzer, which also discovers the logical relations that may exist among the entities belonging to V_N and those belonging to V_{N+1}. For example: the class `Facilitator` from the fourth version includes the meaning of its subclass `Informer` from the third version; the class

`Trainer` includes the meaning of merged classes `Tutor` and `Teacher` belonging to the previous ontology version.

Table 1. UKIs modification for *full compatible* and *backward compatible changes* for which the interpretation of annotated resources via V_N is the same as when using V_{N+1}

Change Example	UKIsModifier function
Add an instance to the class `Content_Expert`	Modifies ONLY the ontology path (i.e. version name and/number) in the UKIs **UKI$_1$ (V$_N$) = *http://www.example.org/eLearningOntology/3.0# Content_Ex*** └→ **UKI$_1$ (V$_{N+1}$) = *http://www.example.org/eLearningOntology/4.0#Content_Ex***
Add class `Manager` to class `Facilitator`	Assists the users in defining new UKIs for the new knowledge (e.g. highlights the new class "Manager" and supports users to bind it to a resource).

Table 2. UKIs modification for *incompatibles changes* for which the interpretation of annotated resources is invalid or the access to them is hindered via V_{N+1}

Change Example	UKIsModifier function
Modifies the ontology path (i.e. version name and/or number) AND the fragment identifier in the UKIs	
Merge classes `Tutor` and `Teacher` into `Trainer`	**UKI$_2$ (V$_N$) = *http://www.example.org/eLearningOntology/3.0# Teacher*** └→ **UKI$_2$ (V$_{N+1}$) = *http://www.example.org/eLearningOntology/4.0#Trainer***
Delete class `Informer` from the subclasses of `Facilitator`	**UKI$_3$ (V$_N$) = *http://www.example.org/eLearningOntology/3.0# Informer*** └→ **UKI$_3$ (V$_{N+1}$) = *http://www.example.org/eLearningOntology/4.0#Facilitator***

3. Conclusion and Future Work

In this article we have presented an RDF-based model for annotating resources in a more fine-grained manner; that is annotating resources by their knowledge or competencies. We have also described a framework for managing ontology changes and for maintaining the integrity of semantic annotation of resources after the ontology evolution. The goal of our future work is to complete the development of the UKIsModifier system and to implement it within the TELOS system being built in the LORNET (Learning Object Repository Network) project.

References

1. G. Paquette and I. Rosca, *Ontology-based indexing for selecting actors, operations and resources in an on-line learning system*, SW-EL'04: Semantic Web for E-Learning, 2004.
2. D. Rogozan, G. Paquette and I. Rosca, *Gestion de l'évolution de l'ontologie utilisée comme référentiel sémantique dans un environnement de téléapprentissage*, TICE International Symposium, 2004, 245-252.
3. G. Paquette, *Instructional engineering for network-based learning*, Wiley-Pfeiffer, 2003.
4. J. Breuker, A. Muntjewerff and B. Bredewej, *Ontological modelling for designing educational systems*, AI-ED'99 Workshop on Ontologies for Educational Systems, IOS Press, 1999.
5. W3C Recommendation, "RDF/XML syntax specification," 2004, http://www.w3.org/TR/rdf-syntax-grammar/.
6. T. Berners-Lee, "Uniform resource identifier (URI): Generic syntax", 2005, http://www.gbiv.com/protocols/uri/rfc/rfc3986.html.
7. M. Klein and D. Fensel, *Ontology versioning for the semantic web*, International Semantic Web Working Symposium (SWWS), 2001.
8. J. Heflin and J. Hendler, *Dynamic ontology on the web*, 17th National Conference on AI, 2000.
9. D. Rogozan and G. Paquette, *Managing ontology changes on the semantic web*, The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05), (submitted paper), 2005.
10. S. Staab, A. Maedche and S. Handschuh, *An annotation framework for the semantic web*, The First International Workshop on MultiMedia Annotation, 2001.

Semi-automated Semantic Annotation of Learning Resources by Identifying Layout Features

Sylvain DEHORS*, Catherine FARON-ZUCKER**, Jean-Paul STROMBONI**, Alain GIBOIN*
**ACACIA, INRIA Sophia Antipolis*
2004 route des Lucioles, 06902 Sophia Antipolis cedex, France
Sylvain.Dehors,alain.Giboin@sophia.inria.fr
***MAINLINE, I3S, UNSA*
930 route des Colles, bât ESSI,06930 Sophia Antipolis cedex, France
faron,stromboni@essi.fr

Abstract. It is now widely accepted that any kind of digital content must be somehow semantically annotated to be intelligently used by computer programs. Annotations can be metadata, descriptions, etc. When dealing with learning, most systems require the author to manually annotate resources so that the system can deploy a navigation strategy, an adaptive behavior etc. However this task is very problematic, and often reveals to be an overwhelming enterprise. In this paper we propose a methodology, based on the reuse of existing pedagogical documents to achieve a semi-automated extraction of semantic annotations by identifying semantic information contained in the layout. We are applying this methodology in the design of a Web Based Learning System.

1 Introduction

The now classical approach to learning systems design is to rely on semantic annotations of pedagogical documents; this applies whether in the domain of LMS (Learning Management Systems), LCMS (Learning Content Management Systems) or LOR (Learning Object Repository). Exchanging metadata that are both understandable by humans and interpretable by machines is also the vision of the Semantic Web and languages, like RDF and OWL, are the key to express semantic annotations in a standard way.
In order to be used by computer systems, annotations must be expressed using a strict vocabulary often related to a model. This model, whether a "thesaurus", a "domain model" [1] or an "ontology" [2], provides common references to annotate resources.
One of the major pitfall, is the creation of those annotations manually (by humans), automatically (through automated programs), or semi-automatically (both by humans and programs). In this paper we first have a look at the existing approaches to annotate learning resources. Then we propose a methodology for semi-automatically extracting semantic annotations from pedagogical documents by identifying the semantic information contained in the layout. Finally we present the application of this method to the design, implementation and use of a simple WBLS based on semantic technology.

2 Existing methods and tools for the annotation of learning resources

As most learning systems use tailored courses, they require teachers to specifically create each document used by the system. Teachers are provided with authoring tools [1] to create new documents. Most of the existing research tools generate information in proprietary formats, whereas international standards like SCORM are being more and more enforced by commercial products. But this requires a lot of work and imposes major constraints upon the author. Moreover it does not take advantage of the huge amount of learning resources already available, both on the web or in the author's personal resources.

Another approach is then to consider reusing existing material by complementing it with extra annotations. To create such annotations, teachers are provided with annotation editors dedicated to learning resources. The interface appears often as a form to fill in and it is quite difficult to fill them with relevant and coherent information. For example several experts might not agree upon a document's content, and what concepts can be used to annotate it. Reusing existing material also implies to work on the size of the content; sometimes a resource can be a complete book [3]. To propose enhanced navigation it is necessary to slice the content into smaller chunks.

Here we propose a methodology for reusing document content and displaying it in a WBLS without relying on a specific annotation tool with form-based annotation.

3 Annotation of learning resources based on document layout features

The basic assumption we rely on is that the annotation task must be straightforward for the teachers and must not impose them to use the underlying formalism chosen for storing the annotation. For example it is not a viable option to manually edit HTML or RDF files.

We argue that every course is based on a learning or pedagogical model, which includes some pedagogical strategy. So first, the teacher is asked to explicit the pedagogical strategy for his/her course. For example in our case study we focused on a question-based approach to motivate learners to read the course. Then, the annotation task consists in interviewing the teacher, who is also the author of the document, and making him/her explicit the model of the existing document and how this model supports the envisioned educational strategy. Once this model is defined, the annotation task consists in identifying in the layout of the document the markers of the elements of this model. For example if the model defines the concept of "important notion of the domain", it is very likely that the corresponding word will appear in bold somewhere in the page, and so on. This kind of "visual" information must be gathered and standardized through a discussion with the author. Then a phase of re-authoring according to this layout principles must take place to ensure that all the visual clues are present to identify each component's role according to the model. The final step consists in formally creating instances of the components of the document through an automated process. Here we argue that the automation task is not very hard as most formats used for courses today (.doc, .ppt, .tex) are well structured. For example Word documents can be exported to XHTML and then treated by XSL transformation to extract annotations in the desired formalism (XML, RDF). We have successfully applied this methodology in the experiment described below.

4 A Web-based Learning System Design Experiment

We have developed a system called QBLS (for Question-Based Learning System) to demonstrate the use of semantic web technologies for setting and running a Web based learning system. The course we took for our first experiment is an introduction to signal analysis for first year computer science french engineering students. This course has previously given rise to thoughts about the use of information technologies in education. The aim of the experiment was to set up a knowledge pool where pedagogical resources are annotated in RDF and semantic queries are performed by the semantic search engine Corese [5].

4.1 Acquisition and representation of pedagogical knowledge

Our original document was a unique Microsoft PowerPoint File, supporting one hour of formal lecture. It was following an implicit model, for example the curriculum objectives were explicitly written at the top of every slide, and a set of relevant questions were given to motivate the students. Our task was then to formalize this underlying model that, we claim, exists in any pedagogical document of reasonable quality. This document was used as a support for oral teaching but was also given to the student as a hard copy course reference. This is a very common practice at university level, so this example is quite significant.
We had to model the document to reflect the pedagogical strategy chosen by the teacher. It is important to notice that our teacher was the author of the document and further investigation is needed to determine if someone else could have done it. This is crucial as reusing material is seen as the way to reduce the high-cost task of creating adaptable digital learning material. At least we demonstrated here that this was possible for the author to re-use his own documents. The defined model has no ambition of being generic, as our rationale is to save time and effort for teachers and the resources are to be used by a specifically designed WBLS.

4.2 Ontology of the pedagogical document

Because the final annotations would be expressed in RDF we formalized the model in an RDFS ontology. The pedagogical roles expressed in the model (like “definition”, “example”, etc.) formed the conceptual vocabulary we used to create the ontology. This ontology is a “pedagogical” one as defined in [2]. In the approach described in this paper the domain is solely represented by a set of concepts. It doesn’t seem to be possible to extract more information about the domain (like relationships between concepts) without a more sophisticated approach and then more work for the teacher. This approach is quite different from those relying on a model/ontology of the domain to “describe the content” of learning resources. Here the domain model comes solely from the document and only pedagogical information is to be used by the targeted WBLS. The ontology designed here is not meant for sharing resources across the globe but just between a teacher and his/her students, for more details see [6]. Once we had defined the ontology, a set of styles was created in Word. The teacher had to apply them on the document to identify the different components of the model. We only relied on the layout to generate annotations. For example words in italic, refer to the concepts in the course. It is important to notice that this was done with the sole use of the usual Microsoft Word program the teacher is familiar with. This technique was applied in [1] but we express much more information here.

4.3 Semantic-based retrieval of resources for visualization and navigation

QBLS uses the semantic search engine Corese [5] to perform queries on the RDF annotation base. We use it to retrieve the annotated resources on demand. Using the system the learner visualizes resources relative to a concept of the domain, and navigates from one to another resource either by following “seeAlso” links which leads to other concepts or by

gathering information on the same concept by accessing the definition, example(s), etc. available. According to the classification proposed by [1] our approach is to set up a “concept-based hyperspace”.
When asking for a new concept a query is sent to the Corese engine through a Web server that sends back an RDF result. That result is then displayed in a browser using an XSLT stylesheet. We must stress here that most of the development effort was spend on the generation of the annotations. The remaining part of presenting the resources in a web browser was done with very little development time, as most of the job is done by the existing search engine. The QBLS system has been used during a one hour formal course and a two hours exercise session by 49 students. The conceptual navigation provided by the system was quite appreciated by the students. They gave a high score (4.2 out of five) to a “system usability” questionnaire given after the session.

5 Conclusion

In this paper we have presented a methodology for semi-automatically extracting annotations from existing pedagogical documents. The corner stone of our method is that it does not require the use of any specific annotation tool but a little re-authoring work for the teacher, just manipulating the layout. For this re-authoring task the teacher can use the tool he is used to, which is a major incentive for him/her. This methodology also relies on the collaboration between the teacher and an ontologist/modeler to decide on the model of the document. This method relieves the teacher from the burden of authoring and heavily annotating the whole course. The application of our method to the design of a Web based leaning system has lead to an effective experiment showing that the semantic expressivity of the annotations acquired from the layout of the original document is quite sufficient to support a dynamic navigation in the so-built QBLS system. This work also pointed out some weaknesses of the existing standard models as they require far too much effort and maybe cannot even be effectively put in practice by a normal teacher. In the next experiment we will refine the methodology sketched out here and show more advantages of using standard semantic web technologies for designing and implementing WBLS.

References

[1] Brusilovsky, P. (2003) Developing adaptive educational hypermedia systems: From design models to authoring tools. In: T. Murray, S. Blessing and S. Ainsworth (eds.): Authoring Tools for Advanced Technology Learning Environment. Dordrecht: Kluwer Academic Publishers, 377-409.

[2] R. Mizoguchi, M. Ikeda, and K. Sinitsa (1997) Roles of Shared Ontology in AI-ED Research -- Intelligence, Conceptualization, Standardization, and Reusability. Presented at AIED-97, Kobe.

[3] P.Rigaux and N.Spyratos (2003) Metada Management and Learning Object Composition in a Self e-Learning Network, in proc. of Workshop on Information Search, Integration and Personalization, Japan.

[4] Stromboni J.P. (2002), Un cours introductif au traitement du signal à travers les applications audio de l’ordinateur multimédia, in Technologies de l’Information et de la Communication dans les Enseignements d’ingénieurs et dans l’industrie (TICE 2002), Lyon.

[5] Corby O., Dieng R. et Faron C. (2004), Querying the semantic web with the Corese search engine, in proc. of the European Conference on Artificial Intelligence (ECAI 2004), subconference PAIS, Valencia.

[6] Dehors S., Faron-Zucker C., Giboin A., Stromboni J.P. (2005), QBLS : web sémantique de formation pour un apprentissage par questionnement, In Environnements Informatiques pour l'Apprentissage Humain (EIAH 2005), Montpellier.

Ontology-based Access to the Resources of a Course Memory

Dominique LENNE, Marie-Hélène ABEL, Claude MOULIN
UMR CNRS 6599 Heudiasyc
University of Technology of Compiègne, France
{Dominique.Lenne,Marie-Helene.Abel,Claude.Moulin}@utc.fr

Abstract. Many documents and resources can be provided to students in the context of distance learning. It is however often difficult to identify pertinent resources and to organise them in coherent sets. A first direction to address this problem is to build resources banks or learning objects repositories. Re-using these resources for a specific course unit often needs an instructional design work. In the Memorae project, we study another direction, which consists in building a course memory or more generally a "learning organisational memory", that can be directly used by students. In this paper, we describe the content of this memory, the ontologies on which it relies, and the navigation possibilities it offers. We show how this memory can be used to support self-regulated learning.

1. Introduction

The development of distance learning reduces contacts between learners and teachers and leads learners to be more autonomous and more active in their learning process. In the same time, they can have access to more and more documents and resources either produced in the context of the course they are involved in, or directly available on the web. It is however often difficult for learners and even for distance learning designers to identify pertinent resources and to organise them in coherent sets. In order to address this problem, projects aiming at building pedagogical resources banks and learning objects repositories have been launched.

Usually, these repositories do not deliver ready-to-use material. An instructional design work is often needed. They are therefore designated rather for instructional designers or teachers in order to allow them to help them to build adapted courses or training. In the framework of the MEMORAe[1] project [1], we propose on the contrary to give the students a direct access to the learning resources. This suppose to do earlier a part of the instructional design work, by selecting pertinent resources, organising them and giving students means to self-regulate their learning process [2].

In this paper, first we present the classic approach of resources banks and learning objects repositories, then we describe the main characteristics and contents of the learning organisational memory we designed in the MEMORAe project. Finally, we present a pilot application that we developed for an applied mathematics course at the university of Picardy.

[1] MEMORAe stands for ORganisational MEMory Applied to e-Learning. This project is supported by the region of Picardie (France)

2. The Memorae Approach

Since a few years, many projects aiming at building bases of learning resources, in order to share and re-use them, have been launched. There are two kinds of resources bases : learning object repositories that group many subject matters (for example Merlot [3] or Ariadne[4]) and thematic resources bases [6]. In both cases the resources are not ready to be used by learners. An instructional design work is usually needed before.

On the contrary, within the MEMORAe project, our goal is to let learners directly access the resources of a course memory. Following a knowledge engineering approach, we organise the resources in an organisational memory. Actually, it is a course memory, in which a course is seen as an organisation (see section 3). A course memory is different from a learning memory [5] because its goal is not to help learners to remember what they previously studied. It can rather be seen as a memory of concepts and resources that teachers or designers find useful in the framework of a particular course.

In order to give learners a direct access to the memory, a part of the instructional design work has to be made earlier. The advantage is that the memory is ready to be used by learners, provided that pedagogical and didactical choices made earlier are acceptable. This can therefore lead to a loss of flexibility, but we make the assumption that these choices can at least by shared by a community of teachers, that could act as a "community of practice".

3. A Course Memory

The environment of a given course or training can be seen as an organisation. Hence, different actors (teachers, learners, administrative staff, etc.) are involved in this environment. That is why, in the MEMORAe project, we propose to manage resources, information and knowledge of this kind of organisation by relying on an organisational memory and more precisely on a "course memory". This memory can be accessed by teachers when they want to re-use resources, as in a thematic resources base. But our main goal is to allow learners to directly use the memory. Let us see the contents of this memory.

3.1. Contents of the Memory

The course memory contains the resources and the notions regarded as pertinent by the teaching team for a given course. It relies on two ontologies allowing to organise and index the resources.

Resources can be very different from one to another. They vary according to their size (web page or book for example), their nature (course, exercises, definitions, case studies, etc.), their form (book, report, web site, etc.), their medium (paper, video, audio, etc.), … A resource can be present in the memory, if it is digital, but it can also only be referenced, in case of non digital or external resources.

Notions are not only chosen because they are related to the course theme. They are selected on the basis of a didactical work. For example, in the context of a course on algorithms and programming, why and how to decide to establish a link between the notions of "array" and "loop" ?

Resources are selected and indexed relying on this work. Indexing is not made in an automatic way. The course manager, with the help of an editing committee if needed, is responsible for the pertinence of this link. It is not because a document treats of a notion that it will automatically be indexed by this notion. This is the result of a choice, that is to say that the document must have been judged suited for the learning of this notion. These decisions result from the pedagogical goal the course manager wants to achieve.

3.2. Ontologies

We chose to model our course memory with the help of ontologies. By using ontologies, our goal is on one hand to define a vocabulary that can be shared by all the actors in order to characterize the notions to learn, and on the other hand, to organise the access to the resources (see section 4). Building an ontology is quite a complex task, which is made easier by using a method. In the Memorae project, we used the OntoSpec [6] method. OntoSpec is a method of semi-informal specification of ontologies. It supposes that a conceptualization is made up of a set of concepts (or conceptual entities) and relations. The concepts in OntoSpec are organized in a taxonomy. Sub-concepts inherit all the properties of their super-concept. The relations make it possible to connect various concepts between them. Sibling concepts are organized in semantic axes according to their similarities.

We separate two ontologies [7] : the domain ontology which concerns the domain of training organisation in a general way and the application ontology which represents what is specific to a given course or training.

The concepts of the domain ontology are answerable to different types: persons (student, teacher, administrative staff, etc.), documents (books, slides, web pages, etc.), resource access (digital, solid), pedagogical features (e.g. activity type), or means to express a point of view (e.g. annotation).

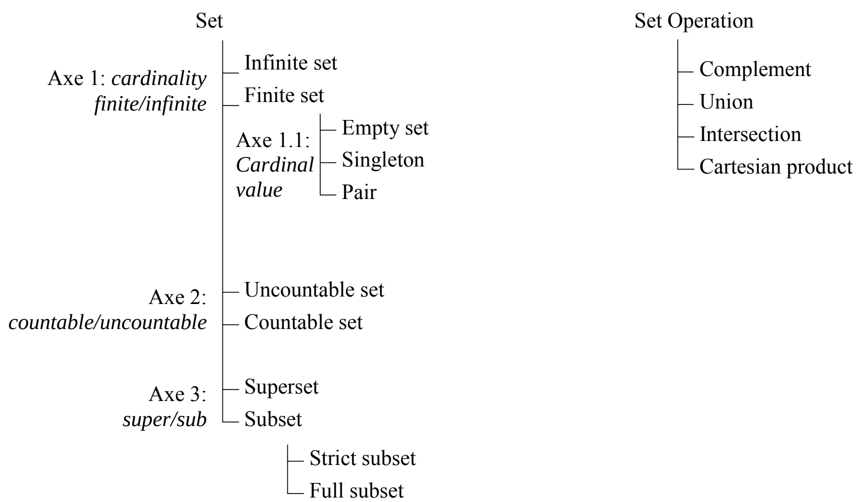


Figure 1. Excerpt of the B31.1 ontology

The application ontology describes the notions associated to a specific course. For example, Figure 1 shows an excerpt of the ontology of a course in applied mathematics (B31.1). An ontology is not only a taxonomy, it also includes a definition for each of the concepts, conditions on these concepts and relations between them.

Let us stress that an ontology is always constructed in connection with the application it will be used for. In the case we consider, the concepts correspond to notions to teach and to learn. There are relations between these notions, for example the "pre-requisite of" relation, that can reveal different visions on the learning domain. Therefore the ontology we have constructed is not an ontology of applied mathematics, it is an ontology of a specific course in applied mathematics. However, we think that this ontology could be reused by teachers that share the same vision of applied mathematics learning.

The domain and application ontologies are not independent; they have to be connected. For example, to express that a document is an introduction to finite sets, the concepts of "introduction" and "finite set", that are not part of the same ontology, have to be linked. Moreover, pedagogical relations such as "pre-requisite of" or "uses" are defined in the domain ontology, while other that are more specific are part of the application ontology (e.g. “has cardinal number” in the B31.1 ontology). See [1] for more details.

4. Navigation in the Memory

The navigation in the memory relies on the two ontologies. The navigation interface presents for each notion a definition, the resources that are related to it and a part of the associated ontology (parent, childs, siblings). It also shows an history of the navigation and provides some entry points that are defined by the course manager. Entry points allow to directly access to a notion in the memory.

The definition of a notion can refer to other notions. This reference corresponds in the application ontology to a neighbourhood relation (except subsumption), such as : “prerequisite of”, “suggestion”, “uses”, etc. To this end we also defined an horizontal navigation allowing to access to these notions.

5. Conclusion

We presented in this paper the course memory we designed in the framework of the MEMORAe project. At the opposite of the approach that is generally adopted with learning objects repositories or thematic resources bases, this course memory is bound to be directly used by learners. This implies to do earlier part of the instructional design work. Let us note however that this approach is only feasible with learners having self-regulating abilities. Some features of the course memory we propose, such as the explicit representation of the notions to learn, the indexation of adequate resources with this notions, the navigation following the relations of the ontology, and the use of a private space, aim at facilitating this self-regulation.

References

[1] Abel, M.-H., Barry, C., Benayache, A., Chaput, B., Lenne, D., and Moulin, C., *Ontology-based Organizational Memory for e-learning.* Educational Technology & Society, 2004. vol 7(3).

[2] Boekaerts, M., *Self regulated learning: a new concept embraced by researchers, policy makers, educators, teachers, and students.* Learning and Instruction, 1997. 7(2): p. 161-186.

[3] MERLOT, *Multimedia Educational Resource for Learning and Online Teaching*. 2004.

[4] ARIADNE, *Alliance of Remote Instructional and Distribution Networks for Europe*. 2004.

[5] Azouaou, F., Desmoulins, C., and Mille, D. Formalismes pour une mémoire à base d'annotations : articuler sémantique implicite et explicite. in *EIAH 2003*. 2003. Strasbourg.

[6] Kassel, G. OntoSpec : une méthode de spécification semi-informelle d'ontologies. in *IC'2002*. 2002. Rouen, France.

[7] Breuker, J. and Muntjewerff, A. Ontological Modelling for Designing Educational Systems. in *Workshop on Ontologies for Intelligent Educational Systems, Ninth International Conference on Artificial Intelligence in Education, AI-ED'99*. 1999. Le Mans, France.

Towards a Corporate Semantic Web Approach in Designing Learning Systems. Review of the TRIAL SOLUTION Project

Michel BUFFA*, Sylvain DEHORS**, Catherine FARON-ZUCKER*, Peter SANDER*
**MAINLINE, I3S, UNSA*
930 route des Colles, bât ESSI, BP 145, 06930 Sophia Antipolis cedex, France
{buffa,faron,sander}@essi.fr
***ACACIA, INRIA Sophia Antipolis*
2004 route des Lucioles, BP 93, 06902 Sophia Antipolis cedex, France
Sylvain.Dehors@sophia.inria.fr

Abstract. The TRIAL SOLUTION EU project focused on the publication of personalized electronic documents based on existing scientific books. Its general approach consists in slicing electronic books into elementary learning resources and annotating them with metadata enabling the retrieval of resources by a semantic search. The annotated resources are published into a repository available for teachers or students to produce personalised teaching or learning materials with delivery tools. In this paper we give an overview of the project, emphasizing the authoring tool we have developed to annotate the learning resources, and we review it by the light of the Semantic Web.

Introduction

TRIAL SOLUTION EU project [1] focuses on the publication of personalized electronic documents based on existing scientific books. Designing new electronic documents from scratch is very expensive and print-oriented authors rarely possess the required competency to perform such a specialized task. On the other hand, most recent regular printed books have a digital format, e.g. Latex or Microsoft Word, that can help in automating the publication of personalised electronic documents from printed books. The benefits of the approach based on re-engineering existing materials have been described in [1] and the process itself is detailed in [2]. This was the starting point of the TRIAL SOLUTION project.

The general approach consists in slicing electronic books into elementary learning resources and re-engineering these resources by refining the slicing and annotating the resources with metadata on content, didactic features and interoperability interfacing, this in order to enable intelligent retrieval. The annotated resources are then published into an online repository available for teachers and students to produce personalised documents, or to just find relevant learning materials.

The TRIAL SOLUTION platform integrates three main services: the automatic extraction and annotation of learning resources from electronic books, the re-engineering of the repository of learning resources and the retrieval of learning resources based on their annotations. Our contribution to the project is the authoring tool for re-engineering the learning resources by improving on their initial slicing and by adding metadata to them. In the next section we will give more details on the book enhancement and learning resource

[1] TRIAL SOLUTION stands for *Tools for Reusable, Integrated, Adaptable Learning - Systems/standards for Open Learning Using Tested, Interoperable Objects and Networking* It was funded by the EU as part of the IST Program within the EU's Fifth RTD Framework Program. http://www.trial-solution.de/

extraction and annotation processes. In section 2 we then review the TRIAL SOLUTION project by emphasizing the similarity of its approach with Semantic Web approaches of automatic annotation of learning resources based on ontologies.

1. Extraction and Annotation of Learning Resources: the Trial Solution Approach

1.1. Automatic Extraction of Learning Resources from Books

The first step of the TRIAL SOLUTION process consists in automatically disaggregating a textual document into a set of slices, using a tool from the SIT enterprise[2] that works with structured format documents such as LaTeX or well-styled Word documents. It extracts an XML table of contents where each entry corresponds to a slice, i.e. a learning resource. Details on the DTD can be found on the project site. Sections, chapters, tables, figures, examples are some of the easiest slices that the tool can identify automatically, and course documents using a standard style can be sliced more efficiently.

The extracted learning resources are automatically annotated with metadata about the author of the original book, the semantic content of the resources, and the relationships between resources in the original book. Hyperlinks or *"see also"* sentences are used to determine relationships while keywords help in classifying some resource contents. The Splitter looks for keywords and sentences specified in a thesaurus for the book's domain. It is based on a Thesaurus Management tool and a Key Phrase Assignment tool. The first tool checks for consistency of the thesaurus and the second one manages a collection of mathematical key phrases, extracted from a standard textbook on mathematics [3].

However, this automatic process is almost always insufficient and may even produce some wrong guesses, as stated in [2]. The extracted learning resources have to be re-engineered using our authoring tool.

1.2. Re-engineering Learning Resources

We have developed the TRIAL SOLUTION re-engineering tool to allow re-engineers to fix and improve the automatically-produced base of annotated and interconnected learning resources. Experiences on book enhancement using this tool are described in [2]. Re-engineers must have good knowledge of the original book content, and are either the authors themselves or (more often) teachers using the book as course material.

Our re-engineering tool is a client of the learning resource server. It talks to the server using a custom protocol where both the requests and the values returned by the server are based on an XML encoding defined in the TRIAL SOLUTION DTD.

When connected to a repository of learning resources automatically extracted from electronic books, our re-engineering tool enables (1) assigning the resources a title, (2) editing their contents, (3) editing the tree structure of the whole repository: resources can be split, merged, deleted; sub-resources can be created or reorganized, and (4) editing the metadata associated to the resources. We distinguish between three kinds of metadata. *Types* represent the pedagogical role of the resource contents, eg, a definition, a theorem, etc. *Keywords* specify the topics that the resource contents address. *Relations* with other resources like "*references*", "*requires*", etc. build up a semantic network between resources.

Our re-engineering tool also enables editing the thesaurus upon which the metadata are built. The thesaurus can hold various relationships between keywords: subsumption, synonymy and relatedness, and a description can be specified for each keyword.

2.1. Related Work

---

[2] Infotech Slicing Technology (SIT) GmbH, http://www.slicing-infotech.de/de/index.php

A similar automatic slicing approach has been applied in the SeLeNe [4] project where documents follow the DocBook DTD. In the domain of Intelligent Tutoring Systems (ITS) and Adaptive Hypermedia Systems (AHS), where using small bits of educational content is a very common practice, the more-common approach is rather to author the content specifically through authoring tools [5].

Other projects like Ariadne [6] or UBP [7] have proposed "learning object repositories" (LOR) where resources are stored and annotated with standard metadata but no assumption is made on the origin of the resources, especially whether they have been authored on purpose or come from existing documents. The OLR [8] repository uses RDF for storing the annotations and takes advantage of the chapter structure of the global course to inherit annotations. All these LORs are provided with annotation tools that help filling in the metadata.

To sum up, the TRIAL SOLUTION re-engineering tool adopts the same strategy as LORs annotation tools, but the specificity of our approach relies in the coupling of a manual annotation process and a preliminary automatic phase of resource extraction and annotation.

2. Review of Trial Solution

2.2. Web Standards

The Trial Solution project is compliant with some major open standards such as the IMS Content Packaging[3] for modelling objects that have to be exchanged between the different tools of the platform.

Metadata for describing the resources are compliant with the Dublin Core Metadata[4], the IMS Learning Resource Metadata[5] and the LOM Learning Object Metadata[6] standards, but deviate when forced to deal with new type of material [9]. During the project, it was necessary to develop our own metadata specification due to shortcomings in the then-available standard specifications.

However the technology has evolved considerably since the design stage of the project. Web services have come into their own and would now be employed for the client-server exchange of complex objects. The semantic web languages RDF(S) and OWL would now best suit the formalisation of the learning resource metadata.

2.2. Towards a Corporate Semantic Web Approach

TRIAL SOLUTION fits right within the scope of the semantic web. We have built a repository of learning resources which are meant to be shared and reused. They can be efficiently retrieved by means of semantic annotations about their domains, their pedagogical roles and the structural and semantic relationships between them. At the end of the project we argue for a homogenous representation of all these metadata in RDF annotations based on ontologies of concepts and relations about the resource structure, domain, and author's pedagogy. In addition to homogeneity for sharing and reusing, this language provides a much richer expressivity than simple keywords for describing resources.

Another conclusion we draw from TRIAL SOLUTION is about the population and annotation of a learning resource repository. The experience convinced us that manual annotation of resources is overwhelming for teachers or other scientists when facing a large

---

[3] IMS Content Packaging Specification: http://www.imsproject.org/content/packaging/index.html
[4] Dublin Core Metadata Element Set, Version 1.1: http://dublincore.org/documents/dces/
[5] IMS Learning Resource Metadata: http://www.imsglobal.org/metadata/index.html
[6] IEEE Standard for Learning Object Metadata (LOM): http://ltsc.ieee.org/wg12/

amount of resources. We aim at automating this process as much as possible; the extraction of knowledge from structured format documents is quite conclusive and could be improved. We tend towards an approach of knowledge extraction from both textual documents and their authors. The acquisition, of the relationships between the author's pedagogical organisation and the layout of the document, will help an automatic interpretation of the layout. By doing so, the annotations about pedagogical features will be much enriched. Here we speak for minimizing domain dependent annotation in advantage of pedagogical annotation. This approach is quite feasible within learning resource repositories about specific domains: in this case some information about the domain becomes implicit.

Finally, in the TRIAL SOLUTION project, the end-user goals were quite general, which made the annotation task difficult: when composing a personalized document, he/she may search for any resources for anything at all. More specific learning scenarii and profiles should improve the adequacy between the annotation contents and the end-user requests.

Conclusion

The Semantic Web was only emerging at the design stage of the TRIAL SOLUTION project. However the TRIAL SOLUTION approach fits right within the scope of the semantic web. We built a repository of annotated learning resources for sharing and reusing. The resources are annotated based on an ontology, with knowledge about their contents, pedagogical features and relationships among them.

At the end of the project, we turn towards a corporate semantic web approach in designing learning systems, where (1) repositories are domain dependent, (2) the document authors are involved in the learning resource acquisition process and (3) user profiles and pedagogical scenarii are taken into account in the annotation contents. We are currently involved in the design of a corporate semantic web for learning experiencing this approach.

References

[1] I.Dahn. Slicing Book Technology – Providing Online Support for Textbooks, in Proc. of the International Conference on Distant Education (ICDE'2001), Dusseldorf, Germany, 2001.

[2] I. Dahn, M. Armbruster, U. Furbach, G. Schwabe. Slicing Books – The Authors' Perspective, in R. Bromme, E. Stahl (eds.): *Writing Hypertext and Learning: Conceptual and Empirical Approaches*, Pergamon, 2001.

[3] Bronstein, Semendjajew, Musiol, Mühlig, *Taschenbuch der Mathematik,* Verlag Harri Deutsch, 2000.

[4] P.Rigaux and N.Spyratos, Metada Management and Learning Object Composition in a Self e-Learning Network, in Proc. of Workshop on Information Search, Integration and Personalization, Japan, 2003.

[5] Brusilovsky, P. Developing adaptive educational hypermedia systems: From design models to authoring tools. In T. Murray, S. Blessing and S. Ainsworth (eds.): *Authoring Tools for Advanced Technology Learning Environment*. Dordrecht: Kluwer Academic Publishers, pp. 377-409, 2003.

[6] E. Duval, E. Forte, K. Cardinaels, B. Verhoeven, R. Van Durm, K. Hendrikx, M. Wentland-Forte, N. Ebel, M. Macowicz, K. Warkentyne, and F. Haenni, The ARIADNE Knowledge Pool System, in Communications of the ACM 44 (2001), no. 5, pp. 73-78.

[7] Simon B., Quemada J., A Reflection of Metadata Standards based on Reference Scenarios, 2002, http://www.wu-wien.ac.at/usr/wi/bsimon/publikationen/SimonQuemada-ReflectionOnMetadataStandards.pdf

[8] J. Brase, W. Nejdl, Ontologies and Metadata for eLearning, Handbook on Ontologies, pp. 579-598, Springer Verlag, 2003.

[9] W. Lenski, E. Wette-Roch. The Trial-Solution Approach to Document Re-use, in Proc. of Workshop on Electronic Media in Mathematics, Coimbra, Portugal, 2001.

Ontological Support for Teaching Strategy in Intelligent Visual Reasoning Tutor

Eric WANG and Yong Se KIM
wang@me.skku.ac.kr, yskim@skku.edu
Creative Design and Intelligent Tutoring Systems Research Center
Sungkyunkwan University, Suwon, Korea

Abstract. IVRT is an ITS for visual reasoning, using the missing view problem. It combines an ITS framework with a solid modeling kernel that supports hint-generating rules using geometric reasoning. We develop an ontology for IVRT's hint generation rules, and a separate ontology for IVRT's teaching strategy. Teaching strategy rules are stored in a custom text format, with compilation to Jess.

The ability to visualize and reason about geometric aspects of 3D objects is critical for success in many disciplines in engineering and architecture. The missing view problem [1], shown in Figure 1, is typically used in visual reasoning instruction. Two consistent, principal orthographic views are given, and the learner must provide the third view corresponding to a valid 3D solid object.

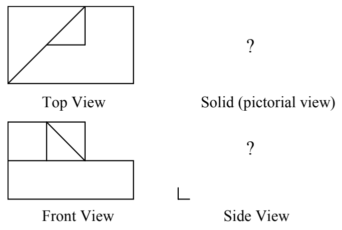


Figure 1. Missing view problem

We have developed an instructional software system called Intelligent Visual Reasoning Tutor (IVRT). IVRT's missing view problem solving module [2][3] implements a solid modeling kernel that can represent and display a partially-constructed 3D solid. It provides interactive sweeping operations [4] to incrementally construct faces of a solution solid.

The missing view module's hint generation knowledge is implemented as a set of rules in CLIPS [5], shown in Table 1. We develop an ontology to represent the knowledge behind these rules, shown in Figure 2, and instantiate a model of these rules, shown in Figure 3. This model explicitly represents the kinds of knowledge used by each rule, and identifies the major programming patterns involved in their implementation.

1. Create visible faces first (top and front faces).
2. Create hidden faces next (bottom and back faces).
3. Create faces using *construct* command last (right faces, then left faces).
4. Prefer the face that is adjacent to the most correct faces.
5. Prefer the face with the most incident edges.
6. Prefer the face whose normal vector contains the most zero components.

Table 1. Hint-generating rules

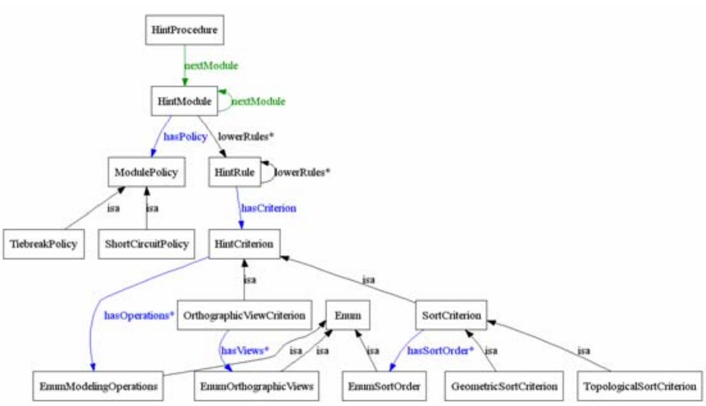


Figure 2. Ontology classes for hint-generating rules

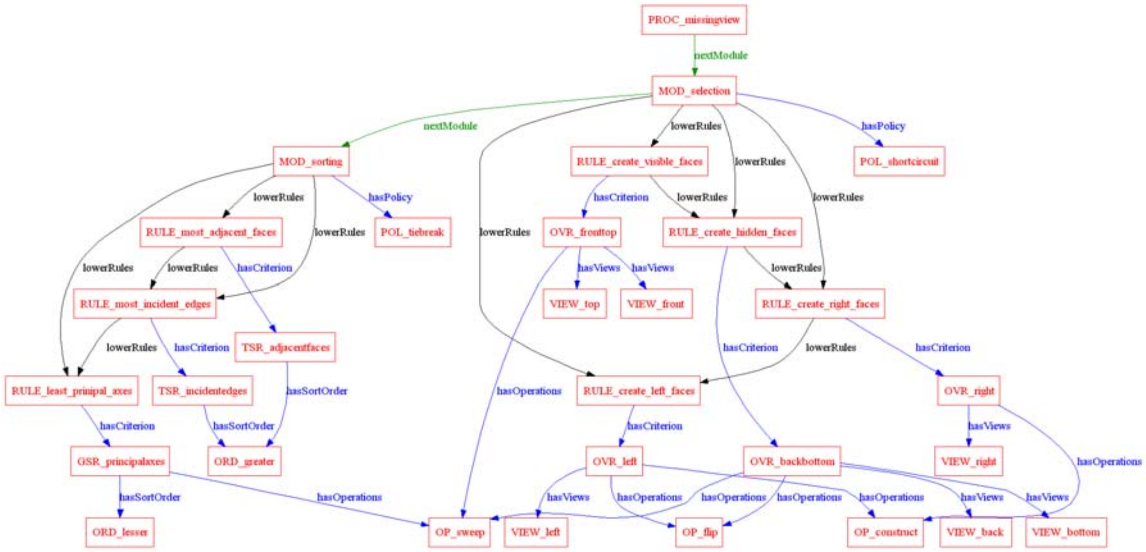


Figure 3. Ontology modeling for hint-generating rules

IVRT's ITS framework uses an ontology of learning contents and learner's status, shown in Figure 4. IVRT's teaching strategy is then represented as rules, which are stored in a custom text format based on the grammar shown in Table 2. This format is a syntactically simplified version of Jess, with quantifiers and implicit variables. Rules are compiled to Jess for execution, using a recursive-descent parsing approach implemented in Jess.

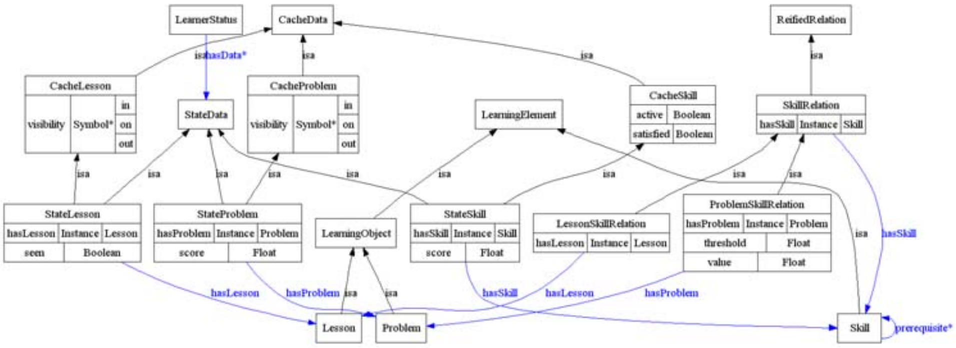


Figure 4. Ontology modeling of IVRT's learning contents and learner status

Rule = RULE $name FOREACH %main-class **HasExpr*** [THRU %prop] [IF **Antecedent**] THEN **Consequent**
HasExpr = HAS %prop = %value
Antecedent = **Quantifier** %reified-rel **HasExpr***
Quantifier = ANY | ALL
Consequent = [IN %modified-class] SET %prop %value

Table 2. Rule grammar for simplified rule format

Rule-3a. For each lesson that uses the conservative approach: if all associated skills are active, show this lesson.
RULE lesson-3a FOREACH Lesson HAS approach = conservative IF ALL LessonSkillRelation HAS active = TRUE THEN IN StateLesson SET visibility on

Table 3. Example of teaching strategy rule

References

[1] Wilde, D. J., "The geometry of spatial visualization: two problems", *8th IFTOM World Congress*, Prague, Aug. 1991.

[2] Kim. Y. S., Moon, C., Chauhan, S., Hubbard, C., Mengshoel, O., and Zhao, H., "Visual Reasoning Tutor (VRT): Instructional Software System for Missing View Problem", *Proc. ASEE Engineering Design Graphics Division Conf.*, Ames, IA, Nov. 1995.

[3] Hubbard, C., Mengshoel, O., Moon, C., and Kim, Y. S., "Visual reasoning instructional software system", *Computers and Education*, Vol. 28, No. 4, pp. 237–250, 1997.

[4] Zhao, H., and Kim, Y. S., "Geometric Operations for Visual Reasoning of a Solid from Orthographic Projections", *Advances in Engineering Software*, Vol.30, No.7, 1999.

[5] Mengshoel, O. J., Chauhan, S., and Kim, Y. S., "Intelligent Critiquing and Tutoring of Spatial Reasoning Skills", *AI in Engineering Design, Analysis, Manufacturing*, Vol.10, 1996.

Agent Based Learning Objects on the Semantic Web

Phaedra Mohammed and Permanand Mohan
Department of Mathematics and Computer Science
The University of the West Indies
St. Augustine, Trinidad and Tobago
fedre@lycos.com
pmohan@tstt.net.tt

Abstract. This paper presents ABLO, a first-attempt at engineering a more active learning object based on agent technologies that allows more sophisticated kinds of learning object reuse than what is currently available.

Introduction

Over the past six years, there has been tremendous interest around the world in the concept of a reusable digital learning resource, usually referred to as a *learning object* [3]. Although there are many advantages of the learning objects' approach, reuse and re-purposing of learning objects is a difficult task that is manually undertaken by content specialists. In [3], an implementation overview of a smart or active learning object is provided, and a range of features are suggested which could allow a learning object to function intelligently on the Semantic Web. Based on these features, we are developing Agent Based Learning Objects (ABLOs), a first-attempt at engineering a more active learning object based on agent technologies that allows more sophisticated kinds of learning object reuse than what is currently available.

Design and Implementation

Some of our design criteria for an active learning object are as follows:

a) It should be aware of itself and its environment and should be able to respond to changes in its environment
b) It should be capable of accepting input and exhibiting goal-oriented behaviour
c) It should be able to recommend itself and act without the direct intervention of humans or other learning objects
d) It should have control over its state and actions.

These characteristics can be met by treating a learning object as a software agent [2], hence the ABLO. In our design, the ABLO encapsulates the learning object and acts on its behalf, imparting agent characteristics such as self-awareness, portability and social interactivity. The ABLO contains a link to an ontology that describes the content of the learning object and provides semantic mark-up for the agent's own understanding of the learning object. The learning material itself needs to be marked up with terminology consistent with the ontology in order for the ABLO to properly identify and understand what parts of the material mean in terms of the context in which they are used.

Within the ABLO, inference rules and an inference engine are used to harness the semantic power of the ABLO and siphon out the relevant pieces of information within the ontology that pertain to a search or query. An ontology editor or a parser is another major component that the ABLO uses for traversal of the ontology and for building a knowledge base. Various agent behaviours are personified by the ABLO. One such behaviour is determining the suitability of a learning object when a request for learning material is made. Another behaviour extracts the specific parts of the learning material in an ABLO that are relevant to a request. A third behaviour allows the ABLO to behave socially whereby other ABLOs that it has been associated with in the past (based on learning material collaboration) are queried for partial contributions to a learner's request. These are assembled with part of its own content to form a new learning object.

Discussion

Our ABLOs can differentiate between learning objects that have material identified by the same syntactic terms but actually refer to entirely different concepts. Consider two learning objects on *Trees* where one deals with living trees and the other with the computer science data structures called Trees. A request with the keywords *Tree, Data Structure* would cause the ABLO that had information on living *Trees* to produce a negative reply to the request since although it matches the 'Tree' search string, the context was different so it gave a negative conclusion. If the request had the keywords 'Tree, Water' then this ABLO would certainly have given a different reply to the request. Our ABLOs are also capable of concluding that concepts are related through chains made up of other concepts and different types of relationships such as subclass, superclass, object-property, datatype-property etc.

Conclusion

Suggestions have been made that including 'application code' within a learning object could enhance its reusability. By using an *Agent Based Learning Object* we are attempting to realize this potential. The coupling of learning content with agent technology is not a new idea and as such the scenario described by Hendler [1] may be achieved in the field of e-Learning by means of ABLOs. The repurposing, fragmentation, and reassembly of learning content can be intelligently undertaken by ABLOs because of the semantic markup associated with the learning content. In the future, we intend to add more behaviours to the ABLO and define new ontologies for communication between ABLOs.

References

[1] Hendler J., (2001) 'Agents and the Semantic Web'. IEEE Intelligent Systems, 16(2), pp.30-37 (March/April 2001)

[2] Jennings N.R., Sycara K., Wooldridge M., (1998) 'A Roadmap of Agent Research and Development'. Journal of Autonomous Agents and Multi-Agent Systems, 1(1), pp.7-38 (July 1998)

[3] Mohan, P., and Brooks, C. (2003) 'Learning Objects on the Semantic Web'. *3rd IEEE International Conference on Advanced Learning Technologies*, Athens, Greece, July 9-11, 2003, 195-199.

Ontology as a Foundation for Knowledge Evaluation in Intelligent E-learning Systems

Slavomir STANKOV, Branko ŽITKO and Ani GRUBIŠIĆ
Faculty of Natural Sciences and Mathematics and Education, University of Split, Teslina 12, 21000 Split, Croatia
slavomir.stankov{branko.zitko, ani.grubisic}@pmfst.hr

Abstract. Services on WWW have enabled development of the e-learning systems which are considered to be direct application of the information and communication technology. The xTEx-Sys system is a Web-based authoring shell with adapted environment to each actor of the system. A formal representation of the course material in the xTEx-Sys involves ontology driven knowledge description. Student's knowledge evaluation in the xTEx-Sys is realized by using dynamic quizzes. This paper describes the ontology's role of the xTEx-Sys knowledge evaluation process.

Introduction

Information and communication technology combined with multimedia, networking and software engineering, have enabled the development of new learning and teaching environment. The last great milestone in this environment was an introduction of the Internet and WWW, and it was expected that all educational systems are to be reengineered. The usage of those technologies enables the development of the Web based authoring shells for constructing the Web oriented intelligent tutoring systems (ITS). According to ITS traditional modular architecture [1] and the idea of the cybernetic model of the system [2, 3] we have developed an intelligent hypermedia authoring shell called the Tutor-Expert System (TEx-Sys) [4]. We have created our own learning and teaching model as well as scenario for the knowledge evaluation by using knowledge bases developed by the TEx-Sys. Nowadays our work is directed on the implementation of a prototype of the extended version of the TEx-Sys, the *eXtended Tutor-Expert System*, xTEx-Sys [5], within a technology project founded by Ministry of Science and Technology of the Republic of Croatia. The xTEx-Sys is an authoring shell with an environment adapted to an every actor of the system: (i) an expert to design the domain knowledge on specially defined ontology for the knowledge representation, (ii) a teacher to design courseware using defined ontology for hierarchical organization of course content on *units, lessons, topics* and *instructional items* for student learning and teaching process as well as *tests of quiz type* for the student knowledge evaluation (the courseware structure elements), (iii) a student to select course and navigate trough the domain knowledge content via didactically prepared course content and (iv) administrator for the system supervision. Scenario for the student knowledge evaluation is of a great interest to us during the TEx-Sys and now with the xTEx-Sys research, implementation as well employment. Supported by our previous experience, a new knowledge evaluation method, based on dynamic quiz, is designed. The structure of knowledge representation in the TEx-Sys points out a motivation for enhanced approach to specially designed didactical ontology. The xTEx-Sys domain knowledge representation is based on OWL Web Ontology Language [6] and such representation makes foundation for teacher's and student's view of the knowledge evaluation process described in first section. Concluding remarks are given in second section.

1. Knowledge Evaluation

Student's knowledge evaluation in the xTEx-Sys is realized by quizzes. Quiz is an implementation of the test where the student gets a set of questions with attached answers which can be correct or incorrect. The teacher is responsible for assigning quizzes in course. Dynamic quizzes, which are generated by the xTEx-Sys, are often used for the fast evaluation of student's knowledge. This kind of quiz has questions structured on queries about concepts and relations.

Dynamic quiz generates questions over some domain knowledge. Considering OWL syntax for the knowledge representation, queries about concepts are translated into questions about classes or individuals, while relations in questions are expanded with properties as a special kind of relation (see Table 1). The xTEx-Sys dynamic quiz has three question categories of different levels of difficulty.

Table 1. Categories and questions in dynamic quiz

1st category	2nd category	3rd category
Recognize class/individual!	What is class/individual?	What are the properties of class?
What kind of relation is between two classes/individuals?	Who is in relation with class/individual?	What is value of individual' property?
Does class has property?	What relation is between two classes/individuals?	Who is and how in relation with class/individual?
Are two classes/individuals in relation?		

The first category contains the easiest questions, third category contains the hardest questions and the questions of middle difficulty level are in second category. Process of the knowledge evaluation can be observed from both the teacher and student point of view. The teacher, as a courseware designer, has to define when the students are to be tested. He prepares starting point for the knowledge evaluation which is performed by the student. In the following, algorithms and tasks from the teacher and student's viewpoint are described.

1.1 Teacher's View of Knowledge Evaluation Process

Test, as part of courseware content, can be created almost in any aggregation of learning objects. Generally, the first condition that teachers meet, while building course, is the existence of domain knowledge. When a teacher wants to publish a test, he must select one aggregation from the set of courseware learning elements, which will hold newly added learning object with a testing functionality. After entering the name of the learning object, the system calculates possible number of question series. In the case of dynamic quiz, chosen aggregation must have at least one learning object because questions are generated over some subset of domain knowledge assigned to aggregations' learning objects. Calculation for proposing amount of question series is based on the number of distinct domain knowledge elements gathered from all learning objects in the same aggregation where test will be put. An algorithm for proposing number of possible question series will count these elements of the domain knowledge: C – *the classes count*, I – *the individuals count*, R – *the relations count*, P – *the properties count*, M – *the media properties count*.

For every dynamic question, a minimum of dynamical generation condition has to be defined. If we look, for example, at the second question type "Are {Class/Individual1} and {Class/Individual2} in relation?" we can see that it is assembled from non changing question text as well as of a dynamic text placeholders. In this template, {Class/Individual1} and {Class/Individual2} are two dynamic text placeholders which are in a process of testing, and what is more filled with name of randomly chosen class or individual. However, answers can also have placeholders, but this template has constant text values which are: (i) No, (ii) Yes, directly and (iii) Yes, indirectly. For this type of question, the number of relations' count (R) has to be above 1 to generate question with "Yes, indirectly" as a correct answer. Moreover, the number of classes/individuals (C+I) has to be greater or equal to 2 so that the question could include two concepts. These two conditions make a minimal dynamical generation condition for that question type. Quiz in the xTEx-Sys must have at least one question type from every category. The minimal dynamical generation condition for category of questions is made by combining minimal dynamical generation conditions of every type of question in that category. Consequently, the minimal condition for the dynamic quiz generation includes minimal conditions of every category.

If minimal condition for dynamic quiz generation is satisfied, then the maximal possible number of question is a minimum of a set of maximum number of generated questions for each question types. For example, the second question type has the minimal conditions R>=1 and (C+I)>=2, so maximal number of generated questions has to be min{R, C+I}. Finally, when all maximal number of questions for every type of question is calculated, then the maximal number of questions that could be dynamically generated in quiz is a minimum of all maximal number of questions that can be generated for each type of question. That number is presented to the teacher; therefore he can select less or equal value of questions for his newly created test.

1.2 Student's view

Afterwards when student selects testing, the system initializes the process of dynamic quiz question's generation and presentation. Dynamic quiz generation in the xTEx-Sys means run-time creation of question text and answers over prepared set of the domain knowledge elements. If there is going to be generated question based on the second question template then algorithm is randomly choosing knowledge domain elements according to the placeholder's requests for particular domain knowledge element (Figure 1).

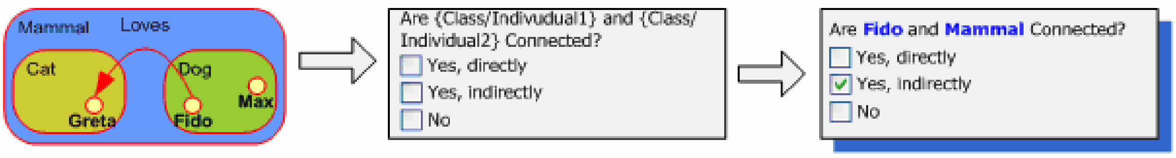


Figure 1. Process of question generation

When a student starts the dynamic quiz, the initial level of difficulty of a problem is sent to the problem generator. According to this difficulty level, the system generates pair of questions and sends it to the student. The first pair consists of two questions from the second category. After solving that pair of questions, the student submits his answers which are going to be evaluated, giving thus partial results of the test. These partial results are used by the system and have a very significant role. The problem generator, according to these partial results feedback, decides from which difficulty category will be the next pair of questions distributed to the student (Figure 2) or, in the worst case, violently interrupts testing and gives unsatisfying mark.

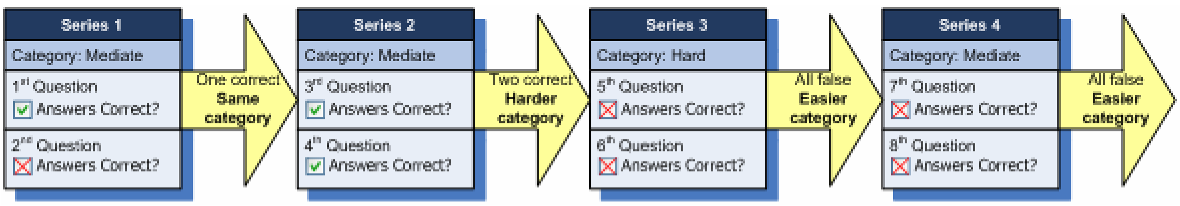


Figure 2. Category shifting in dynamic quiz

After the last series of questions, the overall result is estimated on the basis of calculating the final mark according to the relation between accomplished points and the maximal possible points. Calculated mark varies from an unsatisfying to an excellent. Presenting the result of the test involves not only displaying final mark, but also it gives back set of correct answers as well as the question category sequence. Therefore, the student can actually see where s/he was wrong and afterwards choose concept or relation to see exactly where, how and why s/he had made a mistake.

2. Conclusion

The Web Ontology Language, as a specification for the domain knowledge representation in the xTEx-Sys joins characteristics of network-based and logical-based knowledge representation. By uniting these major approaches to the knowledge representation, the domain knowledge in the xTEx-Sys provides quite good expressive power and computational costs. Ontology as a foundation for knowledge evaluation emphasizes dynamic quiz potential in the process of student testing. From teacher's point of view, the process of defining tests for a student is facilitated to the level where teacher only has to enter the name of the test and choose number of question cycles. At the other side, student will probably never be asked the same question during knowledge evaluation, and the question heaviness will vary depending on the student's correct answers.

Comparing preliminary test results of the xTEx-Sys system usage with results of the TEx-Sys we can say that student's feedback is quite positive in many ways. Major and the most important difference is readability of learning content which implies better understanding of the test questions. The second important difference is a refined user interface and simplified functionalities that make system friendlier and easier for usage.

Acknowledgments

This work has been carried out within projects 0177110 *Computational and didactical aspects of intelligent authoring tools in education*, and TP-02/0177-01 *Web oriented intelligent hypermedial authoring shell*, funded by the Ministry of Science and Technology of the Republic of Croatia.

References

[1] Burns, H., Capps, C., Foundations of Intelligent Tutoring Systems: An Introduction., in M. C. Polson, J. J. Richardson, Eds.: Foundations of Intelligent Tutoring Systems, Lawrence Erlbaum Associates Publishers, 1988, pp. 1-18

[2] Pask, G. A Cybernetic Model of Concept Learning. Proceedings of 3rd. Congress International. Assoc. Cybernetics, 1961.

[3] Božičević, J., Fundamentals of Automatic Control, Part I – System Approach and Control, 10th Edition, Zagreb: Školska knjiga, 1980 (in Croatian).

[4] Stankov, S., Isomorphic Model of the System as the Basis of Teaching Control Principles in an Intelligent Tutoring System, PhD Diss, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Split, Croatia, 1997 (in Croatian)

[5] Stankov, S., Principal Investigating Project TP-02/0177-01 Web oriented intelligent hypermedial authoring shell, Ministry of Science and Technology of the Republic of Croatia, 2003-2005.

[6] xxxx: OWL Web Ontology Language Guide, W3C Recommendation, 2004, http://www.w3.org/TR/2004/REC-owl-guide-20040210/