Proceedings of the 2007 workshop on

Representation models and techniques for improving e-learning

(ReTIeL'07)

Bringing Context into the Web-based Education

Ioannis Hatzilygeroudis
Alvaro Ortigosa
Maria D. Rodriguez-Moreno
(Editors)

AUGUST 2007 ROSKILDE UNIVERSITY COMPUTER SCIENCE RESEARCH REPORT #111

Ioannis Hatzilygeroudis, Alvaro Ortigosa, and Maria D. Rodriguez-Moreno

Computer Science
Roskilde University
P. O. Box 260
DK–4000 Roskilde
Denmark

Telephone: +45 4674 3839
Telefax: +45 4674 3072
Internet: http://www.ruc.dk/dat/
E-mail: datalogi@ruc.dk

ISSN 0109–9779

This research report constitutes the proceedings of the *2007 workshop on REpresentation models and Techniques for Improving e-Learning (ReTIeL'07)* which is held in conjunction with the 6th International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT 2007), Roskilde University, Denmark, August 2007.

Research reports are available electronically from:

http://www.ruc.dk/dat/

International Workshop on

REpresentation models and Techniques for Improving e-Learning: Bringing Context into the Web-based Education (ReTIeL'07)

Roskilde University, Denmark, 20-24 August 2007

Held in the Sixth International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT 07) Conference

PREFACE

The ReTIeL'07 workshop is concerned with the representation, analysis, design, implementation and use of techniques and models to support the creation of advanced e-Learning environments and the realization of activities within these environments. Particularly, the workshop focuses on the use of those techniques and models to provide support for modelling the context of e-learners and helping on the development of contextaware e-learning environments. The aim is to bring together researchers and practitioners that work with or use adaptive hypermedia systems and are interested in durable, reusable solutions, as well as researchers and practitioners that work with or use semantic web technology and are interested in adding to it and defining the (elements of) adaptation and personalization.
Topics of interest include (but not limited to) the following:

- AI techniques in e-Learning
- Research questions unique to the mobile context-aware area, i.e., dialog (speech) based
- Data modelling for e-Learning environments
- Model representation of learning/educational processes
- Interactions and cooperation between users in e-learning environments
- Automatic user models acquisition and learning
- Adaptivity/Reactivity
- User interfaces for VE
- Planning and scheduling in long-life learning
- Planning for communication with students
- Monitoring students for attentiveness, motivation, cognitive load, confusion ...
- Planning for differentiated learning
- Classroom management
- Real applications

ReTIeL'07 PROGRAM COMMITTEE

c

* Ioannis Hatzilygeroudis, University of Patras, Greece (Co-Chair)
* Alvaro Ortigosa, Universidad Autónoma de Madrid, Spain (Co-Chair)
* Mª Dolores Rodriguez-Moreno, Universidad de Alcalá, Spain (Co-Chair)

* Rajendra Akerkar, Technomathematics Research Foundation, India
* Costin Badica, University of Craiova, Romania
* Beatriz Barros, UNED, Spain
* Melquiades Carbajo, Universidad de Alcalá, Spain
* David Camacho, Universidad Autónoma de Madrid, Spain
* Rosa Mª Carro, Universidad Autónoma de Madrid, Spain
* Amedeo Cesta, ISTC-CNR, Rome
* Alexandra I. Cristea, University of Warwick, UK
* Hai-Bin DUAN, Beihang University, China.
* Ramón Fabregat, Universidad de Girona, Spain
* Serge Garlatti, ENST Bretagne, FRANCE
* Elena Gaudioso, National University of Distance Education, Spain
* Jesús González Boticario, UNED, Spain
* Arnon Hershkovitz, School of Education, Tel Aviv University, Israel
* Jamie Kurien, NASA, USA
* Miltiadis D. Lytras, University of Patras, Greece
* George Magoulas, University of London, UK
* Eva Millan, Universidad de Málaga, Spain
* Tanja Mitrovic, University of Canterbury, New Zealand
* Angel Moreno, Universidad de Alcalá, Spain
* Alexander Nareyek, National University of Singapore, Singapore
* Angelo Oddi, ISTC-CNR, Rome
* Estrella Pulido, Universidad Autónoma de Madrid, Spain
* Peter Robinson, NASA, USA
* Carsten Ullrich, German Research Center for Artificial Intelligence, Germany
* Maria Virvou, University of Piraeus, Greece
* Botond Virginas, British Telecom. Adastral Park, U.K
* Kalina Yacef, University of Sydney, Australia

Proceedings available online at CEUR Workshop Proceedings (http://ceur-ws.org/).
ISSN: 1613-0073

III

TABLE OF CONTENTS

c

Multiuser Intelligent M-learning Environment

Angel Moreno†, Melquiades Carbajo†, Bonifacio Castaño‡ and Javier de Pedro†

† Universidad de Alcalá de Henares, Departamento de Automática. Madrid, Spain.
email: angel,mcm, jdp@aut.uah.es

‡ Universidad de Alcalá de Henares, Departamento de Matemáticas. Madrid, Spain.
email: bonifacio.castano@uah.es

Abstract. In this article we present the architecture of an m-learning (mobile-learning) environment using Bluetooth as communications technology. We also describe its practical implementation into a technical laboratory where students can access, work and leave at any time. The system incorporates artificial intelligence techniques in order to adapt itself to the characteristics of each user. This strategy allows us to recognize each student, organize his/her work and evaluate his/her results, without educator intervention. Nevertheless, the teacher will be reported about the student activities and will be advised when the situation requires it. The Bluetooth facilities assure good isolation between different classrooms and multiuser wireless connections.

Keywords: M-learning, AI Planning and Scheduling, Expert systems, Bluetooth.

1 Introduction

Traditional education, where a teacher transmits to their students some knowledge in the classroom, is a very well known communicative process. In the same way that others human situations of communication, teaching and learning are being highly affected by the development of the new technologies of information and communication (TIC). This influence has given rise to the creation of the e-learning concept. Two of the most relevant advantages of this new educational framework are the flexibility and the context adaptation capacity.

It is evident that these features are greatly improved if the potential offered by the wireless communication systems is added (and then the term m-learning is used). Moreover, it is possible to go one step further, integrating also artificial intelligence techniques in order to automate and personalize the learning experience offered to the students, which opens a world of educational possibilities [8].

Among them, this article focuses on the problem of management within a group of students arriving into a laboratory, making practices, getting real time results, asking questions, interacting with the educator -if needed- and leaving at any time. The m-learning environment presented here offers a real solution to this challenge. It uses a Bluetooth scheme and takes advantage of this huge potential. The obtained wireless system is agile, trustworthy and dynamic.

Another fundamental question supporting the versatility and effectiveness of this m-learning answer is the characterization and monitoring of its users. A recognition strategy -based on Artificial Intelligence- allows the system to determine the capacities, preferences and availabilities of the students and to adapt its interaction with them in order to optimize its results.

The paper is structured as follows. First, we provide an overview of the current e-learning and m-learning technologies, in order to settle the context of the work presented. Then, we provide some basic information about planning and scheduling and Bluetooth that are, respectively, the artificial intelligence and communication methodologies supporting the m-learning architecture proposed. In order to present that architecture, first we describe the educational environment where we are going to implement it: a technical laboratory. Then the main modules of the architecture itself are described. Lastly, we summarize the main concepts and propose some challenges for future work.

2 E-learning and M-learning

Current e-Learning and Virtual Educations technologies have experienced an increasing research interest thanks to the use of information technologies and the Internet [5]. These technologies have generated a new kind of tools and frameworks that can be used by educators to design, deploy and control courses. Several well known e-Learning standards, such as IMS [4], SCORM [10] or LOM [6], are currently being used to define and develop new adaptive virtual based education tools. These tools support the creation of personalized learning designs (LD). These new designs make possible to reuse and exchange useful information among different platforms. These new tools can be used by educators (and/or course designers) not only to define the contents of a course (i.e. by using the IMS LD specification), but also to create adaptive and personalized learning flows, so that the educational system can monitor and control the whole learning process.

When these systems incorporate wireless elements, as movable telephones, PDAs or laptops, the term m-learning can be used. In this context, many of the fixed systems advantages can be taken. Among them it is possible to be emphasized: the possibility of giving service to many users simultaneously, the capacity to put the contents to disposition of the students at the opportune time and place and the flexibility in the access to any electronic document. According with some authors [11], the wireless communication with movable devices presents some more details that give it a special character. Some of them are: The possibility to use very small time intervals and the need to simplify the contents presentation due to the limited display and input capabilities of the mobile devices. Nevertheless, it is interesting to mention that in mobile learning several problems exist that are not present in e-learning. One of the most important is the hardness to obtain or send printed material.

The earliest application of mobile computers for teachers and learners started at the beginning of the 1970's at the Xerox Palo Alto Research Center. Nevertheless, the authentic m-learning concept and its technical developing began at the end of the 1990's. From that moment on the number of mobile devices and its applications are growing faster and faster and today they are more than three times the number of personal computers. Some experts even think that mobile phones are going to be an alternative to PC's. In this context, m-learning appears like a very valuable investigation and application field.

3 Artificial Intelligence Techniques

The goal of Artificial Intelligence (AI) [7] is to study how to build machines that perform tasks normally performed by human beings. Within the AI field, our work focuses on *AI planning and scheduling (AI P&S)*. These techniques have been applied to solve complex problems in domains such as robotics, logistics or satellites. In this last domain, it has been an special interest in the development of autonomous architectures that can carry out a large number of functions such as planning activities, tracking the spacecraft's internal hardware, and ensuring correct functioning and repair when possible, without (or little) human intervention. In these new models of operations, the scientists and engineers communicate high-level goals to the spacecraft, these goals are translated into planning and/or scheduling sequences; then a continuous check of the spacecraft status is verified in order to detect any damage and act accordingly.

Then, a *planner* solves a problem by finding a sequence of actions that transform an initial state into a final state. In order for a planner to solve a problem, it is needed (1) to specify the domain that is composed of a set of operators that allow the planner to go from a defined initial state to a state in which a set of goals is fulfilled, and (2) to describe the initial and goal states. The standard language to specify the domain and the problem is PDDL, now in its 3.0 version [3].

A *scheduler* organises activities along the time line by taking into account the resources available. One of the main drawbacks in scheduler systems is the lack of a language that allows us to define the deadlines and resource constraints.

Traditionally, both areas have evolved separately of each other. But nowadays applications require more communication between them. It becomes necessary to take into account inside the planning/scheduling reasoning the time at which the plan/schedule will be provided and executed. This is the case of the architecture described in this paper where activities (i.e. units of learning that the students need to follow) should be planned in a period of time (i.e. the student finishes his/her work in 3 weeks) and depending on the results, some tasks should be generated dynamically (i.e. if the work sent is correct, then he/she starts a new unit of learning).

4 Bluetooth

Bluetooth wireless technology is a short-range communications system intended to replace the cables connecting portable or fixed electronic devices. Bluetooth is now the largest radio-based technology after GSM. Currently, consumers specifically recognize the significant technological advancements of Bluetooth in three markets [1]:

- **Mobile phones / Handsets.** Bluetooth-equipped cell phones are rising quickly, with an estimate of 303.7 million units sold worldwide by 2007.
- **Headsets.** The headset trend is becoming the new wearable technology. Industry experts say Bluetooth headsets will also be able to use with iPods switching from music and calls.
- **Automotive Industry.** The Bluetooth applications for cars are being included in newer car models coming out and also being sold as after market kits. The hands free solution as a safety benefit is one of the most demanded options.

Due to the wide adoption of Bluetooth and its quite interesting properties, a lot of effort has been made trying to evolve it beyond the initially envisioned wire replacement function to a large-scale networking technology. Now it is possible to use Bluetooth communications in a multi-user environment like the one found in a classroom or a library while maintaining short connection times and good performance behavior. Also, using artificial intelligence planning techniques, it is possible to handle the communication needs of an m-learning environment in a very efficient manner.

One of the most interesting features of Bluetooth technology in an m-learning environment is its short range of operation: most portable devices have a coverage area of 10 m, rapidly decreasing when obstacles are present (i.e. walls). It means, for instance, that devices inside a classroom are discoverable and can join the network whereas those outside the classroom are not. It makes this technology suitable for context-dependent applications, where two adjacent rooms can be considered two different m-learning cells, perfectly isolated from each other, and users are clearly identified as forming part of one cell or the other.

5 A Simple Scenario

The chosen scenario is a technical laboratory, which constitutes an m-learning cell, where students can enter or leave at any time during a given schedule. They are supposed to carry their own laptop computer with Bluetooth capabilities. The system detects such students and manages profiles and access rights. Once a student is authenticated, the system sends to the laptop the application needed to interact in the m-learning environment, if it is not already installed. The system also sends to the student the appropriate tasks, according to the schedule and the personal situation. The student can send the solved practices back to the system for evaluation which, in turn, gives him/her feedback about the punctuation obtained, on-line advice on how to proceed if the work done is incomplete or below the minimum required level, or the next task to accomplish if he/she has passed the exam. The student can ask the system for any kind of information about the course, like the practices' program, their content, groups to join, exam dates, and so on. He/she can also ask for on-line advice or for an appointment with the educator at a later date.

The system also interact with the educator, who can enter his agenda, the programmed practices, support material and so on. The system sends the educator various logs, not only about the performance of the whole system, but also about the work done by every student in the laboratory: number of tries for every practice, punctuation obtained, and time needed to accomplish the work, progress made, and so on.

Apart from presenting statistical data in several fashions, the system is also able to learn about the behaviour of the students, valuate the difficulties encountered for every task assigned and detect such things like two students presenting similar solutions to the same task. When the system is not able to evaluate the practice presented by the student, it will ask the educator for advice on how to proceed. If the solution proposed by the educator allows the system to go ahead with the evaluation, it will be incorporated to the knowledge database for future use.

6 M-learning Architecture

Figure 1 shows the basic architecture of the m-learning environment. It comprises a Bluetooth communications module, a Server, an integrated Planner and Scheduler (P&S) and an expert system called Evaluator.

The students interact with the m-learning platform by means of a **Bluetooth communication system**. This system is responsible for: detection of users entering or leaving the m-learning cell, establishment and release of work sessions with these users, and exchange of data between users and the learning environment. It will also manage and optimize in an intelligent manner the communications performance of the m-learning cell. With this intelligent management it is possible to extend the number of simultaneous users of a Bluetooth piconet from 7 to the amount normally found in a laboratory (from 20 to 30 users). At the same time, it also significantly improves the discovery and establishment times of Bluetooth in order to fulfil the real-time needs of the interaction with the users.

The **Server** will interact with the client application in the user's laptop using the Bluetooth module. It is responsible for the dialog and transactions with the students, performing all the tasks not requiring intelligence. A user entering the m-learning cell will be authenticated and registered. The basic m-learning application, if not present, will be sent to the student's laptop, as well as the tasks assigned by the P&S module to the student. It will manage all the requests asked for by the students, sending them to the P&S module or the Evaluator when appropriate. In turn, it is responsible for delivering to the students all the

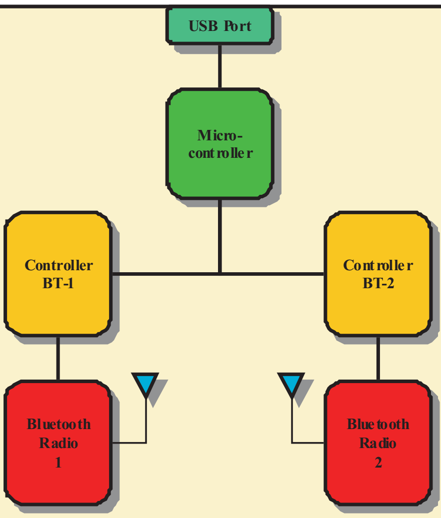

Fig. 1. M-learning environment architecture.

information coming back from the Evaluator or the P&S module. Eventually, the Server will take care of detecting users leaving the m-learning cell and will update the active users' database accordingly.

The Server is also the bridge between the educator and the system. All the information about the courses or the practices, as well as the educator's agenda, the students list or the laboratory groups list is accessed through the Server. Only the information pertaining to the P&S or the Evaluator (domain, constraints, evaluation rules, and so on) will be introduced using the management console module of the Evaluator.

The **Evaluator** is a dedicated PC platform for automatic evaluation of the practices sent by the students [2]. It comprises three main functional blocks:

- **Practice manager.** It implements an automatic service for delivering and collection of practices, without physical intervention. Periodically checks the registered student database, maintained by the server, seeking for new users in the m-learning cell. For any new user, it looks at the student's personalized program and sends him the scheduled practices, together with a practice management agent. The agent allows the student, not only to send the solved practice for evaluation, but also to ask for on-line or physical advice about the work to be done, among several other capabilities. It comprises two main modules, the student module and the tutorial module. The former stores the progress made by the student in his or her interaction with the expert system. This information is useful for chosing the next task to be asigned to a student. The tutorial module is devoted to the course program development and the way the contents are delivered. It controls the progress and sequence of contents, answers the questions raised by the students and detects the kind and level of help they need.

- **Expert system.** Once the student has finished the practice, it is sent for evaluation. In order to do that, the practice management agent delivers the information coming from the student to the expert system which, according to a given set of rules (implementing the evaluation criteria), analyzes the information and reports the qualification obtained. Moreover, it notifies to the student the list of errors found, as well as a series of guidelines in order to improve the student's experience. All the events all collected by a trouble-tracking system and saved in a database, together with the answers recorded by the teacher through the management console. This database is the learning repository for the expert system, which can afterwards propose more elaborate answers and guidelines, based on the answers of the students and the difficulties found. The expert system has also the potential to detect when the answers sent by two different students are significantly similar, reporting the degree of similarity found and notifying the teacher, who can then send a message to the involved students in order to clarify the situation. The expert system does so thanks to the use of NLP (natural language processing) techniques.
- **Management console.** The teacher has at his or her disposal a complete set of tools for performing on-line administrative tasks and track the activities accomplished by all and every student in real time. Among them, it is worth mentioning: a database compiler, in order to incorporate the expert system database to the report manager; an interactive graphical interface for real-time event monitoring and dialog with the students; a query tool, for definition and incorporation of tasks and evaluation rules; an event-reporting module for interaction with the Planner; and a report generator.

Finally, the **P&S module**, by means of the IPSS system [9], detects the characteristics of the different students (observing how they use the system). Then, in every case, the way to interact with each one is determined. In this process of mutual interaction two things may be considered: the objective of learning or formation that is wanted to be reached for the students and the resources susceptible to support them. This way of work allows for a complete adaptation to each student (maximizing the effectiveness of the whole process), also offering the possibility of managing user groups. The result is a personalized program for every student. The general course program is the environment and guideline, giving the boundaries (lower and upper) where the individual programs can be dinamically established. The final result obtained student by student gives feedback in order to improve the overall program for the next season.

In addition to the domain defined and the constraints introduced in the P&S module, the information provided by the Evaluator is very important. The personalized program of a given student may be dynamically altered as a result of the evaluation information. Two students of the same laboratory group may follow similar or very different paths, depending on their own achievements.

7 Conclusions and Future Work

In this paper we have presented a basic m-learning architecture. Bluetooth permits the creation of cells with a classroom scope, allowing for a context-aware approach. The addition of intelligence greatly improves the learning possibilities and the user experience: the Evaluator contributes with an automatic evaluation and feedback methodology and the P&S module improves the system with a student by student approach. This work is just the basic architecture, but offers plenty of opportunities for further research. The management of many adjacent cells, with users roaming between them is one of the issues. Another one is the automatic generation of activities depending on the student's performance. Finally, the modularity of the system and integration with existing or new e-learning components is another field that offers plenty of potential.

Acknowledgements

This work has been funded by the research Junta de Castilla-La Mancha project PAI07-0054-4397 and by the CICYT project CYCIT (TST 2006-12085).

References

1. Bluetooht Interest Group. *www.bluetooth.com*. The Bluetooth technology web site.
2. J. A. Duart and Lupiáñez. La gestión y administración del e-learning en la universidad. conclusiones. *Revista de Universidad Sociedad del Conocimiento (RUSC)*, 2(1), 2005. Las TIC en la universidad: estrategia y transformación institucional (monográfico en línea).
3. A. Gerevini and D. Long. Plan constraints and preferences in pddl3. the language of the fifth international planning competition. Technical report, Department of Electronics for Automation, University of Brescia, Italy, 2005.
4. IMS-LD. Ims learning design. IMS Global Learning Consortium. http://www.imsglobal.org/learningdesign/index.html, 2006.
5. R. Kozma. Learning with media. *Review of Educational Research*, 61(2):179–212, 1991.
6. LOM. Learning object metadata. http://ltsc.ieee.org/wg12/, 2006.
7. N. Nilsson. *Problem-Solving Methods in Artificial Intelligence*. Ed. Feigenbaum, E. and Feldman, J. McGraw-Hill., 1971.
8. A. Patel and Kinshuk. Intelligent tutoring tools in a computer-integrated learning environment for introductory numeric disciplines. *Innovations in Education and Training*, 34(3), 2000.
9. M. D. R-Moreno, A. Oddi, D. Borrajo, and A. Cesta. Ipss: a hybrid approach to planning and scheduling integration. *IEEE Transactions on Knowledge and Data Engineering.*, 18(12):1681 1695, December 2006.
10. SCORM. Sharable courseware object reference model. http://www.academiccolab.org/projects/scorm.html, 2006.
11. J. Zobel. *Mobile Business und M-Commerce*. Hanser, 2001.

Dialectic Approach for Using Viewpoint Discrepancies in Learning

Christiana Panayiotou and Vania Dimitrova

School of Computing,
University of Leeds,
UK
{cpaa,vania}@comp.leeds.ac.uk
http://www.comp.leeds.ac.uk

Abstract. The paper proposes a dialectic approach to exploit discrepancies of viewpoints for learning. The approach is illustrated with an elaborated example. A computational framework of a pedagogical agent capable of interacting with a learner for discussing different viewpoints in the same domain is outlined. The framework employs AI technologies, such as argumentation for defeasible reasoning, situation calculus for contextualized reasoning and dialogue management. The approach can be applied in interactive learning environments to promote awareness, reflection, and conceptual change.

Key words: viewpoints, conceptual discrepancies, argumentation, context, e-learning

1 Introduction

The Semantic Web enables the representation of different conceptualisations in the form of ontologies. However, studies show that conceptualisations may differ between tutors and the resources they prepare [1], as well as between a learner and a tutor [2]. Reasons for this variation can be the intented use of each conceptualisation, the background knowledge of tutors and learners, or the incompleteness of domain ontologies. The awareness of alternative views can bring educational benefits by broadening the perspective of learners.

This paper argues that discrepancies in conceptualisations can be handled constructively to enrich the learning experience in educational systems. We outline an approach where a software agent detects discrepancies in conceptual viewpoints of a learner, tutor, and learning resources, and engages in a dialogue to explore similarities and differences between different viewpoints.

The paper reviews existing approaches for dealing with different viewpoints in learning systems and proposes a dialectic approach for handling viewpoints in educational semantic web applications. The proposed approach will be introduced with the help of an example. Then, the architecture of a dialogue agent that explores different viewpoints in a conversation with a learner will be outlined. We will illustrate the use of AI technologies, such as argumentation for defeasible reasoning, situation calculus for contextualized reasoning and dialogue management, to exploit viewpoints discrepancies in learning.

2 Using Viewpoints in Learning Systems

The first attempts to deal with viewpoints in learning can be traced back to some of the early Intelligent Tutoring Systems (ITSs). Among these, two notable uses of viewpoints are shown in the systems VIPER [3] and DENISE [4]. In VIPER, viewpoints represent different ways of decomposing a domain and provide different interpretations of domain knowledge. However, viewpoints are fixed in advance and refer only to the domain expertise. In contrast, DENISE [4] focuses on student modelling, and considers that the domain model and the learner model may represent different viewpointsA formal way for representing viewpoints in ITS is given in [5] where the viewpoint of an agent a is defined as a triple $Va =< Ba, La, Ma >$ with each element being a subset of the agent's complete belief, logic and meta-logic space, respectively.

While the early ITS research on viewpoints considers different perspectives of the domain and offers representations that distinguish between the tutor's and learner's viewpoints, these projects suffer from two key limitations. Firstly, although the students are considered to have alternative views upon the domain, any deviation from the view of the tutor is considered as a *bug* that needs to be fixed. Secondly, the early ITS systems adopt rather static approaches for dealing with viewpoints, e.g. transmitting the tutor's viewpoint by telling it to the student and assuming that it will overwrite the student's own.

More recent approaches followed in collaborative learning systems which enable the discussion and exchange of different points of view among peers. Based on research in Education which advocates the use of argumentation for constructive learning, collaborative learning systems were implemented to enable and encourage the use of argumentation for joined decision making and sharing of knowledge, e.g. [6, 7]. Empirical evidence from the use of these systems suggests that the exchange and challenge of different viewpoints via argumentation motivates the processes of reflection, articulation and conceptual change. Although these systems aim to sharpen the learner's critical skills, they typically provide very limited analysis of the discussion. They do not model the learners' beliefs during the interaction and do not provide any automatic support to facilitate articulation and clarification of different views about the domain.

Collaborative learning environments have influenced the design of computational approaches for developing intelligent pedagogical agents that support viewpoint clarification. Despite the notable successes, the existing computational approaches do not fully address the problem of identifying and clarifying viewpoints because they do not explore the context in which the views have been formed and ignore what arguments have led the learner/tutor to form a particular position. Moreover, none of these approaches is SW-compliant, so additional work is required to make them ontology-based and to integrate them in educational SW applications, as illustrated in [2].

Proposed Approach. Building on research in dialogue pedagogical agents, Semantic Web, and argumentation, we propose a dialectic approach for exploiting viewpoint discrepancies for learning. The proposed approach caters for the representation of multiple viewpoints of the same domain, treats discrepancies in conceptualisations between the learner and the agent as triggers for dialogue games that clarify different viewpoints and enables the participants to justify their positions via the use of argumentation thus promoting the processes of reflection and articulation.

3 Illustrative Example

We will illustrate our approach with the help of an example that shows how a pedagogical agent A can be integrated in an educational semantic web system, e.g. an adaptive recommender system like OntoAIMS [2]. The agent A has a domain ontology Ω_A representing the main concepts and relations in the domain (we will use here the ontology about Programming Languages from one of the instantiations of STyLE-OLM [8]). The agent recognises that its knowledge can be incomplete and engages in a dialogue to explore different viewpoints.

A **viewpoint** $\mathcal{V}$ is defined as a structure $\mathcal{V} \equiv < s, p, B, \Gamma >$ where: s denotes the source of the viewpoint (e.g. the learner, the agent, or an existing learning resource), p denotes the position of the viewpoint represented with a proposition (e.g. *'VB is OOL'*), B is a set of beliefs of s related to p (e.g. *'All OOL have objects'*), and Γ is a set of arguments of s supporting the position p.

Assume that the learner L is a distance student attending an introductory course on programming languages. She receives most of her learning material online through a list of online resources aiming to enable her to enhance her knowledge on the topic. L comes across a learning resource that states *'Visual Basic is an object-oriented language'*. She believes that an object-oriented language (OOL) needs to satisfy the property of encapsulation but she does not know whether Visual Basic (VB) has this property. Further, she is almost sure that she has read somewhere else that Visual Basic is not an object-oriented language. The learner is confused and invokes a discussion with the pedagogical agent A in order to clarify the position she encountered in a resource R with url URI_1. The agent A accepts the initiated discussion and starts a *viewpoint elicitation game* to find out more about the viewpoint expressed in the resource.

(1) **L:** Let us Discuss the statement in Resource: URI_1: *VB is an OOL*.
(2) **A:** What is the explanation given in the Resource URI_1 for *VB is an OOL*?
(3) **L:** The resource URI_1 states that *VB is an OOL* because *VB contains Objects*.

Based on the above dialogue, the viewpoint $\mathcal{V}_R$ of the resource R is derived:

$\mathcal{V}_R \equiv$ **Viewpoint of resource** R	
Source s_R	R
Position p_R	`instance_of(VB,OOL)`
Belief set B_R**:**	
	`contain(VB, objects)`
Argument set Γ_R**:**	
$claim_1$	`instance_of(VB,OOL)`
$datum_1$	`contain(VB, objects)`
$warrant_1$	$\forall X$: `contain(X, objects)` $\Rightarrow$ `instance_of(X,OOL)`

To find out whether the learner agrees with the viewpoint of the resource, the agent A initiates an *explore viewpoint game*. The interaction continues with:

(4) **A:** So do you agree that *VB includes objects*?
(5) **L:** Yes..I agree that *VB includes objects*
(6) **A:** Do you agree that a *PL is OOL* because it *contains objects*?
(7) **L:** I disagree with this.
(8) **A:** OK, do you agree that *contains objects* is necessary for a PL to be OOL?
(9) **L:** Yes.
(10) **A:** What other condition is needed for a PL to be OOL?
(11) **L:** OOL have the *property of 'encapsulation'*
(12) **A:** So, you think that *IF a PL contains objects AND PL has the property of encapsulation THEN PL is an OOL*
(13) **L:** Yes, that's it!
(14) **A:** Therefore, you think that *IF VB has the property of encapsulation THEN VB is an OO Language*
(15) **L:** Yes.

Initially, A has no information about the viewpoint $\mathcal{V}_L$ of the learner apart from assuming that L is confused about p_R. In the above dialogue, move **(4)** checks whether L supports $\Gamma_R : datum_1$, i.e. the datum in the only argument in $\mathcal{V}_L$, while moves **(6)** and **(8)** clarify whether L supports $\Gamma_R : warrant_1$. Following L's dialogue moves, the agent discovers that there is a discrepancy between viewpoints $\mathcal{V}_L$ and $\mathcal{V}_R$, and attempts to explore the difference between these viewpoints. Move **(10)** asks for an additional belief, based on which a rule is abducted and then checked in move **(12)**. Finally, in move **(14)** the agent confirms the claim of the learner's argument. Hence, the agent derives the following viewpoint of the learner:

$\mathcal{V}_L \equiv$ **Viewpoint of Learner** L	
Source s_L	L
Position p_L	`? instance_of(VB,OOL)`
Belief set B_L**:**	
	`contain(VB, objects)`
	`contain(OOL, objects)`
	`has_property(OOL, encapsulation)`
Argument set Γ_L**:**	
$claim_1$	`has_property(VB,encapsulation)`$\rightarrow$`instance_of(VB,OOL)`
$datum_1$	`contain(VB,objects)`
	`contain(OOL,objects)`
	`has_property(OOL,encapsulation)`
$warrant_1$	$\forall X$: `contain(X,objects)`$\wedge$ `has_property(X,encapsulation)`$\Rightarrow$ `instance_of(X,OOL)`

The question mark in p_L shows that L is undecided about this position, and $\Gamma_L : claim_1$ indicates the condition needed for the learner to support p_L.

To continue the dialogue, the agent compares its own viewpoint with that of the leaner. The agent's viewpoint $\mathcal{V}_A$ is derived from the ontology Ω_A that can be incomplete. Ω_A represents that *'All OOL have the property of inheritance'* and *'Java is an OOL'* but does not have information about VB apart from it being a programming language. Hence, the viewpoint of the agent A is:

$\mathcal{V}_A \equiv$ **Viewpoint of agent** A	
Source s_A	A
Position p_A	`? instance_of(VB,OOL)`
Belief set B_A**:**	

	`instance_of(VB, Programming Language)`
	`instance_of(Java, OOL)`
	`contain(VB, objects)`
	`contain(OOL, objects)`
	`has_property(OOL, inheritance)`
	`class(OOL)`

Argument set Γ_A**:**	
claim	`has_property(VB,inheritance)` $\rightarrow$ `instance_of(VB,OOL)`

datum	`contain(VB,objects)`
	`contain(OOL,objects)`
	`has_property(OOL,inheritance)`

warrant	$\forall X$: `contain(X,objects)` $\wedge$ `has_property(X,inheritance)` $\Rightarrow$ `instance_of(X,OOL)`

Similarly to $\mathcal{V}_L$, $\mathcal{V}_A$ has a question mark in p_A to indicate that A is undecided about this position. $\Gamma_A : claim_1$ indicates the condition needed for the agent to support p_A and $\Gamma_A : warrant_1$ shows the rule which A holds in order to derive that a programming language is object-oriented. The agent notes that the viewpoints $\mathcal{V}_L$ and $\mathcal{V}_A$ have some similarities and differences. Both A and L are undecided about the position *'VB is OOL'* and share some common beliefs. However, A and L have built arguments that support different claims: $\Gamma_L : claim_1 \neq \Gamma_A : claim_1$. The agent cannot accept $\Gamma_L : warrant_1$ because it contradicts with $\Gamma_A : warrant_1$ and A believes that the property of inheritance is necessary. Thus, A initiates an *explore-inform viewpoint game* to find out whether L is aware of the property of inheritance, and if this is not the case, A employs didactic tactics to give guidance to L. This is illustrated in the following dialogue:

(16) **A:** I disagree that IF a PL contains objects AND PL has the property of encapsulation THEN PL is an OOL.
(17) **A:** Let us discuss this.
(18) **A:** Do you think that Java is an OOL?
(19) **L:** Yes, I think that Java is an OOL.
(20) **A:** Can you tell me a property of Java other than encapsulation?
(21) **L:** I can't think of anything else...
(22) **A:** Are you aware of the property of inheritance?
(23) **L:** No.
(24) **A:** Inheritance means that a class obtains the characteristics of it superclasses. All OOL have the property of inheritance. Java has the property of inheritance.

The agent expresses its disagreement with $\Gamma_L : warrant_1$ in move **(16)**. Instead of simply informing the learner that inheritance is a necessary property, A introduces a specialisation of the rule in $\Gamma_A : warrant_1$ and checks whether L the learner can derive the need for inheritance from the specialisations with moves **(18)** and **(20)**. When A discovers that L is unaware that *'Java has inheritance'*, it checks in move **(22)** whether L knows anything about inheritance. Because L is unaware of inheritance, A composes an explanation by combining the statements about inheritance extracted from Ω_A. This ends the dialogue game and updates $\mathcal{V}_L$ to include the new belief about Java. If the answer in move **(23)** was positive, the discussion would have continued with checking whether *'VB has the property of inheritance'* and might have led to updating $\mathcal{V}_L$ or $\mathcal{V}_A$ accordingly.

The above example illustrates a dialogue that explores three viewpoints from a learning resource, the learner, and the tutoring agent. It also illustrate how disagreement in viewpoints can be exploited for learning about the domain. Although the agent is equipped with the domain knowledge to form a viewpoint, this viewpoint is not imposed on the learner and she is only informed of the agent's opinion about inheritance after the viewpoint discrepancies are explored. The agent allows the learner to form her own opinion making her aware of all the relevant knowledge, and the point of view of the agent.

4 Proposed Framework

The goal of our research is to develop a computational framework for the design of a tutoring agent A capable of engaging in discussions to elicit viewpoints and explore similarities and differences between them, as illustrated in the above example. We will outline here the main architecture of our framework and will define its main components. The proposed architecture is illustrated in figure 1.

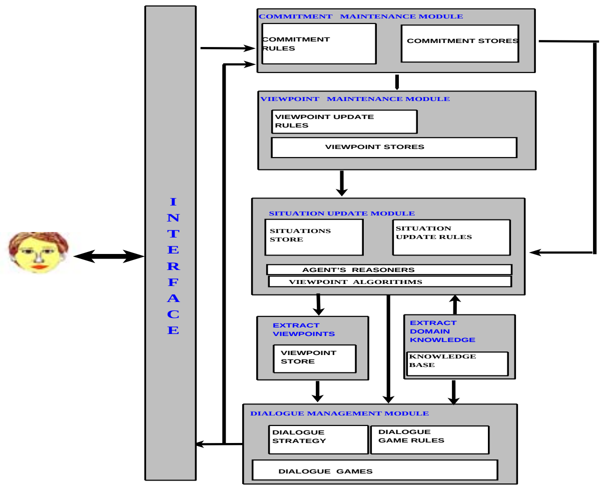

Fig. 1. Proposed Architecture of a Framework for Dialectic Viewpoints Handling

Interface. We assume that both the learner and the tutoring agent are provided with an appropriate interface to compose their utterances that express dialogue moves. In line with existing computational approaches, e.g. [9, 10, 8, 2], we assume that the interaction is restricted to the use of predefined moves where each move is associated with several possible sentence openers. A set of moves and their corresponding sentence openers are illustrated in Section 3.In addition, we assume that the interface provides an appropriate way for the dialogue participants to compose the propositions of their dialogue moves, e.g.

by using structured sentences or graphical statements [2]. Hence, a move m is defined as a tuple $m =< n, a, t, \varphi >$, representing its unique identifier which is a number n, the agent a who produces the move, the move type t that is linked to possible sentence openers, and the statement φ. To make a statement that a proposition p is valid in a particular context C we will use the predicate $ist(C, p)$ [11]. For instance, the first two moves in the example above express statements about the resource R and are defined as follows:

$m_1 =< 1, L, informDiscuss, ist(R, instance\_of(VB, OOL) >$
$m_2 =< 2, A, questionExplore, ist(R, instance\_of(VB, OOL) >$

Commitment maintenance. The beliefs of both participants derived from the dialogue are stored in commitment stores, and are used to compose the viewpoints or to plan the dialogue. Similarly to [8, 2], we employ commitment rules to establish the beliefs to which the participants of the dialogue are committed by taking into account the current dialogue move and the dialogue history. The agent's commitments are also derived from its ontology Ω_A, see Section 3.

Viewpoint maintenance. The viewpoints derived from the dialogue are stored in viewpoint stores. The definition of viewpoints given in Section 3 enables us to compare two viewpoints and identify similarities and differences between them, as shown in the illustrative example above. In addition, the maintenance of viewpoints includes a set of operations over the viewpoint stores to add, delete, update, and revise viewpoints.

Situation update. Based on the commitment stores and the viewpoint stores, the agent obtains information about the current situation which is used for planning the dialogue and update of the existing viewpoints. Situation update is performed after each dialogue move to encounter the changes it brings. For instance, a situation can present that there are discrepancies in two viewpoints (e.g. the situation after move **(15)** in the example will represent that $\mathcal{V}_L \neq \mathcal{V}_A$) or that there is insufficient information about a particular viewpoint (e.g. the situation after move **(3)** in the example will represent that $\mathcal{V}_L$ is still empty).

Dialogue management. The dialogue is organised as a sequence of dialogue games which in turn are sequences of dialogue moves. Each game pursues a particular goal and is initiated and terminated when certain situations occur. For example, the dialogue game in moves **(4)-(15)** in Section 3 has the goal to *explore the viewpoint* $\mathcal{V}_L$ by following viewpoint $\mathcal{V}_R$, and is initiated when there is sufficient information about $\mathcal{V}_R$ and no information about $\mathcal{V}_L$. The dialogue management checks the current situation and initiates or terminates dialogue games, accordingly.

5 Conclusion

The paper proposed a dialectic approach for exploiting viewpoint discrepancies in interactive learning environments. The key characteristics of our approach are that: (a) viewpoints are composed of positions, relevant beliefs and supporting arguments; (b) incompleteness of or discrepancies between viewpoints are used as triggers for argumentative dialogue games; (c) viewpoints represent statements valid in particular contexts, which is explored during the interaction; (d) while discrepancies and similarities between viewpoints are explored, changes in viewpoints are not imposed; (e) viewpoints are accumulated in viewpoint stores and can be shown to a learner to promote domain awareness or to a human tutor to highlight problems with learning resources or existing ontologies.

Currently, we are working on the formal description of our framework by employing argumentation dialogue frameworks based on situational calculus and dialogue games. At the same time, we are developing a Prolog-based proof of concept prototype to illustrate and validate the main definitions. The prototype uses a sample domain ontology about Programming Languages and takes as input Prolog-based definitions of dialogue moves (i.e. it assumes that the moves have been recognised). Once the framework is developed and tuned by using the prototype, we plan to deploy it in an existing educational semantic web system, e.g. OntoAIMS [2], to help learners make links between learning resources and become aware of different perspectives of content and ontologies.

References

1. Dicheva, D., Dichev, C.: Confronting some ontology-building problems in educational topic map authoring. In: Proceedings of Workshop on Semantic Web for Educational Adaptive Hipermedia at AH2006. (2006)
2. Denaux, R., Dimitrova, V., Aroyo, L.: Integrating open user modeling and learning content management for the semanticweb. In: 10th Int. Conference on User Modeling. (2005) 18–22
3. Moyse, R.: A structure and design method for multiple viewpoints. Journal of Artificial Intelligence in Education **3(2)** (1992) 207–233
4. Nichols, D.: Intelligent Student Systems: an Application of Viewpoints to IntelligentLearningEnvironments. PhD thesis, Computing, Lancaster University (1993)
5. Self, J.: Formal approaches to student modelling. In Greer, J., McCalla, G., eds.: Student Modelling: The Key to Individualised Knowledge-based Instruction, Springer-Verlag (1994) 295–352
6. Suthers, D., Weiner, A., Connelly, J., Paolucci, M.: Belvedere: Engaging students in critical discussion of science and publicpolicyissues. In: AIED. (1995)
7. Ravenscroft, A., McAlister, S.: Digital games and learning in cyberspace: a dialogical approach. E-Learning **3**(1) (2006) 37–50
8. Dimitrova, V.: Style-olm: Interactive open learner modelling. International Journal of Artificial Intelligence in Education **13** (2003) 35–78
9. Soller, A.: Computational modeling and analysis of knowledge sharing in collaborative distance learning. User Modeling and User-Adapted Interaction: The Journal of Personalization Research **14** (2004) 351–381
10. Tedesco, P.: Marco: Building an artificial conflict mediator to support group planning interactions. International Journal of Artificial Intelligence in Education **13** (2003) 117–155
11. Guha, R., McCool, R., Fikes, R.: Contexts for the semantic web. In McIlraith, S.A., Plexoudakis, D., van Harmelen, F., eds.: Proceedings of the International Semantic web Conference, Lecture Notes in Computer Science (2004)

CAMOU: A simple integrated eLearning and planning techniques tool

David Camacho† and María D. R-Moreno‡ and Unai Obieta‡

† Computer Science Department. Universidad Autónoma de Madrid. Madrid, Spain.
e-mail: david.camacho@uam.es
‡ Departamento de Automática. Universidad de Alcalá. Madrid, Spain.
email: mdolores@aut.uah.es

Abstract. In this paper we present an educational tool which has been designed to manage (learning) knowledge acquired from the interactions with the students, and to automatically aids educators in the complex process of course design and analysis. In the tool, only some essential learning knowledge will be translated (mapped) and provided to an automatic reasoning system, named IPSS. This system, which integrates Artificial Intelligence Planning and Scheduling, analyzes and detects problems in the current tested course, providing new solutions in form of new learning designs that can be approved (or rejected) by educators.

Keywords: e-Learning, AI Planning and Scheduling, Virtual Education Tools, Learning Designs adaptation.

1 Introduction

Most of the current Virtual Learning Environments (VLE) contain pre-fixed courses where the user navigates and learns the concepts that they have been planned for. Well known educational platforms are: First Class [1], LMS [2], WebCT [3], Moodle [4], or E-ducativa [5].

Those mentioned tools, and platforms, allow the instructors to get statistics as well as other information about the student progress. But there is still a lack of feedback among the previous users, the tool, the instructors and the future users. Among the tools that have worked in this direction we can mention the CourseVis system [11] that visualizes data from a java on-line distance course accessed through WebCT. The tool tracks the students evaluation and takes into account the instructors' requirements. This examination has to be done manually without any tool that can assist the instructor in the decisions that have to be made. Our approach can solve some of the deficiencies of eLearning

[1] http://www.softarc.com
[2] http://www.lotus.com/lotus/offering6.nsf/wdocs/homepage
[3] http://www.webct.com/
[4] http://moodle.org/
[5] http://www.e-ducativa.com/

courses and gives automatic solutions to the improving of existing courses by taking into account student interaction with them.

On the other hand, several (eLearning) standards and guides have been proposed related to learning object metadata, student profiles, course sequencing, etc. The IEEE Learning Technology Standards Committee (LTSC, 2006) has developed the Learning Object Metadata (LOM, 2006) standard which specifies the attributes required to describe a Learning Object (LO), where a LO is defined as any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning. Relevant attributes of learning objects to be described include type of object, author, owner, terms of distribution, format, and pedagogical attributes, such as teaching or interaction style. Another specification which allows the modeling of learning processes is the Learning Design (LD) information model (IMS LD, 2006) from the IMS Global Learning consortium. A learning design (LD) is a description of a method enabling learners to attain certain learning objectives by performing certain learning activities in a certain order in the context of a certain learning environment.

LD integrates other existing specifications. Among these, it is worth mentioning the IMS Content Packaging (IMS CP, 2006), which can be used to describe a learning unit (LU). A LU can have prerequisites which specify the overall entry requirements for learners to follow that unit. In addition, a LU can have different components such as roles and activities. Roles allow the type of participant in a LU to be specified. Activities describe the actions a role has to undertake within a specified environment composed of LO. LD also integrates the IMS Simple Sequencing (IMS SS, 2006), which can be used to sequence the resources within a LO as well as the different LO and services within an environment. Content is organized into a hierarchical structure where each activity may include one or more child activities. The learning process can be described as the process of traversing the activity tree, applying the sequencing rules, to determine the activities to deliver to the learner. However, the increasing interest, and research, in educational standards makes quite difficult to reuse them with other techniques such as Artificial Intelligence based. Currently complex mapping processes are hardly programmed to adapt different aspects from the eLearning standards (LOs, metadata, etc.) into an appropriate AI-based representation (i.e. PDDL planning representation language). Our approach, tries to simplify how to deal with these knowledge using only some statistical and educational interactions among students and educators, to integrate them into a reasoning module, to show how automatic reasoning techniques (i.e. planning and scheduling) can be used.

The paper is structured as follows. Section 2 provides a brief description about the related Artificial Intelligence techniques used. Next, Section 3 describes the learning tool developed to interact and test the educational courses. Then, Section 4 shows both how the integration among the AI reasoning system, and the educational system, has been done and provides a simple execution example. Finally, Section 5 shows the main conclusions and future work of the paper.

2 Automatic Reasoning in VLE

Although, the initial approach that we have followed in [4] integrates the IPSS [12] system in an adaptive (deployed) learning tool, named TANGOW [5]. TANGOW requires tasks and rules. The tasks define the units in the learning process, the rules specify the way of organizing tasks in the course along with information about the task execution (order among tasks, free task selection, prerequisites among tasks, etc.). However, in this work our main motivation is the study of the reasoning techniques to manage, and deal, with the educational problems, for this reason the approach presented in this paper does not need to define rules, since the new tool (CAMOU) does not perform any individual adaptation but the course per se. The new system is used for advising and fault detection and it is based on the statistic results of the students to replan the whole course.

2.1 Brief Introduction to AI Planning & Scheduling Techniques

In the last decades Artificial Intelligence (AI) Planning and Scheduling (P&S) has become a successful, and widely used techniques. It allows us to generate a sequence of activities that achieves a set of goals having in mind the time and resources available.

These techniques have been applied with success in different real (and complex) environments such as, Industry, Robotics, Space missions or Information Retrieval. Traditionally, there is a clear subdivision of techniques and roles that belong to Planning and Scheduling. Planning [2] generates a plan (sequence or parallelization of activities) such that it achieves a set of goals given an initial state and satisfying a set of domain constraints represented in operators schemas. In Scheduling systems, activities are organized along the time line having in mind the resources available. Scheduling has to face the problem of organizing tasks in time. The problem is to locate a set of tasks in time, each task needing one or several resources during its execution. Nowadays it is being an increasing interest to integrate AI P&S because of real domains needs. From this perspective, by combining them the weaknesses of both areas can be solved. In this direction, IPSS [12] has been built. Other approaches that have followed this approach are O-PLAN-2 [14], IxTeT [1] or EUROPA [8]. Using a high level description, the inputs to those kind of systems are:

- *Domain theory*: the STRIPS representation originally proposed by Fikes and Nilsson is one of the most widely used alternatives [7]. In the STRIPS representation, a world state is represented by a set of logical formulae, the conjunction of which is intended to describe the given state. Actions are represented by so-called operators. An operator consists of pre-conditions (conditions that must be true to allow the action execution), and post-conditions or effects (usually constituted of an add list and a delete list). The add list specifies the set of formulae that are true in the resulting state while the delete list specifies the set of formulae that are no longer true and must be deleted from the description of the state.

- *Problem*: is described in terms of an initial state and goals. Those states are represented by a logical formula that specifies a situation for which one is looking for a solution.

As output, the planner generates a plan with the sequence (linear or parallel) of operators that achieves a state (from the initial state) that satisfies the goals.

For scheduling systems, many techniques used in this area come from the Operational Research (OR) area [13] (i.e., branch and bound, simulated annealing or lagrangian relaxation). Lately, Constraint Satisfaction (CSP) [6] has been applied to the different scheduling problems with very good results. A CSP problem has inputs:

- A set of variables.
- A set of domains values containing the possible values for the corresponding variable.
- A set of constraints for the variables.

The output of scheduling systems is a values assignment that fulfills all the constraints in the variables.

As a result of the integration, they generate as an output a plan or set of plans (if a solution exists) time and resource consistent. A plan can be seen as a sequence of operator applications (learning activities) with a specific duration that can lead from the initial state to a state in which the goals are reached with the resources available (i.e. educators available).

In educational environments several works to automatically generate courses based on pedagogical tasks and methods has been performed.For instance, in [15] an AI hierarchical task network (HTN) planner called JSHOP [10] which assembles learning objects retrieved from one or several repositories to create a whole course has been used. The learning objects are linked by taking into account the user knowledge information and the learning goals that the user should achieve. Our approach not only can link learning objects, but also schedule them along a period of time and consider previous student results to generate different LDs.

2.2 Integrating Planning and Scheduling: IPSS

The IPSS system is divided in two blocks as shown in Figure 1. The Plan Reasoner (IPSS-P) composed of an heuristic planner and an a deorder algorithm [3]. The deorder algorithm transforms the sequence of activities given by the planner (Total Order plan) into a parallelization of activities, eliminating the inneccesary precedence constraints (Partial Order plan). And the Scheduler reasoner (IPSS-S), is represented as a Constraint Satisfaction Problem (CSP) partitioned in two sub-problems. A basic Ground-CSP to reason on temporal constraints and a Meta-CSP to reason on resource constraints. Like that, IPSS is able to manage not only simple precedence constraints, but also more complex temporal requirements and multicapacity resource usage/consumption.

Then, the reasoning is subdivided in two levels. The planner focuses on the actions selection (possibly in the optimisation of some quality metric different than time-resource usage), and the scheduler on the time and resource assignments. During the search process, every time the planner chooses to apply an operator, it consults the scheduler for the time and resource consistency. If the resource-time reasoner finds the plan inconsistent, then the planner backtracks. If not, the operator gets applied, and search continues until a solution is found.

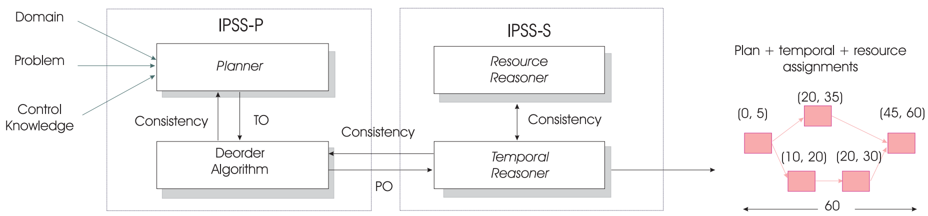

Fig. 1. Planning IPSS architecture.

3 Statistical course redesign based on planning techniques: CAMOU

Using our previous experience, we have designed and implemented a new learning tool which facilitates the definition of LDs and the acquisition of student interactions, both kind of data are later translated to be automatically analyzed by IPSS. The tool, named CAMOU, has been implemented using the following modules (Figure 2):

- *Learning Design Generation Module.* It allows (educators) to manage all the activities related to LDs generation and monitoring (i.e. create a new LD, modify, delete, or listing the stored LD), Figure 3 shows some screenshots of these functionalities, they can be summarized as follows:
  - *Learning Design management.* It allows to define the information related with a particular LD stored in the system (Figure 3 a) and b)), i.e. number of educators, groups...
  - *Unit of Learning management.* It is used to define the Unit of Learning (UL), and their associated pedagogical contents that defines the course. We use a meta-data representation, that can be used by other elements in our system (i.e. IPSS planner) to reason with the stored information (Figure 4 a) and b)). It is quite interesting to remark that some meta-data information related to the maximum and minimum duration for each UL should be provided by the educator (later this information will be used by the reasoner).

```
1- Learning Design Generation Module:
    - Learning Design management
    - Unit of Learning management
    - Dependencies management

2- Students & Educators Management Module:
    - Educators management
    - Student management

3- Exams & Tests Module:
    - Question generation module
    - Answers generation module
    - Exams & test management

4- Statistical Module:
    - Exams & Test statistics
    - Group & Students statistics
    - Questions statistics
```

Fig. 2. CAMOU Architecture.

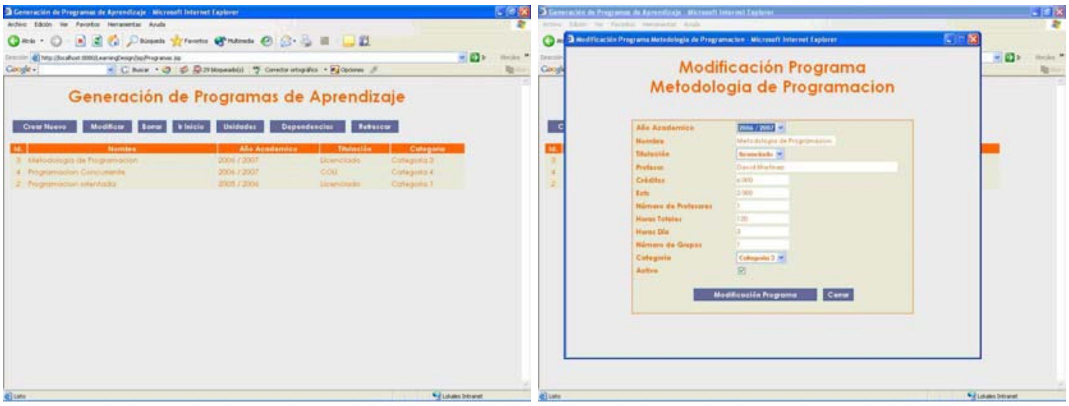

Fig. 3. a) LD listing; b) LD meta-data modification

- *Dependencies management*. This submodule allows to define (or modify) two different kind of dependencies (weaks and strongs) between the different UL that defines the course (Figure 5 a) and b)).

– *Students & Educators Management Module*. This is module allows (using several Web interfaces) to manage the main actors in the system, educators and students. Figure 6 shows both how a particular educator is registry in the system, and the current list of students for a particular course.

– *Exams & Tests Module*. This module allows the educator to generate (modify or delete) both the questions and the related answers that will be used to make the exams and tests to our students. This module incorporates meta-data information related with both UL and LD. Figure 7 shows (a) several

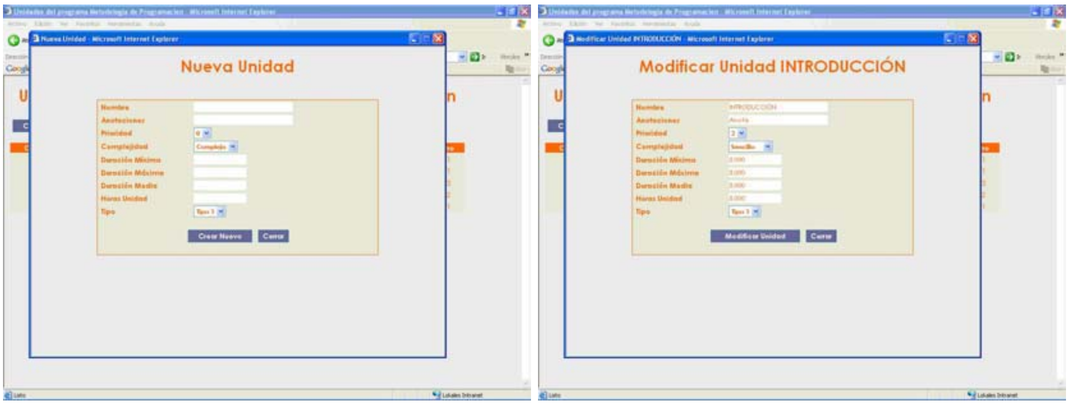

Fig. 4. a) UL definition; b) UL modification

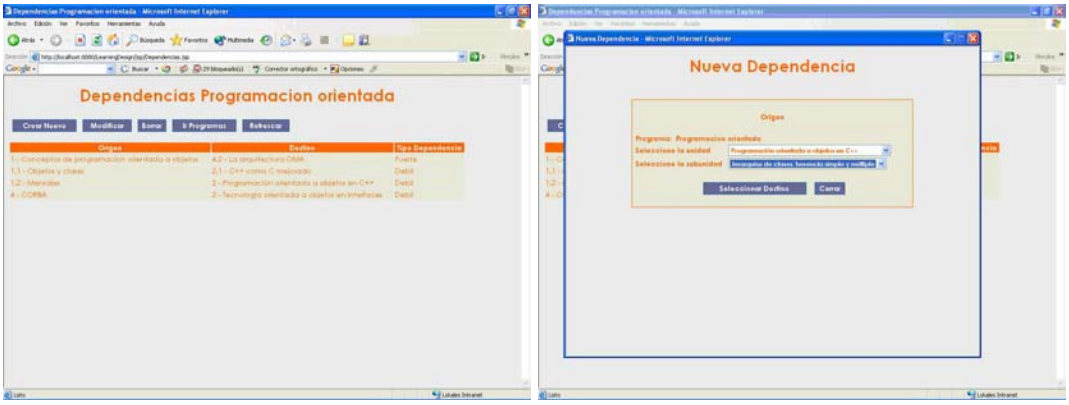

Fig. 5. a) Listing some existing dependencies for a particular LD definition; b) Dependency definition

questions and their UL related that have been created and stored in the system, and (b) how a new question is generated in the system.

– *Statistical Module*. Finally, this module generates a set of classical statistical values for different issues: groups, questions and persons. Figure 8 shows (a) several statistical results for each group, and (b) the statistics for each question.

4 How is the integration between CAMOU and IPSS done?

In this section we show the process that students and educators follow for a particular course, i.e. a LaTeX course [9]. We present this example to illustrate

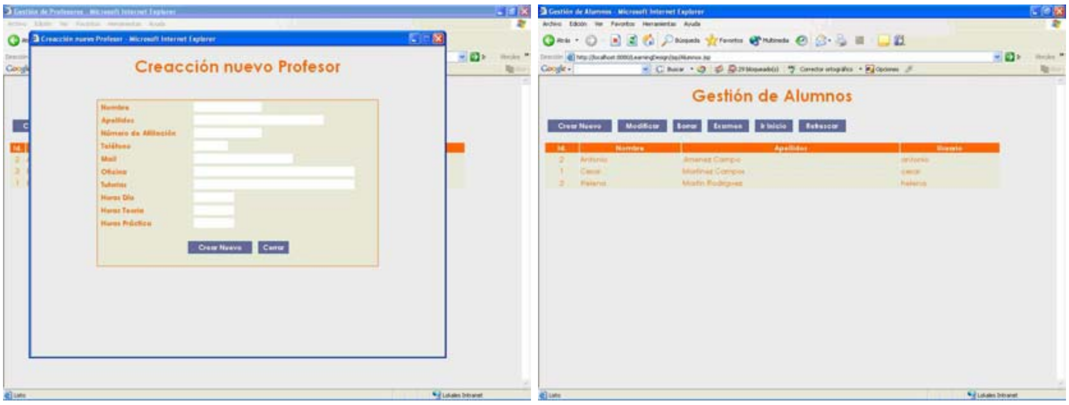

Fig. 6. a) Registration of a new educator; b) Listing some students

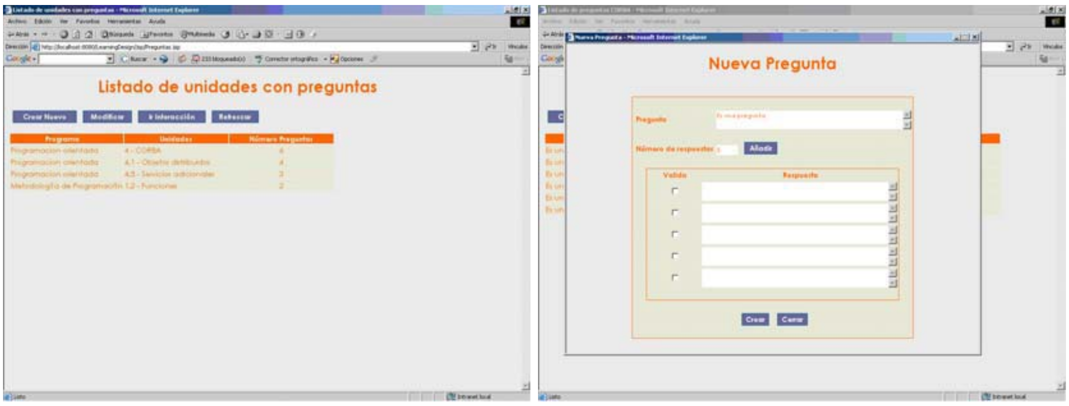

Fig. 7. a) Listing of Question for several UL; b) Generation of a new Exam/Test Question

how the integration is done. The first step is to define all the information about the units that are part of the course and associate to them the contents and exercises. This task will be done by the educators using the *Learning Design Generation Module* described on previous section.

Figure 9 shows the different units and subunits that compose the course, and some annotations such as the minimum and maximum duration, the priority or the complexity. The tool checks that the total course duration (known as a `makespan` in AI terminology) is equal to the sum of the units and subunits. If there is an inconsistency, a message is presented to the educator before the automatic module can be run.

Another information that we should provide is the dependencies (i.e. weak and strong). All this information is needed in order to translate it into IPSS. Units

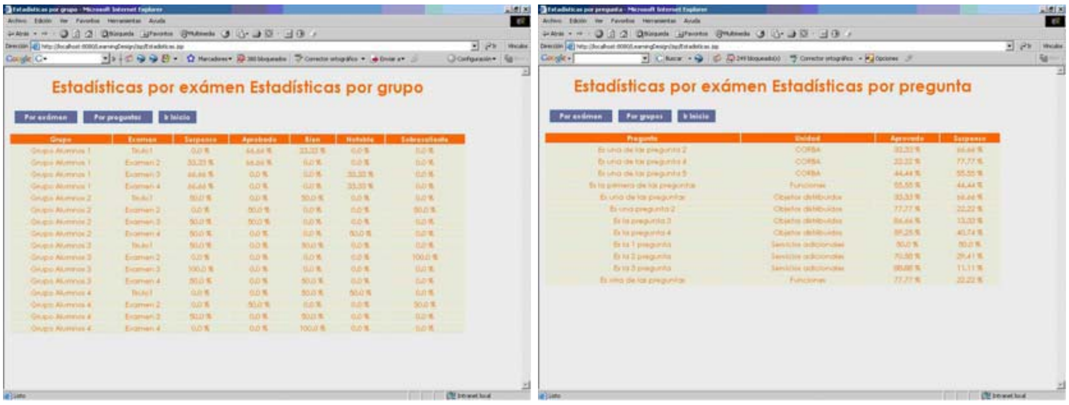

Fig. 8. a) Statistical results for different exams/groups; b) Statistical results for several questions and LD

```
1- Introduction (priority = 3, duration (2,4,6) Complexity = very low):
    1.1- History
    1.2- Components

2- Structure of a Document (priority = 8, duration (7,10,13) Complexity = medium):

3- Basic Formattin Tools (priority = 4, duration (3,4,5) Complexity = medium):

4- The Layout of the page (priority = 5, duration (6,8,10) Complexity = high):

5- Tabular Material (priority = 6, duration (12,14,16) Complexity = high):
    5.1- Tabbing
    5.2- array
    5.3- supertab and longtable
    5.4- Applications

6- Mastering Floats (priority = 4, duration (6,8,10) Complexity = high):
```

Fig. 9. An example of a LaTeXcourse.

with weak dependencies could be eliminated from the course in case some other units require more duration. IPSS will decide based on the dependencies, the minimum and maximum duration, and the priority. The units with high priority and weak dependencies are less probable to be eliminated than the units with low priority and weak dependencies. The base priority, that makes IPSS to decide which units can be part of the course or not, should be provided by the educators for a complete description).

Once the educators have introduced all the information, the students can start using the tool. It is now up to the educators to evaluate the student's knowledge and psychological model. This test that can be performed through the tool, will allow the educators to define and know the student profile. Actually, when a student starts a course, the student previous knowledge is uncertain and the educator does not know what can be the main difficulties that he/she has to face with. Thanks to the new Information Technologies and well made tests, this information can be known almost immediately and it can automatically be translated into the initial state of a planning problem and the preconditions of the operators.

At the beginning all the students will start the course with the first unit of learning: "Introduction". IPSS will assign to the course the minimum time duration that the educator has decided, due to the low priority and complexity values. Until know, there are not many options for the scheduler to plan for different solutions.

After one or several units, let us suppose that an exam is planned. The students are now in the "Tabular Material" unit of learning, and thanks to the tests, we have a personalized knowledge of the weak points of the already learnt subunits.

From the results we can know that 70% of the students have failed the "array" sub-unit. Then, a failure in the Learning Design (LD) has been detected. This information is saved for the future LD revisions. In this situation, the pedagogical responsible can decide to add more examples to this subunit, what implies the increase in the minimum, medium and maximum duration time. This increase of time in one of the modules will produce a reduction in other modules in order to keep consistency with the global course duration (deadline). And in more drastic cases, to eliminate one (or several) subunit(s). That decision will be made automatically by IPSS, but it is the responsibility of the pedagogue to check the consistency from the pedagogical point of view.

5 Conclusions

In this paper we have described both, a simple tool (CAMOU) that has been designed to manage educational knowledge acquired from the interactions with the students, and how it can be integrated with an automated reasoning system (IPSS) to help educators in the complex process of course design. Although there exist some current eLearning standards (i.e. IMS, LOM or SCORM) widely used by the e-Learning community, when these standards are combined, or integrated, with other techniques (i.e. Artificial Intelligence) it can be quite hard to represent, translate and manage the stored knowledge due the complexity of those standards. Our approach tries to simplify how to deal with this knowledge using only some statistical and educational interactions among students and educators, and integrate them into a reasoning module.

The CAMOU tool only uses some essential learning knowledge that is translated (mapped) and given to IPSS. Using only these knowledge we are trying to minimize the bias in the translation/mapping process, because if we try to map all the eLearning knowledge, possibly important semantic and syntactic knowledge will be missing by the reasoning system.

Acknowledgements

This work has been funded by the research projects: CYCIT project TSI 2006-12085 and Junta de Castilla-La Mancha project PAI07-0054-4397.

References

1. P. Albers and M. Ghallab. Context Dependent Effects in Temporal Planning. In *Procs. of the 4th European Conference on Planning, Toulouse, France.*, pages 1–12, 1997.
2. J. F. Allen, B. Hendler, and A. Tate. *Readings in Planning.* Morgan Kaufman, 1990.
3. C. Bäckström. Computational aspects of reordering plans. *Journal of Artificial Intelligence Research*, 9:99–137, 1998.
4. D. Camacho, A. Ortigosa, E. Pulido, and M. D. R-Moreno. *AI techniques for Monitoring Student Learning Process.* Information Science Reference, formerly Idea Group Publishing, Ed. by Francisco J. García, 2008 (to appear).
5. R. Carro, A. Ortigosa, and J. Schlichter. A rule-based formalism for describing collaborative adaptive courses. In V. H.-R. Palade and S.-V. L. Jai, vol. 2774, editors, *Knowledge-Based Intelligent Information and Engineering Systems. Proceedings of the International Conference KES'2003*, pages 252–259, 2003.
6. R. Dechter and F. Rossi. Constraint Satisfaction. *Survey ECS*, 2000.
7. R. Fikes and N. Nilsson. STRIPS: A new Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, 2:189–208, 1971.
8. J. Frank, A. Jónsson, and P. Morris. On Reformulating Planning as Dynamic Constraint Satisfaction. In *Procs. of the Symposium on Abstraction, Reformulation and Aproximation (SARA), Horseshoe Bay (Lake LBJ), Texas, USA*, pages 271–281, 2000.
9. M. Goossens, F. Mittelbach, and A. Samarin. *The Latex Companion.* Addison-Wesley, 1994.
10. O. Ilghami and D. S. Nau. A general approach to synthesize problem-specific planners. Technical Report Technical Report CS-TR-4597, Department of Computer Science, University of Maryland, 2003.
11. R. Mazza and V. Dimitrova. CourseVis: Externalising student information to facilitate instructors in distance learning. In U. Hoppe, F. Verdejo, and J. Kay, editors, *Proceedings of the International conference in Artificial Intelligence in Education*, pages 117–129, 2003.
12. M. D. R-Moreno, A. Oddi, D. Borrajo, and A. Cesta. Ipss: a hybrid approach to planning and scheduling integration. *IEEE Transactions on Knowledge and Data Engineering.*, 18(12):16811695, December 2006.
13. H. A. Taha. *Operations Research: An Introduction.* Prentice Hall. Eighth edition., 2006.
14. A. Tate, B. Drabble, and K. R. *O-Plan2: An Open Architecture for Command, Planning, and Control.* Morgan Kaufman, 1994.
15. C. Ullrich. Course Generation Based on HTN Planning. In *Proceedings of 13th Annual Workshop of the SIG Adaptivity and User Modeling in Interactive Systems*, pages 74–79, 2005.

Activity Systems and Context Working as Core Concepts in Modeling Socio-Technical Systems

Heidrun Allert, Christoph Richter

Research Group Knowledge Media, Upper Austria University of Applied Sciences,
Softwarepark 11, 4232 Hagenberg, Austria.
{heidrun.allert;christoph.richter}@fh-hagenberg.at

Abstract. Current modeling approaches in the field of learning and work resemble the notion of workflows and hence fall short in describing the situated and socially mediated nature of practices. Against this background the paper describes an alternative modeling approach as well as its theoretical foundation and practical implications. As this paper rethinks the epistemological foundation of modeling socio-technical systems, the approach goes beyond specifying specific concepts and relations and addresses the meta-level of modeling. The cultural historical activity theory as well as the theory of social systems work as the theoretical foundation.

Keywords: Knowledge representation, activity theory, socio-technical systems

1 Introduction

The formal description of socio-technical systems as well as learning processes has attracted attention among researchers and developers in recent years and has resulted in a couple of specifications focusing on individual as well as collaborative processes of learning and working. The explicit and formal representation of such processes is relevant for diverse reasons. Besides their technical and economic relevance they also provide an important communicative tool for designers as they allow to share experiences and to coordinate activities among those involved in the design and development process. Furthermore they are of interest for scientists as they provide a frame of reference for the analysis and comparison of different scenarios. While current modeling languages such as e.g. IMS Learning Design [8] overcome the problem of de-contextualized objects by describing the use of these objects within a unit of study, they resemble traditional workflow models and hence reproduce the problem of contextualization on a higher level, as the unit of study is again de-contextualized. Even though these approaches acknowledge the complex nature of situated processes they are reductive in the sense that they equal the processes with the sum of the activities entailed. Thereby the situated and socially mediated character of human action is neglected. Against this background this paper outlines an alternative modeling approach which draws on activity-theoretical as well as systemic theories to depict practices. The formal concept of role types is used to represent the systemic nature of activity and its situatedness adequately. The paper is structured as follows: Key assumptions of the cultural-historical activity theory as well as the theory of social systems are introduced to outline the underlying rationale of the modeling approach. Referring to the theoretical foundation the modeling approach is developed step by step. Finally the practical implications of the modeling approach are discussed.

2 The Concept of Practice

The concept of practice can be defined as *the ways of doing work, grounded in tradition and shared by a group of workers* [4]. In general it has to be distinguished between practices as implemented by a specific group of people (e.g. the way a particular lecture is given at a particular university) and practices as prototypical conceptualizations of a certain activity within a broader community (e.g. the way lectures are given usually). While the concept of practice can basically be defined as a customary way of doing things, it seems worthwhile to have a closer look at this concept from a theoretical point of view. Theories this work is founded in are the activity theory and the theory of social systems. The following is a list of key-assumption on human activity and social systems, which is based on activity-theoretical and system-theoretical (systemic) considerations. A more detailed and extended outline of these key assumptions is given in [2].

2.1 Key Assumptions of Activity Theory

Activity theory is a powerful philosophical framework and descriptive tool focusing on understanding human activity and work practices. It is based upon the anthropological and psychological theory of Leontjew [11] and Vygotsky [19].

(1) Human activity is object-oriented, i.e. it is directed towards a physical or conceptual object that is manipulated or transformed by the activity. It is the *object of activity* and not the goal that allows distinguishing different activities from one another. (2) Activities are always mediated by *tools* and signs, which are constitutive elements of the activity. Tools and signs are mediators which range from material tools over less tangible artifacts like plans and spreadsheets to scientific theories and languages. Tools capture and preserve the socially shared knowledge developed in a given community and mediate the subjects' relation with the object of the activity as well as with other human beings [11], [16]. (3) Activities are shaped by contextual conditions and circumstances. The *subject* has to continuously adapt its actions and operations to external events and circumstances. As a consequence human activity is guided but not predefined and determined by the plans of those engaged in the activity [3]. The variability of contextual conditions and circumstances inevitably results in a variation of the way the activity is carried out and can result in the evolution of the activity system if improper variations are selected and proper variations stabilized [14]. (4) The relationship between subjects, objects, and tools is reciprocal. These elements are mutually interdependent, which means that a change in one of them will inevitably alter the other ones. In this sense the constituents of an activity form a system where each component is defined in relation to the other

components. (5) Activities are hierarchically structured. According to [11] three levels of activities can be distinguished, namely collective *activities* which are carried out on a communal level often involving multiple actors, *actions* that are performed by a single subject to achieve a certain goal relevant to the collective activity, and *operations* in the form of fine grained automated routines. But even though activities are structured hierarchically the relation between operations and actions as well as actions and activities is not an additive one [11]. Therefore it is not possible to simply decompose an activity into a set of actions. (6) Activities are never static but evolve when contradictions or tensions emerge between the elements in an activity system. Human activity whether carried out by an individual or collectively cannot be detached from its social context as its meaning is bound to its interpretation within a collective.

2.2 Key Assumptions of the Theory of Social Systems

The Theory of Social Systems is a descriptive framework presenting a system-centered view. It is a non-prescriptive meta-theory. [20] characterizes this theory as universal regarding domains and disciplines as many disciplines are confronted with similar problems, e.g. the problem of increasing complexity, which can not be reduced to simple categories and principles. A comprehensive introduction into the Theory of Social Systems is given by [20] and [9]; the foundational work is Social Systems [10]. Here only few key assumptions are presented.

(1) Personal systems as well as social systems are meaning processing systems as they process information by constructing meaning. A social system is not equivalent with the group of people in the system, but it is of different quality. Personal systems and social systems reduce environmental complexity by processing environmental complexity selectively. Thus, inner and outer complexity is different. Systems organize their inner complexity and reduce contextual (environmental) complexity. (2) The Theory of Social Systems is a descriptive framework which describes the world in terms of systems, drawing the difference system/environment, whereas in object-oriented modeling objects and categories are defined. *The central paradigm of recent system theory is 'system and environment'. The concepts of function and functional analysis no longer refer to 'the system' (...) but to the relationship between system and environment. (…) This leads to a radical de-ontologizing of objects as such (...)* [10]. The difference system/environment is not an ontological but an epistemological. (3) Systems are closed and self-regulated. Within a system, elements generate each other reciprocally, e.g. in listening, the audience creates the speaker and vice versa. An entity, such as a person (personal system) does not belong to a social system but to its environment [10]. This means, a person (and any other entity/type) does not belong to a system for all intents and purposes but in some respect, filling a specific role [20]. For example: The person Peter and the person John belong to the environment of the system family. Only Peter filling the role son and John filling the role father, belong to the system. A system can not determine another, e.g. a personal system can not determine a social system.

3 Modeling Practices as Coherent Social Systems

This section outlines a modeling approach for modeling socio-technical systems. The approach draws on the concept of practice and refers to activity systems as coherent social systems. As this paper rethinks the epistemological foundation of modeling socio-technical systems, the approach goes beyond specifying specific concepts and relations and addresses the meta-level of modeling. The (meta-) modeling approach is based on three major inputs: (1) distinguishing the meta-level categories `natural type` from `role type` to distinguish between an object and its role within a specific context [7], [17], (2) introducing a system-centered perspective to model a system of elements which reciprocally generate each other, and (3) integrating basic assumptions of activity theory in order to overcome shortcomings of workflow models which work as means-end-models. The structure of this section is as follows: First a basic comprehension of the formal terms natural type and role type is given, based on the work [6], [7], and [17]. Then, a system-centered (systemic) perspective is delineated, modeling socio-technical systems as coherent social systems [1]. Finally the meta-model of the system-centered role-based modeling approach, which reflects an activity-theoretical and system-centered perspective, is presented.

3.1 Meta-Level Categories Natural Type and Role Type

In the context of knowledge representation, meta-level categories are categories used to model the world, such as `concept`, `property`, `state`, `role`, `attribute`, and `relation`. Within this work, distinguishing the meta-level category `natural type` from `role type` is crucial, based on [7] who provides an ontological distinction. This distinction is based on the meta-properties `identity` and `rigidity` (table 1). [18] states that the definition of natural types matches the class construct of object-oriented modeling, as the definition of classes is outside the context of any relationships, and the instances keep their types for their lifetimes (identity). A type is a natural type if *belonging to the type is independent of being engaged in a relationship (except for, perhaps, a whole-part relation) and if an object cannot leave the extension of the type without loosing its identity*. [18]. Natural types relevant in modeling socio-technical systems and (knowledge) practices are e.g. `person` and `artifact` (such as `technology, services, information asset).` Person here represents any meaning processing system (e.g. a group, an organization; according to [10]). The category of role type is as fundamental in object-oriented modeling as the category of natural types, classes, and relations [17]. Due to the fact that usually no difference is made between the concepts of natural types and role types, the concept of role types is relatively unknown. The concept of types normally represents both: natural types and role types. Due to a synopsis prepared by [17] the concept of role type does not play a role in most formal languages, including the logics (cp. modeling and the formal grounding of maths by Frege, 1848-1925), while it plays a major role in linguistics and semantics [5]: In linguistics there is a common theory of formal languages, integrating the role type as fundamental concept complementing the concepts of predicates and objects.

The difference between role types and natural types is in its contents. Syntax allows to work without distinguishing the concepts - but, semantically many problems arise from not drawing the difference between the concepts [17]. Husserl introduces the quality of Fundierung (en: foundation), cp. [17]. [6] (in the context of knowledge representation) specifies semantical and ontological rigidity. A concept is founded if none of its instances can exist alone: Each instance is related to another instance. A concept is semantically or ontologically rigid if an instance can not join and leave the extension of the concept without loosing its identity. If x *has the property of being an apple, it cannot lose this property without losing its identity (...). This observation goes back to Aristotelian essentialism (...).* [7]. [6] founds the concept of role type as an ontological concept and gives a formal definition assigning two conditions. Role types are those concepts which are founded and not semantically rigid. Natural types are those concepts which are semantically rigid and not founded. According to [7], the meta-property rigidity means: *A property P is rigid if, for each x, if P(x) is true in one possible world, then it is also true in all possible worlds. Person and location are rigid, while student and tall are not.* A role type specifies the behavior within a context - a behavior is a contract or relationship between two entities. A role type implies a specific relationship between instances filling the role. Role types require the instance to have an identity apart from its role type. Natural types do not imply a specific relationship with other types (except for whole-part relations). Natural types grant an instance its identity. The concept of role types allows describing the function an object fills within a specific context. [17] states that the standardization of the term *role* (role type) in modeling complements the meta-level categories type and relation. Instances of types can *fill roles*. The classical dichotomy type/relation is extended to the trilogy type/role/relation. [17] works out practical implications for its integration in object-oriented modeling and its representation in the modeling language UML ([13]). Introducing the concept of role types into object-oriented modeling makes possible dynamic modeling approaches. Role types are dependent from relations and context. Each instance of a certain natural type can fill different role types, called polymorphism [17]. Role types and natural types (in the context of object-oriented modeling [17] refers to natural types as classes) are interconnected by the `supports` relationship, specifying which classes support which roles [17]. The role type specifies the behavior instances of a natural type must provide in order to be able to fill the role. How the behavior is achieved is left up to the classes that support it. It depends on the classes' properties and qualities whether its instances can fill a role or not.

Table 1. Distinguishing the ontological concepts natural type from role type.

Natural Type	Role Type
Static	Dynamic
Semantically rigid: An instance of a class once and forever belongs to that class. It cannot change it without loosing its identity.	Not semantically rigid. Instances of natural types can fill, adopt and leave a role without loosing their identity [6].
Not founded	Founded. Role types are defined by context and relation.

Integrating the concept of role types in UML, the notation for role types must be specified. [17] recommends using the lollipop-notation, which in UML represents interfaces. In the following UML diagrams a rectangle indicates a natural type, a circle indicates a role type (fig. 2). The UML diagram specifies role types the instance of a natural type can fill. In specifying metadata, it is necessary to distinguish between static attributes (such as Dublin Core and vCard attributes), which are based on the natural type of a resource, and context- or role-dependent attributes which are based on the role type a resource fills. Natural types such as information assets and persons have context-independent static attributes. These static attributes are independent from the role a resource fills. Regarding an information asset, static attributes are taken from Dublin Core (Dublin Core Metadata Initiative, 2004). Persons are annotated with vCard attributes like vcard:FN (full name) and vcard:EMAIL. Besides static attributes, context-specific role-based attributes are attached to resources. Role-based attributes are specified according to a specific context.

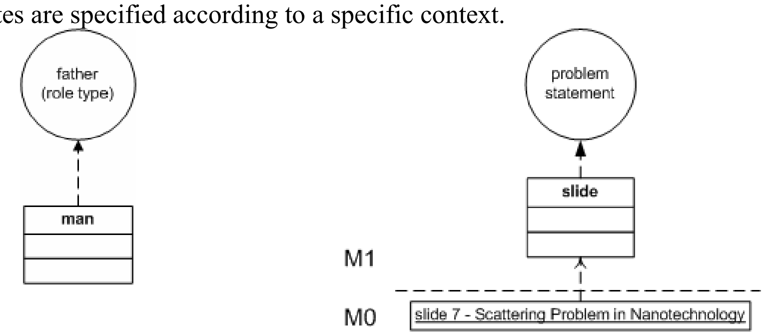

Fig. 2. Notation in UML. Man is a natural type, father is a role type. The role father can be adopted and dropped by instances of the class man. An instance of the natural type `slide` fills the role type `problem statement`. Not distinguishing the natural type from the role type mixes the resource and its function/role within a specific context (as e.g. in LOM's category Learning Resource Type).

The System-Centered Perspective - Modeling Coherent Social Systems

[7], [17] define the meta-level category role type as a binary relation. A role type is defined by its relation to another role type. In this section we complement this definition by a system-centered perspective and define the concept of role type as n-ary relation. We refer to role types as `role` and to natural types as `type`.

Roles within a system are related. They generate each other, as elements within an activity system generate each other reciprocally. For person related roles this means for example: there is no accused without a complainant, no father without a son or daughter. A person (natural type) filling a role within a system has expectations towards the other persons filling roles. The accused has specific expectations towards the judge. An instance of a natural type fills a role as soon as it moves into the system. In case of the natural type person, the concept of role types is intuitively understood (fig. 3). But also further natural types such as information asset (e.g. a picture), behavior, technology, service, etc. fill roles within diverse systems (fig. 4). In the same way activities are related and generated by one another within the system. Within the legal system (which serves as an example here) the type picture does not exist. But a picture which fills the role indication does exist in the legal system. This means: as soon as someone hands in a picture the judge will bring it into the system as indication (the picture filling the role indication) - or refuses to do so. Only filling the role indication (or another) the picture is part of the system. The same with the role evidence: only as the judge accepts an asset as evidence it becomes part of the system. It is not part of the system per se, but filling the role evidence. Types do not belong to a system but to its environment [10]. An instance of the natural type person which fills the role accused in the legal system fills the role father in the system family, each with specific intents, aspects, and purposes. Modeling coherent social systems, this work argues that roles are arbitrary n-ary relations. This is different from the definition of a role as a binary relation [7]. Whereas [7] interprets an unary predicate as a concept (class) and a binary predicate as a role, this work assumes a role as n-ary predicate. The relation `father-son` is insufficiently described by a binary relation as in a system the relation father-son is entirely affected by any other role represented in the system e.g. the role mother. The absence of an instance filling the role mother entirely affects the relation father–son.

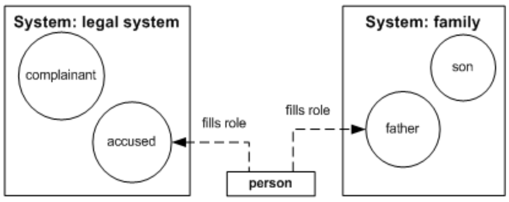

Fig. 3. A person (natural types) filling roles within different systems.

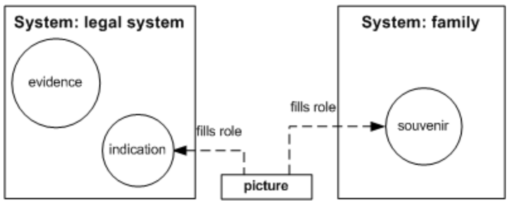

Fig. 4. A picture (natural types) filling roles within different systems.

Modeling an activity system requires a further level of abstraction as it has a (theoretical) foundation and underlying rationale (e.g. the concept of *Bildung* or the concept of *situated learning*). This foundation conceptualizes the system but is not formalized. The underlying rationale of the system is reflected by the *meta-type* in M2 (meta-level 2, fig. 5). The meta-meta-level category meta-type is crucial in modeling socio technical systems, as there always is an underlying rationale which is not formalized. The meta-type reflects central issues/culture/identity of the activity system.

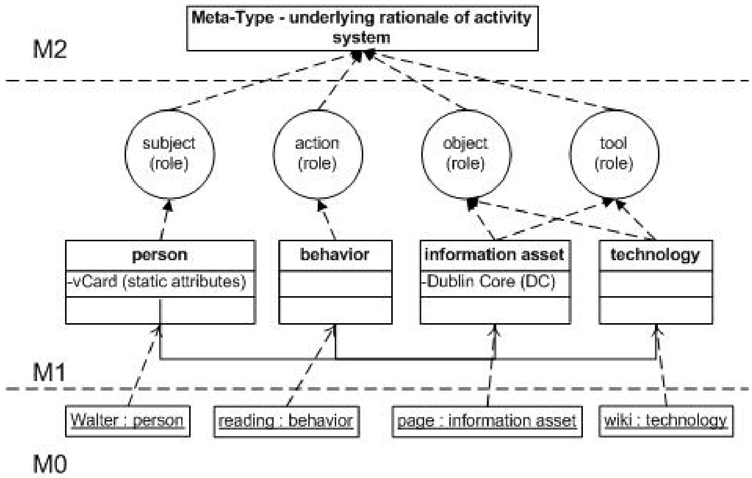

Fig. 5. A set of natural types (person, behavior, information asset, technology), a set of role types (subject, action, object, tool (M1)) and a meta-type (M2) is specified.

The role-based modeling approach allows modeling a natural type filling different roles within different activity systems. Tim, an instance of the natural type `person`, fills the role `manager` at workplace and the role `learner` within the executive MBA program. Interoperability between the activity systems and different contexts is given via the natural type. First, a set of role types and natural types is specified. Then, the roles are related. Typically the natural types filling roles are related in a cause-and-effect chain, forming a process-oriented workflow model. In the context of learning this means, a subject performs an activity to reach a predefined goal (cp. e.g. [8]. Learning is assumed to result from a chain of actions. Such a model would oversimplify learning for several reasons, cp. [14]. To avoid this, this work models `action` as an n-ary relation and `action` and `activity` are modeled on different levels of emergence (fig. 6). Thus the modeling approach addresses several key assumptions of Activity Theory and the Theory of Social Systems: (1) learning is contextualized, (2) activities can not be de-composed to several actions without loss of information - the relation between operations and actions as well as actions and activities is not an additive one, (3) the elements of a system generate each other, (4) learning can not be reduced to a chain of actions - it is not possible to simply decompose learning into a set of actions, (5) social systems are meaning processing systems - the difference between a social system and a group of persons is not a quantitative but a qualitative one. A final example might illustrate this: Taking into account Leontjew's [11] concept of activities, actions and operations, one and the same action is capable to be a component of different activities. An activity can not be decomposed to the actions it contains without loosing information. The action of reading is different depending on the activity the learner carries out (reading a problem statement in a setting of knowledge creation learning, or reading out loud in a setting of instructional design, e.g.).

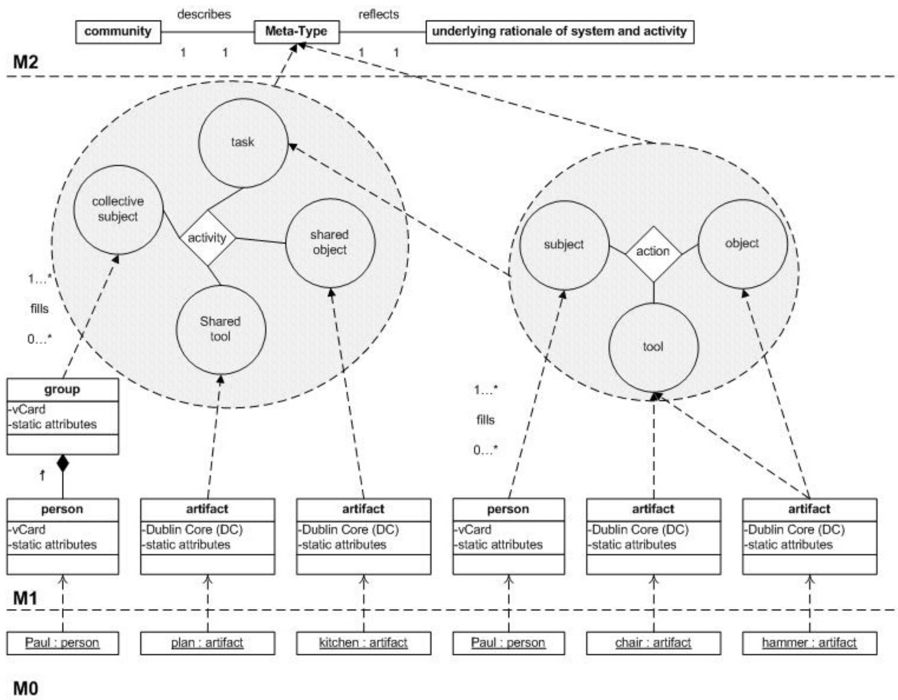

Fig. 6. The meta-model of a system-centered role-based modeling approach. According to activity theory and the theory of social systems, the elements within a system generate each other. Changing one element within the system also effects all the others.

4 Practical Relevance

The aim of this section is to elaborate on the practical implications of the meta-model introduced before. Thereby it is important to note that the model described so far is a meta-model providing the semantics but not the syntax of a respective modeling language. The aim is to demonstrate the general implications of the meta-model proposed. In order to do so, several concrete modeling problems related to learning and education will be discussed.

4.1 Decomposition of activities

The problem of decomposition of activities directly relates to the relationship between activities and actions. The question at stake is whether an activity can be broken down into a set of interrelated actions without loss of information. Common modeling languages such as the Unified Modeling Language [13], the Business Process Modeling Notation [12] and IMS Learning Design [8] are build on the assumption that such a decomposition is possible and hence equate the sequence of actions with the respective activity. Nevertheless this is problematic not only from a theoretical but also form a practical point of view. Given that the assumption would be true it would follow that the sequence of actions including the actors, artifacts and tools used would suffice to describe an activity. Consequently it would be possible to compare two activities by comparing the actions entailed. For example the IMS-LD Best Practice Guide [8] describes a problem based learning scenario as an arrangement of 17 actions, implying that differences between pedagogical scenarios are due to the organization of actions entailed. From this point of view it would not make a difference whether the students would have to solve a well- or ill-structured problem, whether it is a theoretical or practical problem, or whether there is a real customer interested in the outcome or not. All these differences cannot be modeled adequately and therefore result in misleading or even wrong comparisons across pedagogical scenarios in particular and practices in general. On the other hand, the meta-model introduced here can handle these differences as it models activities as entities of their own, which cannot be decomposed.

4.2 Equivalence of actions and its components

Another problem relates to the comparability of actions and its components, i.e. the question whether two actions are equivalent or not. From the practical point of view this question is of interest with regard to the modification of a pedagogical design or the implementation of a given design in another context. Both modification and re-implementation require knowing the constituting elements of the original solution in order to modify or transfer them intentionally. Modeling language that do not distinguish between role- and type-based attributes of the objects involved, such as the ones mentioned above, run into trouble when it comes to the equivalence of actions and its components. The problem is that they either generalize to natural-classes or that they mix up role- and type-based attributes. In both cases the misleading conclusions might be implied. For example in a given pedagogical design the students might be administered a multiple-choice test in order to assess their understanding of the topics addressed in the course. The aim of the multiple-choice test in this case is to provide an overall feedback whether the students understood the core concepts or if remedial activities are required. When modifying or adapting the pedagogical design it might become relevant to replace the multiple-choice test by another instrument and hence to know what is equivalent to this test. In case one generalizes to the natural-class the static attributes of the test, namely that it is a multiple-choice test focused on domain specific topics comes to the fore while its particular purpose is dropped. Consequently the test might be replaced by another test which is not designed to provide formative but summative information on students' performance. Even if no such generalization is made, it still remains unclear which attributes of the test are relevant and which not and hence might lead to the false conclusion that it is important to use a multiple-choice test while in fact any other instrument providing feedback on students understanding would be suitable.

Coupling of actions

The last problem to be discussed here relates to the modeling of interrelations between actions. Following the idea of hierarchical decomposition most of the current modeling languages treat activities and actions as self-contained entities related to other activities and actions via respective pre- and post-conditions. Consequently activities and actions are either organized sequentially or in parallel, while in the later case no direct dependency exists between the actions while being carried out. While these approach allows to depict the overall flow of actions and activities it ignores the fact that actions or activities are often coupled via the persons involved and the artifacts used. In other cases two actions might have to be coupled in order to work correctly. Even though giving a lecture and listening to the lecturer can be decomposed into two distinct actions the coupling of these actions is essential for the overall outcome. The mutual dependency of synchronous actions or activities is of vital importance for understanding the mode of operation. While these dependencies cannot be described adequately when action and activities are treated as self-contained entities, the meta-model introduced here overcomes this problem by allowing a person or artifact to fill different roles in the context of different actions and hence to couple them explicitly.

Further work is to be done in specifying a modeling language which is based on the meta-model presented in this paper. Modeling languages are needed to describe practices and socio technical systems in the field of computer-supported collaborative learning and computer-supported cooperative work.

Acknowledgments. The KP-Lab Integrated Project is sponsored under the 6th EU Framework Program for Research and Development. The authors are solely responsible for the content of this article. It does not represent the opinion of the KP-Lab consortium or the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

1. Allert, H.: Modeling Coherent Social Systems for Learning. Universität Hannover, Hannover (2005)
2. Allert, H., Richter, C.: Drawing on Practices for Modeling Socio-Technical Systems. Proceedings of the Computer Supported Learning Conference (CSCL'07). Kluwer Academic Publishers, Lancaster (2007) in press
3. Bardram, J. E.: Plans as Situated Action: an Activity Theory Approach to Workflow Systems. Proceedings of the 5th European Conference on Computer Supported Cooperative Work (ECSCW'97). Kluwer Academic Publishers, Lancaster (1997) 1732-1740
4. Bødker, S.: Activity Theory as a Challenge to Systems Design. In: H.E. Nissen, H. Klein, R. Hirschheim (eds.). Information Systems Research: Contemporary Approaches and Emergent Traditions. Elsevier, Amsterdam (1991) 551-564
5. Bühler, K.: Sprachtheorie - die Darstellungsform der Sprache. Gustav Fischer, Jena (1934)

6. Guarino, N.: Concepts, Attributes and Arbitrary Relations: Some linguistic and ontological criteria for structuring knowledge bases. Data & Knowledge Engineering, 8 (1992) 249-261
7. Guarino, N., Carrara, M., Giaretta, P.: An ontology of meta-level categories. In: D. J. E. Sandewall, P. Torasso (eds.): Principles of Knowledge Representation and reasoning: Proceedings of the fourth international conference. Morgan Kaufmann, San Mateo (1994) 270-280.
8. IMS Global Learning Consortium: IMS Learning Design Specification, Version 1.0 (2003), http://www.imsglobal. org/learningdesign/index.cfm.
9. Krieger, D.: Einführung in die allgemeine Systemtheorie. UTB, München (1998)
10. Luhmann, N.: Social Systems. Stanford University press, Stanford (1995)
11. Leontjew, A. N.: Activity, Consciousness, and Personality. Prentice Hall, Englewood Cliffs (1978)
12. Object Management Group: Business Process Modeling Notation (BPMN), Version 1.0 (2006), http://www.omg.org/cgi-bin/doc?dtc/2006-02-01
13. Object Management Group: Unified Modeling Language (UML), Version 2.0 (2005), http://www.omg.org/ technology/documents/formal/uml.htm
14. Scheunpflug, A.: Evolutionäre Didaktik: Unterricht aus system- und evolutions-theoretischer Perspektive. Beltz, Weinheim (2001)
15. Sowa, J.: Knowledge representation - logical, philosophical and computational foundations. Brooks/Cole, Pacific Grove (2000)
16. Stahl, G.: Meaning and Interpretation in Collaboration. In: B. Wasson, S. Ludvigsen, U. Hoppe (eds.): Designing for Change in Networked Environments. Kluwer, Dordrecht (2003) 523-532
17. Steimann, F.: Modellierung mit Rollen. Universität Hannover, Hannover (2000)
18. Steimann, F., Siberski, W., & Kühne, T.: Towards the Systematic use of Interfaces in JAVA Programming. Proceedings of the 2nd int. conf. on the principles and practices of programming in java PPPJ 2003, ACM, Kilkenny City (2003) 13-17
19. Vygotsky, L. S.: Mind in Society. Harvard University Press, Cambridge (1978)
20. Willke, H.: Systemtheorie. Fischer, Suttgart (1993)

Adaptive and context-aware scenarios for technology-enhanced learning system based on a didactical theory and a hierarchical task model

Jean-Louis Tetchueng[1], Serge Garlatti[1], Sylvain Laube[2]

[1] ENST Bretagne, GET CS 83818, 29238 Brest Cedex, France
{jl.tetchueng,serge.garlatti}@enst-bretagne.fr
[2] CREAD, IUFM Bretagne, 8, rue d'Avranches, 29200 Brest, France
sylvain.laube@bretagne.iufm.fr

Abstract. Among the main issues of future technology-enhanced learning systems, we can mention the following ones: the ability to reuse learning resources (learning objects, tools and services) from large repositories, to take into account the context and to allow dynamic adaptation to different learners based on substantial advances in pedagogical theories and knowledge models. In our framework, the goal of scenarios is to describe the learning and tutoring activities to acquire some knowledge domain (for instance physics) and know-how to solve a particular problem. The main contribution of this paper is an adaptive and context-aware model of scenario based on a didactical theory and closely related to a domain model, a learner model, a context model. These models are acquired from: i) the know-how and real practices of teachers in a problem-based learning approach in a particular framework: an institution IUFM, different categories of probationary teachers, a course about "the air as gas in its static and dynamic aspects: properties, theory and applications"; ii) the theory in didactic anthropology of knowledge of Chevallard [1]; iii) a hierarchical task model.

Keywords: Adaptation, context-aware, didactical theory, model of scenario, hierarchical task model, Task/Method paradigm.

1 Introduction

Among the main issues of future technology-enhanced learning systems, we can mention the following ones: the ability to reuse learning resources (learning objects, tools and services) from large repositories, to take into account the context and to allow dynamic adaptation to different learners based on substantial advances in pedagogical theories and knowledge models [2].We are interested in technology-enhanced learning systems using a problem-based learning approach, represented by scenarios. In our framework, the goal of scenarios is to describe the learning and tutoring activities to acquire some knowledge domain (for instance physics) and know-how to solve a particular problem. A scenario may depend on several dimensions which describes different learning situations (in some way): the learning domain (course topic), the learner (his know-how and knowledge levels), the tutor/teacher, the learning and tutoring activities (their typology, organization and coordination), the activity distribution among learners, teachers and computers, the learning "procedures" according to a particular school / institution / university and the didactical/pedagogical environment. In order to deal with the broadest range of learning situations, it is necessary to design adaptive learning systems which have the ability to take into account these dimensions. Nevertheless, research on the learning scenario models leads to the standardization of pedagogical approaches - for instance IMS LD [3]. These models require authors/teachers to produce generic and standard models which are neutral on a pedagogical and/or didactical point of view [4]. For instance, learner and tutor activities and adaptation cannot be sufficiently accommodated. It is not possible to specify the management of knowledge and know-how levels of the learners according to the knowledge domain and the context. In other words, these scenario models are unable to deal with the different dimensions previously introduced.

The main contribution of this paper is an adaptive and context-aware model of scenario based on a didactical theory and closely related to a domain model, a learner model, a context model. These models represent the different dimensions and are acquired from: i) the know-how and real practices of teachers in a problem-based learning approach in a particular framework: an institution IUFM[1], different categories of probationary teachers, a course about "the air as gas in its static and dynamic aspects: properties, theory and applications"; ii) the theory in didactic anthropology of knowledge of Chevallard [1]; iii) a hierarchical task model. A co-design methodology has been used to articulate teacher real practices, the Chevallard theory and the hierarchical task model to define the different models [5]. The hierarchical task model enables us to define the learning and tutoring activities, the activity distribution among learners, teachers and computers and also to transpose the main concepts of the Chevallard theory. The context model implements the didactical environment acquired from the Chevallard theory and the teacher real practices and know-how.

First of all, we briefly present the MODALES project in which our research takes place. Secondly, we present the main contributions of the didactic anthropology of knowledge of Chevallard theory in the acquisition teacher real practices and know-how. Thirdly, the computer-based model of scenario is presented and detailed. Finally, the conclusion highlights the main results of this study and point out the next research issues.

2. The MODALES Project

The MODALES project is aimed at designing an adaptive learning system for probationary teachers, based on real practices and teacher know-how. The course topic is about "the air as gas in its static and dynamic aspects: properties, theory and applications" for different categories of learners. They are probationary teachers:

[1] IUFM : Institut Universitaire de Formation des Maîtres

primary school teachers (called PE for "professeur des Ecoles" and secondary school teachers (called PLC for "professeur des Lycées et Collèges": earth/biology sciences and physics. The teachers are considered as experts in education. In MODALES, scenarios may change according to the following features: i) the category of learners having intra and inter category variability; ii) the available resources from different domains - physics, didactic and epistemology - which can be determined by teachers iii) distance or face-to-face activity according to learner needs, learning policy and didactical environment constraints iv) the sharing of activities between teachers, learners and computers according to learner needs and learning policies. The main issue is to design a generic scenario which can deal with the broadest range of learning situations (from a computer science viewpoint).

3. Acquisition of teacher practices and know-how

Firstly, several scenarios based on a common learning scenario P_o (whose variables are learners, the expert teacher and the available resources) were built [6].

Secondly, we use the theory in didactic anthropology of knowledge of Chevallard to go further [6]. The praxeology system ($T/\tau/\theta/\Theta$) of the Chevallard theory enables us to acquire the scenario model and the didactical environment. According to Chevallard, teacher and learner activities can be described in terms of types of tasks T_c achieved by techniques τ which may be recursively achieved by subtasks $T_{c'}$. Thus, a Task/Technique system (T/τ) has a hierarchical structure. This hierarchical structure (T/τ) defines a know-how that leans on an environment composed of a technology θ (discourse that justifies and explains techniques) and a theory Θ justifying and highlighting the technology. In other words, a Task/Technique system (T/τ) describes a type of problem (T) to solve and the technique (τ) describes how to solve it (T).

We can observe six different moments in the didactical organization [1]: i) the first encounter with the type of tasks T_c (M1); ii) the exploration of the type of tasks T_c and the construction of techniques τ (M2); iii) the technique work that improves the technique and makes it more efficient (M3); iv); the construction of a Technology/Theory related to technique τ (M4) v) the institutionalization of the system ($T/\tau/\theta/\Theta$) by the teacher (M5); vi) the evaluation (M6) (cf. Figure 1). For a given technique, a task can be decomposed into sub-tasks which are achieved according to specified operators. At present, three different operators are used: sequence, alternative and parallel.

Moreover, the scenario analysis shows different categories of learning and tutoring tasks, organized at different levels of the task hierarchy: scenario, phase, moment, learning task, routine task and tutoring task. A scenario is generally composed two phases: 1) Phase 1: construction of professional references for teaching (cf.figure1), 2) Phase 2: development of a training sequence implemented in classrooms.

The adaptation of scenarios leads to choose the relevant technique according to the learners and the didactical environment. According to the Task/Technique system, the choice can be done by the computer, the learner or the teacher. The selection of the relevant technique depends on the following properties: the Task/Technique system, the learner category (PE, PLC, type of PLC, etc.), the learner curriculum and the didactical environment. From the Chevallard theory and the teacher real practices and know-how, we define the didactical environment as follows: type of classrooms (virtual classroom, scientific laboratory with or without computers and/or with or without internet access, associated CITT[2] tools (chat, email, forum, etc.), technical instruments (thermometer, barometer, etc.), resources (documents, experiments, etc.) and face to face or at distance.

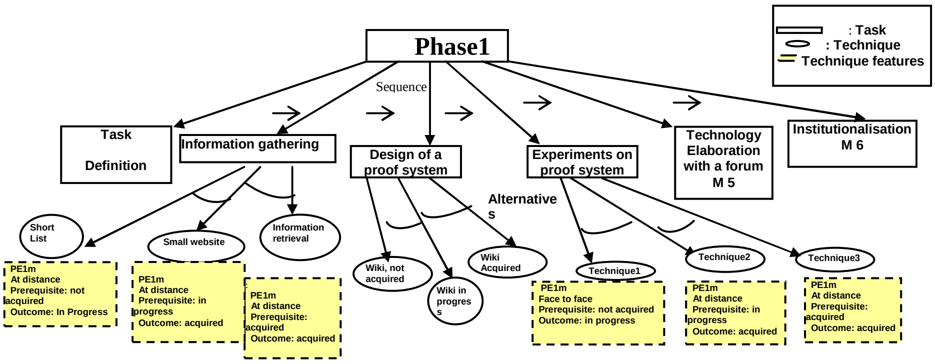

Figure 1. Description of the phase 1 for a PE learner.

First of all, we explain how the learner and the technique properties are used to choose the relevant technique in a given didactical environment. Secondly, we detail the different roles of the didactical environment features.

To illustrate the Chevallard's theory and its concepts, we choose a particular case study for a PE learner in which we detail the task "phase 1" composed of several sub-tasks. Some of them have alternative techniques. We assume the learner states for the concepts "P", "V" and "T" are "acquired" (otherwise more techniques must be added and consist of sub-tasks dedicated to the acquisition of the corresponding knowledge).

The course topic is about "the air as gas in its static and dynamic aspects: properties, theory and applications". In the Chevallard framework, the considered theory is thermodynamics. In physics, theories can be "evaluated" by means of different laws. In our case, it is the Boyle-Mariotte law which is represented as follows ($PV/T = K$) for PE Learners. The knowledge domain is composed of the thermodynamic theory, the corresponding laws, the related concepts (Pressure P, Volume V, and Temperature T) and their relationships. To deal with the learner knowledge and know-how levels, the knowledge domain entities (theories, laws, concepts and relationships) and the type of tasks may have three different states: "not acquired", "in progress", "acquired". For a given type of task, the state "not acquired", correspond to the moment M1 and the states "in progress" and "acquired" correspond respectively to the moment M2 and M3. After a successful evaluation task, a teacher or the computer can update the learner know-how and knowledge levels for some domain entities and for a task, for instance from "in progress" to "acquired" if the corresponding know-how is considered as acquired.

In Figure 1, several techniques are annotated with the knowledge and know-how levels: the prerequisite and outcome states of the learner. When it is the first encounter of the type of task "experiments on proof system", the corresponding

[2] Communication and Information Technologies for Teaching

learner state is “not acquired”. Thus, the relevant technique is “Technique 1”. After a successful evaluation sub-task, his outcome state will be “in progress” for the task. When the learner state for the type of task “experiments on proof system” is “in progress”, the relevant technique is “Technique 2”. After a successful evaluation sub-task, his outcome state will be “acquired” for the task. If the evaluation task fails, a remediation task is used (not described in figure 1). The type of task “experiments on proof system” can be worked several times a year in different modules about astronomy, thermodynamic, etc. in physics. Thus, the relevant technique may change according to the moment at which the type of task “experiments on proof system” is worked in a particular module. Thus, several alternatives are provided for a given type of task.

From the didactical environment, we firstly explain the role of the technical instruments. An historical and epistemological analysis of several historical and didactical situations shows that laws in physics are tested by means of technical instruments [7]; For instance, the technical instruments could be a thermometer and a barometer or a simulation tool. Thus, the learners must have or acquire know-how to use these technical instruments to solve the problem related to the task “phase 1”. Whether the learner state for these tasks “temperature and pressure measurements” are “not acquired” or “in progress”, the relevant technique must have the corresponding prerequisite states and must consist of sub-tasks dedicated to the acquisition of the corresponding know-how.

The “face to face” or “at distance” feature change the Task/Technique system and the activity distribution among learners, teachers and computers. It is the same for the type of classrooms and the CITT tools. Moreover, some specific know-how may be assumed (internet access and information gathering, forum, chat, etc.) to achieve communication tasks or information retrieval tasks. Thus, such know-how must be routine tasks or at least acquired. Otherwise, it is necessary to have sub-tasks to acquire such know-how.

In conclusion, we show that, it is necessary to describe the different techniques according to the learner and the didactical environment features to be able to choose the relevant technique.

4. Adaptive and context-aware model of scenarios

From the acquisition of teacher real practices by means of the Chevallard theory, the didactic-based scenario model is transposed into a computer-based hierarchical task model. Firstly, we describe and justify the transposition of the Task/Technique systems and their hierarchical structure. Secondly, we analyze the representation of the typology of learning and tutoring activities. Finally, we show how the adaptation is formalized according to parameters describing the learner, the context.

Teaching and learning activities of scenarios have been described in terms of type of tasks T_c and techniques τ. The type of tasks T_c describes the teaching and learning activities, while techniques τ describe a way of achieving the types of task T_c. We transpose the resulting Task/Technique system (T_c/τ) in the task/method paradigm of the hierarchical task model. Therefore, we can represent in these model, the Task/Technique system (T_c/τ) of Chevallard [1] fitted with its hierarchical structure and didactics properties describing scenarios while we preserve its initial properties and semantics.

Several research studies in AI[3] focus on the hierarchical task model using the tasks/method paradigm [8-12]. The mechanism of hierarchical and recursive decomposition of a problem into sub-problems is one of the basic characteristics of the hierarchical task model [8-12]. The hierarchical task model consists of abstract and atomic tasks and methods. In a particular task, a method represents the various ways of achieving this task. A method describes the decomposition of its task into sub-tasks. The execution of these sub-tasks is done through a control structure which is composing of the following operators: sequence, parallel, choice. Their respective specifications are quite the same as those of 'seq', 'par' and 'alt' presented in the paragraph 3. Thus, an abstract task can be broken down into abstract or atomic sub tasks through its associated methods. An atomic task is not composed of sub-tasks. It can be achieved by a simple procedure – for instance, an information retrieval process, a particular human computer interaction, etc. The task/method paradigm has respectively a semantic and a hierarchical structure similar to those of the Task/Technique systems (T_c/τ) of Chevallard. Moreover, we have to refine the task and method concepts of our model (specialization) to take into account adaptation and sharing of activities.

The typology of tasks of our computer-based model identifies the various types of tasks T_c which compose the scenarios described in paragraph 3: scenario, phase, moment, learning tasks, routine tasks, tutoring tasks.

These types of tasks are transposed in the computer-based model and are respectively named «ScenarioTasks», «PhaseTasks», «MomentTasks», «LearningTasks», «RoutineTasks», «RoutineTasks». One of the main criteria of the formalization of tasks is their atomic character or not - respectively abstract or not. The tasks «ScenarioTasks», «PhaseTasks», «MomentTasks», «RoutineTasks» are represented by abstract tasks since a scenario consists of two phases which are broken down into moments while each moment consists of learning tasks, routine tasks, and/or tutoring tasks. Tasks «LearningTasks» are also represented as abstract tasks, because they represent a Task/Technique system which can be broken down into others sub Task/Technique systems. Tasks «RoutineTasks» are only composed of atomic tasks. The tasks «TutoringTask» are atomic tasks. They correspond to tutoring activities of the teacher or of the computer system. In both cases, these tasks are seen as "simple procedures".

From the Chevallard theory viewpoint, the relevant technique must be selected according to the current learner and the didactical environment. From a computer-based viewpoint, the adaptation process can be viewed as the selection of the relevant method which represents the Chevallard concept of techniques. It aims at a dynamic selection of the relevant methods according to the context and the current learner. The know-how and knowledge levels of the learner are represented by an overlay model [13] associated to the learner model as described in the paragraph 3.

The context model represents the didactic environment as described in the paragraph 3. It is described by the type of classroom in which the learning activities will take place, the associated CITT tools and devices, a list of technical instruments which are a subset of those in the domain, "face-to-face" or "at distance". The domain

[3] Artificial Intelligence

model consists of the thermodynamic theory, the corresponding laws, the related concepts and their relationships. The learner is described by his curriculum, his category (PE, PLC, type of PLC, etc.) and his knowledge and know-how levels (an overlay model): a set of states (“not_acquired”, “in_progress”, “acquired”) for some domain entities and know-how (tasks). These states are assigned to the learner and are updated.

The context, learner and domain models will be represented by means of ontologies within SCARCE (SemantiC and Adaptive Retrieval and Composition Engine) environment [14]. The adaptation process in SCARCE consists of two stages: firstly, resources are evaluated and classified in one equivalence class according to class membership rules. In this paper, we only need two equivalence classes (“good” and “bad”); secondly, one adaptation technique is chosen for the current learner (annotation, hiding, sorting, direct guidance, etc.). All methods, belonging to the class “good”, are selected for the learner. The membership rules define necessary and sufficient conditions to belong to an equivalence class. Rules are declarative predicates using context, learner and method features (which are binary relationships).

Thus, let T_a be a task, Ci be a context, L be a learner, SL the current set of states describing the knowledge and know-how levels of L. The adaptation process is as follows:

1) If SL does not have a state for the task T_a, the corresponding state is added to SL with value: $SL.T_a$ = “not acquired” (the task T_a does not be worked).

2) Membership rules: all methods of T_a for which the context and the learner features match up to the corresponding method features (or “belong to” for multiple-valued features) belong to the class “good” and others belong to the class “bad”.

3) If the class “good” is empty, it is considered as a problematic situation and required a teacher action to remediate or to provide a new method and context adapted to the learner and the task T_a. Otherwise, all methods, belonging to the class “good”, can be provided to the learner.

5. Conclusion

The design of technology-enhanced learning systems must be considered as a transdisciplinary problem requiring the integration of different scientific approaches - from computer science, didactic, education, etc. It is also necessary to take into account real practices of teachers. We propose an adaptive and context-aware model of scenario based on a didactical theory and closely related to a domain model, a learner model, a context model. The properties of the model presented in this paper have been acquired by means of a co-design methodology in which the real practices of teachers, knowledge and know-how are acquired by means of the theory in didactic anthropology of knowledge. Nevertheless, the model is not finished. At present, we only manage one category of adaptation. In other word, we need to continue the co-design process in order to precise the other adaptation categories and to refine the different models.

Acknowledgments

The project MODALES receives funding from Brittany region as a PRIR project, and belongs to the ACI GUPTEN Project.

6. References

1. Chevallard, Y., *L'analyse des pratiques enseignantes en théorie anthropologique du didactique.* La Pensée sauvage, Grenoble, Recherches en didactique des mathématiques, 1999. **19**(2): p. 221-226.
2. Balacheff, N., *10 issues to think about the future of research on TEL.* Les Cahiers Leibniz, Kaleidoscope Research Report, 2006(147).
3. IMS, *IMS Learning Design Information Model, IMS Global Learning Consortium*. 2003.
4. Nodenot, T., *Etude du potentiel du langage IMS-LD pour scénariser des situations d'apprentissage : résultats et propositions*. 2006, in Pernin J-P. et Godinet H. (dir.) , actes électroniques du colloque Scénarios, p. 57-63.
5. Garlatti, S., et al. *The Co-Design of Scenarios for a Didactic-based E-learning System viewed as an Adaptive Virtual Document*. in *Satellite Workshop E-Learning, 2nd IEEE International Conference on Information & Communication Technologies : from Theory to Applications*. 2006. Damascus, Syria.
6. Laubé, S., S. Garlatti, and Al. *Scénarios intégrant les TICE : les méthodologies et les cadres théoriques à l'œuvre dans la recherche MODALES*. in *8ème Biennale de l'Education, Colloque "Scénariser l'enseignement et l'apprentissage : une nouvelle compétence pour le praticien ?" INRP - ERTé E-Praxis*. 2006. Paris.
7. Guedj M., L.S.e.S.P., *De l'analyse historique au profit de l'analyse des situations didactiques.* Numéro spécial des Cahiers du Centre F. Viète, ouvrage collectif, ReForEHST Ed. (à paraître), 2007.
8. Willamowski, J., F. Chevenet, and J.M. François, *A development shell for cooperative problem-solving environments.* Mathematics and computers in simulation, 1994. **36**(4-6): p. 361-379.
9. Ullrich, C., *Course generation based on HTN planning*, in *Proceedings of 13th annual Workshop of the SIG Adaptivity and User Modeling in Interactive Systems*, J. A. and B. B., Editors. 2005. p. 75-79.
10. Trichet, F. and P. Tchounikine, *DSTM: a Framework to Operationalize and Refine a Problem-Solving Method modeled in terms of Tasks and Methods.* International Journal of Expert Systems With Applications, Elsevier Science, 1999. **16**: p. 105-120.
11. Erol, K., D. Nau, and J. Hendler. *HTN Planning: Complexity and Expressivity*. in *AAAI*. 1994. Seatle.
12. Wielinga, B., et al., *The KADS Knowledge Modelling Approach*, in *Proceedings of the 2nd Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop*, R. Mizoguchi and H. Motoda, Editors. 1992: Hitachi, Advanced Research Laboratory, Hatoyama, Saitama, Japan. p. 23-42.
13. Kobsa, A. and W. Wahlster, *User Models in Dialog Systems*. 1989: Springer Verlag.
14. Garlatti, S., S. Iksal, and P. Tanguy, *SCARCE: an Adptive Hypermedia Environment Based on Virtual Documents and Semantic Web*, in *Adaptable and Adaptive Hypermedia Systems*, S.Y. Chen and G.D. Magoulas., Editors. 2004, Idea Group Inc. p. 206-224.

Authoring Collaborative Graphical Editors for Adaptive Context-based Learning Environments

Estefanía Martín[1], Néstor Carrasco[2], Rosa M. Carro[1]

[1] Escuela Politécnica Superior, Universidad Autónoma de Madrid
28049 Madrid, Spain
{estefania.martin, rosa.carro}@uam.es

[2] Escuela Superior de Ciencias Experimentales y Tecnología, Universidad Rey Juan Carlos
28933 Móstoles, Spain
cmnestor@gmail.com

Abstract. This paper presents an authoring tool that supports the specification of different collaborative activities and the configuration of collaborative graphical editors to be incorporated in dynamically generated collaborative workspaces. This tool allows teachers to reusing existing elements and multimedia material, and also saves them from learning the details about the underlying software. Collaborative workspaces are dynamically generated at run-time to support the accomplishment of collaborative activities by students. These activities can be incorporated into adaptive context-based learning environments, such as CoMoLE, which proposes each student the most suitable individual and collaborative activities according to her personal features, needs and context at each time, and supports their accomplishment.

Keywords: context-based learning, collaborative editors, authoring tools, CSCL

1 Motivation

The number of people that uses mobile devices to connect to the Internet from different places is exponentially increasing. This fact has motivated the development of mobile learning environments in which students can access to different learning resources and accomplish different activities though diverse devices [1][2][3]. User context (device, location, available time) can influence the accomplishment of learning activities. It would be inappropriate to propose complex collaborative activities to students when they have not a suitable device to accomplish them (i.e. if they are using their PDA or mobile phone), when they have not enough available time, or when their partner/s are not connected at the same time, in the case of synchronous collaborative ones. Context-based adaptation techniques support the

[1] This work has been partially supported by the Spanish Ministry of Science and Education, project number TIN2004-03140

selection of the most suitable activities according to the current user's context in mobile learning environments [4].

Regarding collaborative learning [5], during the last years many collaborative tools and systems have emerged, such as the one described in [6] for *Computer Programming Learning*. A selection of systems supporting collaborative learning interaction management is presented in [7]. Concerning adaptation in CSCL, some works have been done, such as [8] [9] or [10].

Designing and developing collaborative learning environments is not an easy task. Focusing on synchronous collaborative editors, it is a fact that, on the one hand, managing issues such as synchronism or concurrency is pretty hard. On the other hand, describing collaborative activities in terms of computer software, organising the work groups, as well as developing the multimedia materials to be used for task accomplishment (not only those used by the teachers to illustrate the activity, but those used by the students to represent the solution) also requires a lot of effort. For these reasons, we consider convenient to provide solutions that: i) are transparent to teachers regarding low-level collaborative issues, and ii) gives them the possibility of either creating or reusing already defined activities, text wordings, multimedia elements, icons to be used during the collaboration, etc., and including them in the graphical editor. In this way, they do not need to start from the scratch when specifying new collaborative workspaces, but can take advantage of elements stored. Furthermore, taking into account that maybe not all the activities are suitable of being accomplished by every user at each time (because of the user context), it would be more appropriate to generate different collaborative workspaces and to select the activities to be tackled according to their current context.

The work presented in this paper focuses on helping teachers to specify different collaborative activities to be performed through dynamically generated workspaces that include collaborative graphical editors. Our main goal is to support, on the one hand, design and specification of collaborative workspaces including graphical editors by non technical experts and, on the other hand, dynamic generation of these workspaces starting from the teachers' descriptions, taking into consideration the students connected at a certain time and their context.

This paper is structured as follows: the bases of our approach are presented in section 2. The authoring tool developed to support the specification of collaborative workspaces and the reuse of activities and multimedia materials is described in section 3. Dynamic generation of collaborative workspaces is explained in section 4. And, finally, conclusions and future work are presented in section 5.

2 The Approach

The main goal of the approach presented in this paper is to support: i) dynamic generation of collaborative workspaces in which activities are performed through collaborative graphical editors, and ii) the realization of the most suitable collaborative activities through the graphical editors. These editors are created once at design time and configured at runtime according to the activities to be supported. The purpose is to support the generation of editors with specific icons to build a solution for a certain problem (i.e., if the problem consists of building a logical circuit, the icons to be used to represent the solution to the problem will be logical gates with linkers to connect them). The collaborative editor toolbar is made up of elements, as well as links to mark the relationships between them. With the purpose of supporting the creation of this type of editors, an authoring tool has been developed. An application to support dynamic workspace generation has been implemented too.

The authoring tool helps teachers to specify the activities to be accomplished through collaborative graphical editors, as well as to describe the editor features. When creating a new activity, teachers must: i) provide the wording that describes the activity and ii) specify the features of the collaborative graphical editor to be offered to the students to support their collaborative work. Teachers can reuse, insert, modify or delete each of these elements, create workgroups and, in such a way, create different collaborative workspaces.

When collaborative activities have been described through the authoring tool, different collaborative workspaces are dynamically generated for each student at each time. The application developed generates each workspace starting from its description and also from the information about both the students that want to interact through it and the activities in which they are involved. The collaborative workspace presented to each student includes a list with the collaborative activities to be performed by her. When she selects one of them, the workspace is modified accordingly, in order to provide the corresponding collaborative graphical editor, which includes the activity wording, the icons to be inserted by the students in the shared work area to compose the solution and the shared work area itself. Next, the details of authoring tool and the dynamic generation of workspaces are explained.

3 Authoring Tool

The authoring tool is focused on helping teachers to define collaborative activities to be proposed to students when interacting with context-based adaptive environments. These activities, that will be selected according to the user's context at each time (device, available time, location), will be accomplished though collaborative graphical editors. The activities can be related among them, giving rise to sets of activities related to a certain subject.

A collaborative activity is defined by its attributes: i) name; ii) short description; iii) wording describing the activity, specifying the problem to be solved by students or the activity to be done; iv) icons to be used when performing the collaborative activity (images or pictures representing elements and linkers between them, to be included into the shared work area by the students to represent the solution of the problem); and v) deadline, indicating when the collaborative activity should be finished.

Teachers can take different actions though the authoring tool, such as adding, changing and removing elements (activities, problem wordings, icons, students and workgroups). When using this tool, they must choose between creating a new set of activities and managing an existing one. In the following subsections, the actions to be performed in order to create a new set of activities are explained in detail.

3.1 Defining Sets of Collaborative Activities

When a teacher wants to propose a new set of collaborative activities to the students, she must create a new *set*, in which she can define several *topics*. Each *topic* can contain one or more collaborative activities. In order to create a new collaborative activity, the teacher must select the corresponding *set* and *topic*, and provide both the wording of the problem to be solved and the components to be used for representing its solution in the configurable collaborative graphical editor (icons). When creating an activity, the teacher must give value to its attributes through the corresponding interface. The number of components to be used in each activity can vary. The teacher must either provide new icons/images or select them from those stored in the system (used in previous activities and uploaded by any teacher previously).

When a new activity has been created, the authoring tool gives the teacher the opportunity of managing and interacting with it by: i) modifying the activity wording and the icons/images associated with it, ii) deleting collaborative activities, and, more interesting, iii) simulating the collaborative workspaces that will be dynamically built.

In figure 1, a snapshot of the authoring tool is presented. In the left half of the interface it is shown how the teacher creates new collaborative activities. As it can be seen, three activities have been defined. The first one, *Logical Circuits with AND, OR and NOT gates*, (the *active* one at this time) has four icons associated. Three of them correspond to logical gates. The fourth represents a linker to be used to connect logical gates. The buttons situated at the bottom of the interface allow teachers to adding and deleting activities to the list. The buttons at the upper left-hand side of the page allow teachers to accessing to the main menu, as well as to adding new icons to the active activity.

3.2 Collaborative Workgroups

The next step of the design phase consists of specifying the workgroups involved in the different activities. For instance, let us suppose that a teacher creates three different collaborative activities (Actv1, Actv2 and Actv3) for three students (St1, St2 and St3). Activity Actv1 will be proposed to students St1 and St3. Activity Actv2 will be performed by students St1 and St2. And activity Actv3 will be accomplished by all the students. In this situation, student St1 will be involved in the three collaborative activities, while students St2 and St3 will be proposed with two different activities from the three activities belonging to the same set. The workspaces to be generated at runtime will contain the corresponding list of pending activities for each user.

It is also possible to associate diverse roles to students in each workgroup. In this application, two roles have been defined: team-leader and regular-member of the group. If a student has the team-leader role associated in a certain activity, he has the privilege to mark this activity as finished. A student can be team-leader in a certain group and regular-member in other workgroups. If homogeneous groups are desired, the team-leader role can be associated to all the group members. In the low right-hand side of figure 1, a snapshot of the workgroup creation interface is shown (over the main interface snapshot).

Although groups are currently formed by teachers, automatic-grouping mechanisms can also be incorporated, such as those developed previously [10], in which different grouping criteria can be used [11].

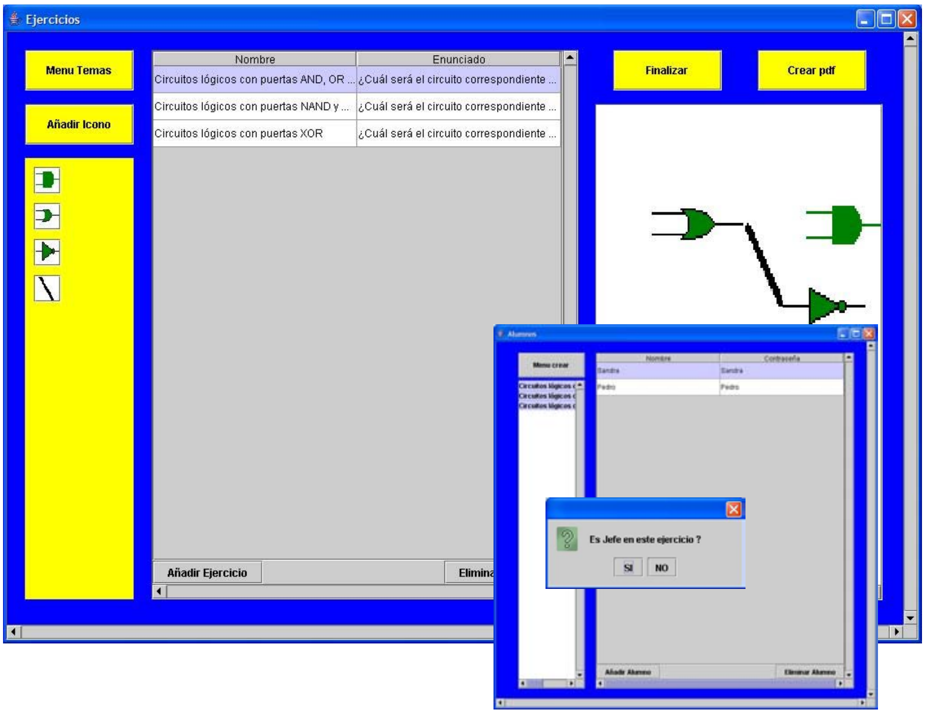

Fig. 1. Authoring tool: editing and monitoring

4 Dynamic Generation of Collaborative Workspaces

Once collaborative activities have been defined and workgroups have been established, students can start to accomplish the activities in which they are involved. When a student accesses to the application that supports the accomplishment of collaborative graphical activities, a workspace is dynamically generated. This workspace is composed by five areas (see figure 2), each of them containing:

- List of the collaborative activities to be performed by this student, presented in area one (see ① in figure 2).
- Statement or wording of the activity selected by the student, presented in area ②.
- Shared working area with the collaborative graphical editor, placed in the main area (③ in figure 2).
- Toolbar, including components (icons and images, representing elements and linkers) to be used for composing the solution of the collaborative activity. Other

icons, such as hand-icon, resize-icon and close-icon, are also included in this bar to allow the student to, respectively, moving, resizing or deleting elements and linkers in/from the collaborative workspace. For team leaders, the finish-icon is also included to allow them finishing the collaborative activity. All the components are part of the toolbar on the left side of the collaborative workspace (④ in figure 2).

– Finally, information messages sent by the application to connected users are presented at the bottom of the interface, in area ⑤.

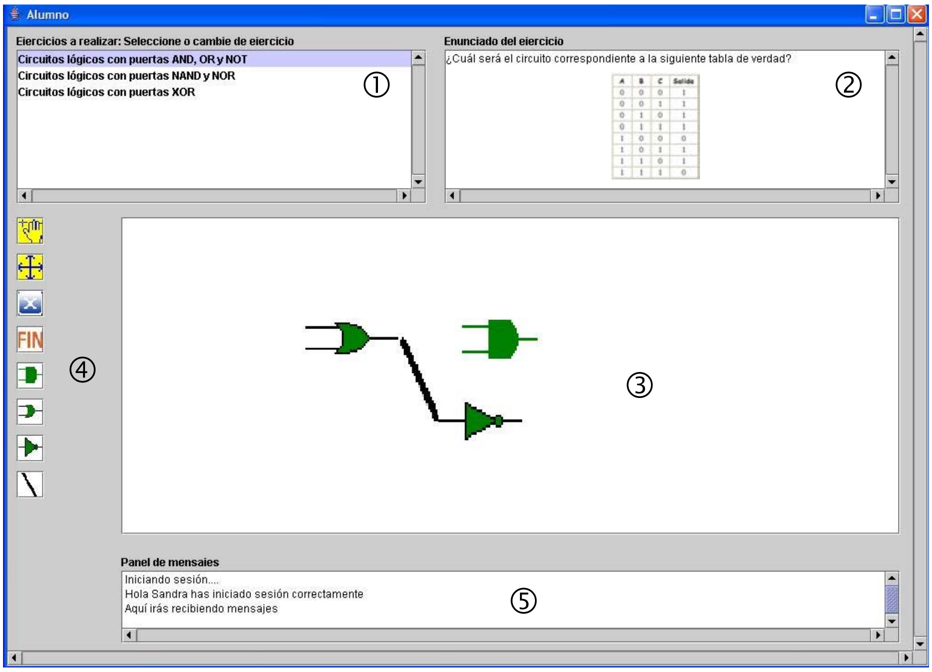

Fig. 2. Example of Collaborative Workspace Generated

Initially, the information presented in this page (problem wording and components to be used) is related to the first activity of the list of student pending activities. If the student wants to work in another activity, he must choose it from the list of activities presented in area ①. Then, the wording of the problem selected is presented in area ②, the icons related to this activity are presented in area ④, and the current state of the activity is loaded in the main area of the interface in the collaborative shared work area ③.

This application manages all the actions performed by students in different workspaces, and stores the state of the shared work area for each activity and workgroup each time a change is made. When a student moves from one activity to other, the application checks the current state of the second task and presents it to the student, including the work developed by all the members of the same workgroup until that precise moment.

When performing a collaborative activity, students can take the following actions: i) inserting a new element in the collaborative workspace, ii) selecting, moving or resizing one element from those already included in the shared work area, or iii) deleting one of the elements already included in the shared area.

The results of each action are immediately shown to all the group members. Students connected to the application receive feedback about both the actions performed by their partners and their connection status in the message area, situated at the bottom of the page (see area ⑤ in figure 2).

Team leaders can annotate an activity as finished by clicking on the finish-button. When taking this action, all the group members receive a message in area ⑤ and a popup window is opened to inform them that the task is finishing within five minutes, since a new and close deadline has been established by the team leader. When this period of time passes, the activity is marked as finished and no member of the group can modify the solution to this problem. Apart from team leaders, teachers can also decide the ending of activities. They can also take other actions such as monitoring the students' actions, looking at the content of the shared work area or generating a PDF file with the evolution and result of each activity (see right-hand side of the tool main interface in figure 1).

Figure 2 shows an example of a collaborative workspace generated for a student who plays the team-leader role in activity *Logic Circuits for AND, OR and NOT gates*. As it can be seen, the problem statement related to the first activity, is shown in area number ②. Icons representing logical gates are available in the tool bar, as well as icons to move, resize or delete components (area ④). As this student is the team-leader, the finish-button is also included. In the main area (number ③), the current state of the activity is shown. At the bottom of the page (area ⑤), the messages of the interaction with other members appear.

The implementation of the authoring tool and the application to support dynamic workspace generation is Java based. A specific package has been developed to control concurrency; it models the producer-consumer problem and implements the reader-writer problem when two students are interacting at the same time. Library iText has been used in order to support the generation of PDF files from the information stored about the solutions provided by the students. The content of these PDF files can be evaluated by teachers.

5 Conclusions

The work presented in this paper supports design and dynamic generation of collaborative workspaces including configurable graphical collaborative editors to support the realization of different collaborative activities in different contexts.

The authoring tool described in the paper helps teachers to create collaborative activities in an easy way by supporting both the configuration of graphical editors to be used for each activity and the reuse of multimedia material associated to previous collaborative activities (problem wordings and graphical elements), reducing the time teachers spend in this labor and also saving them from learning the details about the underlying software.

The activities designed through this authoring tool can be incorporated into CoMoLE [12], a context-based adaptive mobile learning environment able to recommend the most suitable individual and collaborative activities to be accomplished by each student at each time according to her preferences, needs and, particularly, context at that time (location, available time and devices). Regarding the adaptation of collaborative activities, it is possible to adapt the problem or activity proposed, as well as the collaborative workspace to work on, according not only to the student context but also to, i.e., her preferences or learning style.

References

1. Verdejo, M.F., Celorrio, C., Lorenzo, E., Sastre-Toral, T.: An Educational Networking Infrastructure Supporting Ubiquitous Learning for School Students. In: Kinshuk, Koper, R., Kommers, P., Kirschner, P., Sampson, D.G., Didderen, W. (eds.) 6th IEEE Int. Conference on Advanced Learning Technologies, pp. 174-178. IEEE Computer Society (2006)
2. Zimmermann, A., Specht, M. and Lorenz, A.: Personalization and Context Management. User Modeling and User-Adapted Interaction, 15, pp. 275-302. Springer (2005)
3. World Conference on mLearning, mLearn, http://www.mlearn2006.org/
4. Martín, E., Carro, R.M., Rodríguez, P.: A Mechanism to Support Context-based Adaptation in M-Learning. In: Nejdl, Wolfgang; Tochtermann, Klaus (eds.) Innovative Approaches for Learning and Knowledge Sharing. LNCS, vol. 4227, pp. 302-315. Springer (2006)
5. Dillenbourg, P.. Collaborative learning: cognitive and computational approaches. Elsevier, Oxford (1999)
6. Bravo, C., Marcelino, M.J., Gomes, A., Esteves, M., Mendes, A.J.: Integrating educational tools for collaborative Computer Programming learning. Journal of Universal Computer Science, 11 (9), pp. 1505-1517. Know Center, Springer Verlag (2005)
7. Soller, A., Martínez-Monés, A., Jermann, P., Muehlenbrock, M..: From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. International Journal of Artificial Intelligence in Education, 15 (4), pp. 261-290 (2005)
8. Furugori, N., Sato, H., Ogata, H., Ochi, Y., Yano, Y.: COALE: Collaborative and Adaptive Learning Environment. Proceedings of CSCL 2002, pp. 493-494 (2002)
9. Gaudioso, E., Boticario, J.G.: Supporting personalization in virtual communities in distance education. In: Jain, L.C., Howlett, R.J. (eds.) Virtual Environments for Teaching and Learning. World Scientific Publishing Company Pte Ltd., pp. 327-362 (2002)
10. Carro, R.M., Ortigosa, A., Martín, E., Schlichter, J.: Dynamic Generation of Adaptive Web-based Collaborative Courses. In: Decouchant, D. and Favela, J. (eds.) Groupware: Design, Implementation and Use, LNCS, vol. 2806, pp. 191-198. Springer-Verlag (2003)
11. Alfonseca, E., Carro, R.M., Martín, E., Ortigosa, A., Paredes, P.: The Impact of Learning Styles on Student Grouping for Collaborative Learning: A case study. User Modeling and User-Adapted Interaction, Vol. 16, 3-4, pp. 377-401. Special Issue: User Modeling to Support Groups, Communities and Collaboration. Springer Netherlands (2006)
12. Martín, E., Carro, R.M., Rodríguez, P.: CoMoLE: A Context-based Adaptive Hypermedia System for M-Learning. In: Procs. of VIII Simposio Nacional de Tecnologías de la Información y las Comunicaciones en la Educación at CEDI 2007. In press (Sep 2007)

Linking educational specifications and standards for dynamic modelling in ADAPTAPlan

Silvia Baldiris[1], Olga C. Santos[2], Carmen Barrera[2], Jesús G. Boticario[2,] Jeimy Velez[1,3], Ramón Fabregat[1]

[1] Insitute of Informatics and Aplications (IIiA), Universidad de Girona, Spain
{baldiris, jeimy}@eia.udg.es, ramon.fabregat@udg.es
http://bcds.udg.es/

[2] aDeNu Research Group, Artificial Intelligence Department, Computer Science School, UNED, C/Juan del Rosal, 16. 28040 Madrid, Spain
{ocsantos, jgb}@dia.uned.es, cbarrera@invi.uned.es
http://adenu.ia.uned.es/

[3] Universidad Pontificia Bolivariana, Córdoba, Colombia

Abstract. ADAPTAPlan project provides dynamic assistance to support the author in developing instructional design tasks which are included in learning design templates generated in terms of user modelling, planning and machine learning techniques, and making a pervasive use of educational specifications (IMS family) and standards (IEEE-LOM, ISO PNP). In this paper we describe how these standards and specifications are linked to support the dynamic modelling. Three types of user characteristics are considered in order to generate adaptation: i) Felder learning styles, ii) the knowledge level based on Bloom's Taxonomy and iii) collaborative competency levels. The modelling is performed in ADA+, a multi-agent architecture that applies collaborative filtering, machine learning and fuzzy logic techniques on the learners' interactions to support the development of personalised learning paths and to generate dynamic recommendations to be provided during the course execution.

Keywords: Learning objects, Metadata, Learning design, Competences, IMS Global Learning Consortium, Knowledge representation, Educational standards, Educational specifications, Felder learning styles, Bloom Taxonomy, Multi-agent systems, Machine learning, Fuzzy logic, JADE agents, Weka.

1 Introduction

ADAPTAPlan project provides dynamic assistance to authors to support the authoring of instructional design tasks in terms of learning design templates generated with user modelling, planning and machine learning techniques and making a pervasive use of educational specifications and standards. The purpose is to reduce the design effort, which is proven as a major bottleneck in adaptive standard-based learning management systems that support the full life cycle of eLearning [1]. Current educational specifications assume an ideal design scenario where all required elements can be managed at design time. Nevertheless, diverse issues makes unaffordable to design in advance all possible situations: a) learners' performance, b) synchronization and termporization issues, c) evolving learners' needs and preferences, d) adaptation process sustainable over time, e) pedagogical requirements affected by runtime adaptations, f) dynamic modelling.

To cope with these issues, ADAPTAPlan approach relies on a pervasive use of educational specifications and asks the author to add semantic on those elements that the author has traditionally defined (e.g. materials, learners, competences, objectives, …) and exempts him/her from describing alternative learning routes for different types of learners according to their features [2]. In turn, a planning engine takes as input the information provided by the author and the user model dynamically built from the learner's interactions to generate a personalized Unit of Learning (UoL) described in terms of IMS Learning Design specification [3]. ADA+ multi-agent architecture is used to build the user model and provide the dynamic support to learners. It applies collaborative filtering, machine learning and fuzzy logic techniques on the learners' interactions to support the development of personalised learning paths and to generate dynamic recommendations to be provided during the course execution. Details are given elsewhere [4].

2 Educational Specifications and Standards

Specifications describe in a precise, complete and verifiable way the requirements, design and behaviour of a system [5]. If they pass a validation process, they become standards.

To support design time adaptations and improve accessibility, reusability and maintenance in the ADAPTAPlan project we are using in an intensive way the specifications generated by the IMS Global Learning Consortium. In particular, IMS Learner Information Profile (IMS-LIP) [6], IMS Access For All (IMS-AccLIP) [7], IMS Question and Test Interoperability (IMS-QTI) [8], IMS Learning Design (IMS-LD) [9], IMS Reusable Definition of Competency or Educational Objective (IMS–RDCEO) [10], IMS Content Packaging (IMS-CP) [11] and IMS Metadata (IMS-MD) [12]. The later is superseded by IEEE LOM standard [13]. Furthermore, ISO standard on Individualized Adaptability and Accessibility in e-Learning, Education and Training (Personal Needs and Preferences), which is derived from IMS Access For All, will be considered when it is publicly available[1]. Each of them focuses on specific functions in the design and execution of the learning process in the context of a virtual learning environment.

IMS-LIP provides the general framework to define the general user characteristic, such as identification, goals, certification and licenses, acquired competencies, interests, etc. It can be linked to other specifications like IMS-RDCEO, which define the user competences.

IMS-AccLIP is an extension of IMS-LIP that considers the users preference regarding accessibility. IMS-AccLIP modifies the <*accessibility*> element in IMS-

[1] http://jtc1sc36.org/doc/36N1139.pdf, 36N1140.pdf and 36N1141.pdf at the same base URL, visited 6th May 2007 (expected to be publicly available at the end of 2007).

LIP, by removing the <*disability*> element and by addition of the <*AccessForAll*> element in this label. This new element considers information about how the materials are displayed, how the learner interacts with the system and the learner's preference about the content.

IMS-QTI uses the ASI model (Assessment-Section-Item) to define reusable evaluations. These evaluations and its parts can be interchange between different kinds of systems.

IMS-LD formalizes the design of a learning process in a Unit of Learning (UoL). The specification defines three levels of detail. Level A offers the necessary vocabulary to express a general learning process. It considers the definition of different user roles in the process (e.g. teacher and learner), the creation of activities composed by scenarios or environments and the utilization of learning objects in these environments. The second level, level B, adds the possibility of defining conditions based in properties about the individual user or roles. Finally, the level C allows the definition of a notification mechanism between roles.

IMS-LD can be linked from the <*environment*> element to IMS-QTI specifications. The evaluations are considered resources in IMS-LD. Moreover, the properties in IMS-LD can refer to attributes of the IMS-LIP or IMS-AccLIP specifications. Thus, it facilitates personalisation at course level or assessment level.

IMS RDCEO is a minimalist but extensible-based XML data model to define competencies or learning objectives. With this model it is possible to achieve a clear definition of competencies. It does not adjust to any particular curricular model and depending of the author different characteristic elements of the competency can be considered. Each UoL in a LD refers to objectives that can be associated to an IMS-RDCEO competence definition.

Additional to the above specifications, we are also using IEEE LOM standard / IMS-MD specification to characterize the learning objects and IMS-CP specification to generate or import packages with different kind of resources, such as courses and evaluations.

3. User characteristics for Adaptation

Three user characteristics are considered in ADAPTAPlan project [3] in order to generate adaptation: 1) Felder Learning Styles [14], the Knowledge Level based on Bloom's Taxonomy [15] and the Collaborative Competency Levels [16].

In [4] we introduced how we are managing the learning styles in the project. We have defined clusters for each of the 4 Felder's dimensions (Input, Processing, Understanding and Perception) in order to clearly separate the preference of different students. Table 1 shows how the clusters are assigned for the Perception dimension.

Table 1. Clusters for Felder's Learning Styles (Perception dimension)

CLUSTER	VALUES	STYLE DESCRIPTION
Balanced	1s, 3s, -3i, -1i	Sensitive / Intuitive
Moderated	5s, 7s, -7i, -5i	Sensitive / Intuitive
Strong	9s, 11s, -9i, -11i	Sensitive / Intuitive

The user's knowledge model is based in the Bloom Taxonomy [15]. It considers six levels of knowledge (Knowledge, Understanding, Application, Analysis, Synthesis and Evaluation). The student acquires these levels through the learning process by the study of the learning objects for the subjects of the course and the performance of the associated activities. The knowledge is the main element of a competency (although not the only one) since it influences the adequate performance of a person in a specific context. For this reason, we relate the student knowledge with a level of a specific competency.

Finally, we consider the six Collaborative Competency Levels defined in [16] (see table 2). We decide to separate this type of competency because it defines important aspects in the collaborative and cooperative behavior of the student. We are interested in modelling these user characteristics in order to establishing their relation with the success of the learning process.

Table 2. Level for the Collaborative Competency Table

LEVEL	OBJETIVE	DESCRIPTION
1	Participative_Learner	Interacts frequently in the course
2	Non_Colaborative_ Learner	Behaves as if there are no collaboration facilities.
3	Comunicative_Learner	Shares information with other learners using the available communication tools.
4	With_iniciative_Learner	Starts the proposed activities without waiting for other student's contributions.
5	Insightful_Learner	Makes contributions and comments on activities from other learners that later receive high scores.
6	Useful_Learner	Makes comments and contributions that are considered by other learners.

These competency levels have to be promoted for each student. Monitoring their achievement by the system can facilitate the generation of recommendations to encourage collaboration when needed.

Now that we have defined the learners' characteristics used for the adaptation, it is necessary to establish the relationship between these characteristics and the attributes in each of specifications mentioned above, which we are using to model the learning process (see table 3).

Table 3. User characteristics vs. IMS-LIP and IMS-RDCEO Specifications.

USERS CHARACTERISTICS	LIP – RDCEO SPECIFICATIONS ELEMENTS	POSSIBILITIES
FELDER LEARNING STYLES	accessibility.preference.typename.tyvalue accessibility.preference.prefcode	- Learner_Style_Processing - Learner_Style_Understanding - Learner_Style_Perception - Learner_Style_Entry
COLLABORATION COMPETENCY LEVELS	IMS – RDCEO Rdceo.identifier Rdceo.statement.statementname Rdceo.statement.statementtoken	- Participative_Learner - Non_Colaborative_ Learner - Comunicative_Learner - With_iniciative_Learner - Insightful_Learner - Useful_Learner
KNOWLEDGE COMPETENCY LEVEL	IMS – LIP Lip.competency.contentype.referential.indexid Lip.competency.exrefrecord	- Novice_Level (Bloom Knowledge and Comprension Levels) - Mean_Level (Bloom Application Analysis and Synthesis Levels) - Expert_Level (Bloom Evaluation Level)

The learning styles are linked to the <*preference*> element in IMS-LIP, which "it can be used to describe the physical environment required, the input/output technology required and also the learning styles that best suit the individual" [6]. For each learner there are four instances of this element, one by each dimension of Felder theory. The attribute *prefcode* stores the value of the dimension (balanced, moderate or strong). The learning styles are obtained from Felder Test [14].

The definition of the competencies is performed using the IMS-RDCEO specification. For each competency, an identifier is defined. In the <*statement*> element, specifically in statementtoken, the level of the competency is established. These values are dynamically generated through the analysis of learner interactions with the learning objects and activities and the evaluations results. Each competency is also referred in the <*competency*> element present in IMS-LIP.

In the table 4 examples of these definitions are presented.

Table 4. Examples of elements definitions

USERS CHARACTERISTICS	LIP – RDCEO SPECIFICATIONS ELEMENTS
FELDER LEARNING STYLES	accessibility.preference.typename.tyvalue= Learner_Style_Processing accessibility.preference.prefcode=visual.strong
COMPETENCIES	rdceo.statement.statementname = collaborative competency rdceo.statement.statementtoken = Participative_Learner

4. Adaptation generation in ADAPTAPlan

Adaptation in ADAPTAPlan is two fold. On the one hand, it consists in the generation of personalized learning routes in IMS-LD adjusted to the users' characteristics. On the other hand, dynamic recommendations to learners are provided during the course execution. In this paper we focus on the first one.

The personalized learning routes are generated by the planning engine [3]. The system should identify the adequate learning objects, collaborative tasks and evaluations in order to present them to a particular learner. For these reason, it is necessary to define the following set of properties in the IMS-LD:

- Four global and personal properties to model Felder's learning style for each learner. These properties are related to the IMS-LIP attributes defined in Table 4.
- Six local and personal properties to model the different knowledge levels. These properties are related to a specific knowledge body and to the level of competency.
- Six global and personal properties to model collaborative competency level.

The values of these properties constitute the input for the planner to generate a learning route adjusted to the user preferences and their characteristics. However, this process is only possible if there is an explicit relationship between the users characteristics and the different kinds of resources and activities associated to the learning design [17,18]. If the resources are characterized with metadata, rules can be applied to assign the resources to the activities in the UoL. In particular, IEEE LOM is used to characterize the learning objects. In Table 5, we present the relationship between the different Felder's dimensions for the learning style and the metadata attributes of the learning objects. This information facilitates the automatic generation of environments in the UoL selecting the appropriate learning objects for each particular learner. An appropriate learning object is one which addresses at least one characteristics of the specific user.

In the case of the knowledge level, each IMS-QTI evaluation is related to a specific concept of the knowledge body and to a specific knowledge level through its associated metadata. Each learning object addresses a specific level of knowledge, too. In this way, the evaluation process updates the knowledge properties in the UoL. Depending on the values of these properties, the learning objects are selected.

The collaborative competency levels are obtained by monitoring the learners' behaviour and their interactions in the system. This task is done by ADA+ multi-agent systems.

Table 5. Relating Users Characteristics with specifications attributes

LIP	LOM Attributes
– Learner_Style_Processing (Sequential - Global) – Learner_Style_Perception (Intuitive - Sensitive) – Learner_Style_Understanding (Active - Reflective)	Learning Resource Type – Exercise (Active,Intuitive,Verbal, Sequential) – Simulation (Active,Sensitive,Visual) – Questionerie (Active, Verbal, Sequential) – Diagram (Visual, Global, Intuitive) – Figure (Visual, Global,Sensitive) – Graph (Visual, Global,Sensitive) – Index (Global,Verbal) – Slide (Verbal, Sequential) – Table (Global, Sensitive) – Narrative text (Verbal,Reflective,Intuitive) – Exam (Active,) – Experiment (Active,Sensitive) – Problem statemen (Active, Sensitive,Verbal) – Self assessment (Active, Sequential) – Lecture (Verbal,Reflective,Intuitive)
– Learner_Style_Entry (Visual - Verbal)	Format (are free defined). It can be: – Text (Reflective, Intuitive,Verbals, Sequential) – Multimedia (Sensitive, Visual) – Graphics (Sensitive, Visual, Global) – Movies (Sensitive, Visual) – Sound (Sensitive, Verbal,Sequential)
– Learner_Style_Understanding (Active - Reflective)	Interactivity Type – Active (Simulation, questionnaires, exercises, problems) – Expositive (Hypertext, video, graphics and audio) – Mixed
– Learner_Style_Perception (Intuitive - Sensitive)	Density of Semantic – Very Low (Intuitive) – Low (Intuitive) – Medium (Sensitive) – High (Sensitive) – Very High (Sensitive)
– Level of Knowledge	Difficulty – Very Easy (Knowledge Level) – Easy (Comprenssion Level) – Medium (Application Level) – Difficult (Analysis and Synthesis Level) – Very difficult (Evaluation Level)

Some rules to define what learning objects are presented to each learner are described in Table 6.

Table 6. Rules to assign learning objects to learner's features

USER FEATURES	RULES
FELDER LEARNING STYLE	IF accessibility.preference.typename.tyvalue = Learner_Style_Entry AND accessibility.preference.prefcode = A THEN lom.format = Graphics, Multimedia,Movies IF accessibility.preference.typename.tyvalue = Learner_Style_Processing AND accessibility.preference.prefcode = A THEN lom.learning.resource.type = exercise,simulatons
KNOWLEDGE LEVEL	IF locpers-property.title=" Knowledg_Varieble" and locpers-property.value > 80 THEN Rdceo.statement.statementtoken.value = Expert_Level IF locpers-property.title=" Knowledg_Varieble" and locpers-property.value < 30 THEN Rdceo.statement.statementtoken.value = Novate_Level Now, IF Rdceo.statement.statementtoken.value = Expert_Level THEN lom.dificultty = difficult,very_difficult IF Rdceo.statement.statementtoken.value = Novate_Level THEN lom.dificultty = easy,very_easy

5. Conclusions

Having in mind a general approach to provide design time and run time adaptations in open and standard-based virtual learning environments [1] in this paper we focused on design issues. More specifically, our approach supports current educational specifications (IMS family) and has been integrated in dotLRN LMS through a web services interface. In this way, interoperability and extensibility is guarantee.

In this paper we defined the user characteristics required to generate adaptations according to learning styles, knowledge level and collaborative competences. Furthermore, we described the mechanism to link together those features with learning objects and resources to be integrated in the final learning design specification.

Our approach supports different educational specifications and standards in order to generate different kinds of adaptations and is intended to lessen the workload of the authoring process directing authors' attention to those elements they are used to manage and control in learning scenarios, like the specification of learning activities, temporal restrictions, evaluations, and not so much on a thorough description of alternative learning routes for different types of learners according to their features, which in any case are strongly dependent on learners' interactions and their evolution over time.

To date we have been exploring the application of this approach to several courses. First, a course on How to teach through the Internet taught in the on-going education program at UNED from year 2000. Second, an Object Oriented Programming Course (OOPC) developed in the Shaboo Project [19]. Our initial experiences has shown that course authors are much more predisposed to provide this set of information via a web-based interface rather than defining the whole IMS-LD design.

Acknowledgments. Authors would like to thank the Spanish Science and Education Ministry for the financial support of ADAPTAPlan project. Thanks to the Programme Alban, the European Union Programme of High Level Scholarships for Latin America, scholarship No. E06D103680CO”. Also, thanks to Fundación Carolina and Universidad Pontificia Bolivariana Montería for their support.

References

1. Boticario, J.G., Santos, O.C.: An open IMS-based user modelling approach for developing adaptive learning management systems. In J. of Interactive Media in Education (in press)
2. Boticario, J.G., Santos, O.C.: A Dynamic assistance approach to support the development and modelling of adaptive learning scenarios based on educational standards. Fifth International Workshop on Authoring of Adaptive and Adaptable Hypermedia. International Conference on User Modelling 2007 (2007).
3. Santos, O.C.,Boticario, J.G. Supporting Learning Design via dynamic generation of learning routes in ADAPTAPlan. 13th Int. Conf. on Artificial Intelligence in Education. (in press).
4. Santos, O.C., Baldiris, S., Velez, J., Boticario, J.G., Fabregat, R. Dynamic Support in ADAPTAPlan: ADA+. Proceedings of CAEPIA (in press)
5. Beshears. F.M. Open Standards and Open Source Development Strategies for e-Learning. Presentation for IS224 Strategic Computing and Communications Technology. Berkeley: Educational Technology Services. 2003.
6. IMS Learner Information Package. Version 1.0 Final Specification 2001.
7. IMS Learner Information Package Accessibility for LIP. Version 1.0 Final Specification 2003.
8. IMS Question and Test Interoperability. Version 1.2.1 Final Specification, 2003.
9. IMS Learning Design. Version 1.0 Final Specification, 2003.
10. IMS Reusable Definition of Competency. Versin1.0. Final Specification. 2002.
11. IMS Content Packaging Specification. v1.1.4 final specification. 2004.
12. IMS Metadata 1.2.1. Final Specification. 2001.
13. Learning Technology Standards Committee. Draft Standard for Learning Object Metadata Final version 1.2. 2002.
14. Felder R. M., Silverman L. K., ‘Learning and Teaching Styles In Engineering Education’, Engr. Education, 78(7), 674–681 (1988) – Preface: Felder R. M., June 2002.
15. Bloom, B.S. Taxonomy of Educational Objectives. New York: David Mckay, 1956.
16. Santos, O.C., Boticario, J.G. Supporting a collaborative task in a web-based learning environment with Artificial Intelligence and User Modelling techniques. Proceedings of the VI International Simposium on Eductive Informatics (SIIE’04), 2004.
17. Karagiannidis C. and Sampson D. Adaptation rules relating learning styles research and learning object meta-datas. Workshop on Individual Differences in Adaptive Hypermedia. 3rd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH2004), Eidhoven, Netherlands. 2004.
18. Peña, Clara I. PhD Thesis: Intelligent agents to improve adaptivity in a web-based learning environment. Universidad de Girona, 2004.
19. Moreno, German D. Baldiris, Silvia M. Degree project memories: Sistema Hipermedia Adaptativo para la Enseñanza de la Programación Orientada a Objetos. Universidad Industrial de Santander, 2003.

RECENT RESEARCH REPORTS

#116 Marco Baroni, Alessandro Lenci, and Magnus Sahlgren, editors. *Proceedings of the 2007 Workshop on Contextual Information in Semantic Space Models: Beyond Words and Documents*, Roskilde, Denmark, August 2007.

#115 Paolo Bouquet, Jérôme Euzenat, Chiara Ghidini, Deborah L. McGuinness, Valeria de Paiva, Luciano Serafini, Pavel Shvaiko, and Holger Wache, editors. *Proceedings of the 2007 workshop on Contexts and Ontologies Representation and Reasoning (C&O:RR-2007)*, Roskilde, Denmark, August 2007.

#114 Bich-Liên Doan, Joemon Jose, and Massimo Melucci, editors. *Proceedings of the 2nd International Workshop on Context-Based Information Retrieval*, Roskilde, Denmark, August 2007.

#113 Henning Christiansen and Jørgen Villadsen, editors. *Proceedings of the 4th International Workshop on Constraints and Language Processing (CSLP 2007)*, Roskilde, Denmark, August 2007.

#112 Anders Kofod-Petersen, Jörg Cassens, David B. Leake, and Stefan Schulz, editors. *Proceedings of the 4th International Workshop on Modeling and Reasoning in Context (MRC 2007) with Special Session on the Role of Contextualization in Human Tasks (CHUT)*, Roskilde, Denmark, August 2007.

#111 Ioannis Hatzilygeroudis, Alvaro Ortigosa, and Maria D. Rodriguez-Moreno, editors. *Proceedings of the 2007 workshop on REpresentation models and Techniques for Improving e-Learning: Bringing Context into the Web-based Education (ReTIeL'07)*, Roskilde, Denmark, August 2007.

#110 Markus Rohde. *Integrated Organization and Technology Development (OTD) and the Impact of Socio-Cultural Concepts — A CSCW Perspective*. PhD thesis, Roskilde University, Roskilde, Denmark, 2007.

#109 Keld Helsgaun. An effective implementation of *k*-opt moves for the Lin-Kernighan TSP heuristic. 2006, Roskilde University, Roskilde, Denmark.

#108 Pernille Bjørn. *Virtual Project Teams — Distant Collaborative Practice and Groupware Adaptation*. PhD thesis, Roskilde University, Roskilde, Denmark, 2006.

#107 Henrik Bulskov Styltsvig. *Ontology-based Information Retrieval*. PhD thesis, Roskilde University, Roskilde, Denmark, 2006.

#106 Rasmus Knappe. *Measures of Semantic Similarity and Relatedness for Use in Ontology-based information Retrieval*. PhD thesis, Roskilde University, Roskilde, Denmark, 2006.

#105 Davide Martinenghi. *Advanced Techniques for Efficient Data Integrity Checking*. PhD thesis, Roskilde University, Roskilde, Denmark, 2005.