

Karampiperis P. and Sampson D. (2005). Designing Learning Services for Open Learning Systems utilizing IMS Learning Design. *In Proc. of the 4th IASTED International Conference on web-based Education (WBE 2005)*, Special Session on Designing Learning Activities: From content-based to context-based Learning Services, Grindelwald, Switzerland.

DESIGNING LEARNING SERVICES FOR OPEN LEARNING SYSTEMS UTILIZING IMS LEARNING DESIGN

Pythagoras Karampiperis and Demetrios Sampson

Department of Technology Education and Digital Systems, University of Piraeus
150, Androutsou Street, Piraeus, GR-18534 Greece
and
Informatics and Telematics Institute, Centre for Research and Technology Hellas,
42, Arkadias Street, Athens, GR-15234, Greece
e-mail: {pythk,sampson}@iti.gr

ABSTRACT

The need for e-learning systems that support a diverse set of pedagogical requirements has been identified as an important issue in web-based education. Until now, significant R&D effort has been devoted aiming towards web-based educational systems tailored to specific pedagogical approaches. The most advanced of them are based on the ADL Sharable Content Object Reference Model in order to enable sharing and reusability of the learning content. However, sharing of learning activities among different web-based educational systems still remains an open issue. In this paper we first examine the limitations of the current state-of-the art authoring tools and investigate the use of the Learning Design framework as a mean to address those limitations. Then, we propose a high-level architecture of a learning activities generator and the supporting SCORM compatible authoring system that utilizes the Learning Design principles to provide the means for designing web-based pedagogical scenarios. Finally, we introduce the ASK-LDT, an authoring system developed based on the proposed architecture.

KEY WORDS

Learning Activities, Learning Design, Authoring, Web-based Educational Systems.

1. Introduction

During the last years, several tools have been developed in order to support the process of web-based authoring and several learning systems have been implemented tailored to specific pedagogical approaches. Several systems already exist supporting the process of web-based authoring for providing active learning, constructive learning, collaborative learning, intentional learning,

conversational learning, contextualized learning, reflective learning etc. [1, 2 and 3].

Currently, there are several educational e-content repositories and networked infrastructures available ranging from federated or distributed learning repositories to brokerage platforms [4, 5]. Nevertheless, the level of learning content reusability remains relatively low, due to the fact that sharing of learning activities across systems has not been addressed yet. This limitation prevents systems from reusing the same learning scenarios, leading to significant extra pedagogical effort for reusing learning content in different contexts. On the other hand, existing Learning Management Systems (LMS) provide tools for web-based authoring that are tailored to the capabilities of the specific system in hand. As a result, reusing and repurposing of learning activities and content is not supported in a consistent manner [6].

In this paper we first examine the limitations of the current state-of-the art authoring tools and investigate how the Learning Design framework can be incorporated in the architecture of a learning activities generator. To this end, we define an abstract high-level SCORM compatible authoring system architecture that utilizes the Learning Design principles to provide the means for designing pedagogical scenarios that can be reused in different contexts and across different web-based educational platforms.

The paper is structured as follows: In section 2, we present the current state-of-the-art authoring tools, investigating the limitations of those systems. Section 3 presents the principles of the Learning Design framework, emphasizing in modeling learning activities. In Section 4, we propose an architectural definition of a SCORM compatible authoring system that addresses the identified limitations incorporating the Learning Design framework. Finally, we present the ASK-LDT, an authoring system

for learning scenarios developed to implement the proposed architecture.

2. Current State-of-the-Art Authoring Tools

Nowadays, there are several vendors that provide platforms (such as Blackboard, WebCT, Lotus Learning Space, learn eXact etc.) being standard conformant. Those platforms provide authoring tools that are based on the ADL Sharable Content Object Reference Model (SCORM). The SCORM reference model provides a reference interaction model between learners and learning content, and describes within a common technical framework for computer and web-based learning the creation process of reusable learning content as "instructional objects", referred to as *sharable content objects* (SCOs).

The main limitation of the SCORM-based courseware authoring tools is that they are based on a "single learner model". This model assumes that a learner interacts only with content objects and that the learning activities are content-based activities engaging the learner in the learning process. Thus, the support provided by SCORM-based courseware authoring tools in the authoring process is limited in supporting the creation and sequencing of single learner, content-based learning activities. To this end, such authoring tools exclude the design of activities based on state-of-the-art pedagogical approaches such as constructive learning, collaborative learning etc. Moreover, since interactions between individual learners and/or between a learner and a tutor are abstracted in the SCORM reference model with no reference on how a learning management system could support those interactions, SCORM-based authoring tools limit the interoperability between systems to only content interoperability. These tools do not allow the description of such interactions leading to possibly different interpretation of learning activities between different systems.

On the other hand, a wide variety of non-SCORM conformant systems exist providing specific pedagogical approaches including active learning, constructive learning, collaborative learning, intentional learning, conversational learning, contextualized learning, reflective learning etc. (such as ActiveMath [7], MetaLinks [8], NetCoach [9], DCG [10], Interbook [11] etc.).

The main drawback of those systems is that they are closed, self-contained systems that cannot be used as service components (*lack of reuse support*). Additionally, due to their close architecture they cannot support all the required functionalities in a learning process since they cannot use external services (*lack of integration*). On the other hand, even if an open and scalable environment has been implemented, the supported content and learning

scenarios are a-priori designed to serve and support a specific pedagogical approach. As a result they are non-flexible in supporting different pedagogical approaches and they require extensive redesign effort in order to be used in different domains.

3. Modeling Learning Activities

For the purpose of our work, a *learning activity* can be formally defined as a triple containing the content that is delivered by an educational system, the actors participating in the learning activity (such as the learner or a group of learners, the tutor etc.) and their corresponding interactions. These interactions include three types, namely, interactions with the learning content, interactions with the educational environment and interactions between the participating actors.

Reusing learning activities across different web-based educational platforms requires that all components of a learning activity can be modeled in a commonly understandable form [12, 13] and that those platforms include the structural components required for the support of learning activities [14]. A first step is to agree on common ways for representing learning scenarios and describing the interactions between participating roles (learners, tutors, etc) and educational systems' services. The ultimate goal is to structure the learning scenarios in such a way that separates them from the learning resources. Thus, learning resources can be reused within different scenarios and scenarios can be also reused when populated with different resources. Moreover, repurposing of learning activities in a consistent manner requires authoring tools that are capable of handling the machine understandable representation form of learning activities [15].

To this end, standardization efforts on learning technologies have led to the IMS Learning Design specification [16] which provides a standard notation language for the description of learning scenarios. This specification promises the capability of describing a wide variety of learning activities based on different pedagogies. IMS Learning Design has the ability to define the role of different actors in the learning process, enabling the definition of both teacher-led and student-led scenarios. This specification builds upon the IMS Content Packaging specification, thus can integrate single learner (SCORM-compatible) activities in more complex activity-based learning scenarios.

3.1 Learning Design Framework

The core concept of the Learning Design framework is that regardless of the pedagogical strategy, learners attain learning objectives by performing a specific order of learning activities. *Multiple roles* can participate in *learning* and/or *support activities* of the learning process.

This formalization has the potential to describe a wide variety of learning activities based on different pedagogies.

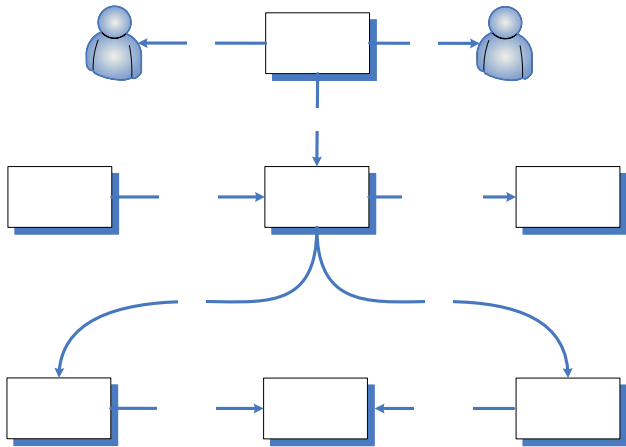


Figure 1: Learning Design Framework

The core components of the Learning Design Framework are shown in figure 1 and summarized below:

- **Role** component specifies the participating roles in learning activity. There are two basic Role types: the *Learner* and the *Support*. These roles can be sub-typed to allow learners to play different roles in certain types of learning activity such as task-based, role-play and simulations. Similarly, support staff can be sub-typed and given more specialized roles, such as Tutor, Teaching Assistant, Mentor, etc. Thus, Roles set the basis for multi-user models of learning. The name of a certain role is given depends on the underline pedagogy and the setting in use. In some instances a learner is called a 'student', whereas in others a 'participant'. The names of staff roles can be even more diverse, e.g., teacher, trainer, tutor, facilitator, mentor, assessor.
- **Activities** are one of the core structural elements of the 'learning workflow' model for Learning Design. They form the link between the roles and the setting in the learning environment. They describe the activities that a certain role can undertake in a specified environment. They also specify their termination conditions and the actions to be taken upon termination. There are two basic types of activities: *Learning Activities* and *Support Activities*. A Learning Activity is directed at attaining a learning objective per individual actor. A support activity is meant to facilitate a role performing one or more learning activities.

The Learning Design framework is implemented in the IMS LD specification at three Levels. Learning Design Level A includes the following elements: a series of activities (assessment, discussion, simulation), performed by one or more actors (learners, teachers etc.) - roles, in an environment consisting of learning objects or services. Level B adds properties (storing information about a

person or group), and conditions (placing constraints upon flow). Level C adds notifications (triggered events - e.g., if a student asks a question, the teacher needs to be notified that a response is needed).

4. The Proposed Authoring Architecture

4.1 Automatic Course Sequencing

In automatic course sequencing, the main idea is to generate a course suited to the needs of the learners. As described in the literature, two main approaches for automatic course sequencing have been identified [17]: *Adaptive Courseware Generation* and *Dynamic Courseware Generation*

In *Adaptive Courseware Generation* the goal is to generate an individualized course taking into account specific learning goals, as well as, the initial level of the student's knowledge. The entire course is adaptively generated before presenting it to the learner, instead of generating a course incrementally, as in a traditional sequencing context. In *Dynamic Courseware Generation* on the other hand, the system observes the student progress during his interaction with the course and dynamically adapts the course according to the specific student needs and requirements. If the student's performance does not meet the expectations, the course is dynamically re-planned. The benefit of this approach is that it applies as much adaptivity to an individual student as possible.

Both the above mentioned techniques are based on a three stage authoring process that is presented in the next section.

4.2 Authoring Process

The authoring process (presented in figure 2) in automatic course sequencing includes three main processes, namely:

- **Concept Selection.** During this process a learning goal is defined over a concept hierarchy graph (e.g. an ontology representing a curriculum). The learning goal corresponds to the desirable by the learner knowledge, or the specified learning goal, related concepts are selected based on the structure of the domain knowledge.

Learning
Activity

Contain

Resource

Is-a

Perform

Contain

Activity

Learner

Is-a

Role

Activity

Structure

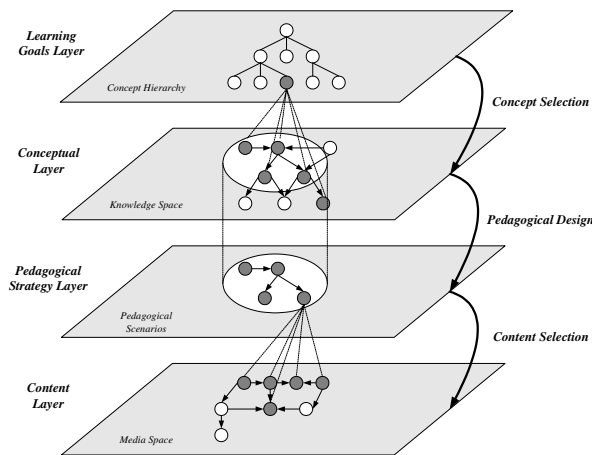


Figure 2: Abstract Authoring Process

- *Pedagogical Design.* This process is based on the use of template learning scenarios allowing the definition of generic learning activities. These templates define the participating roles in an activity sequence, as well as, the interactions between them during the learning process. They contain descriptions of the learning process independently from the learning content that will be used to support this process. The main advantage of using such templates is that they can be processed by a content selection mechanism in order to provide learning services based on a diverse set of pedagogical strategies.
- *Content Selection.* After the selection of the pedagogical strategy, the *content selection process* is applied based on learner characteristics such as accessibility and competency characteristics or even historical information about related learning activities, included in the User Model. During this process learning resources that are related to the previously selected concepts (*concept selection process*) are selected. The output of this process is a learning service following the selected pedagogical strategy that covers the selected concepts, thus, reaching the learning goal.

Figure 3 presents the architecture of a generation engine for learning services following the authoring process described above.

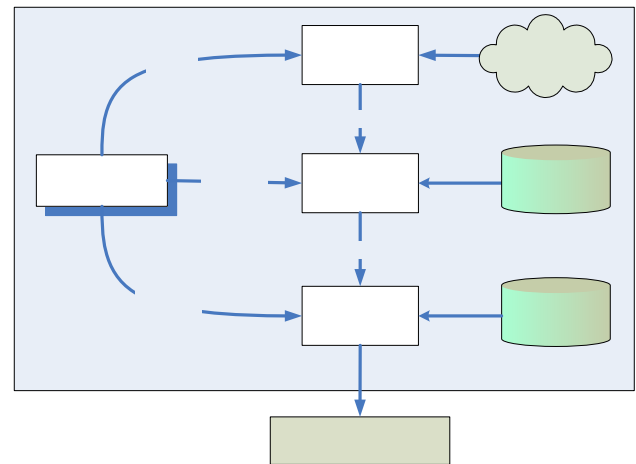


Figure 3: Learning Services Generation Engine

4.3 Authoring System Architecture

In this section, we present the high-level architecture for authoring of learning activities that combines the components of a SCORM compatible authoring system with the Learning Design framework. The key design principle in this architecture is the separation of the learning design process from the content packaging process. This separation enables the design of learning scenarios by defining the participating actors, the response of a learning system to their interaction with the learning content and the services provided by the learning system in such a way that is independent from the learning content. Thus, it enables the same learning scenario to be used with different content, as well as, different learning scenarios to use the same content objects.

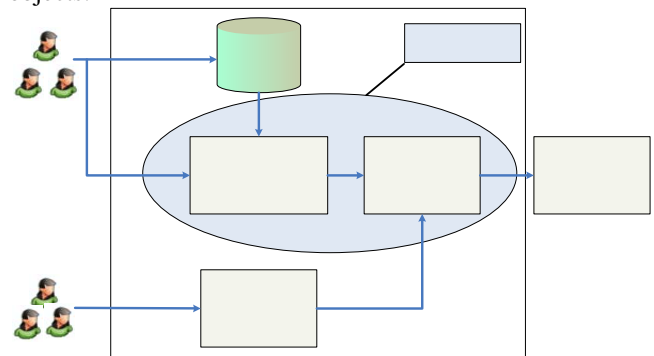


Figure 4: Authoring System Architecture incorporating Learning Design

Figure 4 presents the proposed authoring architecture based on the Learning Design framework described in the previous section of the paper. This figure shows the structural components of the authoring system and their interconnection paths. Interconnection between components is modeled by associations (directed arrows). These associations represent direct connections or they abstract away details of more complex connection and communication patterns (e.g. indirect communication

based on events).

The main components of the proposed architecture are the following:

- *Learning Design subsystem.* This part of the authoring system is based on the use of IMS Learning Design specification in order to provide the pedagogical designer with the environment for defining learning scenarios. The main scope is to enable the definition of generic, domain independent learning scenarios that can be used by the content packaging system in order to create learning activities based on the use of the learning objects stored in the content repository.
- *Content Packaging subsystem.* This part of the authoring system enables the population and packaging of learning scenarios with the learning content. In our system implementation, the development of such packaging tool is based on the commonly used IMS Content Packaging v1.1.3 specification and the supported metadata for indexing the content components of learning activity is based on the IEEE Learning Object Metadata standard.
- *Learning Resources Metadata Authoring & Management subsystem.* This part of the authoring system supports the metadata authoring and repository management. The main goal of this component is to provide an easy-to-use and accessible from anywhere platform capable of authoring, storing, managing and delivering the educational metadata produced.

5. The Proposed Learning Designer Toolkit (ASK-LDT)

The ASK Learning Designer Toolkit (ASK-LDT) introduced in this paper, is a tool supporting the proposed architectural approach for learning activities authoring (see figure 5).

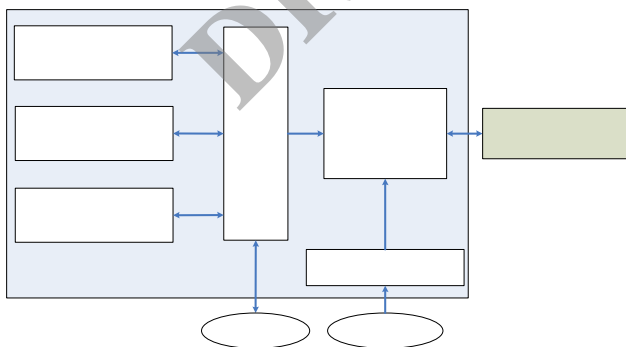


Figure 5: Internal Architecture of ASK-LDT

The ASK-LDT is based on the use of IMS Learning Design specification in order to provide to a pedagogical designer the environment for defining complex learning scenarios. The produced learning scenarios conform to the IMS Learning Design v1.0 Level B specification. The ASK-LDT, also, supports metadata for learning resources

that conform to the IEEE Learning Object Metadata 1484.12.1-2002 standard.

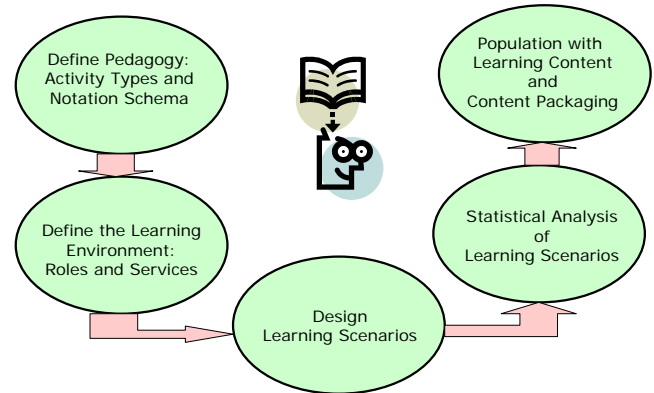


Figure 6: ASK-LDT authoring process

Based on the Learning Design framework principles, the authoring process that the ASK-LDT supports consists of the following steps (figure 6):

- *Definition of Pedagogical Elements.* At this step the ASK-LDT supports the pedagogical designer in defining the activity types he/she wants to support in a learning scenario, as well as, in defining a notation schema for each activity type specified. During this step the designer has the ability to characterize each activity type as “learning” or “support” activity.
- *Definition of the Environment.* At this step a designer defines the participating roles in the desired learning scenario, as well as, the environments in which the activities are taking place. An environment can be a virtual environment such as a virtual laboratory, an on-line chat, a discussion forum etc., or a software tool exposed as a service such as an annotation tool, a search engine etc.
- *Learning Scenario Design.* During this step the designer specifies the activity sequence of a learning scenario using a graphical user interface. For each activity the designer defines the participating roles, the environment in which the specific activity is taken place, as well as, the method by which this activity will be completed and/or terminated (user choice or time limit).
- *Statistical Analysis.* At this step the ASK-LDT provides statistics of the use of each activity type and environment in the learning design specified, in order to visualize the designer’s decisions.
- *Content Packaging.* This is the final step in the authoring process, in which the content components required to support the designed activities are specified. The output of this step is content packages conforming to the IMS Content Packaging v1.1.3 specification.

The core design concept of the ASK-LDT is to provide a graphical user interface for the design and sequencing of learning activities, which, on one hand uses a standard low-level notation language for the description of learning scenarios (so as to be able to inter-exchange learning

activities between different systems), and on the other hand enables pedagogical designers to use their own design notation (high-level notation) for the definition of learning scenarios (see Figure 7).

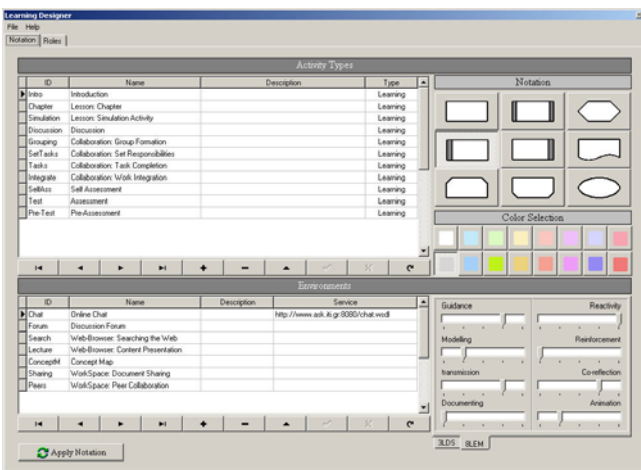


Figure 7: Supporting high level notation with the ASK-LDT

The ASK-LDT supports the learning design process by allowing the design of sequences of learning activities, as well as, the definition of the participating roles and the corresponding environments for each part of these activities. It provides several advanced features (figure 8) including the capability to define participating roles based on specific attributes of a user model and offers advanced control of learning activities sequencing based on the definition of properties and conditions upon learning flow.

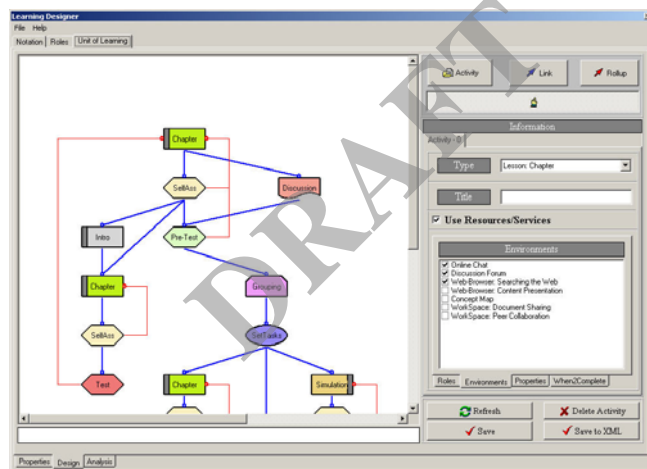


Figure 8: Supporting learning scenarios design and role definition through a graphical design interface.

6. Conclusions

In this paper we presented the limitations of the state-of-the-art authoring tools and discussed open issues and problems concerning the support of learning activities. Based on this discussion, we investigated how the Learning Design framework can be incorporated in the architecture of a SCORM compatible authoring so as to address those issues. Then, we proposed a high-level

system architecture that utilizes the Learning Design principles to provide the means for designing activity-based learning systems. Finally, we introduced a tool, referred to as, the ASK-Learning Designer Toolkit, which implements the presented architectural approach.

7. Acknowledgements

The work presented in this paper is partially supported by European Community under the Information Society Technologies (IST) programme of the 6th FP for RTD - project ICLASS contract IST-507922, the Leonardo da Vinci programme - project eAccess contract EL/2003/B/F/148233. and the Greek Ministry of Development - General Secretariat for Research and Technology - project MindPort contract el03..

References:

- [1] R. M. Marra, & D. H. Jonassen, Limitations of online courses for supporting constructive learning, *Quarterly Review of Distance Education*, 2 (4), 2001, 303-317.
- [2] C.S. Carr, Computer-Supported Collaborative Argumentation: Supporting Problem-based Learning in Legal Education, *Proc. of the Computer support for Collaborative Learning Conference (CSCL)*, Maastricht University, Holland, 2001.
- [3] M. A. Constantino-Gonzales, D. Suthers, & J. Escamilla, Coaching Web-based Collaborative Learning based on Problem Solution Differences and Participation, *International Journal of Artificial Intelligence in Education*, 13(2-4), 2003, 263-299.
- [4] P. Dolog, N. Henze, W. Nejdil, & M. Sintek, Personalization in Distributed eLearning Environments. *Proc. of the Thirteen International World Wide Web Conference*, New York, USA, 2004, 170-179.
- [5] B. Simon, P. Dolog, Z. Miklos, M. Sintek, & D. Olmedilla, Conceptualising Smart Spaces for Learning. *Journal of Interactive Media in Education*, Special Issue on the Educational Semantic Web, 9, 2004.
- [6] P. Brusilovsky, & H. Nijhawan, A Framework for Adaptive E-Learning Based on Distributed Re-usable Learning Activities, *Proc. of the World Conference on E-Learning*, 2002, 154-161.
- [7] P. Libbrecht, Authoring Web Content in ActiveMath: From Developer Tools and Further. *Proc. of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH2004)*, Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia, Eindhoven, Netherlands, 2004.
- [8] T. Murray, MetaLinks: Authoring and affordances for conceptual and narrative flow in adaptive hyperbooks. *International Journal of Artificial Intelligence in Education*, Special Issue on Adaptive and Intelligent Web-Based Systems, 13 (II), 2003, 199-233.
- [9] G. Weber, H.-C. Kuhl, & S. Weibelzahl, Developing adaptive internet based courses with the authoring system NetCoach. *Proc. of the 8th International Conference on User Modeling (UM2001)*, Workshop on Adaptive Hypertext and Hypermedia, Sonthofen, Germany, 2001, 35-48.

- [10] J. Vassileva, & R. Deters, Dynamic courseware generation on the WWW, *British Journal of Educational Technology*, 29(1), 1998, 5–14.
- [11] P. Brusilovsky, J. Eklund, & E. Schwarz, Web-based education for all: a tool for developing adaptive courseware, *Computer Networks and ISDN Systems*, 30, 1998, 291–300.
- [12] H. Hummel, J. M. Manderveld, C. Tattersall, & E. J. R. Koper, Educational modelling language and learning design: new opportunities for instructional reusability and personalised learning. *International Journal on Learning Technology*, 1, 2004, 111-126.
- [13] E. J. R. Koper, & J. M. Manderveld, Educational modelling language: modelling reusable, interoperable, rich and personalised units of learning, *British Journal of Educational Technology*, 35(5), 2004, 537-551.
- [14] P. Brusilovsky, KnowledgeTree: A distributed architecture for adaptive e-learning. *Proc. of the Thirteenth International World Wide Web Conference*, New York, USA, 2004, 104-113.
- [15] P. McAndrew, W. I. S. Woods, A. Little, M. Weller, E. J. R. Koper, & H. Vogten, Implementing Learning Design to support web-based learning. *Proc. of the 10th Australian World Wide Web Conference*, 2004.
- [16] IMS Learning Design Specification Version 1.0. [Online]. Available at: <http://www.imsglobal.org/learningdesign/index.cfm>
- [17] P. Brusilovsky, & J. Vassileva, Course sequencing techniques for large-scale Web-based education. *International Journal of Continuing Engineering Education and Life-long Learning*, 13(1/2), 2003, 75-94.