

ALEF: A Framework for Adaptive Web-based Learning 2.0

Marián Šimko, Michal Barla, Mária Bieliková

Institute of Informatics and Software Engineering,
Faculty of Informatics and Information Technologies,
Slovak University of Technology, Ilkovičova 3, 842 47 Bratislava, Slovakia
{simko, barla, bielik}@fiit.stuba.sk

Abstract. State-of-the-art learning management systems provide their stakeholders with many features coming from Web 2.0 paradigm, but often ignore the need for personalization and adaptation during the learning. More, learning activities are often fragmented – a student needs to make a decision whether he or she wants to take questions or read explanatory materials. In addition, majority of current solutions do not provide a truly interactive environment, where students are allowed to participate in content creation and maintenance. In this paper, we address these issues by proposing and developing a framework for Adaptive Web-based Learning 2.0. We describe basic requirements for such a framework and provide an overview of all its important underlying models and functionality.

Keywords: Adaptive web-based learning, framework, social learning, collaboration, personalization, 2.0

1 Introduction

Computers and broadband Internet connection strongly influenced the way we learn and shifted the traditional in-class learning towards anytime and anywhere web-based learning. We all witnessed the emergence of complex Learning Management Systems (LMS) and e-learning standards, along with unbounded increases in available educational resources on the Web and in digital repositories. However, most of the state-of-the-art approaches just serve electronically the same "one-size-fits-all" static content, which was previously printed in student books, without actually using the benefits of computer and web-based education, namely higher degree of interaction and personalization of the learning flow.

Students, who interact with a teacher and other students in the class should be able to interact also within a virtual learning environment provided by a LMS. Such collaborative learning would not only increase the student's motivation to study (doing something in a group is mostly more fun to do than doing it alone) but also naturally brings him or her more benefits by leveraging the experience of others.

Nowadays, we are all used to the Web 2.0 concepts and are expecting web-based applications to be built and used in a similar way. In some sense, Web 2.0 represents a

challenge for adaptive learning systems and LMS should provide students with means for their active contribution to the presented content (in the form of tagging, commenting and other annotating mechanisms), its sharing and organization (Fig. 1).

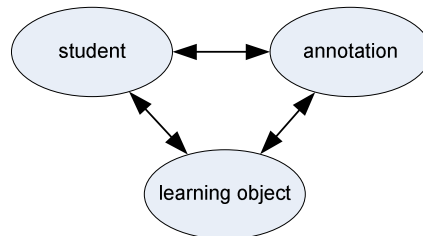


Fig. 1. The Web 2.0 core concept of annotations adapted to e-learning. By a term annotation, we mean any type of data or metadata which can be added to a learning object, e.g., a tag, comment, question.

An advantageous type of interaction, which cannot be achieved in printed books but is feasible in the virtual educational web environment, is instant feedback. All questions, quizzes and exercises can be made interactive, giving the student immediate feedback on the chosen and correct answer along with clarifications or further explanations about reasons behind the question.

An important aspect of web-based learning is personalization. With the increasing number of available learning materials it is becoming crucial to be able to support students in their way through the course, to locate, recognize and understand information, which is the most relevant, considering the given time and progress of the student. Without any support, the student can only with difficulties identify which parts of the course are relevant and which are presenting only additional, not that important information. More, if we consider that the student is not only new to the domain but also to the actual learning system, increased complexity of navigational possibilities may result in "lost-in-hyperspace" problem [1].

Looking at the educational systems from the teachers' point of view, one of the major drawbacks of adaptive course authoring, negatively affecting the spread of adaptive learning systems, is the complexity of metadata used for the personalization in both: definition of such metadata as well as their further maintenance (which is even more challenging in a collaborative environment). Some standardized domain ontologies may be used, but they only exceptionally fit to the author's needs at the desired level of granularity. The domain model has to be often created "from scratch". Unfortunately, manual construction is a demanding task even for small domains. The number of necessary metadata descriptions is counted by hundreds or even thousands.

Our contribution presented in this paper can be seen from two views. First, we propose a schema for Adaptive Web-based Learning 2.0, which new generation LMS creators should follow. This schema takes advantage of all the possibilities that the Web 2.0 currently offers and combines them with main principles of personalized access to educational resources while retaining reusability, easy maintenance and manageability of the learning content and metadata. Second, based on the proposed schema, we present ALEF (Adaptive LEarning Framework), a framework for creating adaptive and highly interactive web-based learning systems.

The framework and its underlying models are designed with respect to the course authoring that is intended to be as flexible as possible. The domain model is simple enough to allow for efficient usage of methods for automatic domain metadata extraction based on lightweight semantics acquisition.

2 Related Work

Probably the most relevant related work is a distributed architecture for adaptive e-learning, KnowledgeTree [1]. The authors identified the main drawbacks of adaptive web-based educational systems (lack of integration of multiple aspects of learning into one system, poor reuse support) and proposed a distributed architecture which allows a student to reach various interactive learning activities from one single point. However, a drawback of the whole approach is that the activities are still kept somehow separated – the system provides a student with all relevant links pointing to distinct services needs to make a decision whether he or she wants to study the explanatory materials, take some questions and quizzes or practice the acquired knowledge on several exercises. Our approach integrates all available learning and supporting activities into one single framework, which makes them easily accessible from any point of the learning flow.

Related issues were addressed also by Meccawy et al. [2]. Similarly to our work, the authors identified that adaptive educational web-based systems should follow Web 2.0 principles in order to be successful. To achieve it, the authors proposed a framework consisting of five web services integrated into the Moodle LMS, extend this LMS by personalization and adaptation features and take advantage of its Web 2.0 presentation layer, which provides collaborative parts of the course (activities, blog etc.). The weak point of the proposed solution is that student modeling process as well as whole personalization is separated from collaborative processes – the collaboration within the Moodle environment cannot directly influence any of the five web services nor their underlying models, and thus drive the behavior of the system.

When considering educational course authoring support, state-of-the-art LMS provide only a partial (and insufficient) assistance, which is related only to models derived from the domain model (e.g., adaptation model [3], goals model [4]). While the domain model forms a basis for domain knowledge representation, its automatic creation is supported very poorly, e.g. by generating course prerequisites [5]. The only relevant evidence of (semi-)automatic domain model generation we are aware of is presented by Cristea and De Mooij [6], where relationships between domain entities are acquired based on the comparison of their domain attributes (albeit these must be entered manually).

One of the reasons that only a few approaches devoted to automated domain model generation exist might be an overwhelming complexity of the used (ontological) models, which are hard to fill manually, not speaking about their automatic population. On the other hand, courses built on the top of simplified domain models with a lightweight semantics (which we are more likely to produce automatically) might give comparable learning outcomes to the students. The support for metadata authoring based rather on lightweight semantic models of the domain is even more

desirable for the case of dynamic content. When a new content emerges as an effect of the collaborative effort of students, it has to be properly assigned to relevant metadata in order to use it throughout the course (e.g., for adaptation realization).

3 Principles of Adaptive Web-based Learning 2.0

The challenge for next generation of LMS lies in the adoption of Web 2.0 concepts into the adaptive web-based learning. We identified three key principles that are required for a design of LMS:

- *Domain modeling* with respect to (i) possibility to automate certain domain model parts creation, (ii) collaborative social aspect and the need to modify or alter domain model by students themselves.
- *Extensible personalization and course adaptation* based on comprehensive user model, which allows for simultaneous employment of different adaptive techniques to enhance the student's learning experience
- *Student active participation in a learning process* with the ability to collaborate, interact and create the content by means of read-write web vision, mainly by different types of annotations allowing for rich interactions on the top of the presented content.

These three principles ensure that a learning environment is no longer seen only as a mean for educational material presentation. It is a place where students collaborate, create, edit, share and organize the educational content according to their learning needs. The notion of a learning management system is shifted towards an integrated learning environment.

Based on these principles, we designed ALEF, an adaptive learning framework for creating adaptive and highly interactive integrated learning environments. The flow of activities that take place within an adaptive Web 2.0-based learning environment and that are addressed by the framework can be seen from two perspectives (Fig. 2):

- learning flow and
- collaborating/creating flow.

The learning flow (Fig. 2, solid line) covers the entire learning process with one *presenter* module, several *personalizers* providing personalization services, respective *user* and *domain models* along with user modeling tools including *semantic logger* and *user model inferencers*. Students learn by interacting with presented materials accessed from the domain model and tailored to user needs by the personalizers taking into account students' characteristics (such as domain knowledge) present in their *user model*. All students' actions are logged and result in appropriate updates of this model.

Personalization services, which play a crucial role in an enhancement of student learning experience, can be related to any stage of the learning process – while student is reading explanatory texts, is taking quick self-assessments to get feedback about freshly acquired knowledge, is practicing on exercises etc. Our architecture

allows for a composition of such personalization services by chaining and combining different personalizers.

The collaborating/creating flow (Fig. 2, dashed line) covers activities related to learning materials enrichment by student themselves. Learning object presentation is obtained from *presenter*. Using a *collaborative adaptive content creator*, supported content type is created by a student. Content type added by the student varies depending on particular collaborative adaptive content creator used. Enrichment can be realized e.g., by assigning annotations that can have different forms: highlighted text, tags, comments or discussion threads. The content is created with respect to the student context, obtained from his or her user model. And vice versa, performing an action related to the content creation reflects into the *user model* update.

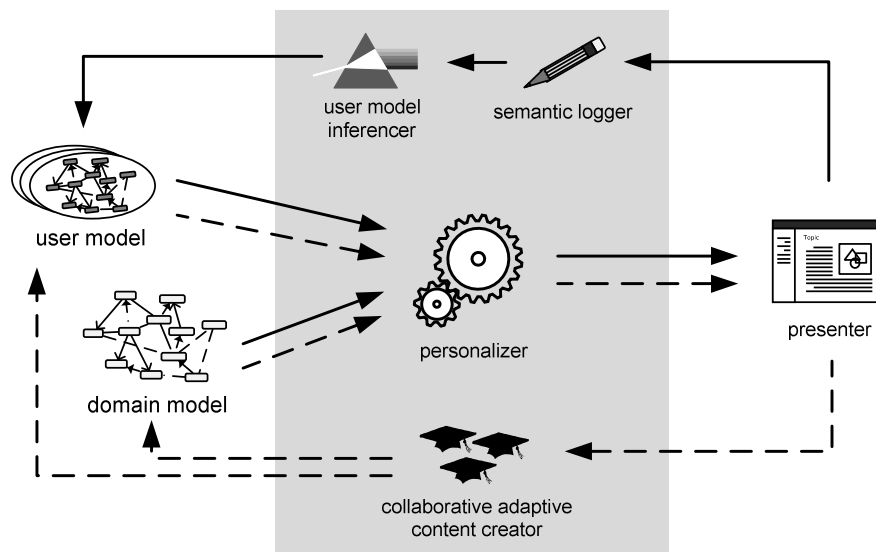


Fig. 2. Activity flows within the framework (learning – solid line, collaborating/creating – dashed line).

4 Models for Adaptive Web-based Learning 2.0

Similarly to any adaptive web-based system there are two basic models in the ALEF framework: domain model and user model. However, we adapted them to address the principles of adaptive web-based learning 2.0 considering the possibility to automate their creation and collaborative aspect of learning.

4.1 Domain model

Domain model of an adaptive educational course represents an area that is a subject of learning. In general, it consists of content entities and metadata entities (in ALEF both seen as resources) that are connected via various types of relationships (Fig. 3).

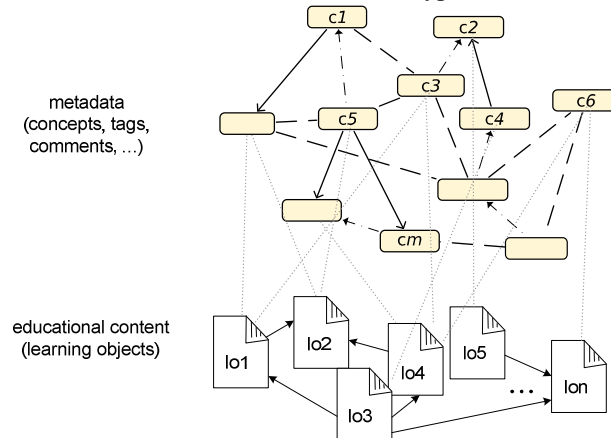


Fig. 3. An example of the domain model of ALEF. Core metadata entities – concepts – are assigned to learning objects. Concepts, like other metadata entities, are interlinked via various types of relationships.

Content

The term *content* refers mainly to a learning object as to fundamental course content representation. However, in a collaborative and read-write environment we support also other types of content entities such as blog (created by both students and teachers) or discussion thread (composed of interlinked comments added by students).

Learning objects are represented by entity `LearningObject` and are further divided into the following types (see Fig. 4):

- Explanation,
- Question,
- Exercise.

Different learning object types and their seamless combination is an important distinguishing feature of our approach. The student is not required to read explanations in one system and practice the acquired knowledge in another one. Different forms of interaction within one system improve student's learning experience and learning outcome.

`Explanation` represents instructional content that describes a subject domain. `Question` and `exercise` represent interactive part of a course. Questions aim to provide students with an immediate feedback on their knowledge with further explanation of respective answer choices. Exercises allow students to practice gained knowledge. Besides enriching the learning process, gathered information is also used to update student model to keep track of user's knowledge.

The domain model covers two distinct parts of every learning object:

- Actual content (text) – stored in XML files using DocBook templates. Allows for easy authoring, maintenance and supports re-usability,

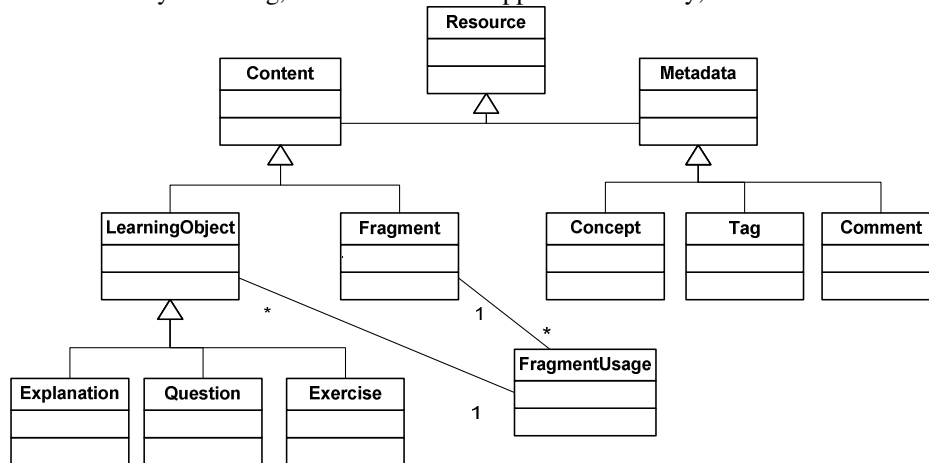


Fig. 4. Domain object model portion – *Resource* types.

- Additional metadata – stored within a relational database holding learning object’s identifier, type and relationships to other domain entities. Contains all information which is relevant for personalization services.

The chosen representation allows for an easy definition of other types of learning objects. The actual content goes to the textual files while its metadata comes to the model as a new subtype of `LearningObject`.

Learning objects are composed of reusable parts referred to as fragments. Fragments are smallest pedagogically coherent units allowing content-based adaptation. Fragments are further subtyped according to a type of particular learning object. Explanation can be composed of one or many Definition fragments. Exercise can be composed of Definition, Hint, Solution and/or Clarification fragments. Similarly, question can be composed of Description and Answer fragments. Each fragment type has its pedagogical role in learning flow and “assists” a student to achieve his or her learning goals.

The order of fragments and metadata related to fragment usage within a learning object is defined by an entity `FragmentUsage` (see Fig. 4).

Metadata

Every content entity is associated with respective metadata. Comparing to other existing approaches, the notion of metadata in ALEF is quite simplified in order to achieve the degree of complexity, which is manageable by ordinary users (both teachers and students). This allows for automatic construction of domain model, and, on the other hand, it provides a solid basis for reasoning resulting in advanced operations such as metadata-based navigation recommendation.

We consider following metadata types:

- Concept,
- Tag,

- `Comment`.

The basic metadata entity is *Concept*. It is a domain knowledge element usually representing topic or subtopic contained within the content (learning object). *Tag* is a keyword or term assigned by a student when organizing their domain knowledge. *Comment* is an advanced type of metadata created by the user in order to annotate the content in a more specific fashion.

Metadata can be interconnected by different types of weighted relationships. We defined `RelatedToRelation`, which represents fundamental connection between any metadata entities. Other types of relations (e.g. prerequisite) are defined by extending `WeightedRelation` or `UnweightedRelation`.

In ALEF, both content entities and metadata entities are considered resources. This generalization is based on the fact that the boundary between content and metadata is very thin (e.g. a comment assigned to learning object can be itself assigned other comments resulting into discussion thread formation). Second reason to generalize *Content* and *Metadata* into *Resource* is the way we model interaction in the framework – relationships between user and content and between user and metadata are very similar.

4.2 User model

Our approach to user modeling within ALEF framework is built on a well-established conception of overlay models [7] but adds several important features to it. Our user model conception is similarly to [8], separated into two layers:

- Evidence (observations) layer storing history of user's interaction with the learning environment – what actions the user has performed in relation to a certain learning object (e.g., user has seen an explanation of *lisp abstract data types* or responded wrongly a question *id42* related to concept of *recursion*).
- Inferred layer storing user characteristics derived from available observations, such as level of mastery (knowledge) or interest in certain concepts.

Moreover, the ALEF framework allows for defining and storing user's attitude (relation) to any type of object present within the learning environment:

- content – primarily learning objects such as explanations, questions, exercises,
- metadata – concepts, annotations and
- other users (students) – effectively creating social network within the system.

Relations of a user to domain objects stored in both user model layers are semantically described using typed connections to ensure the high degree of separation of further processing tools performing inference of additional knowledge or providing user model-based personalization from the particular implementation of the system and its components.

Apart from the mentioned overlay approach to user modeling, which is present in various types of user's relations to domain objects, we provide also means to represent explicit user features which are not connected to a particular object from the learning domain such as goals or backgrounds.

5 Learning Flow

The learning flow in ALEF (outlined in section 3), is based on two central concepts:

- adapting the content,
- tracking the user behavior.

It was already shown [1] that the efficient way to deliver educational content is tailoring it according to the current user knowledge level, interests and needs by employing methods and techniques for personalization and adaptation. These are based on the underlying domain model, which represents the area of learning, and a user model storing user's actual knowledge or goals. The more pedagogically complex a learning environment is, the more adaptation possibilities there exist.

The modern LMS can integrate the study of explanatory texts, quick self-assessment (in order to acquire immediate feedback), and the practice over the set of exercises during one learning session. All these forms of learning are target for personalization and adaptation. The ALEF framework addresses these requirements, as it enables to use and integrate several personalization services by means of *personalizers*. Personalizers can be either chained together to form a processing pipe or can be combined using a voting mechanism (e.g., for a combination of various content recommenders).

For our evaluation domain of learning programming, we developed four personalizers: Content Recommender, Personalized Annotator, Adaptive Test Question Selector and Sidebar Navigator (see an example of screenshot in Fig. 5).

Content Recommender. *Content Recommender* serves as a general personalizer of the content and navigation. Based on the student's knowledge it selects and recommends learning objects (in fact, any `Content` entity, see Fig. 4) tailored to the student needs. Recommendation covers the learning object instance selection and its fragments visibility preconfiguration (e.g., `Hint` fragments in the case of `Exercise` learning objects are hidden for students whose knowledge about associated concepts exceeds a certain threshold).

Personalized Annotator. *Personalized Annotator* filters actually visited learning object's annotations (tags, comments) only to those relevant for the student. It can be viewed as a general personalizer of `Metadata` (see Fig. 4). Its personalization function is similar to personalization function of the *Content Recommender*. However, different adaptation mechanisms are used.

Adaptive Test Question Selector. *Adaptive Test Question Selector* aims at selecting questions, which allow students to obtain immediate feedback about their current progress, and as a side effect it informs the system about student's knowledge, which is stored in the student model. Question selection is based on Item Response Theory (IRT) that enables an adaptive selection of questions based on the student's level of knowledge and question difficulty combined with the topic selection and prioritization following the answer history.

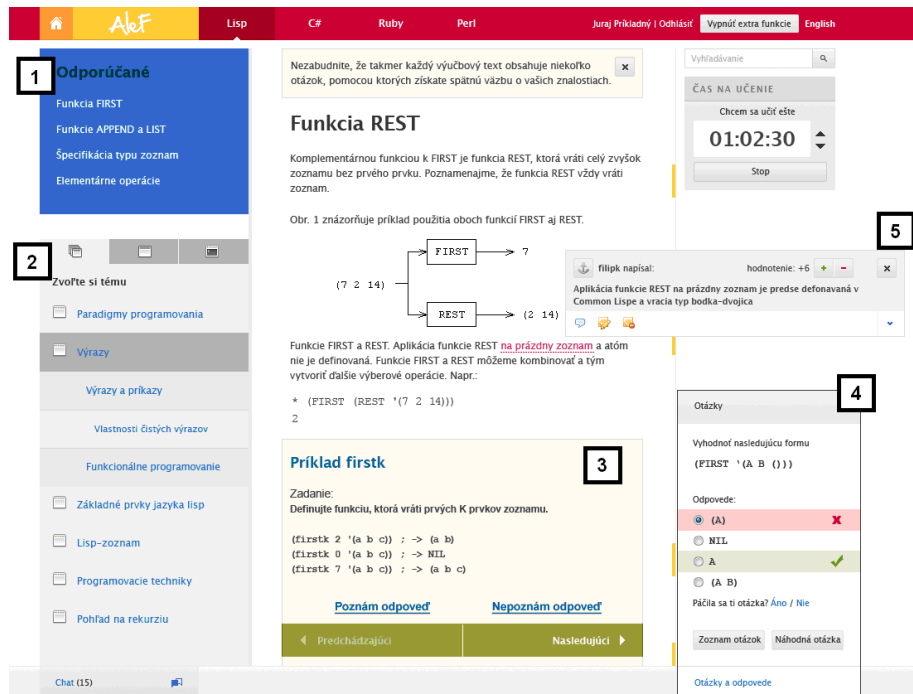


Fig. 5. Screenshot of ALEF user interface for learning Lisp programming (in Slovak). Recommendations coming from Content Recommender are presented either in a separate box (1) above the tabbed-menu or can be embedded within the main content in the form of interactive examples (3). The tabbed-menu (2) provides different navigational styles according to different learning object types and relationships. Collaboratively created questions related to current learning object are visualized on-demand in a pop-up widget (4). Displayed content can be furthermore enriched by adding different types of annotations, which are subsequently accessed by hovering the mouse over the underlined sections of text (5).

Sidebar Navigator. *Sidebar Navigator* focuses on tracing student’s navigation within the learning object presentation page (i.e., scrolling, mouse movements). Based on collected usage data from different users and the student’s social network, it personalizes the visualization of certain learning object parts in the sidebar (e.g. by emphasizing mostly visited text).

6 Creating/Collaborating Flow

The creating/collaborating flow is the ALEF’s realization of the Web 2.0 “read-write” concept. It focuses on social learning and supports students by allowing them to actively participate in the learning process. The traditional web-based learning is shifted towards integrated learning environment where students collaborate, create, edit, share and organize content. The creating/collaborating flow covers the two aspects of student participation:

- contribution to the content (metadata/annotations),
- user model update.

To support the first aspect of student participation, ALEF provides the write access to the course content with so called *collaborative adaptive content creators* entry point. There could be multiple creators, each responsible for other type of annotations. Currently we developed three collaborative adaptive content creators: Tagger, Annotation Creator and Collaborative Question Creator.

Tagger. *Tagger* is a simple form of collaborative adaptive content creator typical for any “2.0” application. It allows students to assign tags to any *Resource* entity (see Fig. 4).

Annotation Creator. *Annotation Creator* serves as a general purpose annotator of learning materials. Students can assign annotations in the form of text comments to selected *part* of any *Resource* entity. Comments may be assigned to other comments resulting into a discussion thread.

Collaborative Question Creator. *Collaborative Question Creator* extends the *Annotation Creator* and allows for creation of special annotations holding questions. During learning, students are encouraged to create testing questions. This way students themselves become the creators of the pedagogical content. Collaboratively created questions support learning as they are related to the content that is the most important, unclear, or controversial (from students’ point of view).

The second aspect of Creating/Collaborating flow is a user model update. The ALEF captures actions related to the flow (e.g. assigning a tag, adding a comment or creating a question) and reflects these actions in the student model. For example, assigning a tag by a student is often interpreted as an increased interest in a given resource (learning object). Thus, we can define an update of user interest in concepts that are related to the given learning object.

Although ALEF allows students to participate in course enrichment, it does not support learning objects authoring. Course authoring from teacher’s point of view (explanatory text creation, course sequencing, domain description with metadata, etc.) is left to the external authoring tools and pedagogical supervision of a teacher.

7 Conclusions

In this paper, we presented a schema for Adaptive Web-based Learning 2.0, which defines next generation of LMS, delivering experience of collaborative, creative and personalized learning. We defined generic-enough models used by such a system, namely *domain model* based on lightweight semantics which opens new possibilities of automated course metadata creation [9] and *student model*, which acts as a basis of personalization of the whole learning flow.

We developed an adaptive learning framework ALEF, which follows the mentioned schema and combines different learning activities (such as learning from

explanatory texts, questions or exercises) along with highly interactive and social environment of the Web 2.0. The contribution of our approach is the integration of the learning and supporting activities into one single framework, making them easily accessible during learning. We already proved on our earlier educational system that such a personalized combination of texts with interactive objects like questions and exercises are improving the efficiency of learning and raise the learning outcome [10].

We proved the feasibility of the framework by using it to create a course for learning programming in Lisp language, which is used in a standard course of Functional and logic programming at our institute.

Acknowledgments. This work was partially supported by the grant KEGA 028-025STU-4/2010 and it is the partial result of the Research & Development Operational Programme for the project Support of Center of Excellence for Smart Technologies, Systems and Services, ITMS 26240120029, co-funded by the ERDF.

References

1. Brusilovsky, P.: KnowledgeTree: A Distributed Architecture for Adaptive E-Learning. In Proc. of the 13th Int. World Wide Web Conf., WWW'04, New York, NY, USA, ACM Press, pp. 104–113 (2004)
2. Meccawy, M. et al.: WHURLE 2.0: Adaptive Learning Meets Web 2.0. In Times of Convergence. Technologies Across Learning Contexts, LNCS 5192, Springer (2008)
3. De Bra, P., et al. Authoring and Management Tools for Adaptive Educational Hypermedia Systems: The AHA! Case Study. In Studies in Computational Intelligence (SCI), no. 62, Springer-Verlag, pp. 285–308 (2007)
4. Cristea, A. I., De Mooij, A.: LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators. In Proc. of the 12th Int. World Wide Web Conf., WWW'03, Budapest, Hungary, ACM (2003)
5. Sosnovsky, S., Brusilovsky, P., Yudelso, M.: Supporting Adaptive Hypermedia Authors with Automated Content Indexing. In 2nd Int. Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia at AH'04, pp 23–26 (2004)
6. Cristea, A. I., De Mooij, A.: Designer Adaptation in Adaptive Hypermedia Authoring. In Proc. of the Int. Conf. on Information Technology: Computers and Communications ITCC'03. Las Vegas, US, IEEE CS, 2003, pp. 444–448 (2003)
7. Brusilovsky, P. Millán, E.: User Models for Adaptive Hypermedia and Adaptive Educational Systems. In P. Brusilovsky, A. Kobsa and W. Neidl (eds.): The Adaptive Web, LNCS 4321, Springer-Verlag, pp. 3–53 (2007)
8. Šimko, M. Bieliková, M.: Automated Educational Course Metadata Generation Based on Semantics Discovery. In Proc. of European Conf. on Technology Enhanced Learning, EC-TEL'09, Nice, France. LNCS 5794, Springer, pp. 99–105 (2009)
9. Yudelso, M., et al.: A User Modeling Server for Contemporary Adaptive Hypermedia: An Evaluation of the Push Approach to Evidence Propagation, In User Modeling 2007, LNCS 4511, Springer, pp. 27–36 (2009)
10. Vozár, O., Bieliková, M.: Adaptive Test Question Selection for Web-based Educational System. In: Proc. of SMAP'08, 3rd Int. Workshop on Semantic Media Adaptation and Personalization. Prague, Czech Republic, CS IEEE Press, pp. 164–169 (2008)