WELSA: An Intelligent and Adaptive Web-based Educational System

Elvira Popescu, Costin Badica, and Lucian Moraret

Software Engineering Department, University of Craiova, Romania
{popescu_elvira, badica_costin}@software.ucv.ro, dnlac@yahoo.com

Summary. This paper deals with an intelligent application in the e-learning area (WELSA), aimed at adapting the courses to the learning preferences of each student. The technical and pedagogical principles behind WELSA are presented, outlining the intelligent features of the system. The learner modeling and adaptation methods are also briefly introduced, together with their realization in WELSA. Finally, the platform is validated experimentally, proving its efficiency and effectiveness on the learning process, as well as the high degree of learner satisfaction with the system.

1 Introduction

An important class of intelligent applications in e-learning are the adaptive ones, namely those that aim at individualizing the learning experience to the real needs of each student. The rationale behind them is that accommodating the individual differences of the learners (in terms of knowledge level, goals, learning style, cognitive abilities etc) is beneficial for the student, leading to an increased learning performance and/or learner satisfaction. A common feature of these systems is that they build a model of learner characteristics and use that model throughout the interaction with the learner [2]. An adaptive system must be capable of managing learning paths adapted to each user, monitoring user activities, interpreting them using specific models, inferring user needs and preferences and exploiting user and domain knowledge to dynamically facilitate the learning process [3].

The idea dates back to 1995-1996, when the first intelligent and adaptive Web-based educational systems (AIWBES) were developed [2]. Since then, both the intelligent techniques employed evolved and the range of learner characteristics that the systems adapt to expanded. A relatively recent characteristic that has started to be taken into account is the learning style of the student, i.e. the individual manner in which a person approaches a learning task, the learning strategies activated in order to fulfill that task. More formally, learning styles represent a combination of cognitive, affective and other psychological characteristics that serve as relatively stable indicators of the way a learner perceives, interacts with and responds to the learning environment [8].

This paper deals with an intelligent e-learning platform that adapts to the learning style of the students, as its name suggests: **W**eb-based **E**ducational system with **L**earning **S**tyle **A**daptation (WELSA). The next section gives an overview of the system, briefly introducing its main features: i) an implicit and dynamic learner modeling method; ii) a dynamic adaptation approach; and iii) an intelligent way of indexing and organizing the learning material. The system architecture is also presented, as well as an example of the platform at work. The following 2 sections describe in more detail the main components responsible with the system's functionality: modeling component (section 3) and adaptation component (section 4). Finally, the system usability is evaluated in section 5 and some conclusions are drawn in section 6.

2 WELSA Overview

2.1 Main Features

The main pedagogical goal of WELSA is to provide an educational experience that best suits the learning style of each student. In order to do that, the system first needs to identify these learning preferences. The learning style model employed in WELSA is that described in [14], called ULSM (Unified Learning Style Model). ULSM is actually a collection of learning preferences extracted from the main learning style models proposed in the literature, which was conceived to cover a wide range of characteristics, while at the same time aiming for independence between the learning preferences and the least possible overlap. More specifically, ULSM integrates learning preferences related to: perception modality, way of processing and organizing information, as well as motivational and social aspects. The advantages of ULSM versus traditional learning style models in Web-based learning settings have been outlined in [14].

Regarding the method used for the identification of the students' preferences, an implicit approach was chosen, based on the analysis of the students' interactions with the system. This overcomes the psychometric flaws of the traditional measuring instruments, while at the same time is more user friendly, not requiring any additional work from the part of the student. Furthermore, with this intelligent modeling mechanism, the learner model may be continuously updated (i.e. dynamic modeling method) (see section 3).

As far as the adaptation technologies are concerned, WELSA makes use of both adaptive presentation and adaptive navigation support technologies [2], providing the student with an individualized path through the learning material. The process is fully automated, based on a set of built-in adaptation rules: the course pages are dynamically generated by the system for each student, according to her/his learner model (see section 4).

In order to achieve the modeling and adaptation objectives, the system uses a fine grained representation of the learning content: the most complex learning object (with the coarsest granularity) is the course, while the finest granularity learning object (LO) is the elementary educational resource. This fine-grained representation facilitates the automatic combination of LOs as well as their reuse in different contexts.

Each such elementary LO corresponds to a particular educational resource (e.g. a .txt file containing a definition or a .jpg file containing an example), which has a metadata file associated to it. The set of metadata that we propose describe the learning object from the point of view of instructional role, media type, level of abstractness and formality, type of competence etc. These metadata were created by enhancing core parts of Dublin Core [6] and Ullrich's instructional ontology [19] with some aspects specific to learning styles. It is worth pointing out that these metadata are independent of any particular learning style model, thus ensuring the independence between the domain and adaptation models.

The internal representation of the course is XML-based (more details about the content organization can be found in [12]), with the structure of the course (chapters, sections, subsections...) being defined and stored separately from the actual content (the elementary LOs). The Web pages for each student are generated dynamically by the course player, starting from the structure defined in the XML chapter file and filling it with the corresponding LOs.

2.2 Main Functionalities

WELSA's functionalities are primarily addressed at the students, who can learn by browsing through the course and performing the instructional activities suggested (play simulations, solve exercises etc). They can also communicate and collaborate with their peers by means of the forum and chat. Students' actions are logged and analyzed by the system, in order to create accurate learner models. Based on the identified learning preferences and the built-in adaptation rules, the system offers students individualized courses.

WELSA provides also functionalities for the teachers, who can create courses by means of the dedicated authoring tool; they can also set certain parameters of the modeling process, so that it fits the particularities of their course.

Figure 1 shows how WELSA appears for a learner who is studying a course on Artificial Intelligence (more specifically the chapter on ”Constraint satisfaction problems”, based on the classical textbook of Poole, Mackworth and Goebel [11]).

A few notes should be made regarding the course pages: the first resource (LO) on the page is entirely visible (expanded form), while for the rest of LOs only the title is shown (collapsed form). Of course, the student may choose to expand or collapse any resource, as well as lock them in an expanded state by clicking the corresponding icons. Also, there are specific icons associated to each LO, depending on its instructional role and its media type, in order to help the learner browse more effectively through the resources. Finally, navigation can be done by means of the Next and Previous buttons, the course outline or the left panel with the chapter list.

2.3 Architecture

The overall architecture of WELSA is illustrated in Fig. 2. As can be seen in the figure, WELSA is composed of three main modules:

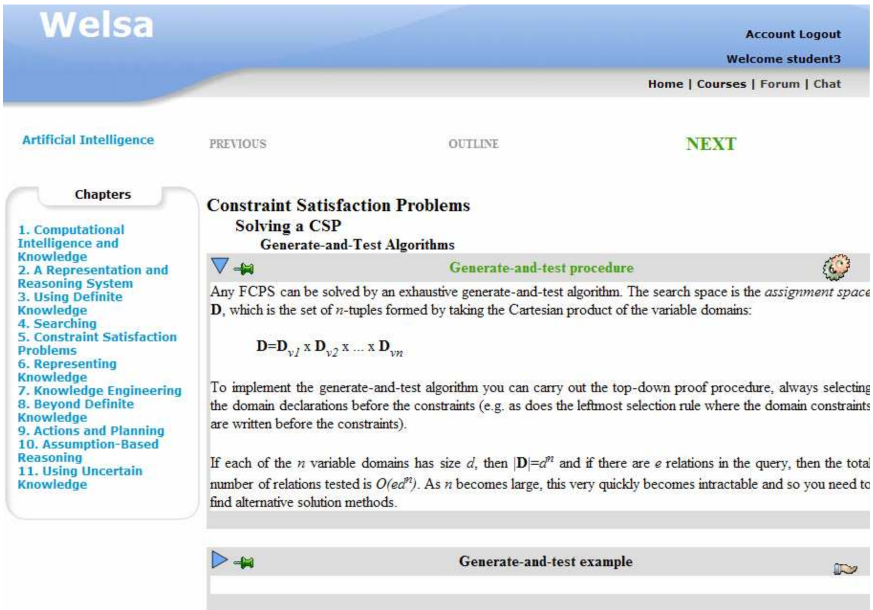

Fig. 1. A snapshot of WELSA (student view)

- an authoring tool for the teachers, allowing them to create courses conforming to the internal WELSA format (XML-based representation)
- a data analysis tool, which is responsible for interpreting the behavior of the students and consequently building and updating the learner model, as well as providing various aggregated information about the learners
- a course player (basic learning management system) for the students, enhanced with two special capabilities: i) learner tracking functionality (monitoring the student interaction with the system); ii) adaptation functionality (incorporating adaptation logic and offering individualized course pages).

The first module was presented in detail in [13]. The other two modules will be described in the following two sections respectively.

As far as the implementation is concerned, Java-based and XML technologies are employed for all WELSA components. Apache Tomcat 6.0 is used as HTTP Web server and servlet container and MySQL 5.0 is used as DBMS.

3 WELSA Analysis Tool

Once the learner actions are recorded by the course player, they have to be processed by the Analysis tool, in order to yield the learning preferences of the students. The modeling mechanism is depicted in Fig. 3

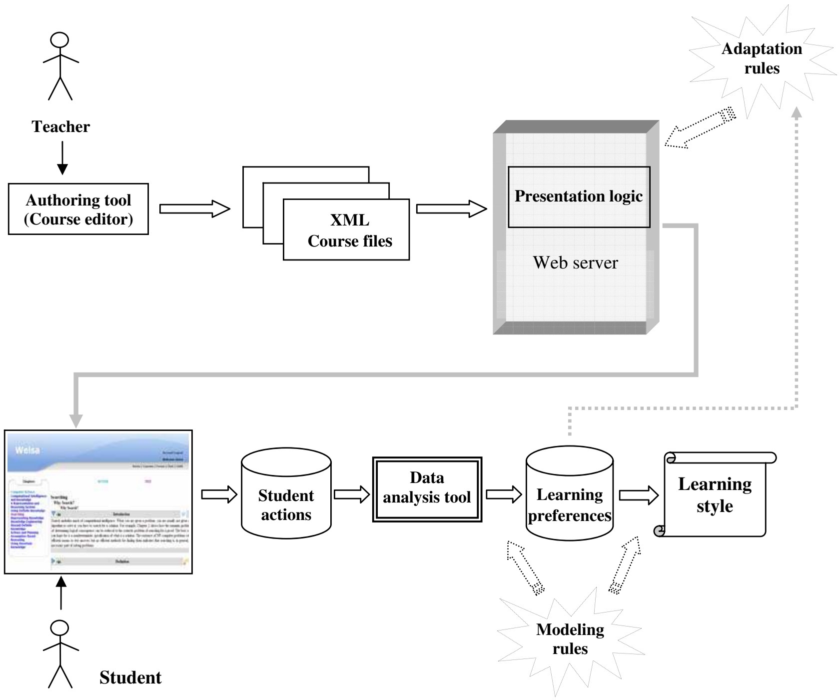

Fig. 2. Schematic WELSA architecture

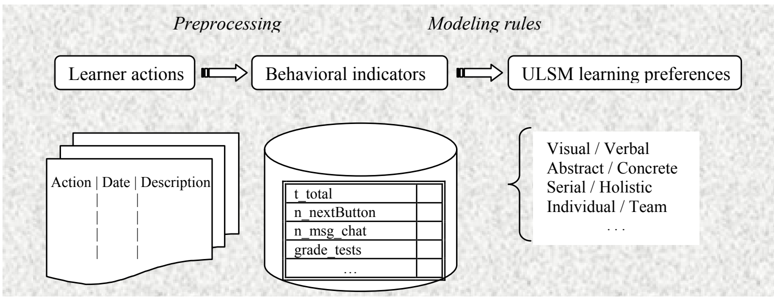

Fig. 3. WELSA learner modeling mechanism

In order to determine the pattern values, a pre-processing phase of the raw data (i.e. the student actions and the associated timestamps) is necessary, for computing LO access times and filtering spurious values. The data are then aggregated to obtain the pattern values for each student (e.g. total time spent on the course, total number of actions performed while logged in, time spent on each type of LO, number of hits on each category of LOs, the order of accessing the LOs, the number of navigation actions of a specific type, the number of messages in chat / forum etc). The reliability levels of these patterns are calculated as well (i.e. the larger the number of available relevant actions, the more reliable the resulted pattern). Next the Analysis tool infers the ULSM preferences of each student, using modeling rules based on the pattern values, their reliability levels and their weights.

It should be noted that these rules also take into account the specificities of each course: the pattern thresholds as well as the importance (weight) of each pattern may vary with the structure and subject of the course. Therefore the teachers should have the possibility to adjust the predefined values to correspond to the particularities of her/his course or even to eliminate some of the patterns, which are not relevant for that course. This is why the Analysis tool has a configuration option, which allows the teacher to modify the pattern weight and threshold values.

4 WELSA Course Player

WELSA doesn't store the course Web pages but instead generates them on the fly, by means of the course player module. The schematic representation of this component's architecture is illustrated in Fig. 4.

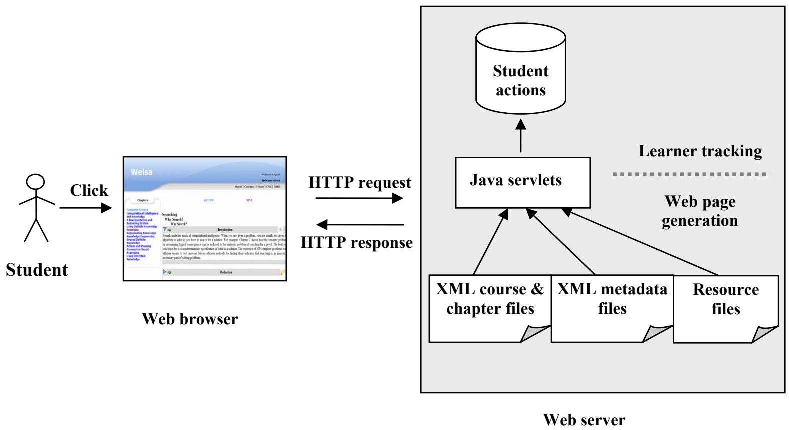

Fig. 4. Course player schematic architecture

The main function of the course player is to generate the Web pages so that they can be visualized by the students. These Web pages are dynamically composed from the elementary learning objects, following the structure indicated in the XML course and chapter files. An example of such a Web page resulted from composing several LOs is included in Fig. 1.

Another function of the course player is to track student actions (down to click level) and record them in a database for further processing. This is done with the help of JavaScript code added to the HTML page, coupled with Ajax technology. Thus the application can communicate with the Web server asynchronously in the background, without interfering with the display and behavior of the existing page.

Apart from the two specific functionalities (Web page generation and learner monitoring), WELSA course player also incorporates some basic LMS (learning management system) functions, such as: administrative support (registration and authentication) and communication and collaboration tools (discussion forum, chat).

This variant of the course player presented so far is used for the non-adaptive version of WELSA (employed for experimental purposes). For the final system however, the player was enhanced with an adaptation mechanism, allowing the dynamic generation of individualized courses for each student. More specifically, the adaptation sub-component queries the learner model database, in order to find the ULSM preferences of the current student. Based on these preferences, the component applies the corresponding adaptation rules and generates the new Web page. As explained in section 2, these adaptation rules involve the use of LO metadata, which are independent of any learning style; however, they convey enough information to allow for the adaptation decision making (i.e. they include essential information related to the media type, the level of abstractness, the instructional role etc).

The conception of these adaptation rules was a delicate task, since it involved interpretation of the learning style literature (which has a rather descriptive nature) in order to extract the prescriptive instructional guidelines. Our pedagogical goal was to offer students recommendations regarding the most suited learning objects and learning path, but let the students decide whether they want to follow our guidelines or not. We therefore decided to rely on sorting and adaptive annotation techniques rather than direct guidance or hiding/removing fragments. We also decided to use the popular ”traffic light metaphor”, to differentiate between recommended LOs (with a highlighted green title), standard LOs (with a black title, as in case of the non-adaptive version of WELSA) and not recommended LOs (with a dimmed light grey title) [16].

The adaptation mechanism is illustrated in Fig. 5, with a fragment of a Web page from the AI course generated for a student with a preference towards *Concrete, practical examples* rather than *Abstract concepts and generalizations*. The page is dynamically composed by selecting the appropriate LOs (mainly of type *Example*), each with its own status (highlighted in case of the LOs of type *Example* and standard in case of LOs of type *Definition*) and ordered correspondingly (first the notion of ”Constraint satisfaction problem” is illustrated by means of two examples and only then a definition is provided).

5 System Validation

Taking into account the structure and mission of the WELSA system, its validation had to be performed on three directions: i) the precision of the modeling method; ii) the efficiency and effectiveness of the adaptation approach; iii) the usability and acceptability of the platform as a whole.

As far as the modeling method is concerned, an experiment involving 71 undergraduate students was realized. The learners studied an AI course module on ”Search strategies and solving problems by search” and all of their interactions with WELSA

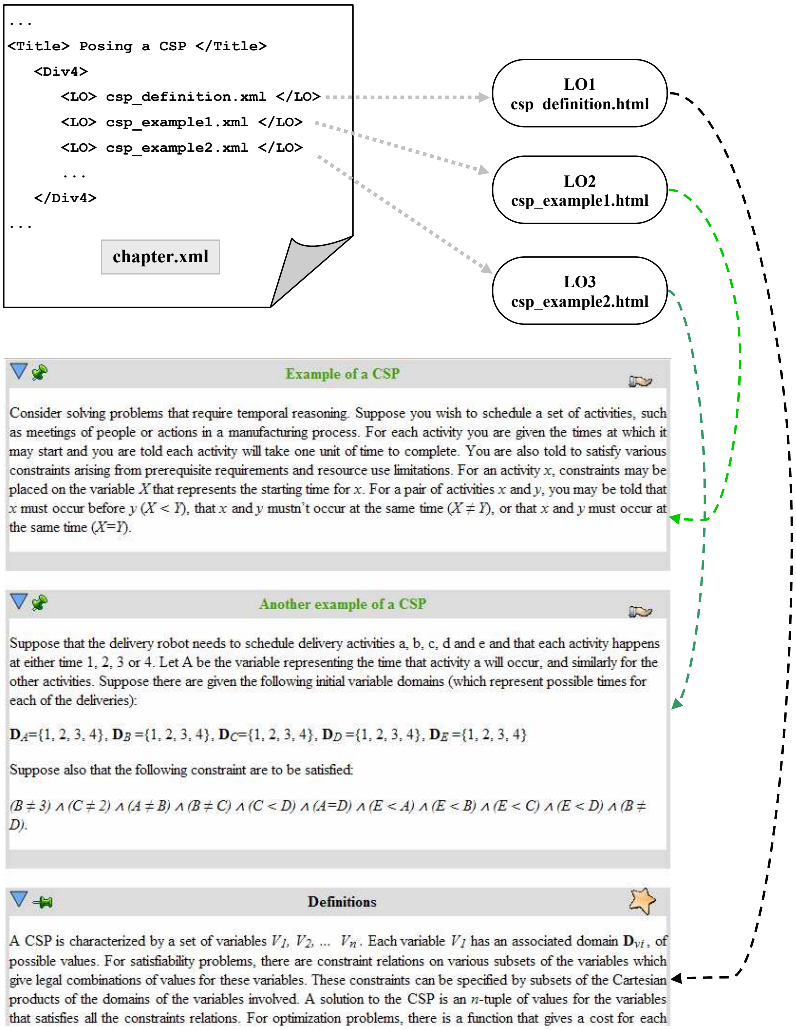

Fig. 5. Composing a page from elementary LOs for a student with *Concrete* preference

were recorded by the course player. Next, the Analysis tool computed the values of the behavioral patterns and applied the modeling rules, inferring the ULSM learning preferences of each student. In order to evaluate the validity of our modeling method, the results obtained by the Analysis tool (implicit modeling method) were compared with the reference results obtained using the ULSM questionnaire (explicit modeling method); good precision results were obtained, with an average accuracy of 75.70%, as reported in [14].

In order to assess the effect of adaptation on the learners, we performed another experiment in which the students had to interact with the adaptive version of

WELSA. After studying another AI course module on ”Constraint satisfaction problems”, the students were asked to fill in an opinion questionnaire, comparing their experiences in the adaptive versus non-adaptive sessions. The results obtained are very encouraging [15], with a perceived improvement in terms of enjoyment and overall satisfaction (for 65.63% of the students), as well as motivation and learning effort (for 56.25% of the students).

The final step of our research was the global evaluation of WELSA system. After following the course sessions, the students were asked to assess various aspects of their learning experience with WELSA, on a 1 to 10 scale (e.g. course content, presentation, platform interface, navigation options, expand/collapse functionality for the resources, communication tools, the course as a whole). All in all, very good marks were assigned to most of the features, with only one feature (the communication tools) receiving lower (but still satisfactory) marks. We can therefore conclude that students had a very positive learning experience with WELSA. These findings are reflected also in the readiness of the students to adopt WELSA system for large scale use, with 87.50% of the students willing to do so and only 6.25% reluctant.

6 Conclusions

The WELSA system described in this paper is an intelligent e-learning platform, aimed at adapting the course to the learning preferences of each student. Unlike similar systems ([1], [4], [5], [7], [9], [10], [17], [18], [20]), WELSA is based not on a single learning style model, but on a distilled complex of features extracted from several such learning style models (ULSM). Furthermore, the identification of the student’s learning style is realized using an implicit modeling method, which only a small number of related systems attempt to use ([5], [7], [17]). Finally, WELSA was thoroughly tested and experimental data is available regarding the efficiency and effectiveness of the adaptation on the learning process.

As future work, the system could be extended by adding more tools and functionalities borrowed from LMSs, such as: more advanced communication and collaboration tools (as the student surveys suggested), student involvement tools (student portfolio, bookmarks, calendar/schedule, searching facilities, context sensitive help etc). Another possible extension could be made to the adaptation component, by incorporating a wider variety of adaptation actions, including also collaboration level adaptation.

References

1. Bajraktarevic, N., Hall, W., Fullick, P.: Incorporating learning styles in hypermedia environment: Empirical evaluation. *Proc. Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems* (2003) 41–52.
2. Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. *International Journal of Artificial Intelligence in Education*, 13 (2-4) (2003) 159–172.

3. Boticario, J.G., Santos, O.C., van Rosmalen P.: Issues in Developing Standard-based Adaptive Learning Management Systems. *EADTU 2005 Working Conference: Towards Lisbon 2010: Collaboration for Innovative Content in Lifelong Open and Flexible Learning* (2005).
4. Carver, C. A., Howard, R. A., Lane, W. D.: Enhancing student learning through hypermedia courseware and incorporation of student learning styles. *IEEE Transactions on Education*, 42 (1999) 33–38.
5. Cha, H. J., Kim, Y. S., Lee, J. H., Yoon, T. B.: An Adaptive Learning System with Learning Style Diagnosis based on Interface Behaviors. *Workshop Proceedings of Intl. Conf. E-learning and Games (Edutainment 2006)* (2006).
6. Dublin Core Metadata Initiative `http://dublincore.org/`.
7. Graf, S.: *Adaptivity in Learning Management Systems Focussing on Learning Styles*. PhD Thesis, Vienna University of Technology, Austria (2007).
8. Keefe, J.W.: Learning style: an overview. *NASSP's Student Learning Styles: Diagnosing and Prescribing Programs* (1979) 1–17.
9. Lee, C. H. M., Cheng, Y. W., Rai, S., Depickere, A.: What Affect Student Cognitive Style in the Development of Hypermedia Learning System?. *Computers & Education*, 45 (2005) 1–19.
10. Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G.D.: Personalizing the interaction in a Web-based educational hypermedia system: the case of INSPIRE. *User-Modeling and User-Adapted Interaction*, 13 (2003) 213–267.
11. Poole, D., Mackworth, A., Goebel, R.: *Computational Intelligence: A Logical Approach*. Oxford University Press (1998).
12. Popescu, E., Badica, C., Trigano, P.: Description and organization of instructional resources in an adaptive educational system focused on learning styles. *Studies in Computational Intelligence*, Vol. 78, Springer (2008) 177–186.
13. Popescu, E., Trigano, P., Badica, C., Butoi, B., Duica, M.: A Course Authoring Tool for WELSA Adaptive Educational System. *Proc. ICCC 2008* (2008) 531–534.
14. Popescu, E.: Diagnosing Students' Learning Style in an Educational Hypermedia System. *Cognitive and Emotional Processes in Web-based Education: Integrating Human Factors and Personalization*, Advances in Web-Based Learning Book Series, IGI Global (2009) 187–208.
15. Popescu, E.: Evaluating the Impact of Adaptation to Learning Styles in a Web-based Educational System. *Proc. ICWL 2009*, LNCS, Springer (in press).
16. Popescu, E., Badica, C.: Providing Personalized Courses in a Web-Supported Learning Environment. *Proc. WI-IAT 2009 (Workshop SPeL)*, IEEE Computer Society Press (in press).
17. Sangineto, E., Capuano, N., Gaeta, M., Micarelli, A.: Adaptive course generation through learning styles representation. *Journal of Universal Access in the Information Society*, 7(1) (2008) 1–23.
18. Triantafillou, E., Pomportsis, A., Demetriadis, S.: The design and the formative evaluation of an adaptive educational system based on cognitive styles. *Computers & Education*, 41 (2003) 87–103.
19. Ullrich, C.: The Learning-Resource-Type is Dead, Long Live the Learning-Resource-Type!. *Learning Objects and Learning Designs*, 1 (2005) 7–15.
20. Wang, T., Wang, K., Huang, Y.: Using a Style-based Ant Colony System for Adaptive Learning. *Expert Systems with Applications*, 34(4) (2008) 2449–2464.