# Personalized Learning in Web-Based Intelligent Educational Systems: Technologies and Techniques

*Ioannis Hatzilygeroudis, Jim Prentzas, John Garofalakis*

University of Patras
Dept of Computer Engineering & Informatics
26500 Patras, Greece
ihatz/prentzas@ ceid.upatras.gr

Research Academic Computer Technology Institute
P.O. Box 1122,
26110 Patras, Greece
ihatz/garofala@cti.gr

## Abstract

In this paper, we present technologies and techniques used in web-based intelligent educational systems (WBIESs). As WBIESs we consider either web-based intelligent tutoring systems (ITSs) or adaptive hypermedia education systems (AHESs) incorporating intelligent techniques. We present technologies and techniques for all the three basic components of a WBIES: the domain knowledge, the student modelling unit and the pedagogical module. The technologies and techniques come from two fields, artificial intelligence (AI) and adaptive hypermedia (AH). At the end, we outline possible key trends and technologies for future WBIESs.

## 1 Introduction

Numerous computer-based systems have been developed for education during the last decades. The first such systems were called Computer Aided Instruction (CAI) systems and were quite effective in helping learners. A major disadvantage, however, was their inability to adapt instruction to the individual's needs. This drawback gave rise to a new generation of education systems encompassing intelligence, in order to increase their effectiveness, called *Intelligent Educational Systems* (IESs).

Intelligent Tutoring Systems (ITSs) constitute a popular type of IESs. ITSs take into account the user's knowledge level and skills and adapt presentation of the teaching material to the needs and abilities of him/her (Polson and Richardson, 1988). This is achieved by using Artificial Intelligence (AI) techniques to represent pedagogical decisions as well as domain knowledge and information regarding each user (student). ITSs were usually developed as stand-alone systems. However, the emergence of the WWW gave rise to a number of Web-based ITSs (Brusilovsky, 1999), a type of *Web-Based Intelligent Educational Systems* (WBIESs) (Hatzilygeroudis, 2004).

Another type of educational systems is Adaptive Educational Hypermedia Systems (AEHSs) (Brusilovsky, Kobsa & Vassileva, 1998). This type of systems is specifically developed for hypertext environments such as the WWW. They use technologies and techniques form Adaptive Hypermedia (AH). The main services offered to their users are adaptive presentation of the teaching content and adaptive navigation by adapting the page hyperlinks. Compared to ITSs, they offer a greater sense of freedom to the user, since they allow a guided navigation to the user-adapted educational pages. Furthermore, they dynamically construct or adapt the educational pages in contrast to ITSs in which the contents of the educational pages are typically static. Enhancing AEHSs with techniques from ITSs creates another type of WBIESs.

So, WBIESs can be considered as either web-based ITSs or AHESs incorporating AI techniques. In this paper, we outline technologies and techniques used in WBIESs, in all the components of such systems. At the end, we specify key trends and technologies for future WBIESs.

The structure of the paper is as follows. Section 2 describes the main components of a WBIES. Sections 3, 4 and 5 presents technologies and techniques used in the development of the three basic components of a WBIES respectively. Section 6 specifies possible key trends and technologies for future WBIESs. Finally, Section 7 concludes.

## 2 Basic Structure of a Web-Based Intelligent Educational System

The basic architecture of a WBIES is actually the same as that of an ITS. So, Figure 1 depicts the basic architecture of an IES. It mainly consists of the following components: (a) the *domain knowledge*, which contains the teaching content and information about the teaching subject, (b) the *student modelling unit*, which records information concerning the student, (c) the *pedagogical module*, which encompasses knowledge regarding various pedagogical decisions, (d) the *user interface*, which communicates with the user.

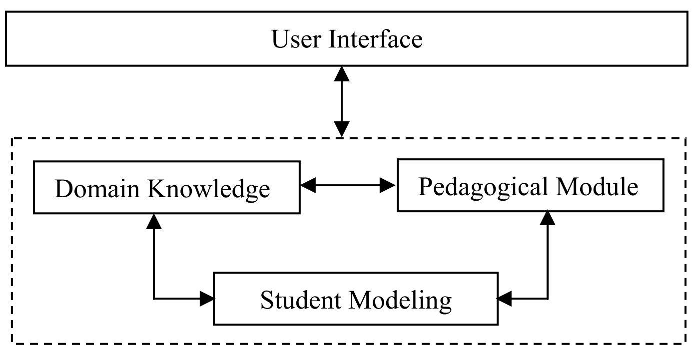


**Figure 1:** Basic Architecture of an Intelligent Educational System

Sometimes, an extra component is considered, namely the *expert module*. Expert module typically deals with interactive problem solving, e.g. with providing help in an intelligent way or giving some examples during a problem solving. It acts as an expert (tutor) who supervises students as they solve problems and give advice, hints, examples etc. This module can be considered as part of the pedagogical module in Figure 1. Typically, domain knowledge, student modelling unit and pedagogical module are sited on the server side, whereas user interface on the client side. In the sequel, we concentrate on the components on the server side.

## 3 Domain Knowledge

Domain knowledge contains knowledge regarding the teaching subject as well as the actual teaching content. It usually consists of two parts: (a) *knowledge concepts* and (b) *learning units*.

Knowledge concepts refer to basic entities of knowledge concerning the domain. Every concept may be described by a number of attributes such as, its name, its level of difficulty etc. Furthermore, various concepts can be linked to other concepts according to their relation. In this way, one or more *concept networks* are formed representing the pedagogical structure of the domain to be taught. The typical relations between the concepts are the following:

- *Prerequisite*: Some concepts are prerequisite of others. A student should know some or all the prerequisite concepts of a concept proceed to it.
- *Part-of*: Many simpler concepts are part of a more complex concept.
- *Is-a*: This relation connects a concept with others that are its typical instances.

The most usual representation schemes for concept networks are AND-OR graphs, semantic nets, frames and conceptual graphs, which are called structured representations. AND-OR graphs are representation schemes containing AND links denoting conjunctive prerequisite relations and OR links, denoting disjunctive prerequisite relations (Vassileva, 1997). Frames and conceptual graphs are more complex representation schemes, often encompassing concepts, learning units and sometimes rules for their selection and ordering.

Learning units constitute the teaching content presented to system users. A learning unit may correspond to theory, examples or exercises. Learning units may be static educational pages or fragments from which educational pages are dynamically generated. The second case is most usual in AEHSs. Each learning unit is associated with one or more knowledge concepts either prerequisite or outcome. The student is required to know the learning unit's prerequisite concepts in order to be able to grasp corresponding knowledge. By studying learning units, the student gains knowledge of its outcome concepts. The distinct representation of the domain's pedagogical structure (concepts) and the actual teaching content (learning units) greatly facilitates updates in domain knowledge.

To facilitate selection and ordering of learning units, domain knowledge frequently includes a *meta-description* of learning units based on their general attributes. There exist *standards* for the meta-description of learning units such as ARIADNE (http://ariadne.unil.ch), IEEE LTSC Learning Object Metadata (http://ltsc.ieee.org/) and Dublin Core (http://purl.org/DC/). Mappings between these standards have also been developed. The primary goal of these standards is to foster the share and reuse of electronic educational material. Key issues that should be dealt with by the standards is the easy creation of metadata by humans and furthermore the efficient exploitation of the metadata by users looking for relevant educational content. The educational metadata is grouped into categories in order to increase their comprehensibility.

## 4 Student Modelling

The student model records information regarding student's knowledge state and traits. This information is vital for the system to adapt to the student's needs. The process of inferring a student model from observable behaviour is called *diagnosis* because it is much like the medical task of inferring a hidden physiological state from observable signs (Polson & Richardson 1988). The term *bandwidth* is used to describe the amount and quality of information available to the diagnosis process. There are many possible student characteristics that can be recorded in the student model and choosing the most appropriate ones can be a problem. If the student model is incomplete, the system's adaptivity will be unsuccessful, whereas if it is complex, the system's operation will be encumbered.

Primary student characteristics recorded in the student model are the following:

- *History of the user's interaction with the system*: It contains information such as the learning units 'visited' by the student or the answers given to exercises.
- *Mental skills*, including characteristics such as learning ability and concentration derived mainly from the student's interaction with the system.
- *Goals*: It is a characteristic that often changes, e.g. from session to session. The goals can be discerned to high level goals, e.g. a concept, or low level goals, e.g. problem solving goals (Brusilovsky 1996).
- *Preferences*: They mainly refer to educational content presentation parameters such as multimedia type preferences (e.g. text, images, or animations) regarding the presented learning units or their level of detail.
- *Background* and *experience*: The student's background refers to experiences beyond the scope of the teaching subject, which are important enough to be considered. Such information includes experience in other relevant fields, experience in using computers, familiarity with the system, etc.
- *Knowledge level*: The student's knowledge of the teaching subject constitutes the most important student characteristic. Well known representation methods are the following: (a) the overlay model, (b) the buggy model, (c) stereotypes, (d) uncertainty models, (e) constraints, which are presented in the remaining part of the section.

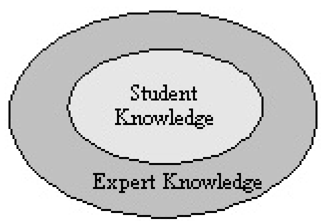


**Figure 2:** The overlay model

The *overlay model* (Polson & Richardson 1988) is the most popular way of representing the student's knowledge. This model is based on the pedagogical structure of the knowledge domain (i.e. knowledge concepts). According to the overlay model, the student's knowledge is considered to be a subset of the expert's knowledge (Figure 2). For each knowledge concept, the model retains a value representing the student's knowledge level. This value can either be boolean (known, not known) or scalar. Based on the overlay model, the system presents educational content until the student's knowledge is identical to the expert's knowledge. The overlay model has been used very often in both ITSs and AEHSs since its simplicity is a great advantage.

A disadvantage of the overlay model is its inability to represent possible student misconceptions. For this purpose, the *buggy model* (Polson & Richardson 1988) is used which represents the student's knowledge as the union of a subset of the domain knowledge and a set of misconceptions (Figure 3). The buggy model provides great assistance in correcting the student's mistakes because recording his/her erroneous knowledge is pedagogically very useful. There are two variations of the buggy model: the *bug catalogue* and the *bug-parts-library* model.

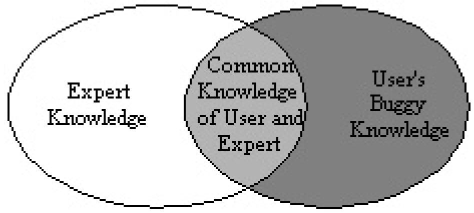


**Figure 3:** The buggy model

In the bug catalogue model there is a library of predefined misconceptions used to insert the student's corresponding misconceptions in the student model. A disadvantage of this model is the difficulty in constructing this library. In case the library is incomplete, the system may misdiagnose the student's misconceptions. In the second variation, the student's misconceptions are created during the teaching process from a library of bug parts. This library usually contains symbolic rules with conditions and actions executed when the conditions are satisfied. Therefore the bug-parts-library is quite smaller and easier to be constructed.

A simpler way of modelling the student's knowledge is to use *stereotypes*. Stereotypes denote predefined classes of users. A stereotype model is represented as a set of pairs 'stereotype-value' where 'value' specifies if the user belongs or not to the specific stereotype. A stereotype is activated and deactivated with the use of triggers. Stereotype models are simpler than the other models and can thus be easier initialized and maintained. Some problems of this model are the difficulty in defining the possible stereotype classes for a specific teaching subject, the difficulty in setting boundaries between different stereotypes and the fact that the simplicity of the model restricts the adaptation power of the intelligent educational systems.

Representation of student knowledge is closely related with student evaluation, i.e. a way to estimate how well a student has learnt specific concepts. Although student evaluation is often used as part of the student modelling unit, we consider more natural to include it in the pedagogical module. Therefore, AI technologies related to student modelling are presented in the next section.

# 5 Pedagogical Module

The pedagogical module represents aspects of the teaching process. It provides the knowledge infrastructure in order to tailor the presentation of teaching content according to the information contained in the student model. The main pedagogical tasks it is called to represent and handle are: *course/lesson plan construction*, *selection of teaching strategy*, *selection of teaching content, student support* and *student evaluation* (see Figure 4). There are various technologies and techniques used to accomplish the above tasks, coming from both AI and AH.

## 5.1 Pedagogical Tasks

A primary task of the pedagogical module is to construct a plan of the course/lesson to be presented to the student, related to the learning goal(s) selected by the student. A course/lesson plan actually consists of an ordered list of concepts. To construct a plan the concept structure and the student model are used. Due to the existence of various relations between concepts, alternate plans, teaching the same learning goal(s), can be derived for different students.

Another task of the pedagogical module is the selection of the instructional strategy for each student. It is important for an IES to offer more than one tutoring strategy, because it gives flexibility. So, there should be an effective mechanism for selecting the appropriate strategy. This is related to the learning style of the student, but it may be necessary to change at a given point of the teaching process. Selection is based on various factors (Tong & Angelides 1999) such as student's advancement, student's preferences, changes in tutorial material, etc.

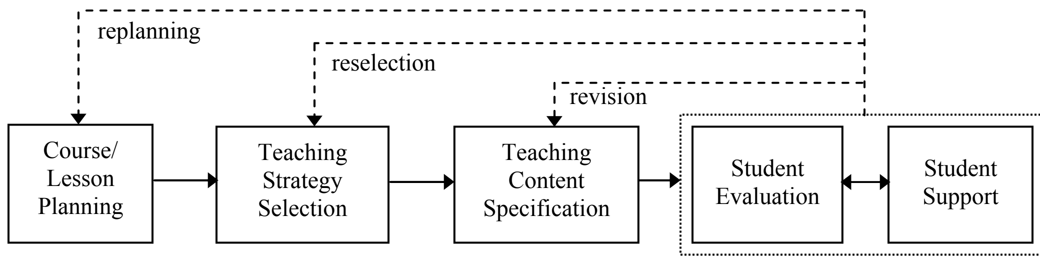


**Figure 4:** Pedagogical Tasks Control Flow

Each concept is related to more than one learning unit. Based on the constructed plan and the selected strategy, the learning units are selected, ordered and presented to the student. For this purpose, the characteristics of the student model (e.g. presentation preferences) as well as the meta-description of the learning units are taken into account. In order to increase the system's teaching effectiveness, construction of the plan as well as selection and ordering of learning units should not be static but updated according to the student's performance.

Another important aspect of the pedagogical model is to evaluate the student's performance based mainly on the answers to questions or exercises. In this way, the student model is updated and presentation of the educational content is affected accordingly. The system should be able to identify on the one hand what is wrong or incomplete in the student's answers and on the other hand the missing knowledge or misconception causing the error. The system may be able to analyze the final solutions to the problems, each individual solution step, or several solution steps and offer student-adapted assistance accordingly. Student support can vary from mere hints to the complete solution to the problem. In case of student errors, remedial tutoring may be necessary causing global or local changes to the course/lesson plan (Vassileva 1997).

## 5.2 Technologies and Techniques

### *5.2.1 From Artificial Intelligence*

There are a number of technologies and techniques that are used in WBIESs, coming from both AI and AH, to achieve realization of the basic pedagogical tasks. ITSs use AI technologies and techniques. Knowledge representation and reasoning (KR&R) is of great importance here, since what is mainly needed is representation of human reasoning. Well-known such techniques, are rule-based reasoning and case-based reasoning. Also, connectionist techniques like neurocomputing are used for representing classification tasks.

*Rule-based reasoning*, usually called the *expert systems approach*, is one of the most popular KR&R methods (Negnevitsky, 2002, ch. 2). Rules represent general domain knowledge in the form of if-then rules: if <conditions> then <conclusion>, where <conditions> represents the conditions and <conclusion> the conclusion of a rule. The conclusion of a rule is derived when the logical function connecting its conditions results to true. The main parts of a typical rule-based expert system are: rule base, inference engine, working memory and explanation mechanism. The inference engine uses the knowledge in the rule base as well as facts about the problem at hand to draw conclusions. The explanation mechanism provides explanations about drawn conclusions. Rules are often used in most pedagogical tasks (Vassileva, 1998), (Simic & Devedzic, 2003).

*Case-based reasoning* (Leake, 1996) store a large set of past cases with their solutions in the case base and use them whenever a similar new case has to be dealt with. A case-based system performs inference in four phases: (i) retrieve, (ii) reuse, (iii) revise and (iv) retain. In the retrieval phase the most relevant stored case(s) to the new case is(are) retrieved. Similarity measures and indexing schemes are used in this context. In the reuse phase, the retrieved case is combined with the new case, to create a solution. The revise phase validates the correctness of the proposed solution. Finally, the retain phase decides on retention (or not) of the new case. Case-based reasoning has been used for instructional tasks (Gilbert, 2000), (Guin-Duclosson, 2002).

*Neural networks* represent a totally different approach to AI, known as *connectionism* (Gallant, 1993). A neural network consists of many simple interconnected processing units called neurons. Each connection from neuron $u_j$ to neuron $u_i$ is associated with a numerical weight $w_{ij}$ corresponding to the influence of $u_j$ to $u_i$. The output of a neuron is based on its inputs and corresponding weights. Usually, neural networks are organized in three levels: input, intermediate (or hidden) and output level. The weights of a neural network are determined via a training process via empirical data. Input neurons are fed with the input values of the problem. These values are propagated through the network and produce the outputs by activating the corresponding neurons. The system in (Tchetagui & Nkambou, 2002) employs a neural network to classify the student into a knowledge level.

Sometimes *uncertainty models* are employed to model student's knowledge and evaluation. Such models use Bayesian networks (Jameson 1995) or fuzzy logic (Nkambou 1999). *Bayesian networks* (or *probabilistic nets*) are graphs, where nodes represent statistical concepts and links represent mainly causal relations between them. Each link is assigned a probability, which represents how certain is that the concept where the link departs from causes (leads to) the concept where the link arrives at. *Fuzzy expert systems* constitute the most popular application of fuzzy logic. In such systems, sets of *fuzzy rules* are used to infer conclusions based on input data. Fuzzy rules include fuzzy variables. Inference process includes three phases: fuzzification of inputs (via membership functions), application of fuzzy rules and defuzzification (to produce the output).

*Constraint-based modelling* (Mitrovic 1999) is also a representation scheme suited for student knowledge and evaluation. A constraint indirectly represents the solutions violating the knowledge domain. The user’s knowledge is represented as a set of constraints that he/she violates or not.

### *5.2.2 From Adaptive Hypermedia*

On the other hand, *adaptive presentation* and *adaptive navigation* techniques are important technologies coming from AH, which we briefly present in the following. Adaptive presentation adapts the contents of an educational page to student model characteristics. A popular technique of adaptive presentation is known as *additional explanations* (Brusilovsky, Kobsa & Vassileva 1998). In this method, the various pieces of information constituting the learning units are associated with conditions. When conditions are satisfied, the corresponding information is presented. In this way, some information, which is incompatible with the student's characteristics, is hidden from him/her. This means that some students will obtain additional information compared to others.

Another popular method of adaptive presentation concerns the *explanation variants* (Brusilovsky, Kobsa & Vassileva 1998) that stores variations of the educational content and selects the most suitable one according to the user model. The method can be implemented in two ways. The simplest way of implementing it is to use *page variants*. In this case, the system retains variants of the same page with different presentations for the same subject. A more specialized way of implementing the 'explanation variants' method is to use *fragment variants*. This specialization is useful when an educational page refers to more than one knowledge concepts. These two ways can be combined in order to enable adaptation according to multiple user characteristics. For instance, the appropriate page can be selected from the page variants according to the user's background and the suitable fragments according to the user's knowledge level.

Adaptive navigation assists users in navigating the hyperspace of the educational system by adapting the page hyperlinks to characteristics of the user model. Its goal is to find 'optimal paths' through the learning material (Brusilovsky 1996), i.e. course/lesson planning. The usual techniques of adaptive navigation are the following:

- *Direct guidance*. It is the simplest adaptive navigation method. It shows the next best learning unit to access. It is best to use direct guidance in conjunction with other adaptive navigation methods so that the user will have more freedom in his navigation.
- *Link sorting*. The links of a specific page are sorted in decreasing relevance order.
- *Link annotation*. According to the appropriateness of the corresponding pages, links are annotated using link colors, icons, etc.
- *Link hiding, removal, disabling*. Links presumed to be of low interest are hidden and presented as simple text, totally removed or disabled.

# 6 Key Trends and Technologies for Future WBIESs

Current WBIESs do not actually combine techniques from both AI and AE in a strong way. Most of existing systems are either ITSs that have been transported into web-based ITSs or AEHSs that have incorporated some kind of "light" intelligent techniques (Brusilovsky & Paylo, 2003). So, a first key trend for future WBIESs seems to be the development of systems with real integration of intelligent and adaptive technologies/techniques. For example, systems that use adaptive presentation and rule-based or case-based reasoning for teaching content specification.

Recently, hybrid KR&R techniques have started to be used in WBIESs. Hybrid KR&R techniques combine more than one KR&R technique and offer a number of advantages in developing WBIESs (Hatzilygeroudis and Prentzas, 2004), (Frias-Martinez et al, 2004). For example, formalisms integrating rules and neural networks, like neurules, are used in representing human reasoning in the pedagogical module (Prentzas, Hatzilygeroudis & Garofalaikis, 2002). Also, neurofuzzy techniques are used for content planning (Magoulas et al, 2001). So, use of hybrid KR&R techniques seems to be a key trend in future WBIESs.

Agent-based technology is also a key technology for future WBIESs. It is generally admitted that agents technology is very important in the context of WWW, but, although it is widely used in other application domains, like e-commerce, its use in web-based education is rather limited. However, agents can offer great flexibility and make dynamic adaptation feasible (Kabassi and Virvou, 2003). Furthermore, pedagogical agents (i.e. human-like artificial characters) would play an important role in this direction (Chou et al, 2003).

Finally, semantic web-based intelligent educational systems (SWBIESs) will be a new category of educational systems. Semantic Web can offer semantic interoperability between different applications. A key technology for Semantic Web towards that direction is ontologies (Staab and Studer, 2004). An ontology formally and declaratively provides machine-interpretable definitions of the basic concepts in a domain and their relations. Given that, domain and pedagogical knowledge, for example, can be shared between different educational systems. Key low-level languages for representing ontologies are XML, XML Schema, RDF and RDF Schema (recommended by the W3C), whereas a higher-level one is OWL (Antoniou & van Harmelen, 2004). Given that OWL is based on a description logic (DL), DL-based reasoning will play an important role in SWBIESs (Krdzavac, Gašević & Devedžić, 2004)

# 7 Conclusions

In this paper, we briefly present the main technologies and techniques used in WBIESs. We define existing WBIESs as being either web-based ITSs or AEHSs incorporating intelligent techniques. So, the technologies and techniques come from both AI and AE and are presented in the context of the basic components of a WBIES. At the end, we specify some possible key trends and technologies for future WBIESs.

# References


Antoniou, G., & van Harmelen, F., (2004). A Semantic Web Primer. MIT Press.

Brusilovsky, P., Kobsa, A., Vassileva, J. (Eds.) (1998). Adaptive Hypertext and Hypermedia. Kluwer Academic Publishers.

Brusilovsky, P. and Paylo, C. (2003). Adaptive and Intelligent Web-Based Educational Systems. *International Journal of AI in Education* 13, 156-169.

Chou, C-Y., Chan, T-W., & Lin, C-J., (2003). Redefining the learning companion: the past, present and future of educational agents. Computers & Education, 40, 255-269.

Frias-Martinez, E., Magoulas, G., Chen, S., & Macredie, R. (2004). Recent Soft Computing Approaches to User Modeling in Adaptive Hypermedia. In De Bra, P., & Nejdl, W. (Eds), Adaptive Hypermedia and Adaptive Web-Based Systems, Third International Conference, AH 2004, Eindhoven, The Netherlands, Aug. 23-26. Proceedings. LNCS, Vol. 3137, 104-114.

Gallant, S.I. (1993). Neural Network Learning and Expert Systems. MIT Press.

Gilbert, J.E. (2000). Case-Based Reasoning Applied to Instruction Method Selection for Intelligent Tutoring Systems. Proceedings of the ITS'2000 Workshop on Case-Based Reasoning in Intelligent Training Systems, 11-15.

Guin-Duclosson, N., Jean-Danbias, S., & Norgy, S. (2002). The AMBRE ILE: How to Use Case-Based Reasoning to Teach Methods. In Cerri, S.A., Gouarderes, G., Paraguacu, F. (eds.): Sixth International Conference on Intelligent Tutoring Systems. Lecture Notes in Computer Science, Vol. 2363. Springer-Verlag, Berlin, 782-791.

Hatzilygeroudis I. (Guest Editor) (2004). Preface. Special Issue on AI Techniques in Web-Based Educational Systems. International Journal on AI Tools (IJAIT), 13(2).

Hatzilygeroudis, I., & Prentzas, J. (2004). Knowledge Representation Requirements for Intelligent Tutoring Systems. In Lester, J. C., Vicari, R. M. and Paraguacu, F. (eds.): 7th International Conference on Intelligent Tutoring Systems (ITS 2004). Proceedings. Lecture Notes in Computer Science, Vol. 3220. Springer-Verlag, Berlin, 87-97.

Jameson, A. (1995). Numerical Uncertainty Management in User and Student Modeling: An Overview of Systems and Issues. User Modeling and User-Adapted Interaction, 5, 193-251.

Kabassi, K., & Virvou, M., (2003). Using Web Services for Personalized Web-Based Learning. Educational Technology & Society, 6(3), 61-71.

Krdzavac, N., Gašević, D., & Devedžić, V., (2004). Description Logics Reasoning in Web-Based Education Environments. International Workshop on Adaptive Hypermedia and Collaborative Web-based Systems (AHCW'04) (http://www.ii.uam.es/~rcarro/AHCW04/).

Leake, D.B. (ed.) (1996). Case-Based Reasoning: Experiences, Lessons & Future Directions. AAAI Press/MIT Press.

Magoulas, G.D., Papanikolaou, K.A., & Grigoriadou, M. (2001). Neuro-fuzzy Synergism for Planning the Content in a Web-based Course. Informatica, 25, 39-48.

Mitrovic, A. (1999). Evaluation of a Constraint-Based Tutor for a Database Language. Journal of Artificial Intelligence in Education, 10, 238-256.

Negnevitsky, M. (2002). Artificial Intelligence, A Guide to Intelligent Systems. Addison Wesley.

Nkambou, R. (1999). Managing Inference Process in Student Modeling for Intelligent Tutoring Systems. In Proceedings of the Eleventh IEEE International Conference on Tools with Artificial Intelligence. IEEE Computer Society Press, 19-23.

Polson, M.C., Richardson, J.J. (1988). Foundations of Intelligent Tutoring Systems. Lawrence Erlbaum Associates.

Prentzas, J., Hatzilygeroudis, I., & Garofalakis J. (2002). A Web-Based Intelligent Tutoring System Using Hybrid Rules as its Representational Basis. In Cerri, S.A., Gouarderes, G., Paraguacu, F. (eds.): Sixth International Conference on Intelligent Tutoring Systems (ITS-2002). Lecture Notes in Computer Science, Vol. 2363. Springer-Verlag, Berlin, 119-128.

Siemer, J., Angelides, M. (1998). A comprehensive method for the evaluation of complete intelligent tutoring systems. Decision Support Systems, 22, 85-102.

Simic, G., & Devedzic, V. (2003). Building an Intelligent System Using Modern Internet Technologies. Expert Systems with Applications, 25, 231-146.

Staab, S., & Studer, R. (Eds) (2004). Handbook on Ontologies. Springer-Verlag.

Tchetagui, J. M. P., & Nkambou R. (2002). Hierarchical Representation and Evaluation of the Students in an Intelligent Tutoring System. In Cerri, S.A., Gouarderes, G., & Paraguacu, F. (eds.): Sixth International Conference on Intelligent Tutoring Systems (ITS-2002). Lecture Notes in Computer Science, Vol. 2363. Springer-Verlag, Berlin, 708-717.

Tong, A., Angelides, M. (1999). Formalizing tutoring strategy selection in multimedia tutoring systems". Information and Software Technology, 41, 67-90.

Vassileva, J. (1997). Dynamic Course Generation. Journal of Computing and Information Technology, 5, 87-102.