Adaptive and Intelligent Technologies for Web-based Education

Peter Brusilovsky

Carnegie Technology Education and
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
plb@cs.cmu.edu

Abstract: The paper provides a review of adaptive and intelligent technologies in a context of Web-based distance education. We analyze what kind of technologies are available right now, how easy they can be implemented on the Web, and what is the place of these technologies in large-scale Web-based education.

1 Introduction

Web-based education (WBE) is currently a hot research and development area. Benefits of Web-based education are clear: classroom independence and platform independence. Web courseware installed and supported in one place can be used by thousands of learners all over the world that are equipped with any kind of Internet-connected computer. Thousands of Web-based courses and other educational applications have been made available on the Web within the last five years. The problem is that most of them are nothing more than a network of static hypertext pages. A challenging research goal is the development of advanced Web-based educational applications that can offer some amount of adaptivity and intelligence. These features are important for WBE applications since distance students usually work on their own (often from home). An intelligent and personalized assistance that a teacher or a peer student can provide in a normal classroom situation is not easy to get. In addition, being adaptive is important for Web-based courseware because it has to be used by a much wider variety of students than any "standalone" educational application. A Web courseware that is designed with a particular class of users in mind may not suit other users.

Since the early days of the Web, a number of research teams have implemented different kinds of adaptive and intelligent systems for on-site and distance WBE. The goal of this paper is to provide a brief review of the work performed so far in his area. The review is centered on different adaptive and intelligent *technologies*. We stay on the level of technologies to provide compatibility with earlier papers on adaptive hypermedia [7] and Web-based ITS [6]. By adaptive and intelligent technologies we mean essentially different ways to add adaptive or intelligent functionality to an educational system. A technology usually could be further dissected into finer grain techniques and methods, which corresponds to different variations of this functionality and different ways of its implementation. In the next section we analyze what kind of technologies are available right now, and how easy they can be implemented on the Web. After that we discuss what is the place of these technologies in large-scale Web-based education.

2 Web-based educational systems: a review of technologies

Web-based Adaptive and Intelligent Educational Systems (AIES) are not an entirely new kind of systems. Historically, almost all Web-based AIES inherit from two earlier kinds of

AIES: *intelligent tutoring systems* (ITS) and *adaptive hypermedia systems*. Most of adaptive and intelligent technologies applied in Web-based AIES systems were directly adopted from either the ITS area or the adaptive hypermedia area. As long as Web-based AIES research get more mature, it will produce original technologies inspired by the Web context. At least one of these Web-inspired technologies could already be identified (model matching). This section provides a review of existing technologies grouped by its origin. For each technology we list existing Web-based AIES and projects, which implements variations of this technology and discuss the ways to implement it on the Web.

2.1 ITS technologies in Web-based education

Intelligent tutoring systems is a traditional area of research that investigates problems of developing AIES [13]. The goal of various ITS is the use the knowledge about the domain, the student, and about teaching strategies to support flexible individualized learning and tutoring. A review of existing intelligent tutoring systems performed by the author in 1990 helped to identify three core ITS technologies: curriculum sequencing, intelligent analysis of student's solutions, and interactive problem solving support. All these technologies were implemented in numerous ITS. Since 1990, only one new technology (example-based problem solving support) was added to the set to classify a functionality that was not covered by the core three. While the proposed set of ITS technologies could be considered subjective and incomplete, it turned out to be very useful for classifying existing Web-based AIES. Web-based AIES that use traditional ITS technologies are usually called Web-based ITS. First Web-based ITS were reported in 1995-1996 [6; 12; 34; 37]. These systems still constitute a rather small stream inside the ITS area.

2.1.1 Curriculum sequencing

The goal of the *curriculum sequencing* technology (also referred to as instructional planning technology) is to provide the student with the most suitable individually planned sequence of knowledge units to learn and sequence of learning tasks (examples, questions, problems, etc.) to work with. In other words, it helps the student to find an "optimal path" through the learning material. The classic example is the BIP system [5]. There are two essentially different kinds of sequencing: active and passive. Active sequencing implies a *learning goal* (a subset of domain concepts or topics to be mastered). Systems with active sequencing can build the best individual path to achieve the goal. Passive sequencing (which is also called *remediation*) is a reactive technology and does not require an active learning goal. It starts when the user is not able to solve a problem or answer a question (questions) correctly. Its goal is to offer the user a subset of available learning material, which can fill the gap in student's knowledge of resolve a misconception. For active sequencing systems, it makes sense to distinguish systems with fixed and adjustable learning goal. Most of existing systems can guide their students to the fixed learning goal - the whole set of domain concepts. A few systems with adjustable learning goal let a teacher or a student to select a subset of the whole set of concepts as the current learning goal. In most of ITS systems with sequencing it is possible to distinguish two levels of sequencing: high and low. High-level sequencing or *knowledge sequencing* determines next learning subgoal: next concept, set of concepts, topic, or lesson to be taught. Low-level sequencing or *task sequencing* determines next learning task (problem, example, test) within current subgoal. High and low level sequencing are often performed by different mechanisms. In many ITS systems only one of these two mechanisms are intelligent, for example, a lesson is selected by a student, while learning tasks within this lesson are adaptively selected by the system. Some systems can only manipulate the order of task of one particular kind: usually problems or questions. In this case it could be also called problem or question sequencing.

Sequencing is currently the most popular technology in Web-based AIES. Almost all kinds of sequencing mentioned above were already implemented on the Web. Active sequencing is a dominated type of sequencing. Only a few systems (InterBook, PAT-InterBook, CALAT, VC

Prolog Tutor, and Remedial Multimedia System) can perform passive remedial sequencing. Among active sequencing systems, only a handful of systems such as ELM-ART-II, AST, ADI, ART-Web, ACE, KBS-Hyperbook, and ILESA are able to perform intelligently both high and low level sequencing. Others, like Manic, leave a choice of activity within a topic to the user. Vice versa, some systems, like Medtec, leave a choice of a topic to the user but can generate an adaptive sequence of problems within the topic. Most of the systems supports sequencing with fixed learning goal (equals to the whole course). Only a few systems support adjustable learning goals enabling a teacher (as in DCG) or a student (as in InterBook and KBS Hyperbook) to select an individual goal. The student can choose a goal as a subset of domain concepts (InterBook) or a project (KBS Hyperbook).

Active sequencing in most of the systems is driven by the students knowledge (more exactly, by the difference between student's knowledge and global goal). A few systems and projects, however, experiment with the use of students' preferences on the type and media of available learning material to drive sequencing of tasks within a topic [14; 15; 45]. Two interesting cases of sequencing could be found in DCG and SIETTE systems. DCG [49] can perform advanced sequencing of educational material adapted to a learning goal. However, the sequencing is performed before students start working with the system producing a static Web-based course. SIETTE [40] is an example of a Web-based adaptive testing system. The only kind of learning material it possesses is questions. The only thing it can do is to generate an adaptive sequence of questions to assess student's knowledge. Systems like SIETTE are incomplete by their nature and have to be used as components in distributed Web-based AIES.

While curriculum sequencing could be considered as the oldest ITS technology (it was implemented in almost all first ITS), for about 20 years it was a Cinderella among other technologies. Very little attention was devoted to it. Mainstream ITS research were centered around problem solving support technologies (which will be analyzed below). Problem solving support was considered as a main duty of an ITS, while delivery and sequencing of education material was though to be performed outside the system (usually, by a human teacher). Naturally, almost no ITS includes educational material itself (other than a set of problems). The situation with Web-based AIES is very different. In the context of Web-based education a solid amount of educational material (usually structured as a hyperspace) is one of the main attractions of an educational system. In this context (with its "lost in hyperspace" problem), curriculum sequencing technology becomes very important to guide the student through the hyperspace of available information. This technology is also natural and easy to implement on the Web: all knowledge could be located on the server and all sequencing could be done by a CGI-script. It's not surprising that, it is not only the oldest, but also the most popular technology of Web-based AIES.

2.1.2 Problem solving support technologies

As it is mentioned above, for many years, problem solving support was considered as a main duty of an ITS system and a main value of an ITS technology. We have identified three problem solving support technologies: intelligent analysis of student solutions, interactive problem solving support, and example-based problem solving support. All these technologies can help a student in a process of solving an educational problem, but they do it by different ways.

Intelligent analysis of student solutions deals with students' final answers to educational problems no matter how these answers were obtained. To be considered as intelligent, a solution analyzer has to decide whether the solution is correct or not, find out what exactly is wrong or incomplete, and possibly identify which missing or incorrect knowledge may be responsible for the error (the last functionality is referred as knowledge diagnosis). Intelligent analyzers can provide the student with extensive error feedback and update the student model. The classic example is PROUST [Johnson, 1986 #681. As it could be seen from the Tables 1 and 3, a number of Web-based AIES implement intelligent analysis of student solutions.

Interactive problem solving support is a more recent and a move powerful technology. Instead of waiting for the final solution, this technology can provide a student with intelligent help on each step of problem solving. The level of help can vary: from signaling about a wrong step, to giving a hint, to executing the next step for the student. The systems which implement this technology (often referred to as *interactive tutors*) can watch the actions of the student, understand them, and use this understanding to provide help and to update the student model. The classic example is the LISP-TUTOR [2]. This technology is also represented by a number of Web-based AIES (Tables 1 and 3).

The example-based problem solving technology is the newest one. This technology is helping students to solve new problems not by articulating their errors, but by suggesting them relevant successful problem solving cases from their earlier experience (it could be examples explained to them or problems solved by them earlier). An example is ELM-PE [51]. In the Web context, this technology is implemented in ELM-ART [12] and ELM-ART-II [53].

In the area of traditional ITS, the interactive problem solving support technology absolutely dominates. Interactive problem solving support is an ultimate goal of almost any ITS, while intelligent analysis of student solutions is often considered imperfect (and example based problem solving support is too rare to consider as a competitor). Again, the Web context changes the situation. Both intelligent analysis of student solutions and example based problem solving support appears to be very natural and useful in Web context. Both technologies are passive (works by student request) and can be relatively easy implemented on the Web using a CGI interface. Moreover, an old standalone AIES, which uses these technologies, could be relatively easy ported to the Web by implementing a CGI gateway to the old standalone program. It is not surprising that these technologies were among the first implemented on the Web. An important benefit of these two technologies in the Web context is their low interactivity: both usually require only one interaction between browser and server for a problem solving cycle. This is very important for the case of slow Internet connection. These technologies can provide intelligent support when a more interactive technology will be hardly useful. Currently, these technology dominates in Web context over more powerful and interaction hungry interactive problem solving support.

Interactive problem solving support technology is the last ITS technology migrated to the Web. The problem here is that the "fast-track" approach of implementing Web-based ITS (developing a CGI interface to an older standalone ITS) used in pioneer systems does not work properly for this technology. It could be well illustrated by the PAT-Online system [41], which was probably the first trial to implement interactive problem solving support on the Web. This system uses a form-based CGI-AppleScript interface to a standalone Practical Algebra Tutor (PAT) system. Since CGI interface is passive, the Web version of the system had to provide a "submit" button for the student to get the feedback from the system. Naturally, it also added another feature, which was essential for students with a slow Internet connection: a possibility to request a feedback once after performing several problem solving steps. As a result, PAT-Online moved to the category of an intelligent problem analyzers, more exactly, to a subcategory of analyzers that are capable to analyze incomplete solutions (ELM-ART also belongs to this subcategory). The intelligent analyzers of this subcategory can be placed between traditional analyzers and interactive tutors (in Tables 1 and 3 they are marked with keyword "partial", however, they can't be considered as real interactive tutors).

A real interactive tutor is expected to be not only interactive, but also active. It should not sleep from one help request to another, but instead should be able to monitor what the student is doing and instantly react to errors. It simply can't be implemented with the traditional server-side CGI interactivity and requires client-side interactivity based on Java. Java technology has matured very recently. Two years ago the review [8] named it as a prospective platform for Web-based AIES and mentioned only three Java-based systems. Now Java provides a reliable solution for Web-based interactive tutors. To be more exact, Java offers two different solutions. One solution is a tutor implemented completely in Java. It could be a Java applet working in a browser, or a Java application. Another solution is a distributed client-server tutor where a part of functionality is implemented in Java and works on the client side, and another part works on the server side. The parts communicate over the Internet. While the pure Java solution looks simpler (just a new language to build an AIES), the client-server architecture offers a more attractive choice for developing Web-based tutors. It is a definite choice for porting a standalone interactive tutor on the Web. D3-WWW-Trainer [20] and AlgeBrain [1] demonstrate how to re-use the intelligent functionality of an earlier standalone tutor by changing it to a server-side application and developing a relatively thin "brainless" Java client that implements interface functions and communicates with an intelligent server. Event relatively small newly implemented interactive tutors such as ADIS [50] and ILESA [30], which could be easily implemented in pure Java, can benefit from client-server architecture for such reasons as central student modeling. Finally, an overhead of the client-server approach (the need to have a distributed system) is not very big since Java naturally supports several ways of client-server communications - HTTP/CGI, sockets, or RMI/CORBA. We think, that the client-server architecture will become very popular in the coming years as a standard way of implementing Web-based interactive tutors and a way to implement all kinds of highly interactive Web-based AIES. We already see examples of using it for implementing pen-based interface in WITS-II [27] and an animated pedagogic agent Vincent in TEMAI [38].

2.2 Adaptive hypermedia technologies in Web-based education

Adaptive hypermedia is a relatively new research area [7]. Adaptive hypermedia systems apply different forms of user models to adapt the content and the links of hypermedia pages to the user. We distinguish two major technologies in adaptive hypermedia: adaptive presentation and adaptive navigation support. Education always was one of the main application areas for adaptive hypermedia. A number of standalone (i.e., non-Web-based) adaptive educational hypermedia systems was built between 1990 and 1996. First Web-based AIES that use adaptive hypermedia technologies were reported in 1996 [12; 17]. Since that the Web has become the primary platform for developing educational adaptive hypermedia systems.

The goal of the *adaptive navigation support technology* is to support the student in hyperspace orientation and navigation by changing the appearance of visible links. Adaptive navigation support (ANS) can be considered as a generalization of curriculum sequencing technology in a hypermedia context. It shares the same goal - to help students to find an "optimal path" through the learning material. At the same time, adaptive navigation support has more options than traditional sequencing: it can guide the students both directly and indirectly. In a WWW context where hypermedia is a basic organizational paradigm, adaptive navigation support can be used very naturally and efficiently. There are several known ways to adapt the links [7]. Two examples of ANS-based standalone systems are ISIS-Tutor [10] with adaptive hiding and adaptive annotation and Hypadapter [24] with adaptive hiding and adaptive sorting. The three ways that are most popular in Web-based AIES are direct guidance, adaptive link annotation, and adaptive link hiding.

Direct guidance implies that the system informs the student which of the links on the current page will drive him or her to the "best" page in the hyperspace (which page is "best" is decided on the basis of student's current knowledge and learning goal). Often, if a link to the next best page is not presented on the current page, the system can generate a dynamic "next" link. As we can see, adaptive navigation support with direct guidance is almost equivalent to curriculum sequencing technology. There are some differences though (in addition to the different origin). A page suggested by a direct guidance technology is always a page of the existing hyperspace. The student usually could reach this page in one or several steps without the system guidance. The guidance just helps the student to realize that this page is "best" and to get there fast. In an ITS with adaptive sequencing a "page" with next best task or presentation could be completely generated from system's knowledge, thus the student has no ways to get to this material others than using sequencing. Also, direct guidance usually applies a one level sequencing mechanism (in comparison with two-level sequencing in most ITS): the best page is simply selected from the set of acceptable pages using some heuristics. We refer to this way of sequencing as page sequencing. InterBook and ELM-ART provide good examples of this technology. However, the difference between these two technologies starts to disappear in the Web context. Web-based ITS systems are naturally moving to hypermedia platform representing at least some part of the learning material as a hyperspace. As long as some type of educational material (presentations, problems, and questions) is represented as a set of nodes in hyperspace, sequencing of it becomes indistinguishable from direct guidance. To stress this similarity we have represented adaptive sequencing and adaptive navigation support with direct guidance in the same column of the tables.

The most popular form of ANS on the Web is annotation. It was used first in ELM-ART [12] and since that applied in all descendants of ELM-ART such as InterBook, AST, ADI, ACE, and ART-Web as well as in some other systems such as WEST-KBNS and KBS HyperBook. ELM-ART and InterBook also use adaptive navigation support by sorting. Another popular technology is hiding and disabling (a variant of hiding that keeps link visible but does not let the user to proceed to the page behind the link if this page is not ready to be learned). The options are either to make the link completely non-functional (nothing happens when the user clicks on it) as implemented, for example, the Remedial Multimedia System [4] or to show the user a list of pages to be read before the goal page as done in Albatros [29]. Tables 1 and 2 list all major systems that use adaptive navigation support and indicates the type of adaptation.

The goal of *the adaptive presentation technology* is to adapt the content of a hypermedia page to the user's goals, knowledge and other information stored in the user model. In a system with adaptive presentation, the pages are not static, but adaptively generated or assembled from pieces for each user. For example, with several adaptive presentation techniques, expert users receive more detailed and deep information, while novices receive more additional explanation. Adaptive presentation is very important in WWW context where the same "page" has to suit to very different students. Only two Web-based AES implement full-fledged adaptive presentation: PT [28] and AHA [16]. Both these systems apply a flexible but low-level conditional text technique. Some other systems use adaptive presentation is special contexts. Medtec [19] is able to generate adaptive summary of book chapters. MetaLinks can generate a special preface to a content page depending on where the student came from to this page. ELM-ART, AST, InterBook and other descendants of ELM-ART use adaptive presentation to provide adaptive insertable warnings about the educational status of a page. For example, if a page is not ready to be learned, ELM-ART and AST insert a textual warning at the end of it and InterBook inserts a warning image in a form of a red bar. A very interesting example of adaptive presentation is suggested in WebPersona project [3] where an individualized presentation of information in an educational hypertext is performed by a life-like agent.

2.3 Web-inspired technologies in Web-based education

The last group of technologies is probably the most exciting one since these technologies has almost no roots in pre-internet educational systems. Currently this group include only one technology. We call this technology *student model matching* (or simply model matching) because the essence of this technology is the ability to analyze and match student models of many students at the same time. Traditional adaptive and intelligent educational systems has no opportunity to explore this technology since they usually work with one student (and one student model) at a time. On the contrary, in the WBE context this opportunity happens naturally because student records are usually stored centrally on a server (at least for administrative reasons). It provides an excellent framework for developing various adaptive and intelligent technologies that can make some use of matching student models of different students. So far, we have identified two examples of student model matching, which we call *adaptive collaboration support* and *intelligent class monitoring*. These examples quite differ from each other and probably could be considered as different technologies within the student model matching group.

Adaptive collaboration support is a very new adaptive technology which was developed within last 5 years along with development of networked educational systems. The goal of adaptive collaboration support is to use system's knowledge about different students to form a matching group for different kinds of collaboration. The pioneering non-WBE (i.e., non-Web, or non-educational) examples of adaptive collaboration support are known for already a few years. These examples include forming a group for collaborative problem solving at a proper moment of time [25; 26] or finding the most competent peer to answer a question about a topic (i.e. finding a person with a model showing good knowledge of this topic) [31]. Less than two years ago Brusilovsky [8] predicted that adaptive collaboration support will become a popular technology. This prediction came true almost immediately. Now we can list already several real examples of adaptive collaboration support in WBE context. The group from University of Saskatchevan has extended their original workplace-oriented peer-help technology developed for PHelpS system [21; 31] to the WBE context in their Intelligent Helpdesk system [22]. Another similar system was developed and evaluated in the University of Central Florida [32]. In addition to that, the group in the University of Duisburg known for their pioneering work on adaptive collaboration support [25] have recently suggested a complete framework for implementation of intelligent support techniques for distributed internet-based education. This framework can naturally support their original adaptive collaboration support techniques and provides a framework for exploring other model matching techniques.

Intelligent class monitoring is also based on the ability to compare records of different students. However, instead of searching for a match, it search for a mismatch. The goal is to identify the students who have learning records essentially different from those of their peers. These students may be different from others in many ways. They cold be progressing too fast, or too slow, or simply have accessed much less material than others. In any case, these students need teacher's attention more than others - to challenge those who can, to provide more explanations for those who can't, and to push those who procrastinate. In a regular classroom the teacher can simply track students attendance and activity to find students who need special attention. In a Web-based classroom, the teacher in the best case has only logging data - tables with numbers which are very hard to grasp. At the same time, the need to identify a small subset of students who need help more than others is more important. In WBE context, communication between teacher and students is usually more time consuming and a distance teacher simply can't individually address more than a small subset of the class. The system HyperClassroom [36] provides an interesting example of using fuzzy mechanisms to identify deadlocked students in a WBE classroom. At the time of writing, it is the only example of the intelligent class monitoring technology known to the author.

3 Adaptive and intelligent technologies for large-scale Web-based education

It should be clear to anyone who is familiar with the needs of Web-based education, that adaptive and intelligent technologies can enhance different sides of Web-based educational systems. Adaptive presentation can improve the usability of course material presentation. Adaptive navigation support and adaptive sequencing can be used for overall course control and for helping the student in selecting most relevant tests and assignments. Problem solving support and intelligent solution analysis can significantly improve the work with assignments providing both interactivity and intelligent feedback while taking a serious grading load from the teachers' shoulders. Model matching technologies can enforce both administration of distance courses and communication / collaboration between students and teachers.

From another side, adaptive and intelligent technologies have not found yet their place in "real" virtual classroom, i.e., as a part of real courseware used by hundreds of distance students.

Most of the systems discussed above are typical "lab" systems, which have never been used for teaching real distance classes. The rest of them, a handful of systems mainly from ELM-ART and AHA families, were used in a few relatively small classes. At the same time, none of the dozens of commercial and "university-grown" Web courseware systems that are used in hundreds of real distance courses applies adaptive and intelligent technologies. Does it mean that research and practice in Web-based education area will never merge together?

The position of the author is the following. Web-based education itself is relatively young. Until now different companies producing Web-based education systems were able to compete on the market with their simple non-adaptive systems. However, a number of research level systems have already clearly demonstrated the benefits of adaptive and intelligent technologies. As long as the competition on the market of Web-based educational system will increase, “being adaptive” or “being intelligent” will become an important factor for winning the customers. Traditional Web-based education companies will start to include adaptive and intelligent functionality. Research teams with solid experience in using adaptive and intelligent technologies will found startup companies to bring their technology to the market. The first technologies to be used in commercial systems will probably be sequencing technologies (page sequencing and question sequencing) since they match very well to the current structure of Web-based education systems. Next will come the turn of adaptive navigation support and model matching. Problem-solving support technologies will stay on research level for longer, though we could expect the market debut of small Web-based tutors that are aimed to support teaching a fragment of some subject. I hope that the next five years will show us a number of examples of commercial-level adaptive and intelligent systems as well as many new and exciting developments on the research level.

System	Ref.	Adaptive sequencing	Adaptive navigation support	Problem solving support	Intelligent solution analysis	Adaptive presentation
ELM-ART	[12]	Page	Annotation	Partial	Server	Some
ELM-ART-II	[53]	Course, tests	Annotation	Partial	Server	Some
PAT-InterBook	[11]	Page, remedial	Annotation	Partial	Server	Some
VC Prolog Tutor	[39]	Task, remedial			Server	

Table 1 Adaptive and intelligent technologies in Web-based educational systems that combine adaptive hypermedia and ITS functionality

System	Ref.	Adaptive sequencing	Adaptive navigation support	Adaptive presentation
InterBook	[9]	Page	Annotation	Some
AST	[46]	Course	Annotation	Some
ADI	[43]	Course (knowledge+interests)	Annotation	Some
ART-Web	[52]	Course, tests	Annotation	Some

ACE	[45]	Course (knowledge+interests)	Annotation Hiding	Some
Remedial Multimedia System	[4]	Course, remedial	Hiding	
PT	[28]		Hiding	Yes
AHA	[16]		Annotation Hiding	Yes
WEST-KBNS	[18]		Annotation	
MetaLinks	[33]	Page		Some (intro)
KBS Hyperbook	[23]	Course	Annotation	

Table 2 Adaptive and intelligent technologies in Web-based adaptive hypermedia systems

System	Ref.	Adaptive sequencing	Problem solving support	Intelligent solution analysis
CALAT	[35]	Course, remedial		
Medtec	[19]	Tasks		
Manic	[47]	Topic		
DCG	[49]	Course		
SIETTE	[40]	Question		
ILESA	[30]	Lesson, problems	Server, Java	
PAT-Online	[41]		Partial	Server
PAT-Java	[42]		Java	
WITS	[37]			Server
WITS-II	[27]			Server, Java
Belvedere	[48]		Server, Java	
ADIS	[50]		Server, Java	
(Yang-Akahori)	[54]			Server
D3-WWW-Trainer	[20]		Server, Java	
AlgeBrain	[1]		Server, Java	
ADELE	[44]		Server, Java	
TEMAI	[38]		Partial	Server, Java

Table 3 Adaptive and intelligent technologies in Web-based ITS systems

References

1. Alpert, S. R., Singley, M. K., and Fairweather, P. G.: Deploying Intelligent Tutors on the Web: An Architecture and an Example. International Journal of Artificial Intelligence in Education **10** (1999) 183-197, available online at http://cbl.leeds.ac.uk/ijaied/abstracts/Vol_10/alpert.html
2. Anderson, J. R. and Reiser, B.: The LISP tutor. Byte **10**, 4 (1985) 159-175
3. André, E., Rist, T., and Müller, J.: WebPersona: A Life-Like Presentation Agent for Educational Applications on the World-Wide Web. In: Brusilovsky, P., Nakabayashi, K. and Ritter, S. (eds.) Proc. of Workshop "Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8th World Conference on Artificial Intelligence in Education, Kobe, Japan, ISIR (1997) 78-85, available online at http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Andre/Andre.html
4. Anjaneyulu, K.: Concept Level Modelling on the WWW. In: Brusilovsky, P., Nakabayashi, K. and Ritter, S. (eds.) Proc. of Workshop "Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8th World Conference on Artificial Intelligence in Education, Kobe, Japan, ISIR (1997) 26-29, available online at http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Anjaneyulu.html
5. Barr, A., Beard, M., and Atkinson, R. C.: The computer as tutorial laboratory: the Stanford BIP project. International Journal on the Man-Machine Studies **8**, 5 (1976) 567-596
6. Brusilovsky, P.: Intelligent tutoring systems for World-Wide Web. In: Holzapfel, R. (ed.) Proc. of Third International WWW Conference (Posters), Darmstadt, Fraunhofer Institute for Computer Graphics (1995) 42-45
7. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction **6**, 2-3 (1996) 87-129
8. Brusilovsky, P.: Adaptive educational systems on the World Wide Web. In: Ayala, G. (ed.) Proc. of Workshop "Current Trends and Applications of Artificial Intelligence in Education" at the 4th World Congress on Expert Systems, Mexico City, Mexico, ITESM (1998) 9-16
9. Brusilovsky, P., Eklund, J., and Schwarz, E.: Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems. **30**, 1-7 (1998) 291-300
10. Brusilovsky, P. and Pesin, L.: An intelligent learning environment for CDS/ISIS users. In: Levonen, J. J. and Tukianinen, M. T. (eds.) Proc. of The interdisciplinary workshop on complex learning in computer environments (CLCE94), Joensuu, Finland, EIC (1994) 29-33, available online at http://cs.joensuu.fi/~mtuki/www_clce.270296/Brusilov.html
11. Brusilovsky, P., Ritter, S., and Schwarz, E.: Distributed intelligent tutoring on the Web. In: du Boulay, B. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997) 482-489
12. Brusilovsky, P., Schwarz, E., and Weber, G.: ELM-ART: An intelligent tutoring system on World Wide Web. In: Frasson, C., Gauthier, G. and Lesgold, A. (eds.) Intelligent Tutoring Systems. Lecture Notes in Computer Science, Vol. 1086. Springer Verlag, Berlin (1996) 261-269
13. Burns, H. L. and Capps, C. G.: Foundations of intelligent tutoring systems: An introduction. In: Polson, M. C. and Richardson, J. J. (eds.): Foundations of intelligent tutoring systems. Lawrence Erlbaum Associates, Hillsdale (1988) 1-19
14. Carver, C. A., Howard, R. A., and Lavelle, E.: Enhancing student learning by incorporating student learning styles into adaptive hypermedia. In: Proc. of ED-MEDIA'96 - World Conference on Educational Multimedia and Hypermedia, Boston, MA, AACE (1996) 118-123

15. Danielson, R.: Learning styles, media preferences, and adaptive education. In: Brusilovsky, P., Fink, J. and Kay, J. (eds.) Proc. of Workshop "Adaptive Systems and User Modeling on the World Wide Web" at 6th International Conference on User Modeling, UM97, Chia Laguna, Sardinia, Italy, Carnegie Mellon Online (1997) 31-35, available online at http://www.contrib.andrew.cmu.edu/~plb/UM97_workshop/Danielson.html
16. De Bra, P. and Calvi, L.: AHA! An open Adaptive Hypermedia Architecture. The New Review of Hypermedia and Multimedia **4** (1998) 115-139
17. De Bra, P. M. E.: Teaching Hypertext and Hypermedia through the Web. Journal of Universal Computer Science **2**, 12 (1996) 797-804, available online at http://www.iicm.edu/jucs_2_12/teaching_hypertext_and_hypermedia
18. Eklund, J. and Sawers, J.: Customising Web-based course delivery in WEST® with navigation support. In: Proc. of WebNet'96, World Conference of the Web Society, San Francisco, CA, AACE (1996) 534-535
19. Eliot, C., Neiman, D., and Lamar, M.: Medtec: A Web-based intelligent tutor for basic anatomy. In: Lobodzinski, S. and Tomek, I. (eds.) Proc. of WebNet'97, World Conference of the WWW, Internet and Intranet, Toronto, Canada, AACE (1997) 161-165
20. Faulhaber, S. and Reinhardt, B.: D3-WWW-Trainer: Entwiklung einer Oberfläche für die Netzanwendung. Technische Universität München, München (1997) 31-40
21. Greer, J., McCalla, G., Collins, J., Kumar, V., Meagher, P., and Vassileva, J.: Supporting Peer Help and Collaboration in Distributed Workplace Environments. International Journal of Artificial Intelligence in Education **9** (1998) 159-177, available online at http://cbl.leeds.ac.uk/ijaied/abstracts/Vol_9/greer.html
22. Greer, J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., and Vassileva, J.: The intelligent helpdesk: Supporting peer-help in a university course. In: Goettl, B. P., Halff, H. M., Redfield, C. L. and Shute, V. J. (eds.) Intelligent Tutoring Systems. Lecture Notes in Computer Science, Vol. 1452. Springer-Verlag, Berlin (1998) 494-503
23. Henze, N., Naceur, K., Nejdl, W., and Wolpers, M.: Adaptive hyperbooks for constructivist teaching. Künstliche Intelligenz , 4 (1999) 26-31
24. Hohl, H., Böcker, H.-D., and Gunzenhäuser, R.: Hypadapter: An adaptive hypertext system for exploratory learning and programming. User Modeling and User-Adapted Interaction **6**, 2-3 (1996) 131-156
25. Hoppe, U.: Use of multiple student modeling to parametrize group learning. In: Greer, J. (ed.) Proc. of AI-ED'95, 7th World Conference on Artificial Intelligence in Education, Washington, DC, AACE (1995) 234-249
26. Ikeda, M., Go, S., and Mizoguchi, R.: Opportunistic group formation. In: Boulay, B. d. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997)
27. Ito, J., Okazaki, Y., Watanabe, K., Kondo, H., and Okamoto, M.: Pen based user interface for an ITS on WWW client. In: Proc. of ICCE'98, Beijing, China, AACE (1998) 324-327
28. Kay, J. and Kummerfeld, B.: User models for customized hypertext. In: Nicholas, C. and Mayfield, J. (eds.): Intelligent hypertext: Advanced techniques for the World Wide Web. Lecture Notes in Computer Science, Vol. 1326. Springer-Verlag, Berlin (1997)
29. Lai, M.-C., Chen, B.-H., and Yuan, S.-M.: Toward a new educational environment. In: Proc. of 4th International World Wide Web Conference (1995), available online at http://www.w3.org/pub/Conferences/WWW4/Papers/238/
30. López, J. M., Millán, E., Pérez-de-la-Cruz, J. L., and Triguero, F.: ILESA: a Web-based Intelligent Learning Environment for the Simplex Algorithm. In: Alvegård, C. (ed.) Proc. of CALISCE'98, 4th International conference on Computer Aided Learning and Instruction in Science and Engineering, Göteborg, Sweden (1998) 399-406

31. McCalla, G. I., Greer, J. E., Kumar, V. S., Meagher, P., Collins, J. A., Tkatch, R., and Parkinson, B.: A peer help sytem for workplace training. In: Boulay, B. d. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997) 183-190
32. Morelos-Borja, H.: Effects of Automatic Assignment of Peers to Tutor/Tutee Roles in Web-based Distance Education, Report, School of Computer Science, University of Central Florida (1998) available online at http://www.cs.ucf.edu/~morelos/papers/Fall98TechReport.rtf
33. Murray, T., Condit, C., and Haugsjaa, E.: MetaLinks: A preliminary framework for concept-based adaptive hypermedia. In: Proc. of Workshop "WWW-Based Tutoring" at 4th International Conference on Intelligent Tutoring Systems (ITS'98), San Antonio, TX (1998), available online at http://www-aml.cs.umass.edu/~stern/webits/itsworkshop/murray.html
34. Nakabayashi, K., Koike, Y., Maruyama, M., Touhei, H., Ishiuchi, S., and Fukuhara, Y.: An intelligent tutoring system on World-Wide Web: Towards an integrated learning environment on a distributed hypermedia. In: Maurer, H. (ed.) Proc. of ED-MEDIA'95 - World conference on educational multimedia and hypermedia, Graz, Austria, AACE (1995) 488-493
35. Nakabayashi, K., Maruyama, M., Kato, Y., Touhei, H., and Fukuhara, Y.: Architecture of an intelligent tutoring system on the WWW. In: Boulay, B. d. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997) 39-46
36. Oda, T., Satoh, H., and Watanabe, S.: Searching deadlocked Web learners by measuring similarity of learning activities. In: Proc. of Workshop "WWW-Based Tutoring" at 4th International Conference on Intelligent Tutoring Systems (ITS'98), San Antonio, TX (1998), available online at http://www.sw.cas.uec.ac.jp/~watanabe/conference/its98workshop1.ps
37. Okazaki, Y., Watanabe, K., and Kondo, H.: An Implementation of an intelligent tutoring system (ITS) on the World-Wide Web (WWW). Educational Technology Research **19**, 1 (1996) 35-44
38. Paiva, A. and Machado, I.: Vincent, an autonomous pedagogical agent for on-the-job training. In: Goettl, B. P., Halff, H. M., Redfield, C. L. and Shute, V. J. (eds.) Intelligent Tutoring Systems. Lecture Notes in Computer Science, Springer-Verlag, Berlin (1998) 584-593
39. Peylo, C., Teiken, W., Rollinger, C., and Gust, H.: Der VC-Prolog-Tutor: Eine Internet-basierte Lernumgebung. Künstliche Intelligenz , (this issue) (1999)
40. Rios, A., Millán, E., Trella, M., J.L., P., and Conejo, R.: Internet based evaluation system. In: Laojie, S. P. and and Vivet, M. (eds.) Artificial Intelligence in Education: Open Learning Environments. IOS Press, Amsterdam (1999) 387-394
41. Ritter, S.: Pat Online: A Model-tracing tutor on the World-wide Web. In: Brusilovsky, P., Nakabayashi, K. and Ritter, S. (eds.) Proc. of Workshop "Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8th World Conference on Artificial Intelligence in Education, Kobe, Japan, ISIR (1997) 11-17, available online at http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Ritter/Ritter.html
42. Ritter, S., Brusilovsky, P., and Medvedeva, O.: Creating more versatile intelligent learning environments with a component-based architecture. In: Goettl, B. P., Halff, H. M., Redfield, C. L. and Shute, V. J. (eds.) Lecture Notes in Computer Science, Vol. 1452. Springer Verlag, Berlin (1998) 554-563
43. Schöch, V., Specht, M., and Weber, G.: "ADI" - an empirical evaluation of a tutorial agent. In: Ottmann, T. and Tomek, I. (eds.) Proc. of ED-MEDIA/ED-TELECOM'98 - 10th World Conference on Educational Multimedia and Hypermedia and World Conference on Educational Telecommunications, Freiburg, Germany, AACE (1998) 1242-1247

44. Shaw, E., Johnson, W. L., and Ganeshan, R.: Pedagogical agents on the Web. In: Proc. of Third International Conference on Autonomous Agents, ACM Press (1999) 289-290, available online at http://www.isi.edu/isd/ADE/papers/agents99/agents99.htm
45. Specht, M. and Oppermann, R.: ACE - Adaptive Courseware Environment. The New Review of Hypermedia and Multimedia **4** (1998) 141-161
46. Specht, M., Weber, G., Heitmeyer, S., and Schöch, V.: AST: Adaptive WWW-Courseware for Statistics. In: Brusilovsky, P., Fink, J. and Kay, J. (eds.) Proc. of Workshop "Adaptive Systems and User Modeling on the World Wide Web" at 6th International Conference on User Modeling, UM97, Chia Laguna, Sardinia, Italy, Carnegie Mellon Online (1997) 91-95, available online at http://www.contrib.andrew.cmu.edu/~plb/UM97_workshop/Specht.html
47. Stern, M. K. and Woolf, B. P.: Curriculum sequencing in a Web-based tutor. In: Goettl, B. P., Halff, H. M., Redfield, C. L. and Shute, V. J. (eds.) Intelligent Tutoring Systems. Lecture Notes in Computer Science, Springer-Verlag, Berlin (1998) 574-583
48. Suthers, D. and Jones, D.: An architecture for intelligent collaborative educational systems. In: Boulay, B. d. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997) 55-62
49. Vassileva, J.: Dynamic Course Generation on the WWW. In: Boulay, B. d. and Mizoguchi, R. (eds.) Artificial Intelligence in Education: Knowledge and Media in Learning Systems. IOS, Amsterdam (1997) 498-505
50. Warendorf, K. and Tan, C.: ADIS - An animated data structure intelligent tutoring system or Putting an interactive tutor on the WWW. In: Brusilovsky, P., Nakabayashi, K. and Ritter, S. (eds.) Proc. of Workshop "Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8th World Conference on Artificial Intelligence in Education, Kobe, Japan, ISIR (1997) 54-60, available online at http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Warendorf/Warendorf.html
51. Weber, G.: Individual selection of examples in an intelligent learning environment. Journal of Artificial Intelligence in Education **7**, 1 (1996) 3-31
52. Weber, G.: ART-WEB, Trier, University of Trier (1999) available online at
53. Weber, G. and Specht, M.: User modeling and adaptive navigation support in WWW-based tutoring systems. In: Jameson, A., Paris, C. and Tasso, C. (eds.) User Modeling. Springer-Verlag, Wien (1997) 289-300
54. Yang, J. C. and Akahori, K.: Development of Computer Assisted Language Learning System for Japanese Writing Using Natural Language Processing Techniques: A Study on Passive Voice. In: Brusilovsky, P., Nakabayashi, K. and Ritter, S. (eds.) Proc. of Workshop "Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8th World Conference on Artificial Intelligence in Education, Kobe, Japan, ISIR (1997) 99-103, available online at http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Yang/Yang.html