Developing Adaptive ElectronicTextbooks on WWW

Peter Brusilovsky

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
plb@cs.cmu.edu

Elmar Schwarz

Department of Psychology,University of Trier
Carnegie Mellon University
Pittsburgh, PA15213, USA

Abstract

An Electronic textbook is a popular kind of educational applications onWorld Wide Web (WWW). We claim that adaptivity is especially important forWWW-based educational applications which are expected to be used by verydifferent groups of users without assistance of a human teacher. In this paper we describe an approach fordeveloping adaptive electronic textbooks and present InterBook - anauthoring tool based on this approach which simplifies the development ofadaptive electronic textbooks on WWW.

Introduction

World Wide Web opens new ways of learning for many people. Now, educationalprograms and learning materials installed and supported in one place can beused by thousands of students from all over the world. However, most of theexisting educational WWW applications use the simplest solutions and are much more limited thanexisting 'on-site' educational systems and tools. For many designers, anideal format of educational WWW material seems to be a static electroniccopy of a regular textbook: chapter by chapter, page by page, picture bypicture. Such electronic textbooks are non-adaptive, i.e., students withdifferent abilities, knowledge, and background get the same educationalmaterial in the same form.

We claim that adaptivity is especially important for educational programson WWW which are expected to be used by very different classes of userswithout assistance of a real teacher (who usually can provide adaptivity ina normal classroom). Currently, we can name very few adaptive educational applications on WWW (Brusilovsky,Schwarz, and Weber 1996a; Nakabayashi et al. 1997; Okazaki, Watanabe, andKondo 1997; Stern, Wolf, and Kuroso 1997). All these applications keep a model of the user between sessions and useit to adapt the teaching sequence and the presentation of the material to agiven user. The problem is that adaptive electronic textbooks are"knowledge-rich" applications and theyare not very easy to design. There are some authorinig tools for developing"static" electronic textbooks on WWW (Goldberg, Salari, and Swoboda 1996;Thimbleby 1996), but there are no tools available to support a designer in creating anadaptive textbook on WWW.

A possible approach for designing adaptiveelectronic textbooks on WWW was suggested recently in (Brusilovsky 1995). This approach was further elaborated by the ELM research group in theUniversity of Trier which applied it for developing an adaptive WWW-basedLISP textbook ELM-ART (Brusilovsky et al. 1996a). Now thisapproach is implemented in InterBook, a subject-independent shell whichsimplifies the process of creating adaptive electronic textbooks on WWW. Inthis paper we present the approach in its current form and describe thesystem InterBook.

TheApproach

Our approach to developingadaptive electronic textbooks on WWW based on the ideas from the areas ofIntelligent Tutoring Systems (Wenger 1987) and Adaptive Hypermedia(Brusilovsky 1996). Our adaptive textbooks use knowledge about its domain (representedin the form of domain model) and about its users (represented in the formof individual user models). The domain model serves as a basis forstructuring the content of an adaptive ET. We distinguish two parts in anadaptive ET: a *glossary* and a *textbook.* Both these parts are based on the domain model. The studentmodel is used by an adaptive ET to adapt its behavior to each particularuser.

The DomainModel and the Student Model

According to our approach, thekey to adaptivity in an adaptive ET are the domain model and the student model.

The simplest form of domain model is just aset of domain concepts. By concepts we mean elementary pieces of knowledgeforthe given domain. Depending on the domain and the application area,concepts can represent bigger orsmaller pieces of domain knowledge. A more advanced form of the domainmodel is a network with nodes corresponding to domain concepts and withlinks reflecting several kinds of relationships between concepts. Thisnetwork represents the structure of the domain covered by a hypermedia system. The domain model provides a structurefor representation of the student's knowledge of the subject. For eachdomain model concept, an individual student's knowledge model stores somevalue which is an estimation of thestudent knowledge level of this concept. This type of model (which iscalled an *overlay model*) is powerful and flexible: it can measureindependently the student's knowledge of different topics.

TheGlossary

The glossary is the central partof the ET. According to our approach, the glossary is considered as a visualized (andexternalized) domain network. Each node of the domain network isrepresented by a node of the hyperspace, while the links between domainnetwork nodes constitute main paths between hyperspace nodes. The structure of the glossary resembles the pedagogicalstructure of the domain knowledge and, vise versa, each glossary entrycorresponds to one of the domain concepts. The links between domain modelconcepts constitute navigation paths between glossary entries. Thus, the structure of the manual resembles thepedagogic structure of the domain knowledge. In addition to providing adescription of a concept, each glossary entry provides links to all booksections which introduce the concept. Itmeans that the glossary integrates traditional features of an index and aglossary.

TheIndexed Textbook

A human-written textbookrepresents human teaching expertise on how to introduce the domain conceptsto the learners. It is usually a real textbook represented in hypermedia form. A textbook is hierarchically structured intounits of different level: chapters, sections, and subsections. To make thetextbook "more intelligent" and to connect it to the glossary, we have tolet the system know what each unit of the textbook is about. It is done by indexing of textbook units withdomain model concepts. For each unit, a list of concepts related with thisunit is provided (we call this list *spectrum* of the unit). For each involved concept, the spectrum of the unit canrepresents also the role of the concept in the unit. Currently we supporttwo roles: each concept can be either a *outcome* concept or a *prerequisite* concept. A concept is included in the spectrum as a outcome concept ifsome part of this page presents the piece of knowledge designated by theconcept. A concept is included in the spectrum as a prerequisite concept ifa student has to know this conceptto understand the content of the page. Indexing is a relatively simple butpowerful mechanism, because it provides the system with knowledge about the content of its pages: thesystem knows which concepts are presented on each page and which conceptshave to be learned before starting to learn each page. It opens the way forseveral adaptation techniquespresented in the next subsection.

Functionality

Domain model-based indexing is arelatively simple but powerful mechanism, because it provides the systemwith knowledge about the content of its pages: the system knows whichconcepts are presented on each page and which concepts have to be learned before starting to learn eachpage. It opens the path for several adaptation techniques presented in thissubsection. All of these techniques were implemented and tested inInterBook system(Brusilovsky, Schwarz, and Weber 1996b).

AdvancedNavigation

The knowledge about the domain and about the textbook content is used toserve a well-structured hyperspace. The system supports sequential andhierarchical links between section. It generates the table of content where all entries are clickable links. In addition, it generates links betweenthe glossary and the textbook. The concept bar provides links from eachtextbook unit to corresponding glossary pages for each involved concept(Figure 1). On the other hand, fromeach glossary page describing a concept the system provides links to alltextbook units which can be used to learn this concept (Figure 2). Theselinks are not stored in an external format but generated on-the-fly by aspecial module which takes into account the student's current state of knowledge represented by the studentmodel. This approach is not only reducing page design time but alsoprovides room for adaptation. In particular, our approach supports twoadaptation techniques: adaptive navigation support and prerequisite-based help.

AdaptiveNavigation Support

Our approach provides many moreopportunities for browsing the course materials than traditional on-linetextbooks. The negative side of it is that there is a higher risk for thestudent to getlost in this complex hyperspace. To support the student navigating throughthe course, the system uses two adaptive hypermedia techniques (Brusilovsky1996): *adaptive annotation* and *direct guidance*. Direct guidance means that the system can suggest to the student the nextunit to be learned. A possible way to provide direct guidance with ourapproach is presented in (Brusilovsky and Schwarz 1997). Adaptive annotation means that the system uses visual cues (icons, fonts,colors) to show the type and the educational state of each link.

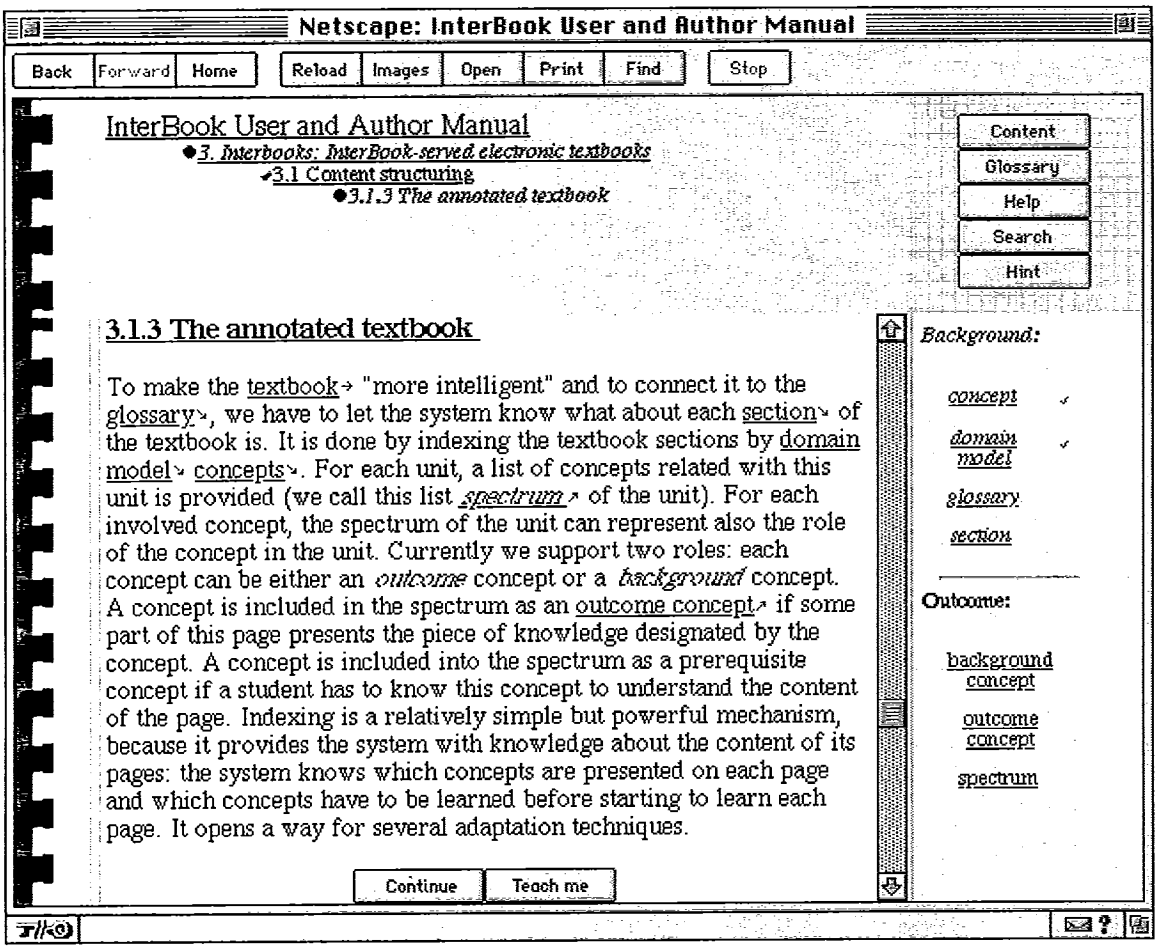

Figure 1. A unit of an adaptive ET prepared and shown with InterBooksystem. Colored balls (up) and checkmarks (right, on the concept bar)provide adaptive annotation. Button "Teach me" (bottom) provides directguidance.

Using the student model, it is possible to distinguish several educationalstates for each unit of ET: the content of the unit can be known to thestudent (all outcome concepts are learned or well-learned), ready to belearned (all prerequisites are learned or well-learned), or not ready to be learned (some prerequisiteconcepts are not yet learned). The icon and the font of each link presentedto the student are computed dynamically from the individual student model.They always inform the student about the type and the educational state of the unit behind the link. InInterBook, a red ball and italic font mean not ready to be learned, a greenball and bold font mean ready and recommended, a white ball means learned,no new information. A checkmark is added for already visited units (Figure 1). The same way can be used todistinguish and show several levels of students knowledge of the conceptsshown on the concept bar. In InterBook, no annotation means "unknown",small checkmark means "known" (learningstarted), medium checkmark means "learned" and big checkmark means "well-learned" (Figure 1).

Prerequisite-based Help

The system knowledge about thecourse material comprises knowledge about what the prerequisite conceptsare for any unit of the textbook.Often, when students have problems with understanding some explanation orexample or solving a problem, the reason is that some prerequisite materialis not understood well. In that case they can request prerequisite-basedhelp (using a special button) and, as an answer to help request, the systemgenerate a list of links to all sections which present some informationabout background concepts of the current section. This list is adaptivelysorted according to the student'sknowledge represented in the student model: more "helpful" sections are listed first. Here "helpful" meanshow informative the section is to learn about the background concepts. Forexample, the section which presents information about an unknown backgroundconcept is more informative than a section presenting information about a known concept. The section whichpresents information about two unknown background concepts is moreinformative than a section presenting information about one concept.

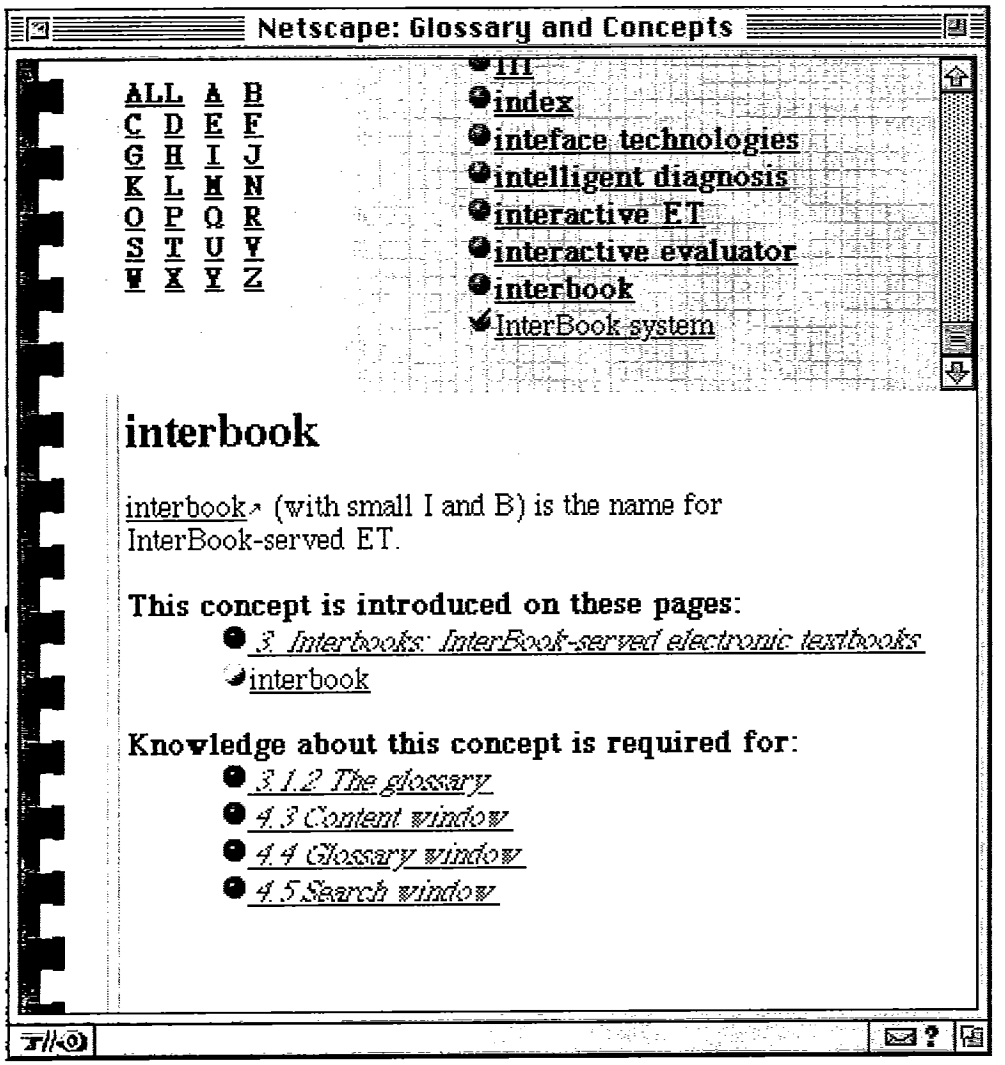

Figure 2. Glossary window of InterBookshowing a page for "interbook". In addition to a short description of theconcept "interbook", annotated links to all pages which introduce"interbook" and all pages which require this concept are shown.

InterBookInterface

InterBook uses advanced features of modern browsers such as multiplewindows and frames to provide the student with useful and powerfulinterface. Main windows used by InterBook are the textbook window (Figure1) and the glossary window (Figure 2).

The Glossary window is used to view the glossary. The upper part of the window is a list of glossary concepts.The lower part of this window is used to show the glossary entry for aconcept. For each concept the system presents the concept description (ifprovided by the author), the list of section titles (selected from all available textbooks) which present theconcept (i.e., which have it as an outcome concept) and the list of sectiontitles which require this concept (i.e., which have it as a backgroundconcept). Section titles are clickable links which makes the corresponding section to be loaded to theTextbook window.

The Textbook window is the most important window in InterBook interface.This window is designed to view the main content of a textbook, section bysection. It is divided into frames performing different functions. Mainframe of the Textbook window is theText window. This window shows a particular section of the textbook whichis called current section. For a terminal section the Text window shows thetitle of the section and the section itself. For a high-level section the Text window shows thetitle, the section preface (if existing) and the full table of content forthe section (i.e. list of hierarchically structured titles of itssubsections down to terminal level). Avertical bar to the right of the Text window is the Concept bar. It is usedto show the concepts related with the current section. All names ofconcepts on the Concept bar are links to the

Glossary. The upper part ofthe Textbook window hosts the navigation center and the toolbox. The navigation center shows the position of thecurrent section in the textbook: it lists the titles of all directpredecessors (father, grandfather, etc.) and all brothers of the currentsection. All names of the sections are clickable links. The navigation center serves for both orientation andnavigation. The toolbox provides a set of buttons which are used to calladditional windows (such as content window, search window, andprerequisite-based help window) which provides additional functionality.

Authoring with InterBook

Authoring an adaptive electronic textbook can be divided into 5 steps whichare described in detail below (see Figure 3). In brief, an ElectronicTextbook is prepared as a specially structured Word file and the task is toconvert this file into InterBook format. The result of this process is a file with the Textbook in InterBookformat which can be served on WWW by the InterBook system.

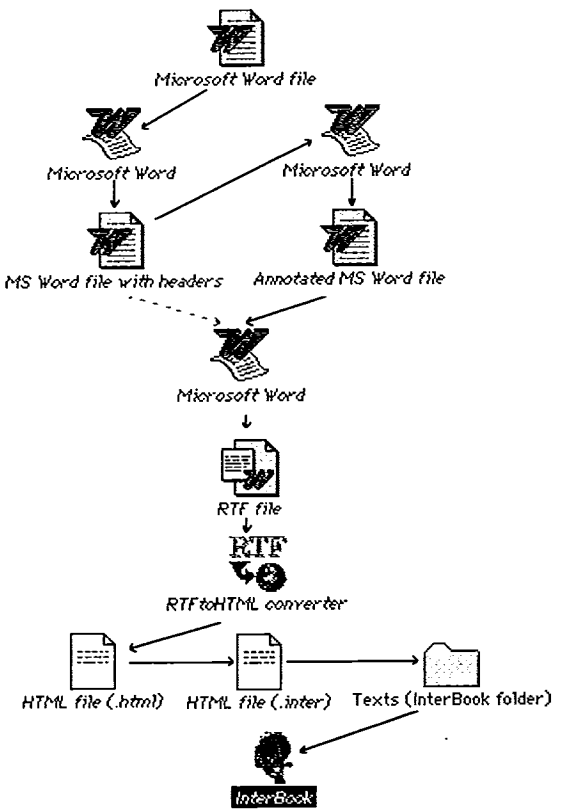

Figure 3. Authoring adaptive ETwith InterBook

InterBook recognizes thestructure of the document through the use of headers. It means that thetitles of the highest level sections should have a pre-defined text style"Header 1", the titles of its subsections should have a pre-definedparagraph style "Header 2", and so forth.The title of the textbook should have paragraph style "Title". The resultof this step will be a properly structured MS Word file.

The second step in the authoring process theninvolves concept-based annotation of the Electronic Textbook (ET) to letInterBook know which concepts stand behind each section. This knowledge allowsInterBook to help the reader of the ET in several ways, and the result ofthis step is an annotated (and structured) MS Word file.

An annotation is a piece of text of special style and format inserted atthe beginning of each section (between the section header and the firstparagraph). Annotations have special character style (hidden + shadowed)which are not visible in the text window to the reader of the ET. For each unit the author provides a set of outcome and background concepts. In this way, eachsection is annotated with a set of prerequisite concepts (or terms whichexist in other sections which should be read before the current section),and a set of outcome concepts (terms which will be assumed known once the reader has visited the section). Theformat for the outcome annotation is: (out: concept-name1, concept-name2,etc.) and the format for the background annotation is: (pre: concept-name1,concept-name2, etc.).

Once the annotations are complete the file is saved in RTF format. The RTFtoHTMLprogram with some special settings is used to convert the ET into HTMLformat. Then the .html extension on the file is manually altered to .interso that it can be recognized by the Interbook system.

Lastly, when the InterBook server starts, it parses all interbook files inits "Texts" folder (i.e. all files with extension .inter) and translates itinto the list of section frames. Each unit frame contains the name and typeof the unit, its spectrum, and its position in the original HTML file. The obtained LISPstructure is used by InterBook to serve all the available textbooks on WWWproviding the advanced navigation and adaptation features. The contentwhich is presented to the student is generated on-the-fly using the knowledge about the textbook, the studentmodel, and HTML fragments extracted from the original HTML file. Thesefeatures of InterBook are based on the functionality of the Common LispHypermedia Server (Mallery 1994).

Aswe can see, our tool seriously simplifies the design of adaptive ET on WWWfor the authors who use the approach presented in section 2. It providesfull support in preparation and serving an ET for the authors who know onlyhow to use the MS Word text processor. An advanced used who have some knowledge on HTML and LISPprogramming can use our tool more flexibly. For example, an author canbypass step 1 and 2 by preparing the textbook directly in HTML format withannotations provided as specially formatted comments. The author can also replace server response functions and HTMLgenerating functions to implement different structure and different "lookand feel" of the be requested by a unique URL. To enable the server torespond to a

particular URL, this URL has to be associated to a response function implemented in LISP which has togenerate an HTML page on the fly as an adaptive response. CL-HTTP includesa set of LISP functions for generating pages.

Discussion:Indexing for "More Intelligent" Authoring on WWW

The main idea behind our tool isusing concept-based *indexing* to make conventional educational material more intelligent and flexible.The idea of indexing is to provide the information about the content ofeach unit of conventional educational material by indexing this unit withrelated domain concepts. Previously, indexing was applied in three authoring contexts: CAI context, hypermediaauthoring context and ITS authoring context.

Indexing was originally suggested in CAI contextby Osin (1976) who suggested a framework for indexing CAI frames by a set of topics which itcovers. Such indexed sets of frames were not related to any pre-scribedorder of presentation. They can be accumulated, stored in speciallibraries, and re-used by different authors to create their own courses. In the multimedia field, a similar idea of are-usable database of multimedia learning material indexed by topics andkeywords is elaborated by Olimpo et al. (1990).

Later indexing was applied in hypermedia and ITSauthoring area. In the hypermedia authoring area, an idea of indexing waselaborated by Mayes, Kibby and Watson (1988) in the StrathTutor system. They stressed additional preference of indexingthe frames of learning material - the possibility to indicate related pairsof frames not by tedious glossary linking of pieces of learning materialtogether, but dynamically, on the basis of similarity of corresponded sets of topics. In the ITS authoringarea, indexing was applied to turn traditional CAI into a "slightlyintelligent" ICAI (Elsom-Cook and O'Malley 1990; Grandbastien and Gavignet1994; Vassileva 1992). "Slightly intelligent" ICAI are based on both the CAI and ITS paradigms.The teaching material is not generated as in 'orthodox' ITS, but stored inCAI-like frames. However, these frames are indexed with the concepts froman explicit domain model network, so they can be selected intelligently. The most recent application ofindexing on the crossroads of the above directions is hypermedia-based ITSwhich use indexing technology to connect the learning material represented in a hypermediaform with the domain knowledge base: SHIVA (Zeiliger 1993), ITEM/PG andISIS-Tutor (Brusilovsky, Pesin, and Zyryanov 1993).

Indexing shows to be a relatively cheap and useful technology for authoring"more intelligent" hypermedia and CAI systems. We argue that it is therelevant technology for developing more adaptive and intelligenteducational applications of WWW. Currently,we can name only ELM-ART (Brusilovsky et al. 1996a) and CALAT (Nakabayashiet al. 1997) as examples of WWW adaptive electronic textbooks based on indexing. Weexpect that the WWW will boost the research and development work onadaptive electronic textbooks.

Acknowledgments

Part of this work was supportedby an Alexander von Humboldt-Stiftung Fellowship and James S. McDonnellFoundation grant to the first author.

References

Brusilovsky, P. 1995. Adaptivelearning with WWW: The Moscow State University Project. In Proceedings ofTelematics for Education and Training Conference, 252-255. Amsterdam:IOS.

Brusilovsky, P. 1996. Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction* 6(2-3):87-129.

Brusilovsky, P.; Pesin, L.; and Zyryanov, M. 1993. Towards an adaptivehypermedia component for an intelligent learning environment. InProceedings of 3rd International Conference on Human-Computer Interaction,EWHCI'93, 348-358. Berlin: Springer-Verlag.

Brusilovsky, P. and Schwarz, E. 1997. User as student: Towards an adaptiveinterface for advanced Web-based applications. In Proceedings of 6thInternational Conference on User Modeling, 177-188. Wien: SpringerWerlag.

Brusilovsky, P.; Schwarz, E.; and Weber, G. 1996a. ELM-ART: An intelligenttutoring system on World Wide Web. In Proceedings of Third InternationalConference on Intelligent Tutoring Systems, ITS-96, 261-269. Berlin:Springer Verlag.

Brusilovsky, P.; Schwarz, E.; and Weber, G. 1996b. A tool for developingadaptive electronic textbooks on WWW. In Proceedings of WebNet'96, WorldConference of the Web Society, 64-69. San Francisco, CA: AACE.

Elsom-Cook, M. T. and O'Malley, C. 1990. ECAL: Bridging the gap between CALand intelligent tutoring systems. *Computers and Education*15(1):69-81.

Goldberg, M. W.; Salari, S.; and Swoboda, P. 1996. World Wide Web - coursetool: An environment for building WWW-based courses. *Computer Networksand ISDN Systems* 28:1219-1231.

Grandbastien, M. and Gavignet, E. 1994. ESCA: An environment to design andinstantiate learning material. In de Jong, T. and Sarti, L. (eds.): *Design and production of multimedia and simulation-based learning material.* 31-44. Dordrecht: Kluwer Academic Publishers.

Mallery, J. C. 1994. A Common LISP hypermedia server. In Proceedings of theFirst International Conference on the World-Wide Web.

Mayes, J. T.; Kibby, M. R.; and Watson, H. 1988. StrathTutor: Thedevelopment and evaluation of a learning-by-browsing on the Macintosh. *Computers and Education* 12(1):221-229.

Nakabayashi, K.; Maruyama, M.; Kato, Y.; Touhei, H.; and Fukuhara, Y. 1997.Architecture of an intelligent tutoring system on the WWW. In Proceedingsof AI-ED'97, World Conference on Artificial Intelligence in Education,39-46. Amsterdam: IOS.

Okazaki, Y.; Watanabe, K.; and Kondo, H. 1997. An Implementation of the WWWBased ITS for Guiding Differential Calculations. In Proceedings of Workshop"Intelligent Educational Systems on the World Wide Web" at AI-ED'97, 8thWorld Conference on ArtificialIntelligence in Education, 18-25. Kobe, Japan: ISIR.

Olimpo, G.; Persico, D.; Sarti, L.; and Tavella, M. 1990. On the concept ofdatabase of multimedia learning material. In Proceedings of WorldConference on Computers and Education, 431-436. Amsterdam: NorthHolland.

Osin, L. 1976. SMITH: How to produce CAI course without programming. *International Journal on the Man-Machine Studies* 8:207-241.

Stern, M.; Wolf, B. P.; and Kuroso, J. 1997. Intelligence on the Web? InProceedings of AI-ED'97, 8th World Conference on Artificial Intelligence inEducation, 490-497. Amsterdam: IOS.

Thimbleby, H. 1996. Systematic web authoring. In Proceedings of Symposium"The Missing Link: Hypermedia Usability Research & The Web" Milton Keynes,U.K.

Vassileva, J. 1992. Dynamic CAL-courseware generation within an ITS-shellarchitecture. In Proceedings of 4th International Conference, ICCAL'92,581-591. Berlin: Springer-Verlag.

Wenger, E. 1987. Artificial intelligence and tutoring systems.Computational approaches to the communication of knowledge. Los Altos:Morgan Kaufmann.

Zeiliger, R. 1993. Adaptive testing: contribution of the SHIVA model. InLeclercq, D. and Bruno, J. (eds.): *Item banking: Interactive testingand self-assessment.* NATO ASI Serie F, Vol. 112, 54-65. Berlin:Springer-Verlag.