Adaptive Hypermedia:
an Attempt to Analyze and Generalize

Peter Brusilovsky

International Center for Scientific and Technical Information,
Kuusinena str. 21b, Moscow 125252, Russia
E-mail: plb@plb.icsti.su

Abstract. Adaptive hypermedia is a new area of research at the crossroads of hypermedia, adaptive systems and intelligent tutoring systems. The goals of this paper are to provide a brief overview of this area and to synthesize a generalized view of the organization of existing adaptive hypermedia systems. We discuss three important questions: *why* do we need adaptive hypermedia (AH), *where* can it be useful, and *what* can be adapted in adaptive hypermedia. Then we introduce a generalized view of internal knowledge structure of AH systems and use it to uncover the basic approaches to hyperspace structuring in AH systems and basic methods of adaptation related with these approaches.

1 Introduction

Hypermedia systems and user-model-based adaptive systems (i.e. intelligent tutoring systems, adaptive information retrieval systems) are most often considered as two different approaches to browsing information spaces and interface organization. Adaptive hypertext and hypermedia systems (AHS) attempt to bridge the gap between the two extremes. AHS enhance classic hypermedia with an intelligent agent which supports a user in her work with hypermedia. The intelligent agent can support the user by adapting the content of a hypermedia page to the user's knowledge and goals and by suggesting the most relevant links to follow. AHS avoid the 'incorrect adaptation' problem of classic adaptive systems by providing space for user-driven adaptation. AHS also avoid the 'lost in hyperspace' problem of classic hypermedia systems by providing intelligent guidance.

The first goal of this paper is to provide a brief overview of recent works on the development of adaptive hypermedia systems. We hope that this overview will be useful for several categories of researchers and will stimulate further work in this area. The review attempts to answer three most important questions: *why* do we need adaptive hypermedia, *where* can it be useful, and *what* can be adapted in adaptive hypermedia. The review does not cover the question *how* adaptive hypermedia can be adapted, this information can be found in another paper [Brusilovsky, 1996].

The second goal and the research contribution of the paper is to generalize the results of AHS design, to find similarities between different system, and to uncover some general approaches which can be used to design new AHS. Existing AHS are oriented to different applications and use different adaptation techniques. What a more detailed analysis shows is that internal features of these systems are more similar than external ones. Due to similar general goals, all these systems appear to use quite similar structures for knowledge representation. The second part of the paper is an attempt to synthesize a generalized view of the internal knowledge structures of adaptive hypermedia systems. This generalized view is used to uncover three basic approaches to hyperspace structuring in AHS and basic methods of adaptation related with these approaches.

2 Why Adaptive Hypermedia?

Unlike other kinds of application systems, any hypermedia system is adaptive in some sense: using free browsing different users can adapt the system to their information needs. Many researchers hold that it is the user who should bring the adaptivity to the man-machine hypermedia system. Why is it necessary any other kind of adaptation? Why do we need that a hypermedia system adapts itself to the particular user? Researchers working on adaptive hypermedia give two main arguments for this.

First, adaptation can solve the problem of hypermedia systems which are used by different classes of users. Users can significantly differ in their goals, computer experience, background, and knowledge of the subject covered by the hypermedia system. Moreover, the same user can have different goals and different knowledge (as a result of learning or forgetting) when using the system at different times. A regular hypermedia system provides the same hypermedia pages and the same set of links to all users, while users with different goals and knowledge may be interested in different pieces of information presented on a regular page and may prefer different links for navigation. A way to overcome this problem is to use the information about a particular user represented in the *user model* to adapt the information being presented to the given user.

Second, adaptation can prevent the user from getting lost in hyperspace, which is a problem for any serious hypermedia system. Knowing user goals and knowledge level, an adaptive hypermedia system can provide navigation support by limiting browsing space (i.e. hiding non-relevant links), suggesting most relevant links to follow, or augmenting the links with some kind of visual cues.

3 Where Adaptive Hypermedia Can Be Helpful

Analysis of existing AH systems allowed us to name five application areas which are used at present in most of the research projects on adaptive hypermedia: educational hypermedia, on-line information systems, on-line help systems, information retrieval hypermedia systems, and institutional information systems (Table 1). Below we characterize each of these application areas pointing out their specific features and identifying problems. In each of these areas adaptive hypermedia can be helpful because it helps to solve the identified problems.

The most popular area for adaptive hypermedia research is *educational hypermedia* [Beaumont, 1994; Brusilovsky & Pesin, 1994; Brusilovsky, Pesin & Zyryanov, 1993; Brusilovsky & Zyryanov, 1993; de La Passardiere & Dufresne, 1992; Gonschorek & Herzog, 1995; Hohl, Böcker & Gunzenhäuser, 1996; Kay & Kummerfeld, 1994; Micarelli & Sciarrone, 1996; Pérez, Gutiérrez & Lopistéguy, 1995]. Existing educational hypermedia systems have relatively small hyperspaces representing a particular course or learning material on a particular subject. The final goal of the student is usually to learn all this material or a reasonable part of it. The role of hypermedia is to support student-driven acquisition of the learning material. The most important user feature in educational hypermedia is user knowledge of the subject being taught. Adaptive hypermedia can be useful to solve the following problems of educational hypermedia. First, the knowledge level of different users can be quite different and the knowledge level of a particular user can grow quite fast. The same page might be unclear for a novice and at the same time trivial and boring for an advanced learner. Second, novices enter the hyperspace of educational material knowing almost nothing about the subject. Most of the offered links from any node lead to material which is completely new for them. They need navigation help to find their way through the hyperspace. Without such help they can "get lost" even in reasonably small hyperspaces, or use very inefficient browsing strategies [Hammond, 1989].

Another popular application for adaptive hypermedia is the area of various *on-line information systems* ranging from on-line documentation to electronic encyclopedias. Examples include Hypadapter [Hohl, Böcker & Gunzenhäuser, 1996], MetaDoc [Boyle & Encarnacion, 1994], HYPERCASE [Micarelli & Sciarrone, 1996], KN-AHS [Kobsa, Müller & Nill, 1994], and HYPERFLEX [Kaplan, Fenwick & Chen, 1993]. Normally, these systems are used for reference access to information and the users usually have some domain knowledge. Each node of the hyperspace usually represents one concept of the subject and contains several pages of information. Depending on the domain, the size of the hyperspace can range from reasonably small to very big. Similar to educational hypermedia, on-line information systems have problems with satisfying the needs of very different users. Users with different knowledge levels and backgrounds usually need different information about a concept at different levels of detail. They usually have no time to browse all the information about the concept looking for the required portion of information. Users also have different goals when accessing an information system. In many cases they know which concepts to access to achieve their goals and do not need any navigation support [Boyle & Encarnacion, 1994; Kobsa, Müller & Nill, 1994]. However, when the goal can not be directly mapped to the structure of the hyperspace or when the hyperspace is too big users need help in finding relevant pieces of information.

Very close to on-line information systems are *on-line help systems*, another application area for AHS. Examples from this area are EPIAIM [de Rosis, De Carolis & Pizzutilo, 1993], Lisp-Critic [Fischer et al., 1990], HyPLAN [Grunst, 1993], and ORIMUHS [Encarnação, 1995]. The difference from the former category is that on-line help systems are not independent like on-line information systems but are attached to some computer application system such as spreadsheets, programming environments, or expert systems. On-line help systems represent all the information about the application system and its domain which is required to help the users. The hyperspace in existing on-line help systems is reasonably small. On-line help systems have the same problems as on-line information systems, however, the problem of helping users to find relevant pieces of information is less important because the hyperspace is not too big and because the user information goal can often be determined by the context of their work in the application system [Encarnação, 1995; Grunst, 1993].

The three application areas listed above belong to classic application areas for hypermedia. The majority of existing hypermedia systems belong to one of these three areas. It is not surprising that most adaptive hypermedia systems also belong to these areas. The areas listed below are more recent application areas for hypermedia. Existing adaptive hypermedia systems show, however, that adaptive hypermedia can be useful in these areas too.

Information retrieval hypermedia systems [Agosti, Melucci & Crestani, 1995] combine traditional information retrieval (IR) techniques with the possibility to browse the hyperspace of documents using static or dynamic similarity links. Browsing can help users to find required documents in the case when they fail to formulate a formal query. The size of hyperspace in regular IR hypermedia is usually huge. Users of IR hypermedia are mostly professionals in different areas who use the system in their everyday work with different IR goals. A very new special kind of IR hypermedia is IR systems on the Word Wide Web which have slightly different types of links and a potentially unlimited hyperspace. Existing adaptive IR systems [Kok, 1991] show some ways to help the user in pure IR settings. Adaptive hypermedia systems such as HYPERFLEX [Kaplan, Fenwick & Chen, 1993], Adaptive HyperMan [Mathé & Chen, 1994], and WebWatcher [Armstrong et al., 1995] can offer some additional help providing individual browsing guidance.

Another new area of application for adaptive hypermedia is *institutional information systems* which serve on-line all the information required for the work of some institution, for example, a hospital [Vassileva, 1994]. Originally, institutional information systems were developed as a set of loosely related databases but in some recent systems such databases are joined into a single hyperspace which can be reasonably big. A specific feature of these systems is that they are a medium for the everyday work of many institution employees. According to their profession they may always use a specific area of hyperspace, and according to their current working goal may need access to a very small subset of it. In this situation the large number of existing navigation opportunities distracts them from their primary work.

It is worthy to mention that all five listed application areas are not completely different. Some of them are pairwise similar and share the same problems. These pairs are: IR hypermedia and on-line information systems, on-line information/help systems and educational hypermedia, educational hypermedia and institutional hypermedia, institutional hypermedia and information space management systems. Also, the difference between neighboring areas is not always clear cut and some systems belong to both areas, for example, Hypadapter [Hohl, Böcker & Gunzenhäuser, 1996] and HYPERCASE [Micarelli & Sciarrone, 1996] share features of educational hypermedia and on-line information systems, and HYPERFLEX shares features of on-line information systems and IR hypermedia.

Table 1. Existing adaptive hypermedia systems classified according their application.

Educational Hypermedia Systems	Anatom-Tutor [Beaumont, 1994], C-Book [Kay & Kummerfeld, 1994], ELM-ART [Schwarz, Brusilovsky & Weber, 1996], ISIS-Tutor [Brusilovsky & Pesin, 1994], ITEM/PG [Brusilovsky, Pesin & Zyryanov, 1993], HyperTutor [Pérez, Gutiérrez & Lopistéguy, 1995], Manuel Excel [de La Passardiere & Dufresne, 1992], SHIVA [Zeiliger, 1993], SYPROS [Gonschorek & Herzog, 1995]
On-line Information Systems	Hypadapter [Hohl, Böcker & Gunzenhäuser, 1996] HYPERCASE [Micarelli & Sciarrone, 1996], KN-AHS [Kobsa, Müller & Nill, 1994], MetaDoc [Boyle & Encarnacion, 1994], PUSH [Höök et al., 1996]
On-line Help Systems	EPIAIM [de Rosis, De Carolis & Pizzutilo, 1993], HyPLAN [Grunst, 1993], Lisp-Critic [Fischer et al., 1990], ORIMUHS [Encarnação, 1995]
Information Retrieval Hypermedia	Adaptive HyperMan [Mathé & Chen, 1994], HYPERFLEX [Kaplan, Fenwick & Chen, 1993], WebWatcher [Armstrong et al., 1995]
Institutional Hypermedia	Hynecosum [Vassileva, 1994]

4 What Can Be Adapted in Adaptive Hypermedia?

In adaptive hypermedia, the space for adaptation is quite limited: there are not so many features which can be altered. At some level of generalization, hypermedia consists of a set of nodes or "pages" connected by links. Each page contains some local information and a number of links to related pages. These links can appear within the content of a page, on a separate menu, or on a separate local map. Hypermedia systems can also include an index and a global map which provide links to all accessible pages. What can be adapted in adaptive hypermedia are the content of regular pages (content-level adaptation) and the links from regular pages, index pages, and maps (link-level adaptation). As a rule, content-level adaptation is used to solve the problem of hypermedia systems which are used by different classes of users, while link-level adaptation is used to provide some kind of navigation support and prevent users from getting lost in hyperspace. We distinguish content-level and link-level adaptation as two different ways of hypermedia adaptation and call the first one *adaptive presentation* and the second one *adaptive navigation support*.

4.1 Adaptive Presentation

The idea of various adaptive presentation techniques is to adapt the content of a page accessed by a particular user to the current knowledge level, goals, and other characteristics of the user. For example, a qualified user can be provided with more detailed and deep information while a novice can receive additional explanations. Existing adaptive presentation techniques are not really different from a "what can be adapted" point of view: the visible result of adaptation is that different users in different time may get different text as the content of the same page. That is why we group all these techniques into one *technology* which we call adaptive text presentation technology. Most of the early work on adaptive hypermedia was centered around adaptive text presentation [Beaumont, 1994; Boyle & Encarnacion, 1994; Brusilovsky, 1992; de Rosis, De Carolis & Pizzutilo, 1993; Fischer et al., 1990].

4.2 Adaptive Navigation Support

The idea of adaptive navigation support techniques is to help users to find their paths in hyperspace by adapting the style of link presentation to the goals, knowledge, and other characteristics of an individual user. Adaptive navigation support techniques can be classified in several groups according to the method they use to adapt presentation of links. We consider these groups of techniques as different technologies for adapting link presentation. The most popular technologies are direct guidance, sorting, hiding, and annotation.

Direct guidance is the most simple technology of adaptive navigation support. Direct guidance can be applied in any system which can decide what is the next "best" node for the user to visit according to the user's goal and other parameters represented in the user model. To provide direct guidance, the system can outline visually the link to the "best" node as is done in Web Watcher [Armstrong et al., 1995], or present an additional dynamic link (usually called "next") which is connected to the "best" node as in ISIS-Tutor [Brusilovsky & Pesin, 1994], SHIVA [Zeiliger, 1993], and HyperTutor [Pérez, Gutiérrez & Lopistéguy, 1995]. A problem of direct guidance is that it provides no support for the users who would not like to follow the system's suggestion. Direct guidance is useful but it has to be used together with a "more supportive" technology.

The idea of *adaptive ordering* technology is to sort all the links of a particular page according to the user model and a user-valuable criteria: the closer to the top, the more relevant the link is. Adaptive ordering has a limited applicability: it can be used with non-contextual links, it can hardly be used for indexes and content pages (which usually have a stable order of links), and it can never be used with contextual links and maps. Another problem with adaptive ordering is that this technology makes the order of links non-stable: it may change each time the user enters the page. At the same time, some recent research shows that the stable order of options in menus is important for novices [Debevc, Rajko & Donlagic, 1994; Kaptelinin, 1993]. However, this technology appears to be useful for information retrieval (IR) applications [Armstrong et al., 1995; Kaplan, Fenwick & Chen, 1993; Mathé & Chen, 1994]. Experimental research [Kaplan, Fenwick & Chen, 1993] showed that adaptive ordering can significantly reduce navigation time in IR applications where each page can have many non-contextual links.

The idea of *hiding* technology is to restrict the navigation space by hiding links to "not relevant" pages. The page can be considered as not relevant for several reasons: for example, if it is not related to the user's current goal [Brusilovsky & Pesin, 1994; Grunst, 1993; Höök et al., 1996; Mathé & Chen, 1994; Vassileva, 1994] or if it presents material which the user is not yet prepared to understand [Brusilovsky & Pesin, 1994; Gonschorek & Herzog, 1995; Pérez, Gutiérrez & Lopistéguy, 1995]. Hiding protects users from the complexity of the unrestricted hyperspace and reduces their cognitive overload. Hiding has a wide applicability: it can be used with all kinds of non-contextual, index, and map links by hiding buttons or menu items [Brusilovsky & Pesin, 1994], and with contextual links by transferring "hot words" into normal text [Gonschorek & Herzog, 1995; Pérez, Gutiérrez & Lopistéguy, 1995]. Hiding has, however, another problem: as noted by some psychologists, hiding can provoke the forming of incorrect mental models of the hyperspace.

The idea of *adaptive annotation* technology is to augment the links with some form of comments which can tell the user more about the current state of the nodes behind the annotated links. These annotations can be provided in textual form [Zhao, O'Shea & Fung, 1993] or in the form of visual cues using, for example, different icons (Manuel Excel, ELM-ART), colors (ITEM/PG, ISIS-Tutor), font sizes (Hypadapter), or font types (ELM-ART). Link annotation is known as an effective way of navigation support in hypermedia [Zhao, O'Shea & Fung, 1993]. The typical kind of annotation considered in traditional hypermedia is static (user independent) annotation. Adaptive navigation support can be provided by dynamic user model-driven annotation. Adaptive annotation in its simplest history-based form (outlining the links to previously visited nodes) has been applied in some hypermedia systems including several World-Wide Web browsers. Even this simplest form of adaptive annotation which can distinguish only two states of links (links to visited/not visited nodes) appears to be quite useful. Current adaptive hypermedia systems (ISIS-Tutor, ELM-ART) can distinguish and annotate differently up to six states on the basis of the user model. Annotation is a very relevant way of adaptive navigation support. Annotation can be naturally used with all possible forms of links. This technique supports the stable order of links and avoids problems with incorrect mental maps. At the same time, annotations do not restrict cognitive overload as much as hiding does, but the hiding technology can be quite well simulated by the annotation technology using a kind of "dimming" instead of hiding for "not relevant" links.

Direct guidance, sorting, hiding, and annotation are the primary technologies for adaptive navigation support. Most existing adaptation techniques use exactly one of these ways to provide adaptive navigation support. However, these technologies are not contradictory and can be used in combinations. For example, ISIS-Tutor uses direct guidance, hiding, and annotation and Hypadapter uses sorting, hiding, and annotation. In particular, the direct guidance technology can be naturally used in combination with any of the three other technologies.

5 Models for Adaptive Hypermedia: The First Generalization

Existing adaptive hypermedia systems are designed for different applications and look quite different externally. However, a careful analysis helps us to find a deep similarity between internal structures of different adaptive hypermedia systems. This similarity becomes clear after some generalization which is introduced below.

5.1 The Domain Model and the User Knowledge Model

The heart of many adaptive hypermedia systems is a set of domain concepts. These concepts can be named differently in different systems - topics, knowledge elements, objects, learning outcomes, but in all the cases they are just elementary pieces of knowledge for the given domain. Depending on the domain and the application area, the concepts can represent bigger or smaller pieces of domain knowledge. This set of domain concepts is usually referred to as *the domain model.* An independent set of concepts is the simplest form of domain model. We call it *a level one model.* Such a model is used in some AHS (SHIVA, MetaDoc). Other AHS use a more advanced domain model where the concepts are related to each other thus forming a kind of semantic network. This network represents the structure of the domain covered by a hypermedia system. We call a network domain model *a level two model.*

Most existing AHS distinguish several types of concepts which represent different kinds of knowledge elements or objects and several kinds of links which represent different kinds of relationships between concepts. For different systems the kinds of concepts are usually different because they depend on the domain and the design choices. The choice of links is more limited. The kinds of links which are popular in many systems are classic conceptual links like "is-a" (Hypadapter, EPIAIM, PUSH, Anatom-Tutor, KN-AHS, ITEM/PG, ELM-ART) and structural links like "part-of" (PUSH, Anatom-Tutor, ELM-ART). Educational systems often use prerequisite links between concepts which represent the fact that one of the related concepts has to be learned before another (ITEM/IP, SHIVA, HyperTutor).

In some systems with a level two model the concepts are real atoms which do not have any internal structure (MetaDoc, KN-AHS, SHIVA); however, AH systems often use a more advanced frame-like knowledge representation, i.e., represent an internal structure of each concept as a set of attributes where different kinds of topics usually have different sets of attributes (Hypadapter, EPIAIM, PUSH, Anatom-Tutor, ISIS-Tutor, ITEM/PG, ELM-ART, HyperTutor). We call such a frame-based domain model *a level three model.*

One of the most important functions of the domain model is to provide a structure for representation of the user's domain knowledge. For each domain model concept, an individual user knowledge model stores some value which is an estimation of the user knowledge level on this concept. This can be a binary value (known – not known), a qualitative value (good-average-poor), or a quantitative value, such as the probability that the user knows the concept. This type of model, which is called an *overlay model,* is most often used in AH systems. An overlay model of user knowledge can be represented as a set of pairs "concept - value", one pair for each domain concept. The overlay model is powerful and flexible, it can measure independently the user's knowledge of different topics.

5.2 Three Approaches to Organization of Adaptive Hypermedia Systems

The domain model and the user knowledge model introduced above provide a generalized level for describing the knowledge structure used in many AH systems. This generalization enables us to reveal a similarity between different AH systems. We have found that almost all existing AH systems which can adapt to the user domain knowledge level can be classified into three groups. Systems from the same group have similar hyperspace structure and share specific adaptation techniques which are based on this structure. The key feature we use to classify a system into one of these groups is the relationship between domain model concepts and hypermedia pages. Three different ways of connecting the concepts with the hypermedia pages define three basic approaches to organization of AH systems.

Page Indexing

The first way is to index all hypermedia pages with domain model concepts which are related to the content of the page. This way is especially popular in education-oriented hypermedia systems (Lisp-Critic, ISIS-Tutor, ELM-ART, SHIVA). There are two major types of indexing: content-based indexing and prerequisite-based indexing. With content-based indexing, a concept is included in a page index if some part of this page presents the piece of knowledge designated by the concept (SHIVA, ISIS-Tutor, ELM-ART). With prerequisite-based indexing, a concept is included in a page index if a student has to know this concept to understand the content of the page (Lisp Critic, ELM-ART). A more general method to index the pages with concepts is to add the role for each concept in the page index, i.e., to specify the type of the relationship between a concept and a page. In particular, the concept can be a content concept or a prerequisite concept. This method is used in ELM-ART which distinguishes several different roles for concepts, including the above two.

Page indexing is a relatively simple mechaizm. It can be applied even with a level one domain model when the domain concepts have no internal structure and there are no links between them. At the same time, indexing is a very powerful mechaizm, because it provides the system with knowledge about the content of its pages. It opens the way for several adaptation techniques. With content-based indexing, the system knows which concepts are presented on a page. It can be used by a direct guidance technique to recommend the most relevant page to read about a particular topic (SHIVA, ISIS-Tutor). It can also be used by an annotation technique to show the educational state of the page. For example, ELM-ART and ISIS-Tutor distinguish and annotate the following cases differently: the page is not ready to be learned (it contains concepts which have unlearned prerequisite concepts), the page is recommended (it contains concepts which are a part of the current learning goal). With a prerequisite-based indexing technique, the system knows which concepts have to be learned before starting to learn the page itself. This knowledge can be used to support adaptive presentation. For example, when LISP-Critic presents a page which has unknown prerequisite concepts, it inserts explanations of all the unknown concepts before the main content of the page. Alternatively, this knowledge can be used by a hiding technique. For example, ISIS-Tutor can hide the pages which have unlearned prerequisite concepts.

Fragment Indexing

The second approach is to divide the content of the hypermedia page into a set of fragments and to index some (or even all) of these fragments with domain model concepts which are related to the content of this fragment (Anatom-Tutor, KN-AHS, MetaDoc). This approach is somewhat similar to the previous one. Similarly, it can be applied even with a level one domain model. The difference is that indexing is done on a more fine-grained level. Sometimes very small text fragments are used, so it is often possible to use exactly one concept to index a fragment. It gives the system a more fine-grained knowledge about the content of the page: the system knows what is presented in each indexed fragment. This knowledge can be effectively used for advanced adaptive presentation. Depending on the level of user knowledge about the concepts presented in a particular fragment, the system can hide this fragment from the user (MetaDoc, KN-AHS) or use an alternative way to present this fragment (Anatom-Tutor).

A good example is provided by MetaDoc, the first system which implements this approach to hypertext organization. MetaDoc not only indexes some text fragments with related concepts, but also distinguishes three types of fragments: general text, additional explanations, and low level details. The system decides whether to present the text fragment to the user or to hide it depending on the user's level of knowledge of the indexing concepts. A user with good knowledge of a particular concept will always get additional explanations of this concept (which can be boring for that user) hidden and all low level details presented. On the contrary, the user with poor knowledge of a concept will always get additional explanations of this concept and all low level details (too complicated for that user) will be hidden. The user with medium level knowledge will see both kinds of text fragments.

Knowledge-based Approach

The third way (which we call *knowledge-based approach*) is quite different from the first two. With the knowledge-based approach, a hypermedia network is built as a visualized and externalized domain model network. Each concept of the domain model network is represented by a node of the hyperspace, while the links between the concepts constitute main paths between hyperspace nodes. Thus the structure of the overall hyperspace resembles the structure of the domain knowledge. With this mechanism, each concept corresponds either to one hypermedia page (Hypadapter, EPIAIM, ISIS-Tutor, ITEM, ELM-ART) or (when the amount of information to present about the concept is too large) to one hyperdocument (PUSH, HyperTutor). The content of a page or a hyperdocument representing a concept can be prepared directly in a text form (ISIS-Tutor, PUSH, HyperTutor). However, most often it is not stored in an external format, but generated on-the-fly from the frame-based internal representation of domain concepts (ITEM/PG, Hypadapter, EPIAIM, ELM-ART).

The knowledge-based approach sets the strongest requirements for the domain model. It always requires a level two model with established links between concepts. To use all its potential knowledge-based approach requires a level three model which represents internal structure of the concepts. Unlike the two previous approaches, this one can not be used "post-hoc" to turn an existing traditional hypermedia system into an AH system. It has to be used from the early steps of a hypermedia system design. However, this approach is the most powerful one and provides the best opportunities for adaptation. With knowledge-based approach, the system knows exactly the type and content of each page and the type of each link. This knowledge can be used by various adaptive navigation support techniques. Annotation is the most suitable technology here. For example, ISIS-Tutor and ELM-ART use different kinds of link annotation to show the current educational state of the concept (not known, known, well-known), to show that the page is not ready to be learned, and to stress that it is a part of the learning goal. Hiding technology can be used in the same context to hide the links to concept pages with unlearned prerequisites (ISIS-Tutor, HyperTutor), or concept pages which do not belong to the current educational goal (ISIS-Tutor).

AHS which use the level three domain model can provide a very sophisticated adaptive presentation by generating pages from concept frames. The most powerful known adaptive presentation techniques (which we call frame-based adaptation) is used in Hypadapter and EPIAIM. These systems use a level three domain model. The strategy of adaptation is defined by a flexible rule-based formalism. Hypadapter has a set of independent "attribute" rules based on the student model. These rules are used to decide whether to present a particular attribute and to calculate its importance. Selected attributes of the frame are ordered by importance and presented to the user. EPIAIM provides an intermediate level for decision making: the presentation schemes. Each schema is just an ordered subset of attributes to present. A set of user model-based rules is used to select the most relevant schema, which is then used to control the concept presentation. Both systems store in different frame slots two or more versions of concept description oriented for users with different knowledge and some additional information about a concept. Thus each user can get the most relevant version of concept description during the presentation.

5.3 Mixing the Approaches

The three above approaches are the basic ones for the organization of hyperspace in AH systems. These approaches, however, do not contradict each other. Moreover, they are really complimentary because they are based on the same domain and user models. Using more than one approach opens the way to use more adaptation techniques because each approach supports its own set of techniques. At present, we know of very few systems which use more then one approach. For example, ISIS-Tutor uses a knowledge-based approach to build a part of the hyperspace representing the concepts of a programming language. Another part of the hyperspace (where the pages are problems and examples) is organized by indexing pages with concepts. This organization lets ISIS-Tutor to use the adaptation techniques from two corresponding groups. ELM-ART, which is based on the same ideas as ISIS-Tutor, uses all three approaches. In addition to concept pages and problem pages it has another kind of page - sections from a programming textbook. To represent the internal structure of these sections, ELM-ART uses fragment indexing with concepts.

6 Generalized Models: The Second Generalization

A generalized model-based view of different AH systems helps us to find close similarities between systems which initially look very different. However, this generalization is based on a domain model and covers only the systems which represent and use for adaptation the domain knowledge of the user. User knowledge is an important part of a user model, but not the only part. A user model can represent user goals, backgrounds, and other individual features. Many AH systems use this information for adaptation. To reveal the similarities between these systems and the systems covered by our first generalization, we suggest below a second generalization which can cover almost all existing AH systems.

6.1 Beyond User Knowledge: User Models in Adaptive Hypermedia Systems

The user's current goal is usually modeled by a mechanism similar to overlay knowledge modeling. As a rule, each system supports a set of possible user goals or tasks which it can recognize (HyPLAN, Orimuhs, PUSH, HYPERCASE, Hynecosum, HYPERFLEX). In some cases, the set of goals is very small and the goals are not related to each other (PUSH, HYPERFLEX). To model the current user goal, the system includes one of these goals into the user model. More advanced goal-based systems (Hynecosum, ORIMUHS, HyPLAN) use a more advanced representation of possible goals and current user goals. The most advanced representation of possible user goals is a hierarchy (a tree) of tasks (Hynecosum) which is quite similar to a level two domain model. The most advanced representation of current user goals is a set of "goal-value" pairs where the value is usually the probability that the corresponding goal is the current goal of the user (ORIMUHS, HYPERCASE, HyPLAN). It is very similar to the overlay knowledge model.

Individual features of a user such as background or experience, are usually modeled by a stereotype user model (MetaDoc, Anatom-Tutor, Hypadapter, EPIAIM, C-Book). A stereotype user model distinguishes several typical or "stereotype" users. For each dimension of user modeling the system can have a set of possible stereotypes. For example, MetaDoc uses two dimensions of classification and two sets of stereotypes (novice - beginner - intermediate - expert): one to represent the user's experience on general computer concepts, another to represent the user's experience with UNIX. A particular user is usually modeled by assigning this user to one of the stereotypes for each dimension of classification (for example, intermediate for general computer concepts and novice for UNIX). Stereotype user models can be represented as a set of pairs "stereotype-value", where the stereotype can be an experience stereotype, a background feature such as profession (EPIAIM), a prospect (Anatom-Tutor), or native language (C-Book). The value of each pair can either be "true" or "false" (which means that the user belongs or does not belong to the stereotype) or some probabilistic value (which represents the probability that the user belongs to the stereotype).

6.2 Generalized Models

It is easy to notice that the ways used to represent knowledge, goals, and individual features of the user are quite similar. Each of these representations is based on a space of possible characteristics (a domain model, a set of possible goals, or a set of possible stereotypes). The individual model of knowledge, goals, and features can be represented as a set of pairs "element-value" where the elements belong to the corresponding space. We suggest that a generalized domain model and a generalized overlay model to be considered. A generalized domain model is a set of *aspects* in which the aspects are all representabe characteristics of the user such as domain concepts, domain tasks and goals, and possible stereotypes. A generalized overlay user model is a set of pairs "aspect-value" in which the value in each pair can be "true" or "false" (indicating if the user has this characteristic) or some qualitative or quantitative value.

On the suggested level of generalization it is easy to find deep similarities between very different AH systems. It is interesting, that the three introduced groups of AH systems, each with different relationships between concepts and pages and with different hyperspace organization can be used to classify almost all existing AH systems if we let aspects be used as concepts to index pages and fragments or to form part of the hyperspace. As previously, systems from the same group will have a similar hyperspace structure and share specific adaptation techniques which are based on this structure.

Let us consider some goal-based and stereotype-based systems from this point of view. A relevant example for stereotypes is page indexing with user stereotypes (EPIAIM, Anatom-Tutor, ORIMUHS, C-Book). It is structurally similar to content-based indexing. The adaptation methods are also similar: this knowledge is used to select a page for a particular stereotype in exactly the same way as content-based indexing is used by a direct guidance technique to select a page containing a particular concept in ISIS-Tutor. An example of page indexing with relevant user tasks is provided by (Hynecosum). This knowledge is used in Hynecosum for adaptive hiding of non-relevant pages (i.e., pages which are not related to the current task) in the same way as it is usually done with concept indexing in ISIS-Tutor. An example of indexing page fragments with relevant user goals is provided by PUSH. Such indexing provides knowledge concerning which page fragments are relevant for a particular goal and which are not. This knowledge is used for content adaptation to present relevant text fragments and hide non relevant in the same way it is done for domain concepts in MetaDoc.

6.3 Mixing the Approaches

On the second level of generalization, we can find more systems which use several approaches. It is interesting, that different approaches are often used in the same system to work with different types of aspects. For example, Anatom-Tutor uses page indexing for backgrounds and fragment indexing for domain concepts. PUSH uses fragment indexing for goals and a knowledge-based approach for domain concepts. EPIAIM uses knowledge-based approach for domain concepts and page indexing for backgrounds.

Conclusions

This paper represents an attempt to analyze and to generalize the experience gained in the area of adaptive hypermedia. Our goal was to present this new area of research to a hypermedia-oriented audience by providing some systematization and generalization of the work done to date. In the first part of the paper, we tried to answer the most important questions: why and where do we need adaptive hypermedia systems and what can be adapted in these systems. In the generalization part we have introduced in two steps a generalized symmetric view on the different kinds of knowledge used in existing AHS. We consider such a view as very fruitful. This view enables us to find close similarities between systems which initially look very different. Using this view we were able to uncover three basic approaches to hyperspace structuring and three groups of adaptation techniques which were applied in nearly all existing AHS. We hope that our analysis and the proposed generalized symmetric view will be helpful for the designers of new AHS. The information provided in this paper can help them to select a proper way for hyperspace structuring and to design a set of adaptation techniques. We expect that our generalized view will help them to uncover the adaptation mechanism of many known adaptation techniques and let them proceed not only by re-using existing techniques, but also by designing deep analogies to known efficient techniques.

Acknowledgments

Part of this work is supported by a Grant from "Alexander von Humboldt" Foundation to the author .

References

Agosti, M., Melucci, M., and Crestani, F. (1995) 'Automatic authoring and construction of hypermedia for information retrieval'. *Multimedia Systems* **3** (1), 15-24.

Armstrong, R., Freitag, D., Joachims, T., and Mitchell, T. (1995) 'WebWatcher: A learning apprentice for the World Wide Web'. *AAAI Spring Symposium on Information Gathering from Distributed, Heterogeneous Environments*, Stanford, CA, http://www.isi.edu/sims/knoblock/sss95/mitchell.ps.

Beaumont, I. (1994) 'User modeling in the interactive anatomy tutoring system ANATOM-TUTOR'. *User Models and User Adapted Interaction* **4** (1), 21-45.

Boyle, C. and Encarnacion, A. O. (1994) 'MetaDoc: an adaptive hypertext reading system'. *User Models and User Adapted Interaction* **4** (1), 1-19.

Brusilovsky, P. (1996) 'Methods and techniques of adaptive hypermedia'. *User Models and User Adapted Interaction* **6** (in press).

Brusilovsky, P. and Pesin, L. (1994) 'ISIS-Tutor: An adaptive hypertext learning environment'. *JCKBSE'94, Japanese-CIS Symposium on knowledge-based software engineering*, Pereslavl-Zalesski, Russia, pp. 83-87.

Brusilovsky, P., Pesin, L., and Zyryanov, M. (1993) 'Towards an adaptive hypermedia component for an intelligent learning environment'. In: L. J. Bass, J. Gornostaev and C. Unger (eds.): *Human-Computer Interaction*. Lecture Notes in Computer Science, Vol. 753, Berlin: Springer-Verlag, pp. 348-358.

Brusilovsky, P. and Zyryanov, M. (1993) 'Intelligent tutor, environment and manual for physical geography'. *Seventh International PEG Conference*, Edinburgh, pp. 63-73.

Brusilovsky, P. L. (1992) 'Intelligent Tutor, Environment and Manual for Introductory Programming'. *Educational and Training Technology International* **29** (1), 26-34.

de La Passardiere, B. and Dufresne, A. (1992) 'Adaptive navigational tools for educational hypermedia'. In: I. Tomek (ed.) *Computer Assisted Learning*. Berlin: Springer-Verlag, pp. 555-567.

de Rosis, F., De Carolis, B., and Pizzutilo, S. (1993) 'User tailored hypermedia explanations'. *INTERCHI'93 Adjunct proceedings*, Amsterdam, pp. 169-170.

Debevc, M., Rajko, S., and Donlagic, D. (1994) 'Adaptive bar implementation and ergonomics'. *Informatica : Journal of Computing and Informatics* **18**, 357-366.

Encarnação, L. M. (1995) 'Adaptivity in graphical user interfaces: An experimental framework'. *Computers & Graphics* **19** (6), 873-884.

Fischer, G., Mastaglio, T., Reeves, B., and Rieman, J. (1990) 'Minimalist explanations in knowledge-based systems'. *23-th Annual Hawaii International Conference on System Sciences*, Kailua-Kona, HI, pp. 309-317.

Gonschorek, M. and Herzog, C. (1995) 'Using hypertext for an adaptive helpsystem in an intelligent tutoring system'. *AI-ED'95, 7th World Conference on Artificial Intelligence in Education*, Washington, DC, pp. 274-281.

Grunst, G. (1993) 'Adaptive hypermedia for support systems'. In: M. Schneider-Hufschmidt, T. Kühme and U. Malinowski (eds.): *Adaptive user interfaces: Principles and practice*. Amsterdam: North-Holland, pp. 269-283.

Hammond, N. (1989) 'Hypermedia and learning: Who guides whom?'. In: H. Maurer (ed.) *Computer Assisted Learning*. Lecture Notes in Computer Science, Vol. 360, Berlin: Springer-Verlag, pp. 167-181.

Hohl, H., Böcker, H.-D., and Gunzenhäuser, R. (1996) 'Hypadapter: An adaptive hypertext system for exploratory learning and programming'. *User Models and User Adapted Interaction* **6** (in press).

Höök, K., Karlgren, J., Wærn, A., Dahlbäck, N., Jansson, C. G., Karlgren, K., and Lemaire, B. (1996) 'A glass box approach to adaptive hypermedia'. *User Models and User Adapted Interaction* **6** (in press).

Kaplan, C., Fenwick, J., and Chen, J. (1993) 'Adaptive hypertext navigation based on user goals and context'. *User Models and User Adapted Interaction* **3** (3), 193-220.

Kaptelinin, V. (1993) 'Item recognition in menu selection: The effect of practice'. *INTERCHI'93 Adjunct Proceedings*, Amsterdam, pp. 183-184.

Kay, J. and Kummerfeld, R. J. (1994) 'An Individualised Course for the C Programming Language'. *Second International WWW Conference "Mosaic and the Web"*, Chicago, IL, http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Educ/kummerfeld/kummerfeld.html.

Kobsa, A., Müller, D., and Nill, A. (1994) 'KN-AHS: An adaptive hypertext klient of the user modeling system BGP-MS'. *4-th International Conference on User Modeling*, Hyannis, MA, pp. 31-36.

Kok, A. J. (1991) 'A review and synthesis of user modelling in intelligent systems'. *The Knowledge Engeneering Review* **6** (1), 21-47.

Mathé, N. and Chen, J. (1994) 'A user-centered approach to adaptive hypertext based on an information relevance model'. *4-th International Conference on User Modeling*, Hyannis, MA, pp. 107-114.

Micarelli, A. and Sciarrone, F. (1996) 'A case-based toolbox for guided hypermedia navigation'. *Fifth International Conference on User Modeling, UM-96*, Kailua-Kona, Hawaii, pp. 129-136.

Pérez, T., Gutiérrez, J., and Lopistéguy, P. (1995) 'An adaptive hypermedia system'. *AI-ED'95, 7th World Conference on Artificial Intelligence in Education*, Washington, DC, pp. 351-358.

Schwarz, E., Brusilovsky, P., and Weber, G. (1996) 'World-wide intelligent textbooks'. *ED-MEDIA'96 - World conference on educational multimedia and hypermedia*, Boston, MA.

Vassileva, J. (1994) 'A practical architecture for user modeling in a hypermedia-based information system'. *4-th International Conference on User Modeling*, Hyannis, MA, pp. 115-120.

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

Zhao, Z., O'Shea, T., and Fung, P. (1993) 'Visualization of semantic relations in hypertext systems'. *ED-MEDIA'93, World conference on educational multimedia and hypermedia*, Orlando, FL, pp. 556-564.