2 Adaptive Hypermedia

P. De Bra

Adaptive hypermedia makes it possible to author learning material once and generate a personalized learning experience for every user. The information that is presented, the way in which it is presented and the possible ways for the user to navigate through it can all be adapted. This chapter presents the most common adaptive hypermedia methods and techniques and shows examples of how they can be (and are) used in existing adaptive hypermedia systems and in adaptive online educational material.

2.1 Introduction

Learning, and education in general, has long been approached with a "one size fits all" attitude. When the Web became popular, authors of educational material started putting textbooks on the Web, thereby using hypertext links to give learners the freedom to study the material in any way (or order) they liked. The newly created navigational freedom was an illusion, however: the textbooks were not written with studying the topics out of the original order in mind. When you open a textbook at chapter 5, you do not really expect to understand everything as you know you have missed chapters 1 through 4. However, when you open an online textbook that shows links to each chapter, you do expect to be able to go directly to chapter 5 and understand what is written there. This is reasonable, because the existence of the link to chapter 5 suggests that the author has written the textbook in such a way that going directly to chapter 5 is "normal". However, writing a textbook in such a way that everything can be understood in any reading order is an impossible task. This is where adaptive hypermedia (or AH) comes to the rescue.

The basic idea in AH is that by observing what the learner does (what she reads, what scores she obtains on tests, etc.), the system can get a detailed impression of the user's knowledge. The presence or absence of knowledge about a topic can be used to insert or remove explanations, depending on whether they are still needed, and it can also be used to guide the user towards topics she can study next. Brusilovsky (2001) has presented an overview of the AH field as it existed in 2001.

Adaptive hypermedia methods and techniques can be used to overcome the difficulties caused by the navigational freedom that is typical for hypertext. AH can be used to guide users towards the topics they are ready to study at any given time. It can be used to insert prerequisite explanations when needed, and to show additional detailed information to interested and advanced learners. It is also possible to change the guidance depending on the learner's cognitive style (or learning style) or to present topics differently, depending on that style, e.g., through text, images or video, when desired.

In order to make it possible to adapt learning material (or any other type of information) to the user, the system must get to know the user in two ways: (a) the system must follow the learning process in order to decide which topics are studied by the user, and (b) the system must know the learning style of the user, or any other aspect of the context in which the system is being used. We will concentrate on adaptation to the evolving knowledge of the user, and not on the learning styles (which are supposedly more stable and only lead to some stereotypical forms of adaptation).

Adaptive hypermedia techniques are appearing in all kinds of applications, such as personalized search engines, personalized advice on online shopping sites, recommender systems (e.g., a personalized TV guide), context-sensitive help systems, etc. However, adaptive educational hypermedia is the most popular application area for this technology, both in research and in actual full-fledged applications.

Adaptive educational applications rely heavily on a structured description of the subject domain of the application or course. In this chapter, we will base adaptation mostly on such a structural description (and only sometimes on domain-independent aspects). The chapters on metadata and on ontologies explain how such descriptions can be created using standards that often came into existence much later than the adaptive hypermedia systems and applications we describe in this chapter.

2.2 Adaptation Methods in Educational Hypermedia

Adaptation is used either to alleviate problems that users encounter (if there is no adaptation) or to improve an otherwise already normal, or at least acceptable, experience.

In most courses, there are not only links to several chapters right from the starting page, but there are also a lot of cross-references between different chapters/sections. In a normal paper textbook, the author knows whether such a reference is a forward or a backward reference, and thus whether or not the reference is to a concept already encountered when following the linear order of the book. In an online course with navigational freedom, the author cannot know whether, for a given learner, a reference will be a forward or backward reference. However, it isn't difficult for a system to track a user's path through the course text, and thus to know whether a reference leads to new information or to a previously visited page or concept. There are now two possibilities:

1. The author can create two versions of a page or the relevant part of a page (also called a fragment): one for learners who have studied the required concepts before, and one for learners who have not (and thus may need a more introductory version or an extra explanation). The system can choose which version to present, based on the user's knowledge when following the reference link.
2. The author can also determine what the conditions are (in terms of knowledge of concepts) under which it is a good idea to follow a cross-reference link. The system can check whether the learner has the required knowledge and can hide, disable or annotate the link accordingly.

These two possibilities represent the two largest categories of adaptation techniques: content adaptation (called adaptive presentation in Brusilovsky (2001)) and link adaptation (called adaptive navigation support in Brusilovsky (2001)).

Content adaptation, based on the learner's knowledge, typically comes in three forms (see also Brusilovsky (2001)):

1. When a page refers to or uses a concept the learner does not yet know, and of which at least some understanding is needed, a short prerequisite explanation can be inserted. This lets the learner continue with the chosen subject, rather than requiring a jump to the prerequisite concept in order to study that in detail first.

2. Sometimes the current concept can be elaborated upon in case a related concept is already known, or when the knowledge level of the learner is already high. For these "expert" users an additional explanation can be given that is beyond the level of the average learner (at the time of visiting the current page).
3. Sometimes an interesting comparison is possible with another concept, but only if that other concept is already known. Such a comparative explanation between the concepts can automatically be shown on the page of the second of the two concepts studied by the learner, regardless of which of the two is second in the chosen reading order.

An additional issue with content adaptation is the issue of stability. When a prerequisite explanation is shown on a page, and the learner later revisits the page after obtaining all prerequisite knowledge, should the prerequisite explanation remain or should it be removed? And if a comparative explanation is shown on the page explaining the second concept, should the comparison also be shown on the page explaining the first concept when that page is revisited later? In our online course on the topic of hypermedia (http://wwwis.win.tue.nl/2L690/), we apply no stability and learners, when revisiting a page, have never commented that the contents had changed. Figure 2.1 shows an example from this course. This of course does not mean that nobody has ever noticed a change.

- Before reading about Xanadu the URL page shows:

  …
  In Xanadu (a fully distributed hypertext system, developed by Ted Nelson at Brown University, from 1965 on) there was only one protocol, so that part could be missing.
  …

- After reading about Xanadu this becomes:

  …
  In Xanadu there was only one protocol, so that part could be missing.
  …

Fig. 2.1. Content adaptation in the hypermedia course

Link adaptation comes in two forms: the system can make suggestions as to which links to follow or avoid, and the system can change links so that they appear, disappear or lead to a different destination depending on the learner's knowledge:

When the system knows the user's goal, because that is defined as a task in the course, because it is specified by the user, or because the system can determine a "preferred" reading order, it may sort or reorder links that appear in a list. Link sorting is always performed by search engines, and it can be made adaptive by taking information about the user's knowledge or goals into account.

In a list of links, it is not only possible to use sorting to suggest a preference for a certain link destination. Links that are deemed inappropriate because the user is missing too much prerequisite knowledge, for instance, can be removed entirely. This adaptive link removal technique should be used with great care. When used on a list of chapters of a course text it may suggest that the course consists of only a few chapters and the learner may be (unpleasantly) surprised to find out there are many more chapters that only become available later. (In a very informal user study we found that users strongly preferred a list with some disabled links over a list with the unsuitable links removed.)

In normal paragraphs of text, sorting and removal are not possible (as they would disrupt the flow of the text). It is still possible to indicate desirability of links by using adaptive link annotation. The suitability of links has already been indicated using link anchor colors in the pre-Web ISIS Tutor (Brusilovsky and Pesin 1994). Later systems, including ELM-ART (Weber and Brusilovsky 2001) and its descendent, Interbook (Brusilovsky et al. 1998), use icons such as colored balls (● ● ●) to indicate whether a link is interesting or not. They may also use other icons such as checkmarks (✓ ✓ ✓) to indicate a knowledge level. Figure 2.2 shows a partial screen shot from an Interbook application.

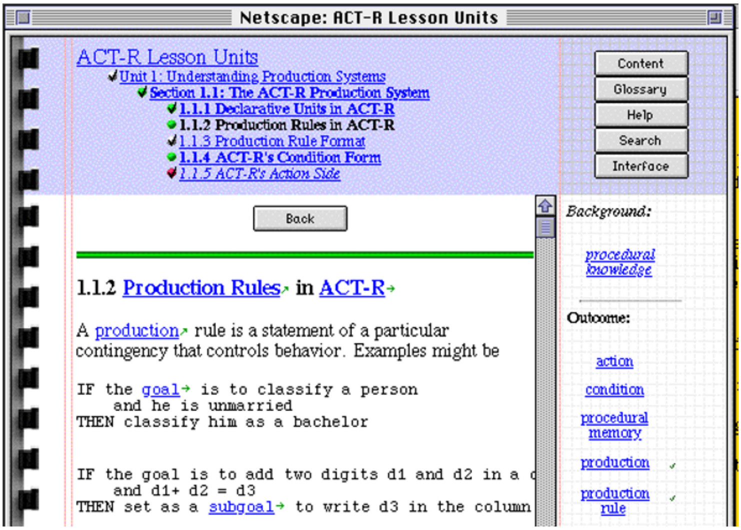

Fig. 2.2. Link annotation in Interbook (*colored balls and checkmarks*)

- Instead of indicating an undesired link through an icon such as the red ball (●), systems like AHA! (De Bra and Calvi 1998; De Bra et al. 2006) can also adaptively hide and/or disable links. When the link anchor is shown in the same color as normal text (and not underlined), the learner will most likely not notice it is there. In addition, the anchor text may also be turned into plain text, removing its link functionality. The advantage of hiding annotation is that the learner is not distracted by non-recommended links. The drawback is that links that are initially hidden and start to appear later may surprise or even confuse the learner, especially when it turns out there is unexpectedly more material to learn.
- Another way to manipulate links is by changing their destination. When a page contains a "next" button for instance, this may lead to the most appropriate page to read next. The learner's goals and knowledge may influence the decision as to which page should be next. But the destination for other links may vary as well. For a novice, a link may lead to an introduction about a topic whereas the same link may lead to a detailed or advanced discussion for a knowledgeable learner. Depending on how such adaptive link destinations are implemented, the different link destination may be visible to an observant user (when the URL is different) or not (when the same URL is used to serve different page content).

2.3 Overlay User Models

Adaptive educational hypermedia applications or courses rely on the existence of a structured description of the domain. The AHAM model (De Bra et al. 1999) shown in Fig. 2.3 shows a domain model and a user model at the core of an AH application. These two models are connected through a teaching model, which we later renamed to adaptation model. AHAM is based on the Dexter Hypertext Reference Model (Halasz and Schwartz 1994), which shows a five-layer architecture of hypermedia applications. We concentrate on the middle layer: the storage layer.

Structure is often thought of as being hierarchical. A course text typically consists of a number of chapters, each divided into sections and subsections, down to small chunks of information such as a definition, a theorem, an example or an explanation of a certain topic. As the learner builds up knowledge of the course by studying the small topics, thus gathering knowledge of larger concepts and eventually the whole course, it makes sense to keep track of the evolving knowledge using a similarly structured user model, which we call an overlay model. (The user model is an overlay of the domain model.)

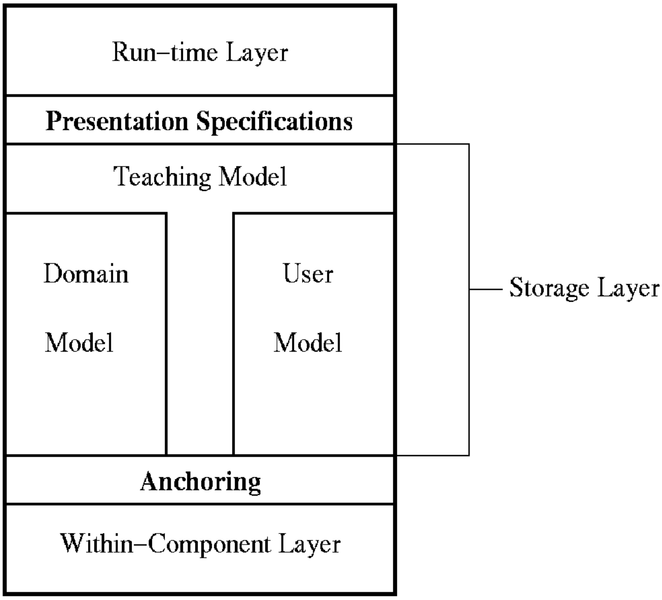

Fig. 2.3. The AHAM reference model

2.3.1 Registering Changes in Learners' Knowledge

In adaptive educational hypermedia applications there are essentially two ways in which the system can know that a learner has acquired knowledge about a topic: by observing that the learner has read a page or by evaluating answers to a test (often multiple-choice).

- Reading pages: When learning is not done through lectures (in class or on video), it is typically at least partially done through reading. (It can also partly be done through exercises and assignment work.) Since the learning material in an online course often consists of web-pages, a server-side program can register the pages a learner accesses, and conclude that the learner will study (and later, has studied) these pages. Whether this means that the learner then has full knowledge of the corresponding concept(s) is impossible to deduce with absolute certainty. If there is missing prerequisite knowledge, the learner will most likely not understand everything the page explains. And if the learner visits the page only briefly, she will not have read everything. A simple approach to decide whether the learner has studied a page is to also register the time between page accesses, in order to deduce the reading time. A more elaborate approach is taken by the AdeLE project (Mödritscher et al. 2006), which uses eye-tracking to decide what the learner has actually seen. It is not yet clear whether registering reading time or even eye movements enables adaptation that is so much better that the overall learning outcome of a course is significantly better.

- Multiple-choice tests: Verification of learners' knowledge has been done through exams for a very long time. A thorough analysis of answers to difficult questions is very hard, if not impossible, to automate. No existing software can analyze an answer to a question like "Why did Germany start World War II?" (or almost any other *why* question). However, skilled teachers often manage to create multiple-choice questions with several believable wrong answers (and one or more correct answers as well). We won't go into detail on how to create good multiple-choice questions, but note that such tests are a popular way to check a learner's knowledge level about a concept. Through such tests an AH system may verify whether its belief about the learner's knowledge is justified.

When the learning process stops, the learner's knowledge of the studies' concepts does not remain constant. Bielikova and Nagy (2006) suggest repeating prerequisite knowledge at the start of a new lesson as a remedy for the learner's imperfect memory (and make other suggestions as well). They have implemented a model for the "decay" of human memory in an extension of the AHA! system. The extended system registers the access time of each page and uses an activity-in-memory attribute of concepts to register that a concept is being used and thus refreshed. Bielikova and Nagy (2006) use an English-Slovak vocabulary application as an example, and each time a word is used it becomes active in memory. In applications with a more complex concept structure, it may be more difficult to implement the activity of refreshing the learner's memory, especially for higher-level concepts.

2.3.2 Deducing Knowledge About Higher Level Concepts

By registering page accesses (or even more detailed information about the learner's interaction with the adaptive application), the AH system creates a very fine-grained model of the learner's knowledge of the subject domain of a course. Using this detailed information in determining whether the learner has enough prerequisite knowledge to visit a page or chapter is impractical. Adaptation is often based on the knowledge of higher-level concepts (with “higher-level” meaning higher up in the concept hierarchy that describes the subject domain of a course). There are essentially two ways to implement a user model in which the knowledge of higher-level concepts is represented. From publications about systems, it isn’t always clear which implementation choice was made in these systems:

- Systems like KBS Hyperbook (Henze and Nejdl 1999) use an inference engine to deduce estimated knowledge of higher-level concepts from knowledge of pages. The basic idea is that by knowing how much knowledge each page (or lowest-level information item) contributes to a higher-level concept (perhaps going through some intermediate levels of concepts as well), the knowledge of a high-level concept can be quickly calculated from its contributing page-level knowledge values. The calculation is only done when the knowledge value is actually needed, for instance, in order to decide whether a prerequisite for a concept is satisfied.
- An alternative is to use the concept hierarchy to transfer knowledge from pages to higher-level concepts each time the knowledge level of a page changes. This may imply that the knowledge of a high-level concept is updated several times without that value being needed. On the other hand, when the knowledge value is needed in order to decide on a prerequisite, it is immediately available and need not be inferred from all the lower-level values. In the AHA! system (De Bra et al. 2006), attributes of concepts (including the knowledge attribute) can be declared volatile or persistent. The value of a volatile attribute is calculated when needed, whereas the values of persistent attributes are stored. AHA! can thus support both implementation choices.

Figure 2.4 shows a screenshot of the Graph Author tool, which is part of AHA! (De Bra et al. 2006). It is used to create the conceptual structure of a course. The left part shows (part of) the concept hierarchy. The right part shows a graph of concept relationships, which in this case are (almost all) prerequisites. The concept hierarchy defines how knowledge is propagated up the concept hierarchy, from pages to sections to chapters to the whole course. This hierarchy thus deals with the conceptual content level. The concept relationship defines a pedagogical structure. In order to decide on how to perform adaptation, we have to consider the following questions: What happens when the learner studies pages with missing prerequisites? How much knowledge is required before we consider a prerequisite satisfied (or should we also consider partially satisfied prerequisites)? Are prerequisites necessarily transitive? We will now deal with each question in turn:

- When a prerequisite for a page is not satisfied, we may expect the learner to not fully understand what the page is trying to explain. If a bit of missing prerequisite knowledge can be compensated for by inserting an additional explanation into the page, we should consider the prerequisites for the page as a whole to be satisfied. When the page as a whole is not suitable for the learner, an AH system typically assigns partial knowledge to the knowledge attribute of the concept. The standard behavior of AHA! (which can be changed) is that revisiting a page while it is still not recommended does not increase the knowledge beyond that partial-knowledge level (which is set to 35 by default, with 100 meaning full knowledge). In Interbook, on the other hand, revisiting a non-recommended page does keep increasing the knowledge, so full knowledge can be obtained by studying the page several times. Links to non-recommended pages are typically hidden or annotated, so the learner clearly knows when she is following a link to such a page. Depending on the AH system used (or how it is configured), the non-recommended status of a page might also be visible on the page itself. In Interbook, a green or red horizontal line near the top of the page indicates whether the page is recommended (green) or not (red).
- When knowledge is propagated from pages to higher-level concepts, a wide range of knowledge values becomes possible. In the hypermedia course 2L690, the first three chapters form a prerequisite for the remaining advanced chapters. Since in AHA! (used to serve the course), knowledge is a value between 0 and 100, we have to decide how much knowledge of the combined first chapters is needed before the learner is advised, or at least allowed, to start studying the advanced chapters. In AHA! prerequisites by default require 50% knowledge in order to be satisfied. (This number can be changed by an author for any or all of the prerequisites.) The choice of 50% knowledge for prerequisites and 35% knowledge for studying non-recommended pages has an interesting consequence: it makes prerequisites transitive. If A is a prerequisite for B and B for C, then 50% knowledge of A is needed to make B recommended, and 50% knowledge of B is needed to make C recommended. Should the learner visit concept B while not knowing enough about A, then only 35% knowledge of B will be recorded, meaning that the prerequisite B for concept C will not be satisfied. So because A was not sufficiently known, concept C will not be recommended even when B is visited. Making prerequisites transitive makes creating a graph of prerequisite relations easy, and results in a graph that is easy to understand.

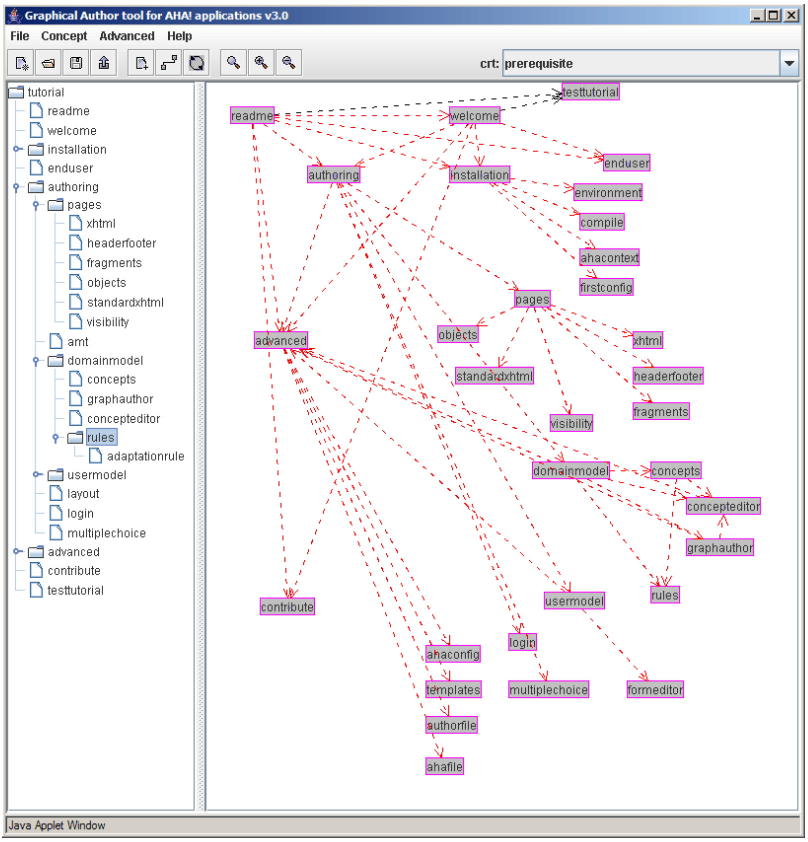

Fig. 2.4. Concept hierarchy and concept relationship graph

2.3.3 Which User Model State to Use in Adaptation?

As we have seen above, the user's knowledge state is updated each time a page is read (or a concept visited, or a test performed), and the user's knowledge state is used to determine how the presented information should be adapted (as shown in Figs. 2.3 and 2.4). But how do the user model updates and the adaptation interact with each other? When the user clicks on a link (anchor), is the requested page adapted to the current user-model state, and is the state updated afterwards? Or is the user-model state updated first and the page adapted to the new state? Both approaches have positive and negative side-effects:

- Suppose we want to use a specific example in several places in a course. The first time a page containing the example is visited, the learner should see the whole example. Subsequent visits to the example (on the same page or on other pages) may briefly recall or summarize the example instead of explaining it completely. If the presentation of the example depends on the knowledge of the example, it will be presented completely the first time, when the knowledge is still 0, and briefly on subsequent visits, when the knowledge is above 0. In order for this scheme to work, information (fragments, pages and concepts) must be presented before the user model is updated.
- Suppose we want to present a page A containing all the prerequisite knowledge for a concept B that should be studied immediately after. Page A will therefore contain a link to B. Since the user will have all the required prerequisite knowledge for B, the link pointing to B should be recommended to this user. But the link to B will only be recommended if the link is adapted to the user's knowledge state after reading page A. Since a page, once presented, cannot be changed while the user is reading it, the link must already be recommended when the user starts reading. Hence the knowledge of A must be updated before the page is presented.
- It is clear that no AH system can implement both approaches simultaneously. The second approach is the most common one. When knowledge is updated first, and adaptation performed second, the link adaptation works as expected. As for the prerequisite explanation (or example), an easy workaround is to count the number of visits, starting at 0, and know that upon the first visit to a concept its visit counter will already be 1 when the concept is presented.

2.4 Adaptation to Other Aspects Besides Knowledge

Brusilovsky (2001) states that in AH, adaptation can be done to a user's individual traits, which includes all user features that together define a user as an individual, and to the user's environment. We will briefly consider adaptation to the learner's cognitive style or learning style in this chapter (and, for simplicity, consider these equivalent), and also look at adaptation to the user's browsing device and network characteristics.

2.4.1 Adaptation to Learning Styles

Many theories on cognitive and learning styles exist. This results in a plethora of terms, many of which are more or less equivalent, and in a lot of advice on how to adapt to a learner's cognitive abilities and preferences, (some also conflicting, and some even suggesting that it may be better to adapt *against* the learner's cognitive style rather than *for* it (Smith et al. 2002)) in order for learners to also train their not-preferred skills. We base our suggestions in this chapter mostly on an extensive report by Coffield et al. (2004) that considers 71 learning-style models worthy of consideration (out of over 100 models they found). Some of the most important models for our description are that of Dunn and Dunn (1978), Witkin et al. (1977), and Honey and Mumford (1992). We highlight a few aspects of these models below:

- Sensory modalities (as used in Dunn and Dunn's study (Dunn and Dunn 1978), for instance) describe how learners best sense or perceive information. Auditory students learn best through verbal lectures. They like listening to and discussing with others. Visual students learn through seeing. This can be through pictoral (images, video) or textual material. Tactile or kinesthetic students learn best through a hands-on approach. When forced to read from paper, they underline or mark with a magic marker.
- Psychological modalities describe how learners best learn (Dunn and Dunn 1978). Analytic students prefer to learn one detail at a time, and later put the parts together to complete the big picture. Global students, on the other hand, first need to see the whole meaning (the big picture) before they deal with the individual details that together form that whole. This classification by Dunn and Dunn roughly corresponds to the field-independent and field-dependent styles of Witkin.
- Honey and Mumford suggest the following terms for the different ways in which a learner likes to learn. An activist likes to "have a go, and see what happens". These students like to experiment, often in groups, in order to deduce a theory. A reflector likes to "gather information and mull things over". These students gather data, analyze it and delay reaching conclusions. A theorist likes to "tidy up and reach some conclusions". These students think things through in logical steps, for models, and only later try to apply them. The pragmatist likes "tried and tested techniques". These students seek quick decisions, resulting in practical step-by-step procedures that work, without long discussions and theory.

- We chose to discuss only these learning styles because we can supply some concrete adaptation for them, as done by Stash and De Bra (2004) and Papanikolaou et al. (2003).
- In order to accommodate the verbalizer and imager styles, an application can offer the same (or similar) information in textual form and through images or video. Auditory users can have the text spoken out loud. Technically this adaptation is very easy using content adaptation, but it does require the information to be produced several times, once for each media type.
- Global/analytical or field dependent/independent users can be helped by changing the order in which concepts are presented. Global and field-dependent users need an overview before studying details. In order to achieve this, the system can guide the users in a breadth-first way through the top-level concepts. These users wish to know what a whole course is about before studying the details of a single topic. Analytical or field-independent users do not need such an overview. They can either navigate freely or receive guidance to navigate in a depth-first way. The required guidance can easily be generated using adaptive link hiding or annotation.
- To accommodate activists, reflectors, theorists and pragmatists, the learning material needs to be divided into objects of different types. Papanikolaou et al. (2003) suggest that an activist start with an activity, such as an exercise. A reflector should see some examples first, followed by the theory, and then solve an exercise. The theorist starts with theory, then gets some examples to illustrate the theory and finally performs some exercises. The pragmatist starts with an exercise, and then goes to examples and finally the theory. If the different pieces are small, they can be presented as fragments of a page and sorted according to the learner's cognitive style. If the pieces are too big to be presented all at once, the system can present the first item and place a sorted list of links to the other items at the bottom of the page that presents the first item.

2.4.2 Adaptation to the Browsing Environment

When dealing with online adaptive systems, the information content and structure should be independent of the actual presentation form. Viewing a website on a large monitor, with a computer on a high-speed connection, provides an experience that is very different from that of viewing the site on a personal digital assistant (PDA) with relatively low processing power and a slow wireless connection. Adaptation (other than the common adaptability by selecting different presentation skins) involves mostly the aspects of size and of bandwidth.

- Adaptation to varying screen sizes involves more than scaling objects. A decision has to be made when to change the presentation structure globally or locally. For instance, text may have to be summarized in order to fit on the screen. A page that displays a number of images may have to be replaced by a slideshow (automated or through links) showing one image at a time. Adaptation to the browsing environment may thus involve more than just adapting each information object, but may involve changing the navigation structure as well. Figure 2.5 from (Fiala et al. 2004) illustrate such adaptation. Research in this area has resulted in the AMACONT (Fiala et al. 2003) and Cuypers engines (van Ossenbruggen et al. 2001).
- Adaptation to variations in bandwidth has been attempted by various research teams. We note the work of Muntean and McManis (2006) because it evaluates the effect of different "quality of experience" factors in courseware based on an extension of the AHA! system. Quality of experience is not just a function of bandwidth but is defined using performance metrics download time, round-trip time, throughput and user tolerance for delays.
- In the study, it turned out that reducing the image or video quality in order to avoid delays allowed students to complete a course (part) equally well and faster, whereas they did not consider the quality of the presentation insufficient. Clearly users are willing to accept less perfect images and videos in order to receive them faster and without interruption.

2.5 Summary/Conclusions

Adaptive hypermedia enables automatic personalization of online course material. In this chapter, we have seen that adaptation can be applied to the information content (in order to ensure that the learner receives understandable), to the navigation structure (either by restricting possible navigation by removing or disabling links or by guiding the user through link annotation and sorting), and also to the layout and presentation (in order to match the capabilities of the browsing device and network).

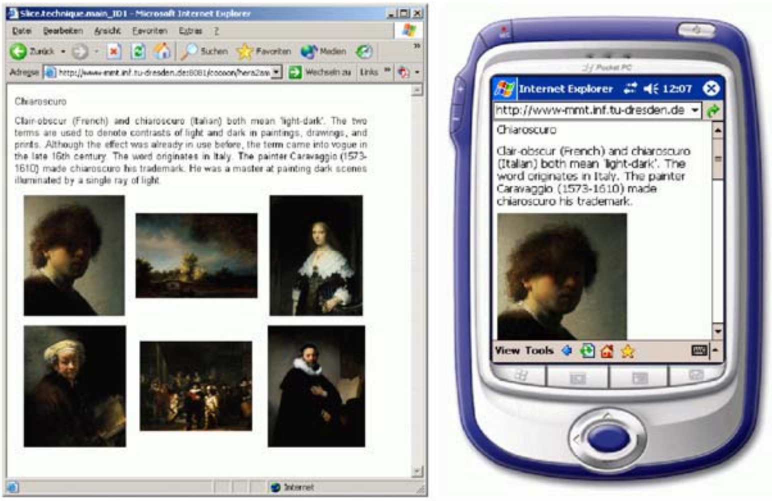

Fig. 2.5. Adaptation to the browsing device (screen size)

Adaptation is typically performed based on an estimate of the learner's knowledge of the subject domain. In order to do so, an overlay user model is constructed and maintained. But adaptation to the learner's cognitive abilities and preferences is also possible. We have shown how an AH system can adapt to various learning styles. Researchers do not yet agree on the question of whether or not adaptation to learning styles is always beneficial, so there is certainly room for future work in this area.

References

Bielikova, M., Nagy, P. (2006). Considering human memory aspects for adaptation and its realization in AHA! In *EC-TEL 2006: Technology Enhanced Learning* (pp. 8–20), LNCS 4227. Berlin Heidelberg New York: Springer.

Brusilovsky P. (2001). Adaptive hypermedia. *User Modeling and User Adapted Interaction*, 11(1–2), 87–110.

Brusilovsky, P., Pesin, L. (1994). ISIS-Tutor: an adaptive hypertext learning environment. In *Japan-CIS Symposium on Knowledge Based Software Engineering, Proc. of JCKBSE'94* (pp. 83–87).

Brusilovsky, P., Eklund, J., Schwarz, E. (1998). Web-based education for all: a tool for developing adaptive courseware. *Computer Networks and ISDN Systems (Proceedings of Seventh International World Wide Web Conference)*, 30(1–7), 291–300.

Coffield, F., Moseley, D., Hall, E., Ecclestone, K. (2004). Learning styles and pedagogy in post-16 learning: a systematic and critical review. Learning and Skills research center. http://www.lsda.org.uk/files/pdf/1543.pdf

De Bra, P., Calvi, L. (1998). AHA! An open adaptive hypermedia architecture. *The New Review of Hypermedia and Multimedia*, 4, 115–139.

De Bra, P., Houben, G. J., Wu, H. (1999). AHAM: a dexter-based reference model for adaptive hypermedia. *ACM Conference on Hypertext and Hypermedia* (pp. 147–156).

De Bra, P., Smits, D., Stash, N. (2006). The Design of AHA!. In *Proceedings of the ACM Conference on Hypertext and Hypermedia* (p. 133) and on-line version at http://aha.win.tue.nl/ahadesign/.

Dunn, R., Dunn, L. (1978). *Teaching Students Through Their Individual Learning Styles: A Practical Approach*. Reston, VA: Reston.

Fiala, Z., Hinz, M., Meissner, K., Wehner, F. (2003). A component-based approach for adaptive dynamic web documents. *Journal of Web Engineering*, 2, 058–073.

Fiala, Z., Frasincar, F., Hinz, M., Houben, G. J., Barna, P., Meissner, K. (2004). Engineering the presentation layer of adaptable web-information systems. In *Proceedings of the International Conference on Web Engineering* (pp. 459–472), LNCS 3140. Munich, Germany, July 2004.

Halasz, F., Schwartz, M. (1994). The dexter hypertext reference model. *Communications of the ACM*, 37(2), 30–39.

Henze, N., Nejdl, W. (1999). Adaptivity in the KBS hyperbook system. In *Second Workshop on Adaptive Systems and User Modeling on the WWW.*

Honey, P., Mumford, A. (1992). *The manual of Learning Styles*. Maidenhead: Peter Honey.

Mödritscher, F., Garcia-Barrios, V. M., Gütl, C., Helic, D. (2006). The first AdeLE prototype at a glance. In *Proceedings of ED-MEDIA* (pp. 791–798).

Muntean, C., McManis, J. (2006). The value of QoE-based adaptation approach in educational hypermedia: empirical evaluation. In *Proceedings of the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 121–130), LNCS 4018. Dublin.

van Ossenbruggen, J., Geurts, J., Cornelissen, F., Hardman, L., Rutledge, L. (2001). Towards second and third generation web-based multimedia. In *The Tenth International Conference on the World Wide Web*, WWW10, (pp. 479–488). Hong Kong: ACM.

Papanikolaou, K., Grigoriadou, M., Kornilakis, H., Magoulas, G. D. (2003). Personalising the interaction in a web-based educational hypermedia system: the case of INSPIRE. *User Modeling and User-Adapted Interaction*, 13(3), 213–267.

Smith, W., Sekar, S., Townsend, K. (2002). The impact of surface and reflective teaching and learning on student academic success. In *Proceedings of the seventh Annual European Learning Styles Information Network Conference* (pp. 407–418). Ghent.

Stash, N., De Bra, P. (2004). Incorporating cognitive styles in AHA! (The Adaptive Hypermedia Architecture). In *Proceedings of the International Conference Web-Based Education* (pp. 378–383). Innsbruck, Austria.

Weber, G., Brusilovsky, P. (2001). ELM-ART: an adaptive versatile system for web-based instruction. *Journal of Artificial Intelligence in Education* 12, 351–384.

Witkin, H., Moore, C., Goodenough, D., Cox, P. (1977). Field-dependent and field-independent cognitive styles and their educational implications. *Review of Educational Research*, 47(1), 1–64.