AHA : a generic adaptive hypermedia system

Eindhoven University of Technology
Department of Mathematics and Computing Science

Proceedings of the Second Workshop on Adaptive Hypertext and Hypermedia

edited by

P. Brusilovsky and P. De Bra

98/12

ISSN 0926-4515

editors: prof.dr. R.C. Backhouse
prof.dr. J.C.M. Baeten

Reports are available at:
http://www.win.tue.nl/win/cs

Computing Science Reports 98/12
Eindhoven, August 1998

Ninth ACM Conference on Hypertext and Hypermedia

Hypertext'98

Pittsburgh, USA, June 20-24, 1998

Second Workshop on Adaptive Hypertext and Hypermedia

Workshop Theme

With the growing size, complexity and heterogeneity of current hypermedia systems, especially the World Wide Web, comes the need to provide more flexible mechanisms for delivering information to the user. That is, we require mechanisms which can modify documents on-the-fly in order to take the user's needs into account. Static hypertext documents suffer from an inability to be all things to all people; document and multimedia authors must write multiple documents for different users rather than a single document which can dynamically modify its content in order to address a particular user's knowledge or the context of delivery.

A possible remedy for the negative effects of the traditional "one-size-fits-all" approach in the development of hypermedia systems is to equip them with the ability to adapt to the needs of their individual users. A possible way for achieving adaptivity is by modeling the users and tailoring the system's interactions to their goals, tasks and interests. In this sense, the notion of adaptive hypertext/hypermedia means a hypertext or hypermedia system which reflects some features of the user and/or characteristics of his/her system usage in a user model, and utilizes this model in order to adapt various aspects of the system's output to the user.

This workshop was intended as an inter-disciplinary exploration into adaptive hypertext and other kinds of flexible hypertext systems. It aimed to draw together a number of research groups taking different approaches to adaptive and flexible hypertext systems, in order to promote the cross-fertilization of ideas and highlight the prospects for future collaboration. The target research areas for the second Adaptive Hypertext and Hypermedia workshop include:

- Adaptive hypertext and hypermedia (adaptive navigation support and adaptive presentation within an existing hypertext network of documents)
- Dynamic hypertext (employing text generation or other techniques to dynamically create both the hypertext network and the documents within the network as the user requests them)
- Information retrieval and filtering (the use of information retrieval or other techniques to determine the relevance of the nodes within a static hypertext network for the individual user)
- Intelligent hypertext (automatic linking, similarity-based navigation, concept-based navigation).

Some related events which have been held in the past include:

- Flexible Hypertext Workshop, held at the Eighth ACM International Hypertext Conference (Hypertext'97)
- Intelligent educational systems on the World-Wide Web, held in conjunction with the 8th World Conference on Artificial Intelligence in Education (AI-ED'97)
- Workshop on Adaptive Systems and User Modeling on the World Wide Web, held in conjunction with the Sixth International Conference on User Modeling (UM'97)
- Workshop on User Modeling for Information Filtering on the World Wide Web, held in conjunction with the Fifth International Conference on User Modeling (UM'96)
- Workshop on Adaptive Hypertext and Hypermedia held in conjunction with the Fourth International Conference on User Modeling (UM'94)

More information about adaptive hypertext systems can be found on Adaptive Hypertext and Hypermedia Home Page at http://www.education.uts.edu.au/projects/ah/.

Workshop Focus

There has been a significant amount of research in this area over the past five years (see workshop theme for more information), but two recurring issues have become increasingly important, and these were the focus of this workshop:

- **World Wide Web**: The Web is both a new application area and a new challenge for adaptive and flexible hypertext research. Web-based applications are expected to be used by a much greater variety of users than any earlier standalone application. Web-based applications naturally need to be flexible; a Web-based hypertext application which is designed with a particular class of users in mind may not suit users of other classes. At the same time, developing adaptive and flexible hypermedia systems on the Web, we can investigate a number of new opportunities such as user model sharing or the use of group models.
- **Evaluation**: One of the key issues which arose from the first flexible hypertext workshop and which has been aired again recently on the adaptive hypertext mailing list is the importance of the evaluation of adaptive and flexible hypertext systems. In particular, since the main goal of these systems is to maximize the suitability of a document to the user's knowledge and needs, evaluation is an essential aspect in the development of these systems. However, very little research has been done which confirms the advantages of such systems or which demonstrates how this evaluation process might be done.

Workshop Format

The workshop ran for one full day before the main Hypertext'98 conference. The number of attendees was limited to 20-25 in order to encourage participation in workshop discussions. Participation was on the basis of submitted position papers or by invitation. The workshop included a limited number of paper presentations and general group discussions. Group discussions focusd on the issues raised in the position papers, as well as on some focus questions. A workshop dinner was also be organized to encourage informal discussion.

The program included:

- Welcome and Introduction
- Workshop Sessions consisting of:
  - 1 to 2 Position Paper Presentations (15-30 minutes), and
  - Group Discussion on the Issues Raised (30-60 minutes)
- Closing:
  - Planning for post-workshop activities
  - Conclusions and Wrap-up
- Workshop Dinner

The proceedings have been compiled into a technical report after the workshop and are available on-line at http://wwwis.win.tue.nl/ah98/.

Program Committee:

- Peter Brusilovsky (chair), School of Computer Science, Carnegie Mellon University, USA
- Paul De Bra, Faculty of Mathematics and Computing Science, Eindhoven University of Technology, the Netherlands
- John Eklund, Sydney University of Technology, Australia
- Kristina Höök, Human-computer interaction and language engineering group (HUMLE), Swedish Institute of Computer Science, Sweden.
- Alfred Kobsa, GMD FIT German National Research Center for Information Technology, Germany
- Maria Milosavljevic, CSIRO Mathematical and Information Sciences, Sydney, Australia
- Johanna D. Moore, Dept. of Computer Science, University of Pittsburgh, USA
- Jon Oberlander, Human Communication Research Centre, University of Edinburgh, U.K.
- Julita Vassileva, University of Saskatchewan, Canada

Ninth ACM Conference on Hypertext and Hypermedia

Hypertext'98

Pittsburgh, USA, June 20-24, 1998

Second Workshop on Adaptive Hypertext and Hypermedia

PROCEEDINGS

Edited by

Peter Brusilovsky Human-Computer Interaction Institute, School of Computer Science, Carnegie Mellon University, USA

Paul De Bra Department of Computing Science, Eindhoven University of Technology, The Netherlands

Papers

Short papers and posters

Position statements

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

AHA: a Generic Adaptive Hypermedia System

Paul De Bra*
Department of Computing Science
Eindhoven University of Technology
PO Box 513
NL 5600 MB Eindhoven
The Netherlands
debra@win.tue.nl

Licia Calvi
Department of Romance Languages
University of Antwerp
Universiteitsplein 1
B 2610 Wilrijk
Belgium
calvi@uia.ua.ac.be

*Paul De Bra is also affiliated with the University of Antwerp and the "Centrum voor Wiskunde en Informatica" in Amsterdam.

Abstract: Since early 1994 the course ``2L670: Hypermedia Structures and Systems" has been available through the Web. It is currently part of the curriculum for computing science and related fields at six universities in The Netherlands and Belgium, and occasionally offered to students from other institutes as well. The software used to deliver this course over the Web has evolved from a static hyperdocument to a versatile adaptive hypermedia system that can be used for many purposes. We call the system AHA, which stands for Adaptive Hypermedia Architecture.

The core of the AHA system consists of an engine which maintains a user-model based on knowledge about concepts. Knowledge is generated by reading pages and by taking tests. The (textual or multimedia) content of a page can be adapted by means of *fragment variants*. The (hyper)links are annotated by changing the color of the link anchor (the link text or the border in case of images). The color scheme can be configured by the author and overridden by the user, to choose between *link annotation* and *link hiding*. When desired, *link removal* can also easily be implemented.

The adaptive hypermedia software can be used for all kinds of applications, not necessarily limited to education (which is what its primary purpose was). It is written (almost) entirely in Java and thus portable to different computing platforms. It is freely available for non-commercial use.

keywords: user modeling, conditional content, link hiding, link annotation.

1. Introduction

Several adaptive hypermedia applications have been introduced over the past few years. The overview article by Brusilovsky [B96] names most of these, and describes the adaptive techniques used in each application. In most cases the software used for maintaining the underlying user models and for generating the adaptive content and link structure is tied closely to the single application for which that software was developed. A notable exception is the Interbook system [BSW96b], which is a descendant from ELM-ART [BSW96a]. While ELM-ART is only a Lisp course, which incidentally also uses Lisp for the software that maintains the user-model and generates the adaptive content, Interbook can be used to create courses on different topics. Still, although more general than ELM-ART, Interbook is still aimed specifically at educational applications. It uses a fixed frames structure to represent an (adaptive) table of contents, a set of known or required concepts, a content page, etc. The front-end of such a system (the user-interface which is realized using HTML) is tied closely to the back-end (the engine which maintains the user model and performs the adaptation).

This paper presents the development at the Eindhoven University of Technology [DC97], which goes one step further: the software for adaptive hypermedia which was originally developed for a course on ``Hypermedia Structures and Systems" [DB94] has been made very generic by concentrating all functionality in the back-end, and leaving presentation issues to the author. The adaptive system, still called AHA (for Adaptive Hypermedia Architecture), can be used with HTML presentations with or without frames. The engine which maintains the user model can be used to generate conditional text and to adapt the link structure through link removal, link hiding and link annotation. In [B96] six application areas for adaptive hypermedia are mentioned. AHA supports five of them: *on-line information systems*, *on-line help*, *educational hypermedia*, *institutional hypermedia* and *personalized* *views*. The one application area which is somewhat more difficult to realize using this software is *information retrieval hypermedia* (because that requires sorting of links, which AHA does not support).

Section 2 presents the structure of a user model and how it is used to generate adaptivity. In Section 3 we show how adaptive content can be realized using standard HTML. Section 4 illustrates the different ways to adapt the link structure through the use of link classes. In Section 5 we present the global architecture of the back-end and show how this engine interacts with the browser to ensure that presented pages always correspond to the correct user model.

2. User Modeling in AHA

Adaptive systems try to anticipate the needs and desires of the user. Any knowledge a system has about the user is based on that user's (prior) actions. The system may simply monitor what a user is doing or it may ask questions. Intelligent tutoring systems are typical: they build a user model based on what reading material is offered to the user, and validates that model by means of (mostly multiple-choice) tests. A simple way to represent a user model is by means of a set of pairs (c,v) where c is a *concept* and v is a *value* which indicates how much (or little) the user knows about that concept. However, a set of subject-value pairs need not represent knowledge about a concept. User preferences can be represented in the same way. Authors can offer variants of hyperdocuments by presetting some pairs. In the sequel we will always use the term *concept*, keeping in mind that it may mean *preference* as well.

The data types used for indicating knowledge (or preferences) are simple: Some systems allow many values, for instance a ``percentage'' (or integer values between 0 and 100), some have a few numeric or named values, like *no knowledge*, *read about* and *knows about*, and others have just Booleans (*true* and *false*). Some applications use only a few concepts (which they have knowledge tests for) while others use many concepts (maybe even one for every page).

Every representation system with a finite number of values can be simulated by a system with just Booleans. For systems with many values, like percentages, such a simulation would be impractical. However, we know of no system which actually calculates percentages to the point that more than a few discrete values can be obtained, or to the point that every discrete value has a different influence on the adaptation. When a system supports three or four values per concept, each concept can be replaced by two. For instance, instead of concept *something* we could create concepts *read-about-something* and *knows-about-something*. In the AHA system we opted for Boolean values.

In AHA knowledge about a concept is generated either by reading a (single) page or by taking a test. This implies that concepts are fairly fine-grained: if the user must read five pages to achieve some desired state, each of the five pages must be associated with a different concept, and the five concepts together define the desired goal.

Apart from a set of known concepts, the AHA software also maintains a logfile for each user. For each time the user accesses a page there are two log entries: one for the start and one for the end of the period the user (supposedly) reads the page. (In this way the reading time for each page is logged.) For each test the score is stored as well. The log is part of the user model so it can be used to mark links to pages the user already read differently from links to unread pages. Most WWW-browsers also change link colors, but this behavior would interfere with the adaptive linking. Also, users may be able to clear the browser's history and associated link coloring scheme and thereby disable the guidance the adaptive hypertext software tries to provide through link colors (see Section 3).

While concepts can be used to represent user preferences the AHA system implements color preferences differently. Colors of links can be selected by the user. A Boolean representation would not be desirable, so explicit color values are stored in the user model.

3. Adaptive Content

Depending on the user's knowledge state information on a given subject may need to be presented in different ways. Students who are first reading about hypertext for instance may be confused when they see the term ``node'' whereas the word ``page'', used in the same context, would be meaningful to them, and probably sufficiently accurate in an introductory text. In the course text for the course ``2L670: Hypermedia Structures and Systems'', the students must first visit a ``readme'' page with instructions on how to use the courseware and how to configure their

WWW-browser. Therefore a short paragraph which tells students to go to the readme page is prominently displayed, along with a link to that page. After reading the instructions the textual content of the start-page of the course is changed automatically. The pointer to the readme page is removed from the top, and a small reminder at the bottom of the page is all that remains.

Such adaptive changes to a page, using what Brusilovsky [B96] calls *fragment variants*, are non-trivial to realize in World Wide Web, because HTML does not allow for text fragments which are *conditionally* made visible or hidden. It would be possible to use *Dynamic HTML* for this purpose, but this would lead to HTML documents that are complicated (because of the JavaScript or VBscript code they need to contain) and thus difficult to author.

AHA implements adaptive content in HTML by means of a preprocessor that filters content fragments by means of conditionals encoded in structured HTML comments:

```
<!-- if definition and history -->
  This part appears if the two "concepts"
  definition and history are both known
  according to the user model.
<!-- else -->
  If this is not the case then this
  alternative is presented instead.
<!-- endif -->
```

Another example is the use of such conditionals is to combine viewgraphs with and without comments in a single source, like:

```
  <LI>This is a viewgraph item.
<!-- if verbose -->
  And this is some additional comment which is
  only shown when "verbose" is true.
<!-- endif -->
```

Similar constructs can be used to choose between a text paragraph, an image, video, etc.

Because the comments have no ``meaning" in HTML, there is no need to respect the proper nesting of conditionals and HTML tags. The following example would not be allowed if the comments would be meaningful HTML tags:

```
  <LI>This is an item in an unordered list.
<!-- if interrupt-list -->
  </UL>
  This conditional text interrupts the list.
  <UL>
<!-- endif -->
  <LI>This is the next item in the list.
```

This flexibility does imply that the author must ensure that the proper begin and end tags for lists or other HTML constructs are included under all circumstances.

4. Adaptive Linking

Depending on the user model an adaptive hypertext system will guide the user towards some "desired" pages, and away from pages that contain information which is not relevant at that time, or for which the user does not have the necessary foreknowledge. Brusilovsky [B96] distinguishes five types of adaptive linking: *direct guidance*, *adaptive link sorting*, *adaptive link annotation*, *adaptive link hiding* and *map adaptation*. Most adaptive hypertext systems offer only one (or maybe two) of these features. Interbook [BSW96b] for instance concentrates on *link annotation*, meaning that desired links are marked differently from undesired links, in this case by means of a colored (big) dot. (It also offers direct guidance through a ``teach me" feature.)

The course 2L670 offered only link hiding, but in AHA this restriction has been lifted. Calvi [C98] has subdivided link hiding into three subclasses:

- (Pure) *link hiding* means that links may be hidden by making the link anchor indistinguishable from the surrounding text (i.e. black in most cases). The link remains functional, so users who know that the link is there can still use it.
- *Link removal* means that the link anchor is not only made to look like normal content, but the link is removed as well. This means that even a user who knows where the link should be cannot follow the link.
- *Link disabling* means that the link is not made invisible but its link functionality is removed. The user still sees the link, but that link doesn't work. (The original version of AHA, used for a version of the course 2L670 inadvertedly used link disabling. Users almost unanimously disliked this feature.)

The latest version of AHA directly supports *link annotation* and *link hiding*, but *link removal* can easily be simulated, and *direct guidance* can be implemented with a little more authoring effort.

- Adaptive link annotation in AHA is realized by means of link classes which are turned into different colors by using *cascading style sheets*. Because of differences in author and user preferences the courseware can be configured to use any color scheme (for desired, undesired, neutral and external links) the author desires, and the user may override these choices through a ``setup" HTML page and a CGI-script. Links that may be desired or undesired at times are marked in the HTML source of the pages using a link class we call ``conditional". The AHA preprocessor translates this to the classes ``good" or ``bad" depending on the user model.
- Adaptive link hiding in AHA is achieved by choosing the color black for one of the link types, typically the ``undesired" links. When the link text is black it cannot be distinguished from the surrounding text. Only by moving the mouse pointer over the link text may the user become aware of the presence of a link, because the browser normally alerts the user by displaying the name of the destination of a link in a message line. In AHA the author chooses between link annotation and link hiding by means of the color scheme. The user can override this choice through a setup form.
- Adaptive link removal in AHA is achieved by turning the HTML anchor tag into conditional text, like in the following example:

  ```
  <!-- if desired -->
    <a href="...">
  <!-- endif -->
    here is the link anchor text
  <!-- if desired -->
    </a>
  <!-- endif -->
  ```

  This trick, with the two ``if" clauses is only needed if you wish to avoid writing the link anchor text twice. A simpler, but redundant form would be:

  ```
  <!-- if desired -->
    <a href="...">here is the link anchor text</a>
  <!-- else -->
    here is the link anchor text
  <!-- endif -->
  ```

- Direct guidance can be achieved through conditional content in a similar (but more laborious) way. Direct guidance means for instance that the adaptive hypertext system offers a button which leads to a suggested page to read next. The URL to which the link leads can be filled out using ``if" comments, depending on logical combinations of concepts. The following example shows how a ``next" button leads to a definition if that hasn't been read before, and otherwise leads to a theorem.

  ```
  <!-- if definition -->
    <a href="theorem.html">
  <!-- else -->
    <a href="definition.html">
  <!-- endif -->
  next
  </a>
  ```

  Offering direct guidance is difficult because the author needs to be aware of what the possible best pages to go to are, depending on each user's knowledge state. Although this may seem to be an authoring problem and not a problem with the AHA software, direct guidance can be delivered automatically (to some extent) by a system by checking the dependencies between concepts. AHA currently does not offer this feature, but Interbook for instance does (through a ``teach me" button).

5. Architecture of the Adaptive Hypermedia System AHA

The AHA system delivers HTML pages that consist of four parts:

- Each page starts with a generated header, which contains the definition of the style sheet with the link coloring scheme.
- The page body starts with a header, which the author creates once, and which is automatically included in every page.
- The page content is authored using a standard HTML editor. (It only consists of a part between the <body> and </body> tags.)
- Every page ends with a common footer, also created by the author and included automatically.

The pages, as they are delivered to the user, are assembled from these parts by means of a CGI-script (or optionally a Fast-CGI script). Configuring the AHA software is done by setting a few variables in a supporting shell script. Variables to set include the course title, the directory on the Web server, the email address of the author, the name of the Web server, etc. (All this is fairly straightforward.)

Each HTML page the author creates (optionally) starts with two comments:

- The first line indicates which Boolean expression on concepts needs to be true in order for a links to that page to be desired. Example:

```
<!-- requires readme and history but not (intro or definition) -->
```

- The second line indicates which concept(s) become ``known" after reading this page:

```
<!-- generates history -->
```

From these lines in every page the system creates:

- A *dependency file*, containing pairs of page names and expressions. Each line indicates which expression must be true for links to that page to be desired.
- A *concept list*, containing all concepts. This list is used in a ``setup" page: the user can change the user model by setting the value for each concept to true or false.

the AHA script recognizes three types of links:

- Links to an absolute URL (containing a `:') are called *external*. These links are colored *red* by default. When an external link has been followed the color changes to *gray* by default.
- Links to a relative URL, of class ``unconditional" (or of no class) are considered to always be desired. The link class is changed to ``good" and the link is initially colored *blue* by default. If the link's destination is a page that was read before the link class becomes ``neutral" and the color changes *purple* by default.
- Links of class ``conditional" are investigated by comparing the user model to the required concept-expression. If the expression evaluates to true the link class is changed to ``good" or ``neutral" (as in unconditional links) and the color becomes *blue* or *purple* by default. If the expression is false, the link class becomes ``bad" and the corresponding color is *black* by default.

If the link anchor consists of an image instead of text it is the color of the image border that changes.

The user can change this color scheme through a ``setup" page. Also, a list of pages that have been read and a list of pages still to be read can be displayed. Multiple-choice tests are available in two versions: one that requires the user to answer correctly (and offers no explanatory feedback to wrong answers) and one that offers explanations of errors and does not require multiple tries. All these features are available through the same CGI-script that generates the pages.

When a user ``logs on" (by filling out a form) the system generates URLs to the pages that contain the name of the CGI-script that generates pages, the identity of the user, and the name of the requested page. Thus a URL may look like:

```
http://servername.domain/2L670/cgi/get.cgi/user-id/page.html
```

Although pages are individualized, they are currently not password protected. (The login form requires a password only to prevent users from accidentally logging on as another user. This may happen occasionally because student's use numbers as their user-id.)

The AHA software is freely available upon request. It is written entirely in Java (1.1) and should be easy to integrate into any Unix-based Web-server that supports CGI or Fast-CGI. (Porting to other platforms may be a bit tricky because of the use of a few Unix utilities to generate the dependency file and concept list, and of a shell script to read environment variables, which is not possible in Java 1.1.)

Visitors are welcome to a few applications of this software: ``http://wwwis.win.tue.nl/2L690/" is the address of the most recent version of the course on ``Hypermedia Structures and Systems". ``http://wwwis.win.tue.nl/2M350/" is the address of a course on ``Graphical User-Interfaces" which uses the adaptivity mostly to set a user-preference for viewgraphs versus full text. ``http://wwwis.win.tue.nl/IShype/" is the address of a small set of information pages (in Dutch) about writing a master's thesis in our Information Systems Group. It illustrates the use of AHA in an application which is not an on-line course.

A final, but important remark is that, like probably all adaptive hypermedia systems, AHA does not reduce the need for skilled authors. While adaptive hypermedia in general, and AHA in particular, offers the possibility to avoid certain usability problems caused by the user taking paths through a hyperdocument which the author did not foresee, it certainly does not eliminate usability problems automatically. Although we have not done this, it is possible to base an adventure game on AHA, using the adaptive features to make it *more difficult* to find your way by changing the contents of pages and the link structure in ways which the player does not expect.

6. Summary and Conclusions

The adaptive hypermedia system AHA offers adaptive content through fragment variants and adaptive link presentation through link annotation, link hiding and/or link removal. Its focus is on simplicity and the use of standards. The AHA engine is written in Java, and works as a CGI or FCGI script. Hyperdocuments for AHA are written in standard HTML. HTML comments are used for delimiting fragment variants and link classes are used to perform the link annotation.

Keeping the AHA system simple, portable and based on standards comes at a price: there are currently no tools for supporting the design of an adaptive hyperdocument, including the definition of concepts and dependencies between concepts. Also, writing documents in HTML may be problematic for some authors. Tools for generating HTML from other document formats are available but not always sufficiently general to support the HTML features used by AHA.

7. References

[Brusilovsky 1996]
Brusilovsky, P., Methods and Techniques of Adaptive Hypermedia. In: User Modeling and User-Adapted Interaction 6: 87-129, Kluwer academic publishers, 1996.
(URL: http://www.contr ib.andrew.cmu.edu/~plb/UMUAI.ps)

[PB96a]
Brusilovsky, P., Schwarz, E., Weber, G., "ELM-ART: An intelligent tutoring system on World Wide Web", Proceedings of the Third International Conference on Intelligent Tutoring Systems, ITS-96, Montreal, 1996. (Lecture Notes in Computing Science, vol. 1086, pp. 261-269).

[PB96b]
Brusilovsky, P., Schwarz, E., Weber, G., "A Tool for Developing Adaptive Electronic Textbooks on WWW",

Proceedings of the WebNet'96 Conference, pp. 64-69, San Francisco, 1996.
[C98]
Calvi, L., "A Proficiency-Adapted Framework for Browsing and Information Filtering in Web-Based Educational Systems. Methodological Implications for Language Learning on the WWW", Doctoral Thesis, University of Antwerp, 1998.
[DB94]
De Bra, P., "2L690: Hypermedia Structures and Systems", Course at the Eindhoven University of Technology, 1994-1998. http://wwwis.win.tue.nl/2L690/
[DC97]
De Bra, P., Calvi, L., "Creating adaptive hyperdocuments for and on the Web", Proceedings of the WebNet'97 Conference, pp. 149-165, Toronto, 1997.

The Value of Adaptivity in Hypermedia Learning Environments: A Short Review of Empirical Evidence

John Eklund
School of Educational Psychology, Measurement & Technology
Faculty of Education
The University Sydney NSW Australia
j.eklund@edfac.usyd.edu.au
http://www.education.uts.edu.au/staff/john_eklund/

Peter Brusilovsky
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA

plb@cs.cmu.edu
http://www.contrib.andrew.cmu.edu/~plb/home.html

Abstract: Adaptivity is a particular functionality of hypermedia that may be applied through a variety of methods in computer-based learning environments used in educational settings, such as the flexible delivery of courses through Web-based instruction. Adaptive link annotation is a specific adaptive hypermedia technology whose aim is to help users find an appropriate path in a learning and information space by adapting link presentation to the goals, knowledge, and other characteristics of an individual user. To date, empirical studies in this area are limited but generally recognised by the Adaptive Hypertext and Hypermedia research community as critically important to validate existing approaches. The purpose of this paper is two fold: to briefly report the results of an experiment to determine the effectiveness of adaptive link annotation in educational hypermedia (fully described in Brusilovsky & Eklund, 1998), and to situate the study within a summarised survey of the literature of adaptive educational hypermedia systems and the empirical studies that have been undertaken to evaluate them.

Keywords: Adaptive, evaluation, hypermedia, navigation, user model, WWW

Adaptive systems using the technique of link annotation

An important technique for navigation support in educational hypermedia is *adaptive annotation* technology. This annotation augments links with a comment which provides users with information about the current state of the nodes behind the links (Brusilovsky, Pesin & Zyryanov, 1993; de La Passardiere & Dufresne, 1992; Hohl, Böcker & Gunzenhäuser, 1996; Schwarz, Brusilovsky & Weber, 1996). It has been convincingly argued (Brusilovsky & Pesin, 1995; Zeiliger, Reggers & Peeters, 1996; Brusilovsky & Eklund, 1998) that this method is desirable for educational hypermedia, and this is the particular technology used in the adaptive learning environments which feature in the review of empirical work covered in this paper. Link annotations can be provided in textual form or in the form of visual cues, for example, using different icons, colours, font sizes, font types etc. This technique can be considered a "soft form" of hiding (Brusilovsky, 1996), but it is considerably more cognitively coherent in that it consistently presents to the user a full view of the hyperspace.

In a comprehensive review of adaptive hypermedia which emphasised adaptive navigation support, Brusilovsky (1996) identified thirteen adaptive hypermedia systems in the educational domain: ANATOM-TUTOR (Beaumont, 1994), C-BOOK (Kay & Kummerfield, 1994), <CLIBBON> (Clibbon, 1995) , ELM-ART (Schwarz et al, 1996), ISIS-TUTOR (Brusilovsky & Pesin, 1994), ITEM/PG (Brusilovsky, Pesin & Zyranov, 1993), HYPERTUTOR (Perez et al, 1995) , LAND USE TUTOR (Kushniruk & Wang, 1994) MANUEL EXCEL (de La Passardiere & Dufresne, 1992), SHIVA (Zeiliger, 1993), SYPROS (Gonschorel & Herzog, 1995), HYPADAPTER (Hohl et al, 1996) and HYPERCASE (Miccarelli & Sciarrone, 1996). Of course we must add to this list InterBook (Brusilovsky, Schwarz & Weber, 1996) which features in this paper. In addition, recent work in adaptive hypermedia systems development also includes 2L670 (De Bra, 1996; Calvi & De Bra, 1997), 2M350 (De Bra, 1998 submitted), WEST-KBNS (Eklund & Sawers, 1996), AST (Specht et al, 1997), ADI (Schöch, Specht & Weber, 1988 in press), ELM-ART-II (Weber & Specht , 1997), and AHM (da Silva et al, 1998 submitted). Of the adaptive education systems listed above, a number use adaptive annotation, either alone or in combination with other adaptive technologies. These are HYPADAPTER, ELM-ART (as well as the ELM-ART derivatives), ISIS-TUTOR, HYPERCASE, AST, ADI, 2M350 and MANUEL EXCEL.

Empirical studies with adaptivity in educational hypermedia

A survey of the literature on adaptive navigation support in educational hypermedia reveals relatively few empirical studies, or at least papers with a non-trivial evaluation component, that serve to set the context for the study reported in this paper. In the first of these, de La Passardiere & Dufresne (1992) conducted experiments with MANUEL EXCEL, the first working system that used adaptive link annotation. Their work focussed on the value of history-based mechanisms such as a three-stage footprint (unseen, partially seen and completed) and the use of adaptive advice. The advice was calculated from an error diagnosis of the student's answer to tests and the number of trials. These two user-model based components of the learning environment were found through the use of audit trails to significantly increase "exploration of explanations". In other words, students were responding to feedback and investigating the responses to their answers. de La Passardiere & Dufresne's (1992) experiment used a "with and without" adaptivity approach, comparing the paths of a group of 20 students using the environment without the adaptive components to a group of 30 students using it with the adaptive advisement and cognitively coherent three stage footprints. It appears that in this experiment the two environments are largely different. One offers feedback while the other does not, and this is a significant difference in the basic functionality of the systems, regardless of adaptive navigation support. The experiment was valuable in the sense that it gave the researchers an indication of how link annotation might work in an educational environment, and an intuition that it could be valuable but it is hardly an empirical result in favour of adaptive link annotation. This work stresses link annotation as being more than just a physical coherence of the information: that footprints should reflect the cognitive structure of the hyperspace. The importance of offering an individualised historical progression, and a diversified dialogue with the learner, were as relevant as being able to let the user "affect" the information space in which they find themselves.

Brusilovsky & Pesin (1995) conducted one of the earliest and most significant studies with the adaptive link annotation and hiding mechanisms in the ISIS-Tutor. ISIS-Tutor is a reasonably old system (first version completed in 1992) which used colours and symbols on a non-graphic DOS screen to adaptively mark the links on a page to related pages. The environment used annotations of "not ready to be learned", "ready to be learned", and "in work", very similar to ELM-ART and InterBook for which Brusilovsky was partially responsible. The first version of the system was reported in (Brusilovsky, Pesin & Zyryanov, 1993). The more advanced version used in the experiment was reported in (Brusilovsky & Pesin, 1994a; Brusilovsky & Pesin, 1994b). The first empirical work with ISIS-Tutor was completed in 1994. The paper (Brusilovsky & Pesin, 1995) written immediately after the experiment reported the very first processed results: "...the overall number of navigation steps, the number of repetitions of previously studied concepts, the number of transitions from concept to concept and from index to concept are seriously less for [adaptive hypermedia]." Later in 1995 these results were analysed in more detail and finally published in (Brusilovsky, 1997). A third, more comprehensive analysis was completed in 1996 and belatedly reported in (Brusilovsky & Pesin, 1998) which presented the carefully processed results from the same experiment. This work showed that due to a great variety of navigation styles within each group very few dependent variables show significant difference between the with/without adaptivity groups. At the same time, the work demonstrated by analysis of variance that the overall number of navigation steps, as well as the number of unforced repetitions of concept and task pages were significantly fewer with the adaptive version of the tutor. No difference was found between annotation and hiding techniques. No difference was also found between all three groups for the quality of mental maps of the hyperspace developed by students.

In 1996 ISIS-Tutor was used again with almost one hundred subjects but at the time of writing this paper (1998) the publication of the results is forthcoming.

In a less formal context, Eklund conducted trials with a group of 30 university students studying HTML using a system called WEST-KBNS (Eklund & Sawers, 1996) which produced a annotated global overview map of the nodes in the courseware. Using questionnaires and focus groups it was found that students thought that the navigable and annotated global map was a useful aid to moving through the hyperspace. Using a scale of 1=strongly disagree to 5=strongly agree a mean of 3.9 and a standard deviation of 1.3 was obtained for students' impressions of the usefulness of this map. Focus groups confirmed that students generally found the feature worthwhile, and the annotations on it which gave them an individual suggestion of where to proceed after each test valuable. Again this work offers no firm empirical evidence in favour of adaptive link annotation but is of value in the sense that it used a range of data obtained from sources that were not generally used in other trials. The use of these sources acknowledges that a qualitative approach can provide very useful and positive feedback about a particular interface feature, in this case link annotation. Audit trails provide a detailed picture of user paths but often the interpretation of them fails to account for a wide range of individual differences in users, and hence they are best used in conjunction with other information sources.

In a recent study using ELM-ART-II, Weber & Specht (1997) compared the mean number of pages and tests that visitors to the ELM-ART-II Web site made (those who made a non-trivial visit), both with and without adaptive link annotation and adaptive curriculum sequencing. It was found that there was a statistically significant effect of adaptive curriculum sequencing at $p<0.05$ but no significant difference in link annotation. Here, it is the ability of these adaptive interface tools to motivate the users to proceed that is being evaluated. Further, Weber & Specht (1997) report that adaptive link annotation had no significant effect on the number of navigation steps. There are two difficulties with this experiment - the small number of participants (a total of twenty-four across four groups), and the fact that users were totally unscreened except for the fact that they visited more than the first few pages. Users' knowledge of the content (LISP programming), the time spent on the computer before the session, the speed of their connection, their motivation, and so forth, are all almost completely uncontrolled variables. They qualify their results as not necessarily supporting the case against adaptive link annotation on the basis of these obvious experimental limitations.

Weber, Brusilovsky & Specht (1998 submitted) used the ELM-ART system to test the effect of both adaptive curriculum sequencing, as mentioned, and link annotation. No significant effect on the number of navigation steps was found through the use of adaptive link annotation. There was some evidence that the adaptive component was useful for beginners, but the small effect faded as the users became more familiar with the system. This is not to say that as a group, the subjects did not benefit from using the system, as they clearly did in this case. They concluded that link annotation appears useful to novice users as it provides an initial support for them in the use of an unfamiliar interface.

Brusilovsky & Pesin's (1995; 1998) study of the efficiency of adaptive link annotation in the ISIS Tutor, as described above, remains the only convincing piece of empirical work with a favourable outcome for the adaptive navigation support technique of link annotation in educational hypermedia. The overall result is that learning using link annotation is faster, more goal-oriented, and significantly reduces the number of steps to cover the hypermedia. It many ways it points to the value of link annotation in the information retrieval domain. Is the fact that students used a "more efficient" path with adaptive navigation support particularly relevant in an educational setting? If it is, could a more efficient path be achieved through a careful layout in a static hyperspace, or through the inclusion of other more traditional non-adaptive strategies such as the use of case studies, interactive examples, or problem solving?

Adaptive annotation in InterBook

InterBook is a system for authoring and delivering adaptive electronic textbooks on WWW (Brusilovsky, Schwarz & Weber, 1996). InterBook is a well-published environment and it is the intention here to provide just the briefest overview. InterBook-served electronic textbooks are based on a domain model represented in a form of concept network. All sections of an electronic textbook are indexed with domain concepts. ie., for each section a list of related concepts with their *roles* is provided (each concept can be either an outcome concept or a background concept). For each domain model concept, an individual student's knowledge model stores some value which is an estimation of the student knowledge level of this concept. InterBook traces student actions and keep the student model updated. Using the student model, InterBook distinguishes several educational states for each unit of ET: the content of the unit can be known to the student (all outcome concepts are learned or well-learned), ready to be learned (all prerequisites are learned or well-learned), or not ready to be learned (some prerequisite concepts are not yet learned). According to the idea or adaptive annotation, the icon and the font of each link presented to 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. In InterBook, a red ball and italic font means "not ready to be learned", a green ball and bold font suggests "ready and recommended", while a white ball means "learned, no new information". A checkmark is added for already visited units. The same way can be used to distinguish and show several levels of students knowledge of the concepts shown on the concept bar. In InterBook, no annotation means "unknown", small checkmark means "known" (learning started), medium checkmark means "learned" and big checkmark means "well-learned".

The Study

In a study involving 25 undergraduate teacher education students in an educational computing elective at the University of Technology, Sydney, students were exposed to two chapters of a textbook about the database and spreadsheet mdoules of the integrated package ClarisWorks, and used the InterBook system both with and without adaptive link annotation. The experiment was created to be in a real-world teaching and learning context, with the use of InterBook as an integral part of a university subject. The goal of this experiment was to assess what impact, if any, user-model based link annotation would have on student's learning and on their paths through the learning space, in this realistic situation. Tests of knowledge were carried out, audit trails and questionnaires were gathered and the results analysed. The experiment took place over a four-week period. In the first two hour session, students were introduced to InterBook and its features explained to them. They used the system for an hour, and answered a questionnaire about its features. This questionnaire showed that almost all students were familiar with what each of the buttons and annotations meant. They were then free to use the system at any time during the following week. In the second session, students were randomly divided into two groups of equal size, one group receiving the link annotation (12 subjects), while the other group did not (13 subjects). They were allowed access to the chapter of the textbook on databases which had been authored into InterBook, and they completed a questionnaire. Students had access to the database chapter for the following week. In the third session, students took a multiple choice test on the database section of the textbook. For the purpose of brevity we omit the description of the following sessions which are outside the goal of this paper.

Results in brief

All student transactions with InterBook were recorded by audit trails which were used to examine how particpants navigated through InterBook with and without ANS. For each user these trails showed the number of times they selected a green ball, red ball, as well as their use of all the other features of the interface. This information was collated with the test results and the questionnaires. We initially found using a two-sided t-test that students with adaptive link annotation performed significantly worse in the database section test and that there was no difference in the spreadsheet section. This unexpected result suggested that further investigation was needed. First, students who did not spend a reasonable time with the system were excluded, and once this was done we found no significant difference in the test results for the group with adaptivity and the group without. A clear correlation (R=0.670) was found between the agreement rate and score in the database tests: the more students agree with system's suggestion, the better is the score - for the group receiving link annotation.

For the database section, four separate groups were identified on the basis of their agreement rate: high positive, low positive, low negative and high negative, and all the users were categorised into one of these groups. We found that for the ANS group a better agreement rate results in generally better test results.

Using sequential navigation (i.e., continue-back) versus non-sequential navigation is known behaviour exposed by novices in hyperspaces. However, we were able to show that ANS encourages the novices to use annotated non-sequential tools more often. This was achieved using the count of hits on annotated links such as table of contents links versus non-annotated links such as Continue button.

This implies that ANS provides the learner with a non-linear guide through the learning space, and learners are more likely to use non-sequential paths with adaptive link annotation. It again reflects the student's trust in the annotations - ANS provides some security for those users who would like to follow non-linear paths but might be afraid of becoming lost.

When the number of hits on pages of various 'states' (suggested, etc) is examined, it is clear that students prefer to visit ready-to-be-learned pages than those which are annotated as "no information". Students spent approximately twice as much time reading ready-to-be-learned pages than reading all other pages combined. So a green and unchecked page is one that students read most. This is naturally due to the fact that the Continue button was used most of the time, bringing the user into a page with a "ready to be learned" status.

What we observed was a mixture of two effects: an effect of page state and an effect of annotation. This means that students of the ANS group noticed the annotations, and may have decided apriori to spend less time on "nothing new" pages and more on those annotated as "not ready". The effect of annotation clearly dominates in the case of "not-ready" pages. Those rare users who selected a page with full understanding that this page is not ready are willing to allocate significantly more time (ANOVA, p=0.012) for reading this page. Again, they understood how the system worked and trusted the integrity of the annotations.

These results are far more fully described in Brusilovsky & Eklund (1998) and Eklund, Brusilovsky & Schwarz (1998, in press). The trial had some clear limitations: Students had varying levels of access to the learning materials in printed form, and some obviously relied on this for their learning. Also, since the non-annotated continue link (which takes the user to the next page) was used in over 90% of transactions, so the effect of link annotation on student paths was relatively small. Further trials with a modified experimental design are currently underway at The University of Technology, Sydney.

Conclusion

The literature on the evaluation of adaptive link annotation in educational hypermedia is limited and sometimes characterised by anecdotal evidence from a variety of less formal experimental designs. Further, the evaluation of adaptive hypermedia systems has generally has been noted by writers in the field as a much under-investigated area. In this paper we have sought to provide a brief survey of adaptive educational systems which offer link annotation, and reviewed the work that has been undertaken to evaluate their adaptive component. We also offer some definite but qualified empirical evidence from a recent study that adaptive link annotation is advantageous to learners who choose to accept the navigation advice. We found that they will exhibit more exploratory paths in the closed corpus educational context, as well as achieve a greater understanding of the content. There is clearly a need for continued studies in this area.

Acknowledgments

We wish to express our thanks to Elmar Schwarz, University of Karlsruhe, Germany, the primary developer of InterBook, for his advice about this research and for making available InterBook as an experimental vehicle. Also Associate Professor Ken Sinclair and Dr Mike Bailey of The University of Sydney for their advice about experimental procedure. Part of this work is supported by grants from Alexander von Humboldt Foundation and MacDonnell Foundation.

References

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

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

Brusilovsky P (1997) Efficient techniques for adaptive hypermedia. In: C. Nicholas and J. Mayfield (eds.): Intelligent hypertext: Advanced techniques for the World Wide Web. *Lecture Notes in Computer Science*, Vol. 1326, Berlin: Springer-Verlag, pp. 12-30.

Brusilovsky P & Eklund (1998) A Study of User Model Based Link Annotation in Educational Hypermedia. *Journal of Universal Computer Science*. **4**, 4. p. 429-448. Springer Science Online. http://www.iicm.edu/jucs_4_4/a_study_of_user/paper.html

Brusilovsky P & Pesin L (1994a) An intelligent learning environment for CDS/ISIS users. In: J. J. Levonen and M. T. Tukianinen (eds.) *Proceedings of The interdisciplinary workshop on complex learning in computer environments (CLCE94)*, Joensuu, Finland, May 16-19, 1994, EIC, pp.29-33. http://cs.joensuu.fi/~mtuki/www_clce.270296/Brusilov.html

Brusilovsky P & Pesin L (1994b) ISIS-Tutor: An adaptive hypertext learning environment. In: H. Ueno and V. Stefanuk (eds.) *Proceedings of JCKBSE'94, Japanese-CIS Symposium on knowledge-based software engineering*, Pereslavl-Zalesski, Russia, May 10-13, 1994, EIC, pp. 83-87.

Brusilovsky P & Pesin L (1995) Visual annotation of links in adaptive hypermedia. *CHI'95 Conference*

Companion (Denver CO, May 1995), ACM Press, 222-223. http://www.acm.org/sigchi/chi95/Electronic/documnts/shortppr/plb_bdy.htm

Brusilovsky P & Pesin L (1998) Adaptive navigation support in educational hypermedia: An evaluation of the ISIS-Tutor. *Journal of Computing and Information Technology.*

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

Brusilovsky P, Schwarz E & Weber G (1996) A tool for developing adaptive electronic textbooks on WWW, in *Proceedings of WebNet'96* (San Francisco CA, October 1996), AACE, 64-69. http://www.contrib.andrew.cmu.edu/~plb/WebNet96.html

Calvi L & De Bra P (1997) Using dynamic hypertext to create multi-purpose textbooks. In: T. Müldner and T. C. Reeves (eds.) *Proceedings of ED-MEDIA/ED-TELECOM'97 - World Conference on Educational Multimedia/Hypermedia and World Conference on Educational Telecommunications*, Calgary, Canada, June 14-19, 1997, AACE, pp. 130-135.

Clibbon K (1995) Conceptually adapted hypertext for learning. In I.Katz, R. Mack, & L. Marks (Eds.), *Proceedings of CHI'95* (pp.224-225). Denver: ACM. http://www.acm.org/sigchi/chi95/Electronic/documnts/shortppr/kc_bdy.htm

da Silva D, Van Durm R, Duval E & Olivie H (1998 submitted) Adaptive Navigation Facilities in Educational Hypermedia. *Paper to be presented at WebNet98.*

De Bra P (1996) Teaching Hypertext and Hypermedia through the Web. *Proceedings of WebNet'96, World Conference of the Web Society.* San Francisco, CA, October 15-19, 1996.

de La Passardiere B & Dufresne A (1992) Adaptive navigational tools for educational hypermedia. In I. Tomek (Ed.), *ICCAL'92, 4-thInternational Conference on Computers and Learning* (pp. 555-567). Berlin: Springer-Verlag. http://mistral.ere.umontreal.ca/~dufresne/Publications/ical91.htm

Eklund J, Brusilovsky P & Schwarz E (1988, in press) A study of adaptive link annotation in educational hypermedia. *Paper to be presented at EDMEDIA'98* Freiburg, Germany. June 1998.

Eklund J & Sawers J (1996) Customising Web-based course delivery in WEST® with navigation support. *Proceedings of WebNet'96, World Conference of the Web Society. San Francisco,* CA, October 15-19, 1996, pp. 534-535.

Gonschorek M & Herzog C (1995) Using hypertext for an adaptive help system in an intelligent tutoring system. In J. Greer (Eds.), *Proceedings of AI-ED'95, 7th World Conference on Artificial Intelligence in Education* (pp. 274-281). Washington, DC: AACE.

Hohl H, Böcker H, Gunzenhäuser R (1996) Hypadapter: An adaptive hypertext system for exploratory learning and programming. *User Modeling and user adapted Interaction* **6**, 2-3, 131-156.

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

Kushniruk A & Wang H (1994) A hypermedia-based educational system with knowledge-based guidance. In T. Ottoman & I. Tomek (Eds.), *Proceedings of ED-MEDIA'94 - World conference on educational multimedia and hypermedia* (pp. 335-340). Vancouver, Canada: AACE.

Micarelli A & Sciarrone F (1996) A case-based toolbox for guided hypermedia navigation. *Proceedings of Fifth International Conference on User Modelling, UM-96.* Kailua-Kona, Hawaii: User Modelling, Inc. (pp. 129-136)

Pérez T, Gutiérrez J & Lopistéguy P (1995) An adaptive hypermedia system. *Proceedings of AI-ED'95, 7th World Conference on Artificial Intelligence in Education.* Edited by J. Greer. Washington, DC, 16-19 August 1995, pp. 351-358.

Schöch V, Specht M & Weber G (1998, in press) ADI - an evaluation of a tutorial agent. *Proceedings of ED-MEDIA/ED-TELECOM'98 - World Conference on Educational Multimedia/Hypermedia and World Conference on Educational Telecommunications*, Freiburg, Germany, June, 1998, AACE.

Schwarz E, Brusilovsky P & Weber G (1996) World-wide intelligent textbooks. In: *Proceedings of ED-TELECOM'96 - World Conference on Educational Telecommunications*, Boston, MA, June 17-22, 1996, AACE, pp. 302-307, http://www.contrib.andrew.cmu.edu/~plb/ED-MEDIA-96.html.

Specht M, Weber G, Heitmeyer S & Schöch V (1997) AST: Adaptive WWW-Courseware for Statistics. In: P. Brusilovsky, J. Fink and J. Kay (eds.) *Proceedings of Workshop "Adaptive Systems and User Modeling on the World Wide Web" at 6th International Conference on User Modeling*, UM97, Chia Laguna, Sardinia, Italy, June 2, 1997, Carnegie Mellon Online, pp. 91-95, http://www.contrib.andrew.cmu.edu/~plb/UM97_workshop/Specht.html.

Weber G & Specht M (1997) User modeling and adaptive navigation support in WWW-based tutoring systems. In: A. Jameson, C. Paris and C. Tasso (eds.) *User Modeling.* (Proceedings of 6th International Conference on User Modeling, Chia Laguna, Sardinia, Italy, June 2-5, 1997) Wien: Springer-Verlag, pp. 289-300.

Weber G, Brusilovsky P & Specht M (in press) ELM-ART: An adaptive educational system on the WWW. In *User Modeling and User-Adapted Interaction* .

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

Zeiliger R, Reggers T & Peeters R (1996) Concept-map based navigation in educational hypermedia : a case study. *Proceedings of ED-MEDIA'96 - World conference on educational multimedia and hypermedia.* Boston, MA, June 17-22, 1996.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Adaptive Communities and Web Places

Paul P. Maglio and Rob Barrett
IBM Almaden Research Center
650 Harry Rd, NWED-B2
San Jose, CA 95120
tel: 408-927-2857
fax: 408-927-2857
email: {pmaglio,barrett}@almaden.ibm.com

Abstract: We describe an approach to personalizing World-Wide Web navigation that combines ideas from social filtering and social navigation. Specifically, we propose a method for (a) automatically finding the boundaries of user communities by clustering users around *web places*, which we define as locations in web space that are bound together by common usage patterns; and then (b) adapting what web users see to the places community members congregate. We describe our implementation of web places based on the Web Browsing Intermediaries (WBI) framework (Barrett, Maglio, & Kellem, 1997; Barrett & Maglio, 1998), and suggest future work to help judge its utility.

Keywords: World-Wide Web, personalization, user modeling, social navigation, social filtering.

Introduction

The holy grail for designers of information delivery systems is to find a mechanism for automatically determining user interests, thus enabling adaptive information delivery. For the World-Wide Web (WWW), this means finding ways to suggest web pages that are interesting and relevant to web users. One promising approach is *social filtering*, in which individual user interests are determined indirectly by clustering users into groups with common interests (Goldberg, Nichols, Oki, & Terry, 1992; Maes, 1994; Resnick & Varian, 1997). The key problem in social filtering systems lies in grouping users. Some systems rely on explicit voting or selection of information items or categories to derive the user profiles that are then clustered (e.g., Konstan, Miller, Maltz, Herlocker, Gordon, & Riedl, 1997; Shardanand & Maes, 1995). Other systems are more passive, relying on ordinary user actions or existing information to derive profiles used to cluster users into groups (e.g., Rucker & Polanco, 1997). Thus, social filtering relies on a *community* to determine what is appropriate, but members of the community generally do not interact directly in forming judgements.

By contrast, *social navigation* refers to cases in which individual users explicitly make recommendations when interacting with others (Dieberger, 1997; Gruen & Moody, 1998; Maltz & Ehrlich, 1995). For social navigation systems, designers do not build in mechanisms for automating information delivery; rather, members of a user community interact directly to deliver information to one another. The main design problem is to provide appropriate affordances to promote information sharing.

In this paper, we suggest an approach to adaptive information delivery that combines ideas from social filtering and social navigation. From social filtering, we take the idea of automating information recommendations using communities bound by common interest. From social navigation, we take the idea that communities naturally grow by interacting on a topic of common interest. Specifically, we propose a method for finding the boundaries of user communities by clustering users around *web places*. We define a web place as a location in web space that is bound together by usage patterns common to a collection of users. Given that users congregate around web places, we describe a method for adapting web pages to the interests of community members.

In what follows, we describe our approach in some detail. We first discuss the idea of web place, and then present our architecture and implementation. In the final section, we suggest follow on work to test the utility of this approach.

Web Places

Although people naturally think of the WWW as a kind of physical space in which they move to obtain information (Maglio & Matlock, 1998), they do not seem to think of the web as a kind *place* where they might meet and interact with others. Of course, the reason is because users on the web generally interact only with information. So what makes a place? In the context of human-computer interfaces, Erickson (1993) defines place as space plus meaning, and Harrison and Dourish (1996) define it "as space which is invested with an understanding of behavioral appropriateness, cultural expectations, and so forth" (p. 69). Thus, a sense of place derives from a shared understanding about a space, and interpersonal interactions are critical in creating shared understanding.

In the simplest case, a single web page might serve as a web place where groups of individuals gather to discuss matters related to the page's topic. For instance, the home page of a particular organization, such as IBM research, could be the focal point for discussions about that organization, such as technology for copper chip fabrication. Or a web page advertising prices on a certin model of computer would be a great place to find people knowledgeable about where to get the best deals on that kind of computer. The key point is that a place provides a context for human interaction. But unlike a coffeehouse, where people gather because of its proximity to home or the taste of the coffee it brews, the information-rich environment of the web naturally provides contexts bound by interest or information.

How to Create Web Places

The WebPath browser (Gruen & Moody, 1998) goes some way toward helping to establish larger places on the web where people can interact. Users start by following a predefined path, such as a path for the morning's news or for learning about snowboarding. Once on the path, the browser application enables users to see where others are browsing, to chat with others while browsing, and to follow another user in real-time. The key idea here is that a community of users gathers around a predefined path, establishing conventions and understandings of the information on the path or related to the path by directly interacting with each other.

Chalmers, Rodden and Brodbeck (1998) describe a method for automatically calculating places on the web where people actually congregate. More precisely, they describe a system that uses web interaction histories of groups of users to determine whether a particular user is travelling a well-worn path. Using this information, their system can suggest further nodes to follow given the previously worn trails as a guide. In this case, communities of users develop naturally around meaningful clumps of information, which are then used to guide information exploration.

Our idea of a web place incorporates both deliberate user-user interaction as well as automatic clustering of users by information usage patterns. In addition, we believe that the most effective way to give users a sense of place is *through the web page itself* (see Barrett, Maglio, & Kellem, 1997, for a discussion of our general design guidelines). Rather than adding features to the browser or using a separate application window, our system actively adapts the form and content of web pages to the communities that view them. In particular, we modify web pages on the fly to enable interaction with other users, and we provide new web functions (i.e., special URLs) for keeping tabs on and for communicating with other users as well.

Web Places System

Our Web Places system transforms web pages into web places by annotating web pages with information about some of the other users who are currently or who have recently followed the same trail. One annotation adds to the bottom of the page links to these other users, along with comments they have made about the page. The links to these other users provide the means for finding out more about them, such as where they have been, as well as for chatting with them or for following them. In addition, users themselves can annotate pages, leaving notes or comments that others can see or even add to.

Another way that Web Places annotates pages is by providing hints on which links to follow. Our approach to calculating these suggestions is similar to that of Chalmers, Rodden and Brodbeck (1998), that is, in terms of the worn trails. We are experimenting with several methods for displaying these: (a) suggested links can be listed at the top of the page, providing explicit suggestions; and (b) the links embedded in the text of the page can be effectively annotated with small images to subtly direct users' attention to some links over others (Campbell & Maglio, 1998).

As an alternative to relying on automatic adaptive communities, Web Places also allows users to define their own communities. For example, if a user meets someone in an adaptive community, they can be added to one another's list of *friends*. Such groups of user-defined friends form a communities that are not confined to a particular Web Place. Whenever someone is using the web, their friends can contact them, find them in whatever Web Place they happen to be exploring, or chat with them. These user-definable communities are similar to environments such as ICQ.

In addition to annotating pages, Web Places provides web pages that allow users to survey the communities of users actively browsing the web. Whereas Gruen and Moody's (1998) WebPath browser bases its notion of community on the set of users following a predetermined path, our system can coordinate several communities whose boundaries are determined on the fly. One benefit of the WebPath approach is that there is a reasonable expectation that some set of users would be following a predetermined path, and so a community would likely exist. For Web Places, communities are determined by the paths users actually travel, and so there is no guarantee others will follow similar trails. We think the key to making Web Places work lies in constraining the larger set of users connected by the system to some natural community, such as researchers at IBM or members of a university computer science department. If the larger community is too broad, then there will be insufficient commonality from which to derive communities. We return to this issue in the final section.

In the next section, we describe the architectural framework behind our Web Places implementation.

Intermediaries Architecture

Intermediaries are computational elements that lie along the path of web transactions (Barrett & Maglio, 1998). They may operate on web data as the request is sent from the browser, passes through firewalls and proxies, is satisfied by the generation of a document, and as the document is returned to the browser. Intermediaries have access to web data at all these points, and are able to observe, respond to requests, and modify both the request and the resulting documents. Our intermediary architecture, Web Browser Intermediaries (WBI), is a programmable proxy server that was designed for easy development and deployment of intermediary applications.

In WBI, intermediary applications are constructed from five basic building blocks: request editors, generators, document editors, monitors, and autonomous functions. We refer to these collectively as MEGs, for Monitor/Editor/Generator. Monitors observe transactions without affecting them. Editors modify outgoing requests or incoming documents. Generators produce documents in response to requests. Autonomous functions run independently of any transaction and perform background tasks. WBI dynamically constructs a data path through the various MEGs for each transaction. To configure the route for a particular request, WBI has a rule associated with each MEG that specifies a boolean condition indicating whether the MEG should be involved in a transaction. An application is usually composed of a number of MEGs that operate in concert to produce a new function.

The first application we constructed with WBI was *personal history*, which uses a monitor that records the sequence of pages visited by each user along with the text of each page (see Barrett, Maglio, & Kellem, 1997). The user can access his or her own personal history through generators that search through the stored text or that display paths taken previously. In addition, the markup of individual pages can be changed to reflect patterns of repeated use, for instance, adding shortcut links to pages that are visited repeatedly within some radius of the current page. We also experimented with a WBI application to cluster a user's history of web usage around key nodes found on the way to some goal (Maglio & Barrett, 1997). In all cases, we were concerned with supporting the web navigation of individual users by providing access to the user's own web history in expected and natural ways. The key point is that the way the web looks to an individual user can be adapted based on the user's history of interactions with the web.

How to Implement Web Places

Our implementation of Web Places takes these kinds of adaptations one step further by using the web histories of a community to change the way the web looks to individuals. We used WBI to implement both a client proxy that runs on an individual user's machine as well as a server that runs on another machine to which all users have access (see Figure 1). On each web transaction, the WBI client uses an editor to modify the returned document to include link annotations and links to people on the same trail. To do this, the WBI client communicates with the WBI server to find out whether there is a community of users around the current page. The server keeps track of all users trails and performs a simple subtrail match with some maximum length (such as five nodes) to determine whether there are common subtrails. If there are, the set of users on the common trail is reported back to client, along with information about which links were most frequently followed. The WBI client uses this information to annotate the current page. Chat is done through a Java applet that the client also embeds in the page. The applet contacts the WBI server through the client to pass text from user to user. In addition to web page annotation, the WBI client also relies on generators to create web pages in response to requests for state information. Again, the client contacts the server to obtain the web locations of other users.

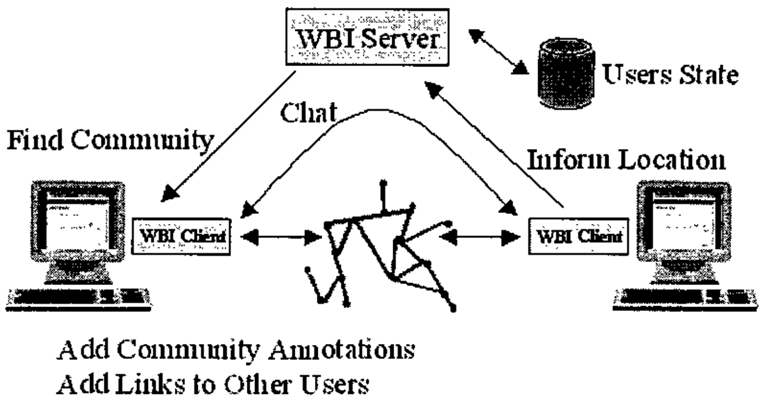


Figure 1: Web Places Architecture.

The Future of Adaptive Communities

Our work with adaptive communities has just begun. The Web Places implementation provides a framework for experimenting with (a) algorithms for forming communities, (b) methods for adapting hypertext documents to individual community members, and (c) schemes for setting up populations from which communities can be formed. We discuss each of these in turn.

As mentioned, our first algorithm for partitioning users into communities is based on finding users with overlapping subtrails. Various parameters can be set on this algorithm, including the percentage of trail matched, the number of other users who must match, how heavily to weigh current browsing habits relative to previous habits, and so on. In addition, other information might be taken into account when forming communities. For instance, the sort of information contained on a user's hard disk, such as email or papers (Nardi, Miller, & Wright, 1998), or the sorts of activities a user routinely performs on his or her own computer, such as applications used and their frequency (Underwood, Maglio, & Barrett, 1998), can be combined with web usage to create a more elaborate and more detailed user model. In any event, as the number of users increases and the length of user trails increase, tradeoffs of computational efficiency will undoubtedly become another issue to investigate.

Following the general design principle of personalizing the web by tailoring it to individual users (rather than by adding additional windows or functions), we have described several simple methods for annotating hypertext to create web places, including adding links to track users on the same trail, and annotating links that previous users have followed. Another method for adapting what users see is to change the connectivity of the web (Brusilovsky, 1996). In this case, for example, we might remove links that no one has followed previously. In general, we prefer the subtler method of annotating links to this kind of heavy-handed approach. Nevertheless, in certain circumstances, such as instruction or for children, adapting the link structure might make sense. In any event, a wide range of possible methods for adapting form, content, and connectivity based on changing communities have yet to be investigated.

Finally, the most interesting questions from a social perspective revolve around whether adaptive communities can in fact take root. What does it take to have a community that people actually want to invest in? Can a meaningful chat result from faceless users gathered around a web page? Or must people have a prior history of involvement? Can computers help form interpersonal groups and web-pals that are persistent, lasting longer than the time it takes to click through a single page? To be sure, chat is extremely popular, and persistent interpersonal communication among faceless individuals occurs all the time on the Internet (Turkle, 1995). However, it is not clear that the same will hold true for adaptive communities. As mentioned, we think that making adaptive communities work on a large scale will require constraining the population that use the same Web Places server to one with a broad common interest, such members of the same university department. It is possible, of course, that no such constraint is necessary. For now, this remains an open question.

References

Barrett, R., Maglio, P. P., & Kellem, D. C. (1997). How to personalize the web. *Proceedings of Human Factors in Computing Systems, CHI '97*. New York: ACM Press.

Barrett, R. & Maglio, P. P. (1998). Intermediaries: New places for producing and manipulating web content. *Seventh International World Wide Web Conference*, Brisbane, Australia.

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction, 6*, 87-128.

Campbell, C. S. & Maglio, P. P. (1998). How to facilitate information navigation without disrupting perception. Manuscript submitted for publication. IBM Almaden Research Center.

Chalmers, M. Rodden, K. & Brodbeck, D. (1998). The order of things: Activity-centered information access. *Seventh International World Wide Web Conference*, Brisbane, Australia.

Dieberger, A. (1997). Supporting social navigation on the World Wide Web. *International Journal of Human-Computer Studies, 46*, 805-825.

Erickson, T. (1993). From interface to interplace: The spatial environment as a medium of interaction. In A. U. Frank & I. Campari (Eds)., *Spatial information theory: A theoretical basis for GIS*, Berlin: Springer-Verlag.

Goldberg, D., Nichols, D. Oki, B. M. & Terry, D. (1992). Using collaborative filtering to weave and information tapestry. *Communications of the ACM, 35*, 61-70.

Gruen D.& Moody, P. (1998). Synchronous WebPath as a tool for leveraging expertise. Paper presented at Human Computer Interaction Consortium Conference, Snow Mountain Ranch, CO.

Harrison, S. & Dourish, P. (1996). Re-placing space: The roles of place and space in collaborative systems. *Computer Supported Cooperative Work '96*, Cambridge, MA.

ICQ - The world's largest online communication network. Available as http://www.icq.com.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Reidl, J. (1997). GroupLens: Applying collaborative filtering to usenet news. *Communications of the ACM, 37*, 77-87.

Maes, P. (1994). Agents that reduce work and information overload. *Communications of the ACM, 37*, 31-40.

Maglio, P. P. & Barrett, R. (1997). How build modeling agents to support web searchers. *Proceedings of the Sixth International Conference on User Modeling*. New York: Springer Wien.

Maglio, P. P. & Matlock T. (1998). Metaphors we surf the web by. *Workshop on Personalized and Social Navigation in Information Space*, Stockholm, Sweden.

Maltz, D. & Ehrlich, K. (1995). Pointing the way: Active collaborative filtering. *Proceedings of Human Factors in Computing Systems, CHI '95*. New York: ACM Press.

Nardi, B. A., Miller, J. R. & Wright, D. J. (1998). Collaborative, programmable intelligent agents. *Communications of the ACM, 41*, 96-104.

Resnick P. & Varian, H. R. (1997). Recommender systems. *Communications of the ACM, 40*, 56-58.

Rucker, J. & Polanco, M .J. (1997). Siteseer: Personalized navigation for the web. *Communications of the ACM, 40*, 73-75.

Shardanand, U. & Maes, P. (1995). Social information filtering: Algorithms for automating "word of mouth". *Proceedings of Human Factors in Computing Systems, CHI '95*. New York: ACM Press.

Turkle, S. (1995). *Life on the screen: Identity in the Internet*. New York: Simon and Schuster.

Underwood, G. M., Maglio, P. P. & Barrett, R. (1998). User-centered push for timely information delivery. . *Seventh International World Wide Web Conference*, Brisbane, Australia.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Content Adaptation for Audio-based Hypertexts in Physical Environments

Elena Not, Massimo Zancanaro
IRST/ITC - Cognitive and Communication Technology Division
38050 Povo - Trento -Italy
phone: +39-461-314444
{not,zancana}@irst.itc.it

Find a postscript version here (gzipped, 56k)

Abstract: The most important new issue emerging when allowing the fruition of a hypermedia repository of information while the user is moving in a physical space is the fact that information is presented in different situational contexts. Also, an additional perceptual dimension comes into play, providing stimuli, attention grasping and feedback. Emphasis should be put on *integrating* the perceptual experience with helpful information, *without competing* with the original exhibit items for visitor's attention.
In this paper we shall discuss some of the critical issues about content adaptation emerging in physical hypernavigation, presenting the approach adopted in the HyperAudio project.

Keywords: content adaptation, physical hypernavigation

Introduction: navigating a physical hyperspace

New hardware technology allows the fruition of virtual repositories of information while enjoying the physical space: for example, kiosks or portable devices may allow the access to a portion of a virtual information space relevant for the object in front of the visitor.

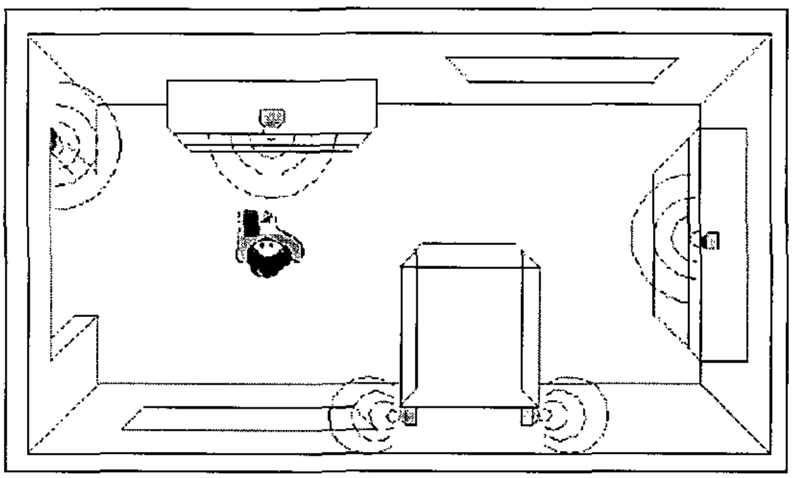


Figure 1: A visitor exploring an augmented room in a museum. (full size, 8k)

Figure 1 suggests one possible scenario of this kind, where visitors of a museum or an exhibition may enjoy a personalized tour through the interaction with the physical space augmented with an overlapped synchronized informational hyperspace (we will call this interaction context a *physical hypernavigation*). Each visitor is equipped with a palmtop computer endowed with headphones, on which an infrared receiver is mounted. Each meaningful physical location has a small (power-autonomous) infrared emitter, sending a code that uniquely identifies it.

Exploiting the infrared signals, the system is able to identify when the visitor reaches a certain physical location and can activate a relevant portion of the information repository loaded on the palmtop. Meaningful information are selected and organized to be played as audio messages or displayed on the palmtop screen. Adaptive and dynamic hypertext technology can be exploited to tailor a presentation according to the visitor interests, the actual context of the visit and so on.

We are currently investigating the features of this scenario inside the HyperAudio project [Not *et al.* 1997], a project IRST is developing in collaboration with the Civic Museum of Natural Sciences in Rovereto (Italy). The results gained inside HyperAudio will contribute to more advanced research efforts towards a richer interaction scenario, to be explored jointly with other partners inside the HIPS European project[*].

Navigating in a physical hyperspace is quite different from browsing an electronic book or a traditional (even adaptive) hypermedia running on a stationary console. The cognitive problems that may arise when a person is moving in a virtual information space (e.g., being lost in the hyperspace or the cognitive overload, [Conklin1987]) are different from those of a person moving in a real space, since an additional perceptual dimension comes into play, providing stimuli, attention grasping and feedback.

In this paper we shall discuss some of the critical issues about content adaptation emerging in physical hypernavigation. We shall particularly focus our discussion on the museum environment, which introduces additional stimulating challenges for effective content adaptivity.

Content Adaptation in a physical hypernavigation

The problem of adapting content for (cultural) information presentations in physical hypernavigation shares many features with the problem of producing adaptive and dynamic hypermedia for virtual museums (e.g. [Mellish *et al.* 1997]) or dynamic encyclopedias (e.g. [Milosavljevic *et al.* 1996]), as confirmed by many psychological studies made on museum visitors. For example: information should be stated in terms the visitor can understand (adaptation to expertise/knowledge level) and should help the visitor connect the new information to what he already knows ([Hood1993]); information should include references to visitor's interests ([Serrell1996]); content should be provided in a form that best stimulates learning on the part of the hearer ([Serrell1996]).

However, content adaptation in a physical environment poses some original problems which are related to the fact that the visitor is experiencing a ``real" situation: moving in a real environment, looking for concrete objects to observe, and receiving perceptual orientation feedback. The peculiarity of inserting a hypertextual structure onto a physical space generates new ways of navigating information:

- moving around the physical space, approaching the various cases, the visitor implicitly "clicks" on meaningful points of the hypertext (the system is able to track the visitor's position by means of sensors);
- as in solely information spaces, the visitor may explore the sub-network of the hypertext nodes related to the physical object he is standing in front of; in addition he can proceed with the exploration within the physical dimension:
- after getting information about the object, the visitor may decide to move in a direction that was explicitly or implicitly suggested by the message (for example because a comparative description was heard that introduces a new interesting and related object). But the visitor may also decide to suddenly change the suggested tour thread. Physical hints may attract his interest more than the proposed hypertextual links: he may be distracted by interesting objects close by or he may have personal intuitions about semantic relations between objects that make him stray from the undertaken path. The system can try to cope with movements that drop out from the hypertextual structure applying techniques coming from the area of dialogue modeling: for example, tracking topic shifting or modifying its assumptions on the user's preferences.

When assembling information presentations the system should take into account the prominence of the situational context, *integrating* the perceptual experience with helpful information, *without competing* with the original exhibit items for visitor's attention:

- The system should provide information to help direct visitor's attention to and stimulate interest in the objects ([Bitgood and Patterson1993], [Boisvert and Slez1994]). This means that messages offered to the user should

---

*The HIPS consortium includes: University of Siena (coordinating partner), CB&J (France), GMD (Germany), IRST (Italy), SIETTE-Alcatel (Italy), SINTEF (Norway), University of Dublin and University of Edinburgh.

directly refer to what the user is seeing (also exploiting appropriate linguistic forms, e.g. deictic references as ``this object" or ``the object on your left") and should help the user to identify the object described (and its importance) among the others displayed.

- The information should be preferably conveyed via audio messages, allowing the user to freely concentrate on the concrete objects.
- The system should not overwhelm users with information ([Finn1985]), though providing opportunities for the interested visitor to easily find new and more detailed information on a subject ([Serrell1996]). Although this issue is relevant for presentation systems in general, it becomes crucial when the user is physically moving through museum rooms and standing in front of exhibits. In fact, physical tiredness might appreaciably affect user's attitude toward long commentaries, as well as his satisfaction and learning.
- The system should integrate information with directions that help the user orient himself in the physical space (e.g. how to reach an interesting object, a friend, the exit, ...) and decide where to go next.
- The system should adapt its behaviour according to a user model dynamically updated interpreting either the user explicit interaction (clicking on the palmtop screen) and his movements.

It may be argued that nowadays virtual reality technology allows the realization of virtual museums in which visitors navigate in a synthesized 3-D environment. Many of the issues above are relevant to this scenario too. However, the human computer interaction in augmented space is still substantially different at least for the presence of physical fatigue associated to movement.

Two dimensions of content adaptation: the user and the situation

In traditional adaptive hypermedia (either running on a stand alone computing station or accessed from a stationary console through the WWW) the most important adaptation factor considered is the user. The interaction context is quite constrained, with the user sit in front of the screen on which the system interface is displayed and with the following possible modalities to interact with the system: clicking on hyperlinks; typesetting information requests; gesturing (if suitable devices are available, e.g. a touch screen).

The most important new issue emerging when allowing the fruition of a hypermedia repository of information while the user is moving in a physical space is the fact that information is presented in different situational contexts. The main factors determining the situational context are (i) the user position and movements (whether he is in front of an object or whether he is simply walking around a room); (ii) the structure of the surrounding physical space; (iii) whether other people are examining the same item or not; (iv) whether the user came alone or not. Even though mobile computing nowadays allows to access hypermedia while the user is moving in the physical space (as in the case of a visitor walking around a museum and browsing on his palmtop computer the museum's web pages), existing systems do not offer real dynamic adaptation with respect to the situational context.

A system effectively supporting physical hypernavigation should integrate the individual, dynamic modeling of the user (his knowledge, interests, goals, integrated with abilities, attitudes and preferences) with a general model of the environment, of user's movements and user's social context.

Importance of audio modality

In a traditional hypermedia, where the user is typically expected to browse around and *read* written information, though enriched with images and sounds, the system can hardly be sure that the user has really read (let alone assimilated [Mellish and O'Donnell1998]) the message. Permanence time associated to hypertext pages is usually not significant, unless some additional feedback is exploited (e.g. mouse moves).

When the audio output modality is available, combined with a position locating system tracking the user movements in the physical space, more information can be inferred about how the user is receiving the messages. For example, if the user is standing in front of the object currently described by the system and does not explicitly stop the presentation or does not move, the system could guess a high assimilation score.

But even if audio output could be a plus, we must be careful not to waste the positive aspects of this resource. In fact, if message content is not properly adapted to the audio modality the risk occurs of introducing an additional disorientation effect or dissatisfaction in the user. This phenomenon is more evident here than in traditional hypermedia because the user can not skip uninteresting or unsuitably tailored information as easily as in a hypertext.

HyperAudio Presentation Composer

In HyperAudio, information presentations are built by the system and are provided to the visitor whenever he reaches meaningful locations or when he explicitly asks for information. Each *presentation* is a structure containing (i) a sequence of audio files which will be played through the headphones, (ii) a set of relevant concepts worth of further elaborations which will be depicted on the display as clickable buttons, and possibly (iii) images related to the object or concept described and a properly oriented map displaying the visitor's current position.

The audio presentations are built concatenating precanned audio files selected from the informational space. In HyperAudio, the informational space is designed so that its contents and its structure can be used in an adaptive way. The information unit is the *macronode* (see fig. 2). Each macronode includes a network of audio files, a list of pointers to other relevant macronodes (including the particular rhetorical relations between them), the type of text (e.g., general introduction, detailed description, ...), a pointer to the relevant semantic concept in the ontology, and possibly a link to a physical location for which the message would be pertinent.

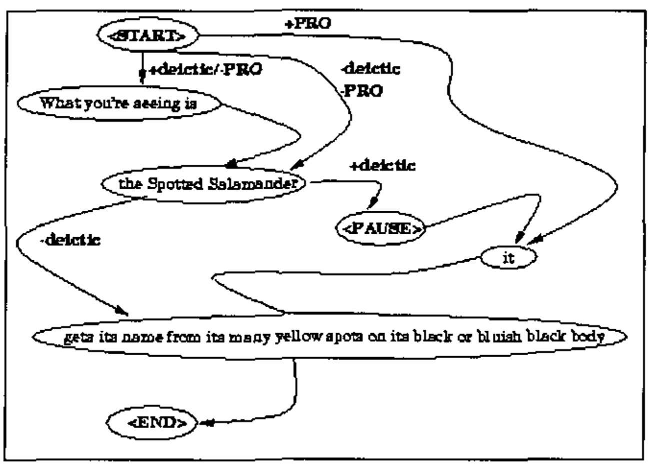


Figure 2: Sketch of macronode network (full size,5k)

More than one macronode can be selected to build the actual presentation. A macronode is atomic with respect to its content but it can have some optional parts that are selected only in some particular discourse contexts. For example, the macronode audio file network in figure 3 can be instantiated in the following messages:

- when the feature `deictic` is selected (i.e. when the visitor is in front of the object being described): *What you're seeing is the Spotted Salamander <PAUSE> it gets its name from its many yellow spots on its black or bluish black body*;
- when the feature `deictic` is deselected: *The Spotted Salamander gets its name from its many yellow spots on its black or bluish black body*;
- when the feature `PRO` is selected (i.e. when the message has to be joined to another one with the same topic): *It gets its name from its many yellow spots on its black or bluish black body.*

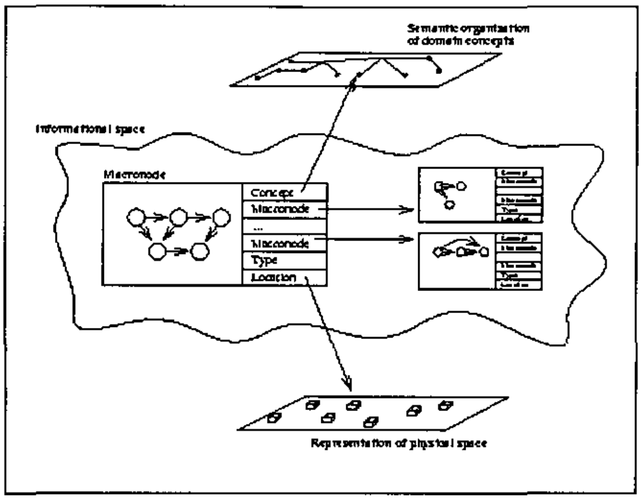


Figure 3: The audio files network for the Spotted Salamander macronode
(in the figure, each circle in the network represents an audio file or a command to the audio player). (full size, 3k)

An audio file network encodes lexical variations of the same message (that means that whatever instantiation you choose, the content of the text is the same). The task of instantiating a network performs in a simplified way some of the choices typically performed by the tactical component of an automatic natural language generator.

An important piece of information included in the macronode is the type of the message stored in the audio file network. Borrowing from [Serrell1996], we have defined the following message types occurring in the museum setting:

- **introductory labels**: descriptions of exhibition goal and extent; they have to be slow enough to help the visitor get used to audio modality and must contain invitation to look on the screen to request further information and exploit the maps;
- **section labels**: descriptions of rationale behind subgrouping of objects (including area descriptions or overviews and historic/social background); they can be played also while the visitor is moving;
- **captions**: object descriptions; they should contain visual, concrete information and make use of deictic language;
- **follow-up information to captions**: information related to objects (for example, general descriptions, anecdotes or similar); they are meant to elaborate information in caption messages;
- **way-finding and orientation messages**: directions for reaching a physical location; they should make use of orientation hints (it is important a correct assumption on user position).

The type describes the purpose of the message and it is exploited in the decision process aimed at building communicatively effective presentations (as discussed below).

Another important information contained in a macronode is the set of macronodes semantically related to the current one. For each related macronode the kind of connecting relation is also indicated. At present, we are using a limited set of relations to capture the ways in which **follow-up information to caption** add information to a **caption** message: among others we consider `elaboration-general` for general new information, `elaboration-part` for part-whole descriptions (it is useful in natural science domains), `elaboration-legend` for legends, anecdotes, and so on.

How an adapted presentation is assembled

Each time the system has to describe a concept, it collects all the macronodes which refer to that concept and exploits a set of heuristics to decide what macronodes have to be discarded, what have to be played as the current audio presentation (and in what order) and what will be realized as textual anchors on the screen. Some of the heuristics are similar to those used by other adaptive systems, exploiting the user knowledge model (for example to select comparisons to already known concepts or introducing explanations for unfamiliar terms), the user interest model (to stress user preferred topics), discourse history (both to not select macronodes already presented and to introduce references to already seen objects)[Mellish and O'Donnell1998].

Other heuristics are more specific for the task of hypernavigation in physical space. For example, adaptation to the physical situation (discussed in section 2.1) can be addressed by:

- (i) selecting the feature `deictic` whenever possible (in particular, for **captions** and **follow-up to captions** but also for **section labels** in order to enrich presentations with references to the physical environment);
- (ii) for **section label** presentation: adding **orientation messages** to highlight interesting spots nearby (exploiting the user interest model [Sarini and Strapparava1998], and a model of the physical environment to compute distances between interesting objects);

Of course, the visual part of the presentation is exploited as well: in our current implementation, **introductory labels** and **section labels** have associated a map which is displayed on the screen and is always maintained oriented consistently with the user's current orientation, and **way-finding messages** are played on demand.

In order to maximize the communicative efficacy of the presentations, we have implemented a set of heuristics that constrain the ordering and the type of messages that are concatenated in a single audio presentation. For example,

- in front of an object, only one **caption** can be selected for an audio presentation, and it can be followed by one or more **follow-up to captions** with the same associated concept (selecting the feature PRO);
- when entering a new room or exposition area, only one **section label** can be selected, and it can be followed by **captions** and **follow-up to captions** preferably with the same associated concept (selecting the feature PRO); if needed the presentation can end with a **way-finding and orientation message**;
- if **way-finding and orientation message** is the primary communicative goal of the presentation, it should not be followed by any other message.

The length of audio presentations should be carefully constrained: presentations based on **captions** should be short (relying on the visual anchors on the screen if the visitor needs more information), presentations based on **section labels** can be longer. **Way-finding and orientation messages** should be very short when associated with other messages while can be longer when they represent the primary goal of the presentation.

Conclusion

Even though we focussed our discussion on the museum setting, many of the critical issues discussed in this paper about content adaptation beyond the traditional stationary hypermedia setting do apply to any physical hypernavigation setting (e.g. being guided in tourist/cultural sites, in airports or in complex buildings).

We are currently designing a set of experiments with real users both in a laboratory setting and in the Civic Museum in Rovereto in order to test the validity of the proposed heuristics as well as the user acceptability of HyperAudio.

We wish to acknowledge the contribution of the other members of the HyperAudio team (Gregorio Convertino, Daniela Petrelli, Marcello Sarini, Oliviero Stock and Carlo Strapparava) to the discussion of the ideas presented in this paper. In particular, many thanks to Marcello Sarini and Carlo Strapparava for useful comments and suggestions on previous versions of this paper. Valuable input was also provided by the discussions with the other members of the HIPS consortium.

References

[Bitgood and Patterson1993] Stephen C. Bitgood and Donald D. Patterson. The effects of gallery changes on visitor reading and object viewing time. *Environment and Behaviour*, 25(6), November 1993. Special Issue on Environmental Design and Evaluation in Museums.

[Boisvert and Slez1994] Dorothy Lozowski Boisvert and Brenda Jochums Slez. The relationship between visitor characteristics and learning-associated behaviours in a science museum discovery space. *Science Education*, 78(2):137-148, 1994.

[Conklin1987] Jeff Conklin. *Hypertext: An Introduction and Survey.* Computer, Survey & Tutorial Series, 1987.

[Finn1985] David Finn. *How to visit a museum.* Harry N. Abrams, Inc. Publishers, 1985.

[Hood1993] Marilyn G. Hood. Comfort and caring: Two essential environment factors. *Environment and Behaviour*, 25(6), November 1993. Special Issue on Environmental Design and Evaluation in Museums.

[Mellish and O'Donnell1998] Chris Mellish and Mick O'Donnell. An architecture for opportunistic text generation. In *Proceedings of the 9th International Workshop on Natural Language Generation*, Niagara-on-the-Lake, Ontario, Canada, August 1998.

[Mellish *et al.* 1997] C. Mellish, J. Oberlander, M. O'Donnell, and A. Knott. Exploring a gallery with intelligent labels. In *Proceedings of the Fourth International Conference on Hypermedia and Interactivity in Museums (ICHIM97)*, Paris, September 1997.

[Milosavljevic *et al.* 1996] Maria Milosavljevic, Adrian Tulloch, and Robert Dale. Text generation in a dynamic hypertext environment. In *Proceedings of the 19th Australasian Computer Science Conference*, Melbourne, Australia, 1996.

[Not *et al.* 1997] Elena Not, Daniela Petrelli, Oliviero Stock, Carlo Strapparava, and Massimo Zancanaro. Person-oriented guided visits in a physical environment. In *Proceedings of the Fourth International Conference on Hypermedia and Interactivity in Museums (ICHIM97)*, Paris, September 1997.

[Sarini and Strapparava1998] Marcello Sarini and Carlo Strapparava. Preliminary notes for a user model in a museum exploration and information-providing adaptive system, 1998. IRST Internal report.

[Serrell1996] Beverly Serrell. *Exhibit Labels.* AltaMira Press, 1996.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Concepts and documents for adaptive educational hypermedia: a model and a prototype

Denise Pilar da Silva, Rafaël Van Durm, Erik Duval, Henk Olivié
Departement Computerwetenschappen
Katholieke Universiteit Leuven
Celestijnenlaan 200 A, B-3001 Leuven (Heverlee), Belgium
Tel: +32-16-32-7659
{Denise.PilarDaSilva, Rafael.VanDurm, Erik.Duval, Henk.Olivie}@cs.kuleuven.ac.be

Abstract: In this paper we describe our approach to develop adaptive hypermedia courseware and a prototype implementation. We applied the approach on a few topics of a university course on 'multimedia modeling and programming'. The fundamental structure of the domain is based on concepts, which are explained by documents. All information is stored in a database. Concepts are linked to documents and to other concepts through typed and weighted links. Each document has an associated level of difficulty. The student is guided towards appropriate documents based on information about his knowledge of each concept.
Keywords: Adaptiveness, navigation, educational hypermedia, link hiding

1 Introduction

Adaptive hypermedia (AH) systems instantiate a relatively recent area of research integrating two distinct technologies in computer assisted instruction, Intelligent Tutoring Systems and Hypermedia Systems. This is in effect a combination of two opposed approaches to computer assisted learning: the more directive tutor-centered style of traditional AI based systems and the flexible student centered browsing approach of a hypermedia system [Eklund & Zeilliger 1996].

Hypermedia users with different goals and knowledge may be interested in different pieces of information and may use different links for navigation. Irrelevant information and links overload their working memories and screen [Brusilovsky 1996]. In order to overcome this problem, it is possible to use information represented in a user model and then adapt the content and/or the links to be presented to that user. Adaptive hypermedia systems build such a model with the goal of personalizing hypermedia.

Adaptation can be done either at content level (*adaptive presentation*), or at link level (*adaptive navigation*). In this paper, we focus on adaptive navigation support. More specifically, we want to reduce the cognitive overload in order to facilitate learning.

Before analyzing our approach we will start with some background information on existing adaptive navigation techniques. In the section 3 we present our system, called AHM, which consists of three main parts: the domain model, the user model and the adaptive engine. Then, we discuss some implementation issues and the drawbacks we have encountered. Section 4 is dedicated to an informal evaluation of the system and the approach itself. Finally, in section 5, we come to our conclusions and present the main open issues in this research.

2 Background

According to [Brusilovsky 1996], there are several goals that can be achieved with adaptive navigation support techniques, though they are not clearly distinct:

- global or local guidance: *global guidance* aims at helping the user to find the shortest way to the information goal with minimal floundering, whilst the goal of *local guidance* is to help the user to make one navigation step.
- local and global orientation support: the goal of *local orientation support* is to help the user to understand what is around and what is his/her relative position in the hyperspace. On the other hand, *global orientation support* aims at helping the user to understand the structure of the overall hyperspace.

Such adaptive navigation techniques comprehend direct guidance; sorting, hiding and annotation of links; and map adaptation. Most of the existing AH systems use link hiding or link annotation in order to provide adaptive navigation support. The techniques of link hiding and annotation are proved to be efficient for learning applications [Brusilovsky 1997], they are briefly explained in the next two paragraphs.

Our main goal with adaptation is to provide both local and global orientation support. This is currently done through the use of link hiding, but it will be reinforced by including adaptive annotation and concept maps in the model.

Link hiding is currently the most frequently used technique for adaptive navigation support. The idea is to restrict the navigation space by hiding links that do not lead to "relevant" pages, i.e. not related to the user's current goal or not ready to be seen. All kinds of links can be adapted according to this scheme by real hiding or by displaying hot words as normal text. Hiding can help to support both local and global orientation. Local orientation is achieved when, by limiting the number of navigation opportunities to reduce cognitive overload, it enables the users to focus on analyzing the most relevant links. On the other hand, by hiding links the size of the visible hyperspace is reduced and thus, global orientation is simplified. The courseware on Hypermedia Structures and Systems (2L670) from the Eindhoven University of Technology is a good example of link hiding. This course is totally on-line and uses a system developed there to track student progress and based on that, generate document and link structure adapted to each particular student. Links to nodes that are no longer relevant/necessary or links to information that the student is not yet ready to access are either physically removed or displayed as normal text [Calvi & DeBra 1997].

The main difference between their approach and our own, as far as adaptation is concerned, is that our links are *typed* and *weighted* and our nodes are also typed. In other words, as nodes we have *concepts* and *documents;* and as links we have *relationships* between nodes, which have associated weight (*threshold* for concept-concept relationships and *difficulty-level* for document-concept relationships. This will be explained in section 3.

Adaptive annotation is the augmentation of links with some form of comments, which can tell the user more about the current state of the nodes behind the annotated links. The annotations can be either textual or visual and can be used with all possible forms of links, like for instance, indexes or contextual links. Link annotation can be used to support local orientation by providing additional information about the nodes available from the current node. In order to provide global orientation support, annotation of links can function as a landmark, i.e., keep the same annotation for a node when the user looks at it from different positions in the hyperspace. Both hiding and annotation can help to provide local guidance, that is, to help the user to make *one* navigation step by suggesting the most relevant links to follow from the current node. Interbook illustrates the use of link annotation [Brusilovsky 1996a]. It is a tool for authoring and delivering adaptive electronic textbooks on the WWW. For each registered user, an Interbook server maintains an individual model of the user and applies it to provide adaptive guidance, navigation support and help.

Currently, annotation is not being used in AHM, but we plan to use it to replace the removal of links to nodes considered *too easy.*

There are other techniques to provide adaptive navigation support, such as direct guidance, sorting, and map adaptation. We believe that these techniques are not the most suitable ones for educational purposes because they can lead to incorrect mental maps [Calvi 1997], and so we are not going to elaborate on them.

3 AHM Approach

In the context of Education, in contrast to hypertext design [Signore et al. 1997], one of the objectives is to ensure that the user not only has access to, but also learns a certain amount of relevant information. Certainly, learning processes can benefit from hypertext facilities and its inherent non-linear order in which content is presented.

However, if the user is left completely free to explore the hyperspace according to his/her preferences and needs, it may be more difficult for him/her to learn efficiently. Some time would surely be spent on irrelevant nodes, i.e. nodes which contain material that is unrelated or too difficult (according to his/her present learning goal). From this perspective, it is interesting to have an interface which ensures that information is seen in a relevant order and that user's level of knowledge and/or goal are taken into account by the system, in order to support navigation and promote efficient learning. [DeLaPassardiere & Dufresne 1992].

The following subsections detail the AHM system components. The architecture of the system is summarized in Figure 1. The end user accesses the courseware through the web server, which launches the adaptive engine. The adaptive engine then accesses the database containing the user and domain models, selects the appropriate information for this user according to his/her user model and returns the information to the user via web server.

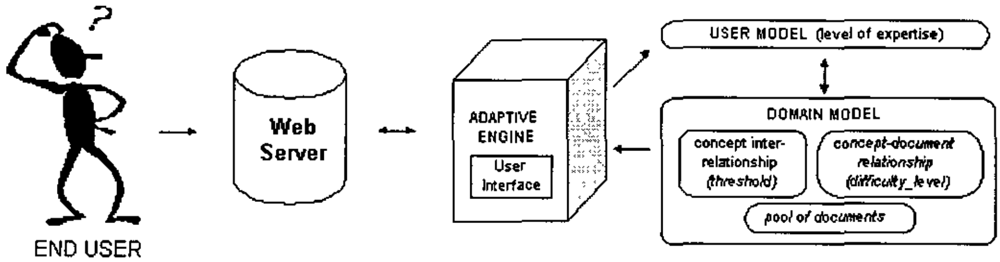


Figure 1 - Architecture of the system

3.1 Domain model

Our system provides adaptive navigation support through the use of *link hiding*. For this purpose, we rely on *typed nodes* and *typed and weighted links* to represent the structure of the domain, which is organized as a set of *concepts* explained by *documents*. The core of the approach has already been explained in [Pilar da Silva et al. 1997].

There are two possible types of nodes in our model: they represent either *concepts* or *documents*. Links are typed too and represent semantic relationships between nodes. Concepts can be related to other concepts (concept-concept relationship) or to documents (concept-document relationship). At present, we use only two link types for the concept-concept relationship: *is_prereq_of* and *is_specialized_by*, but there are other kinds of relationships which could be useful to enrich the model, such as *is_related_to* or *contrasts_with*. For document-concept links, there are also several possibilities that we have been considering, like for instance: *explains*, *illustrates*, *tests*, etc. However, for simplicity, only the type *explains* is being used currently.

The links in the model are also weighted. Associated to the *is_prereq_of* relationship between two concepts, there is a *threshold*, that represents the minimal expertise a student must attain on the prerequisite concept in order to access the more advanced concept. Each document has an associated *level of difficulty* with respect to the concept it explains, which varies from *0* to *99*, and where a higher value means '*more difficult*'.

The *documents* are multimedia objects, such as text segments, figures or interactive demonstrations whose URL is stored in a database. A document may be related to more than one concept, since relationships are stored separately.

We explain our approach to adaptive hypermedia courseware by means of an example courseware on 'multimedia modeling and programming', which in fact contains only some topics from the actual course, hereafter called *the course*.

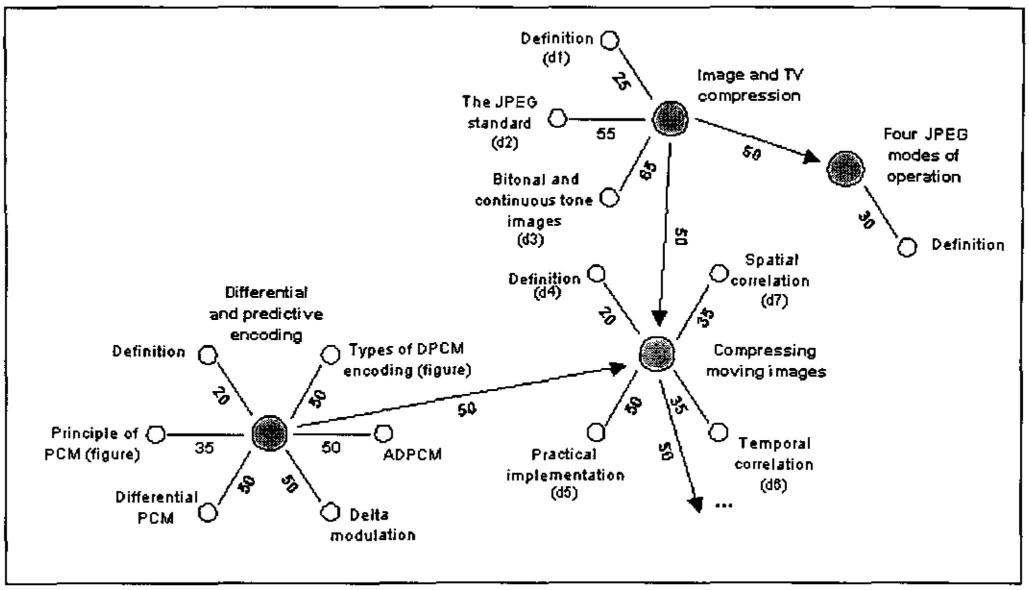


Figure 2 - example course structure

Figure 2 illustrates the course structure. *Concepts* are represented by large circles, and *documents* by small ones. A document explains the concept(s) it is linked to. The number associated to the link between a document and a concept indicates the *level of difficulty* of this document with respect to this concept. Edges linking two concepts represent the relationship *is_prerequisite_of* and the labels correspond to the *level_of_expertise* a student must have for crossing the link between two concepts. For example, in Figure 2, the concept *Differential and Predictive Encoding* (c1) is explained by 6 documents with levels of difficulty ranging from 20 to 50. This concept is a prerequisite for concept *Compressing Moving Images* (c2) with a threshold of 50, which means that, in order to access c2, the user must have accessed at least one of the documents with difficulty level 50, be it *Differential PCM, Types of DPCM encoding* or *ADPCM.*

It is also important to say that the adaptiveness in our approach affects only the so-called *non-contextual* links. Links embedded in text content are not reflected in the user level of expertise about the current concept.

3.2 User model and adaptive navigation

A primary concern for educational hypertext is the definition of an appropriate structure, so that a student can easily and naturally find the most relevant information depending on his/her needs. We believe that the techniques of link hiding and annotation are well suited to achieve our goal, which is to reach a compromise between stability of the environment and adaptiveness.

The user model currently deals only with knowledge about each concept, which can vary from *0 to 99*. We initialize student knowledge (or *level of expertise*) for a particular user as *0* for every concept [see also Calvi 1997] (see example in Table 2 (c) at instant *t0*) and update this value after the student has visited a document related to the concept. When a student visits a document, his/her level of expertise is updated in the following way: when a document is visited, the difficulty level of the document updates the user expertise about the current concept if the former is superior to the latter. Tests are not being used yet, but they can be easily accommodated in the model as special documents, making it possible to evaluate previous or acquired knowledge.

The user model determines which documents are available to a student. Basic concepts, that have no prerequisites, can be accessed by a new student. Acquiring *basic concepts* enables the student to consult documents related to more advanced concepts. Associated to the *is_prereq_of* relationship between two concepts, there is a *threshold,* that represents the minimal level of expertise a student must have attained on the prerequisite concept in order to access the more advanced concept. By acquiring a concept we mean to get a level of expertise for it that is equal or superior to the associated threshold value.

The set of relevant concepts for a student at a particular moment includes basic concepts, as well as concepts whose prerequisites he/she masters sufficiently well, i.e. concepts whose prerequisites he/she has already acquired. The documents accessible to a particular student are called *relevant documents.* Relevant documents are those that explain a relevant concept. Additionally, among all documents referring to relevant concepts, the system chooses those having a difficulty level considered to be appropriate for that user at that particular instant. By appropriate we mean documents comprised in a calculated interval that excludes documents considered being *too basic* or *too advanced*. In practice, when appropriate documents are recalculated, the result is shown in the user interface. Links to '*too easy*' documents are removed from the screen while links to more difficult documents - which have just become 'appropriate' - are added to it.

The user model is illustrated in Table 1. Each user is identified by his/her *user_id*. Then each cell in the column associated to one user represents a different concept already visited by this user and contains the *level of expertise*, which represents the user's knowledge for that concept.

user-ids **concepts**	u1 - Denise	u2 - Erik	u3 - Rafael
Differential and Predictive Encoding	50	50	35
Image and TV Compression	60	--	65
Compressing Moving Images	50	50	35

Table 1 - Snapshot of the user model

As an example of a basic concept, see concept *Image and TV Compression* in Table 2 (a). To see more clearly how a concept is acquired, let's look at Figure 2, the concept *Image and TV Compression* is a prerequisite for concept *Compressing Moving Images* with a threshold of 50 (directed arrow). Thus, in order to be able to access the latter, the user must first access, at least, the document *The JPEG standard.*

Things become more clear through the scenario shown in Table 2.

concepts	*Image and TV Compression*			*Compressing Moving Images*			
time	**d1** (25)	**d2** (65)	**d3** (55)	**d4** (20)	**d5** (35)	**d6** (50)	**d7** (35)
t0	x	-	-	-	-	-	-
t1	x	-	-				
t2	x	-	x				
t3	-	x	x	x	-	-	-
t4	-	x	x	x	x	-	-
t5	-	x	x	-	x	x	x
t6	-	x	x	-	x	x	x

(a) Accessible documents
(' x ' = accessible, ' - ' = not accessible)

concepts	*Image and TV Compression*			*Compressing Moving Images*			
time	**d1** (25)	**d2** (65)	**d3** (55)	**d4** (20)	**d5** (35)	**d6** (50)	**d7** (35)
t0	-	-	-	-	-	-	-
t1	x	-	-	-	-	-	-
t2	x	-	x	-	-	-	-
t3	x	-	x	x	-	-	-
t4	x	-	x	x	x	-	-
t5	x	-	x	x	x	-	x
t6	x	-	x	x	x	x	x

(b) Visited documents
(' x ' = "current node", ' x ' = visited, ' - ' = not-visited yet)

concepts time	*Image and TV Compression*			*Compressing Moving Images*			
	d1 (25)	*d2* (65)	d3 (55)	d4 (20)	d5 (35)	d6 (50)	d7 (35)
t0	0			0			
t1	25						
t2	55						
t3				20			
t4				35			
t5				50			
t6				35			

(c) User model updates
Table 2 - Scenario

Let's consider user *u1* from Table 1, as well as concept *Image and TV Compression* with its documents *Definition, The JPEG Standard,* and *Bitonal and continuous tone images* (here respectively called d1, d2 and d3) and concept *Compressing Moving Images* with its documents *Definition, Spatial correlation, Practical implementation* and *Temporal correlation* (here respectively called d4, d5, d6 and d7) from Figure 2 and for the sake of space we use only document identifiers.

When the user *u1* starts using the system (instant *t0*), her knowledge for all available concepts is 0 (Table 2 (c)). At instant *t1*, *u1* chooses concept *Image and TV Compression* and visits the document *d1* (Table 2 (b)), which is the only visible document for this concept at this moment (Table 2 (a)). *d1* has a difficulty level of 30, thus, *u1*'s level of expertise for concept *Image and TV Compression* becomes 30 (Table 2 (c)). *u1*'s knowledge for other concepts remains unaltered. At *t2*, both documents *d2* and *d3* become accessible (Table 2 (a)), but *u1* chooses to visit *d3* first (Table 2 (b)). Her knowledge for *Image and TV Compression* is then updated to 60, which corresponds to *d3*'s difficulty level (Table 2 (c)). We have shown here how the level of expertise of user *u1* for concept *Image and TV Compression* in Table 1 was achieved. The procedure also holds for the other concepts.

Looking at Table 2 (a) at instant *t5*, we notice that document *d4* which had been accessible till the instant before, has just become inaccessible. The reason is that the system now considers it *too basic* for this user. We may also notice that one or more documents became accessible at instants *t4* and *t5*. A formal definition of the rules to generate both sets of relevant concepts and available documents as well as a rule to define the appropriate documents among the available ones can be found in [Pilar da Silva et al. 1997].

In Table 2 (c), the number shown below documents corresponds to their level of difficulty. It is clear then that user level of expertise for the concept *Image and TV Compression* at instant *t2* reflects the difficulty level of the document she has just visited (see Table 2 (b) instant *t2*).

3.3 Implementation issues

Currently, all information is stored in an Access database and dynamically translated into HTML. There are several approaches to do this, varying from the more traditional CGI approach to server-side includes and newer technologies like *servlets,* which we favor. At the moment, however, we rely on the Active Server Page (ASP) technology developed by Microsoft. The application has been informally tested recently with students at K.U.Leuven.

3.3.1 Technical Grounds

ASP files are server-side scripts which consist of simple HTML with server side statements in VisualBasic syntax. Cookies are used to store a session number on the client, which is then passed to the server each time the browser connects.

ASP uses ActiveX Data Objects (ADO) as sources of data to retrieve information from a database. Rather than retrieving a large volume of data from a database, ADO first makes a connection to the database through a Connection Interface. The Connection Interface establishes the data source and its capabilities. This is done in a script by creating a connection object and setting that object to a particular data source. After the connection is established, a recordset object is created. A recordset object is a set of records (from a specified connection to a database) which is used to perform operations on only one record at a time. Finally, the recordset is based on a connection, and specify which records the recordset should contain. Once the information is retrieved, the script contained in an ASP page uses the ADO recordset to view, add, edit or delete data contained in the recordset's fields. Since connections and recordsets are separate objects, we can have multiple recordsets for each connection. The recordsets can be manipulated to be displayed as HTML or serve as parameters for a JAVA applet, for instance - as we do in our concept maps, which will be explained later in this section.

3.3.2 User Interface

The user interface of the current implementation makes use of HTML frames and Javascript. In Figure 3, we see what the user interface looks like. The frame on the upper-left contains the list of concepts that are available for this student at this particular moment, that is to say, the calculated *set of relevant concepts*. At present, concepts that are not accessible yet are also shown, but as normal text.

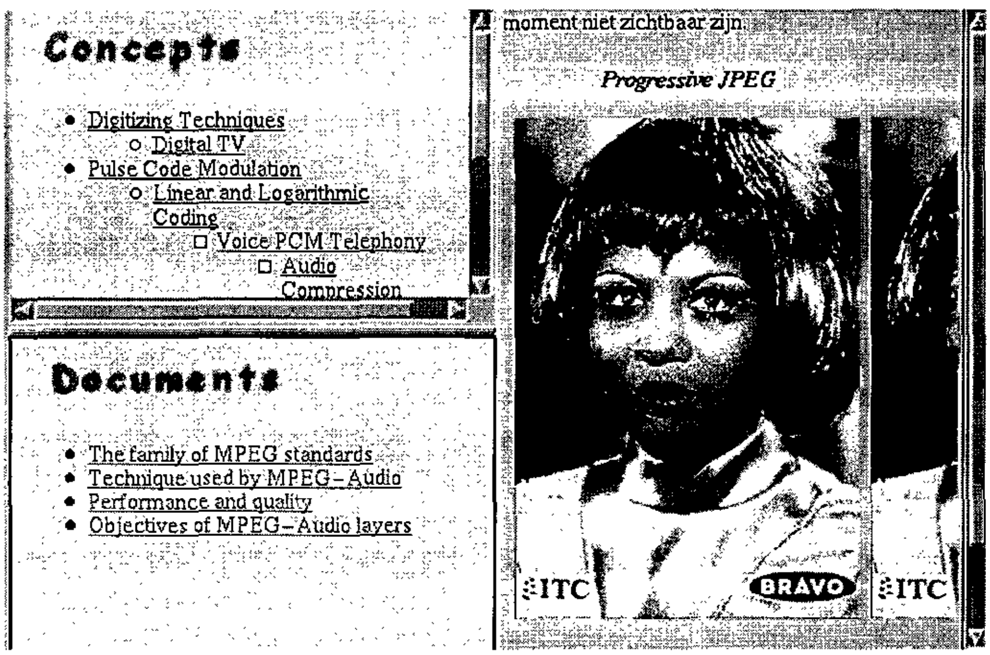


Figure 3 - Current user interface

In the lower-left frame we display a list of accessible documents (the calculated *set of relevant documents*) for the current concept and the current user. The concept name has been recently added to the document frame. Clicking on a document on this list brings its content to the frame on the right. When a student accesses a document, his/her user model updated, and a series of consequences are triggered and reflected in the user interface, that is, new sets of relevant concepts and accessible documents are calculated and links to them are added to (or removed from) the screen.

Currently, in our model we have three concept maps. Two of the maps are static: the first one contains the basic concepts of the course with their documents. The second map contains all concepts of the course but no documents. The third map, however, is dynamically created while the user is using the course. It is called *'Where am I?'* map and represents the current concept and its immediate neighborhood. This map contains the current concept with its documents, as well as the concepts which lead to current concept and the concepts to where the user can go from the current concept. At present, the maps are not 'clickable', i.e. they can't be used as a navigation tool yet.

4 Some experience...

According to [Brusilovsky 1997] lack of experimental investigation is the weak point of adaptive hypermedia. Very few reported AH techniques have been validated by a special study. Even though, the significant results reported by these studies are quite promising. It was shown that adaptive presentation can reduce the time for learning by improving user comprehension of the hypertext material. On the other hand, adaptive navigation support helps to decrease user search and navigation efforts, by reducing the number of navigation steps (or visited nodes) the user makes in order to find the desired or necessary information. Adaptive presentation and navigational support seem to be complementary and if used together may additionally improve the effectiveness of learning with hypermedia.

Though our system is still a prototype, we decided to apply a preliminary experiment with the main goal of evaluating the system in general and re-direct the future of the work based on feelings of real users. Since the experiment and specially the content were not optimally prepared, we are not considering it as evaluation of the system and we cannot draw any conclusions from it. However, we have learned from the experience and based on that we have decided to incorporate some changes to the model. The lessons learned are listed as follows:

- students seem to like the layout and user interface of the system, especially the fact of having three frames (concepts, documents and content) in the same window;
- hiding links to already seen information confuses the user, and visual annotation will be preferred from now on;
- a general overview to provide the users with information about the "size" of the course will be an asset.

5 Conclusions

A model and a prototype system for adaptive courseware navigation has been presented. The underlying domain structure allows us to offer different navigational possibilities to different users based upon a user model. The user interface has been improved, but we still plan some enhancements to it. For instance, the dynamic drawing of local overview diagrams or concept maps that show the immediate neighborhood in order to help minimize cognitive overhead, is now being developed. These concept maps may as well be an alternative as "general overview", though they are not "clickable" yet. The results of the preliminary experience with students, though not very conclusive, were very useful in defining new directions for our work.

At present, the definition of concept structure, concept interrelationships and associated thresholds, document-concept relationship and associated difficulty levels are all defined manually.

As we were preparing our simple test of the system we realized that *difficulty level* of documents is not sufficient to measure user expertise about a concept and we shall include an additional attribute to the relationship document-concept, called *coverage* for the time being, that will state how important the document is for this concept or rather, "how much" of the concept can be considered known if the document is known. Both *difficulty level* and *coverage* will then be taken into account together in order to define user expertise. Furthermore we think that in order for the adaptiveness to be efficient, the user model needs to be elaborated and other individual differences must also be considered together with knowledge. Another important point in educational hypermedia is that the content should be prepared very carefully, for a badly prepared content can have a very negative impact on users' feelings towards the system.

The main problem we have encountered so far is how to define relationships and deriving difficulty ratings and thresholds, since both tasks require very good knowledge of the content.

6 Future Work

First of all, we plan to add some enhancements to the model, which we summarize as follows:

- include visual annotation to:
  - replace the removal of links to nodes considered *too easy*;
  - enrich both concept and document lists by including a kind of sneak-preview (through *rollovers*);
- include more different link types, both for concept-concept and document-concept relationships;
- include an additional measure to document-concept relationships which will be used to compute user

knowledge, together with *difficulty level;*
- find a new criterion for measuring acquired knowledge, as a replacement for only 'visiting' a node.

Furthermore, we intend to investigate the possibilities and the relevance of elaborating the user model by including some personality traits that are relevant for learning processes. We are also studying the possibility of integrating our approach to the system used in the Museum of Science and Technology of the Pontifical Catholic University (PUC/RS) in Porto Alegre, Brazil; which concerns adaptive presentation.

Acknowledgements

First author is supported by Brazilian scholarship (CNPq). The Socrates program of the European Commission provided financial support for the elaboration of the courseware, in the framework of the EONT project.
Third author is a Postdoctoral fellow of the Fund for Scientific Research - Flanders (Belgium) (F. W. O.)

References

[Brusilovsky 1997] Brusilovsky, P., Efficient Techniques for Adaptive Hypermedia. *Intelligent hypertext: Advanced techniques for the World Wide Web.* Lecture Notes in Computer Science. Vol. 1326, pp. 12-30, 1997.

[Brusilovsky 1996] Brusilovsky, P., Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction Journal* Vol. 6, pp. 87-129, 1996.

[Brusilovsky 1996a] Brusilovsky, P., Schwarz, E. and Weber, G. *A Tool for Developing Adaptive Electronic Textbooks on WWW* . Proceedings of WebNet'96 - World Conference of the Web Society, October 16-19, 1996. San Francisco, CA, AACE. - pp. 64-69. http://www.contrib.andrew.cmu.edu/~plb/WebNet96.html

[Calvi 1997] Calvi, L. *Navigation and Disorientation: A Case Study.* Journal of Educational Multimedia and Hypermedia. V.6(3/4). pp. 305-320. 1997.

[Calvi & DeBra 1997] Calvi, L. & De Bra, P. *Improving the Usability of Hypertext Courseware through Adaptive Linking* . The Eighth ACM Conference on Hypertext - Hypertext'97. pp. 224-225. Southampton, UK. April 1997.

[De La Passardière & Dufresne 1992] De La Passardière, B. & Dufresne, A. *Adaptive Navigational Tools for Educational Hypermedia.* Computer Assisted Learning. pp. 555-567. Berlin, New York: Springer-Verlag. 1992. http://mistral.ere.umontreal.ca/~dufresne/Publications/ical91.htm

[Eklund & Zeilliger 1996] Eklund, J., Zeilliger, R. *Navigating the Web: Possibilities and Practicalities for Adaptive Navigation Support.* Second Australian World Wide Web Conference (AusWeb96). Southern Cross University. Lismore. Australia. 1996.

[Pilar da Silva et al. 1997] Pilar da Silva, D., Van Durm, R. Hendrikx, K. Duval, E. Olivié, H. *A Simple Model for Adaptive Courseware Navigation.* WebNet'97 World Conference. Toronto, Canada. November 1-5, 1997.

[Signore et al. 1997] Signore, O. et al. *Tailoring Web Pages to Users' Needs* . Proc. of the workshop "Adaptive Systems and User Modeling on the WWW" - 6th International. Conference on User Modeling. June 1997.

URLs:
Denise Pilar da Silva: http://www.cs.kuleuven.ac.be/~denisep/
Erik Duval: http://www.cs.kuleuven.ac.be/~erikd
Computer Science Department: http://www.cs.kuleuven.ac.be/cwis/dept-E.html
Katholieke Universiteit Leuven: http://www.kuleuven.ac.be/kuleuven/KUL_en.html

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

An Evaluation of Adapted Hypermedia Techniques Using Static User Modelling

Jatinder Hothi and Wendy Hall
Southampton University, Electronics and Computer Science
University Road, Southampton, Hampshire, UK
{jh95r, wh}@ecs.soton.ac.uk

Abstract: Adaptive hypermedia has the potential to break through traditional educational barriers by allowing the tailoring of applications to specific user needs and requirements, to do this effectively the application must hold a model of the user. In this paper we present an empirical evaluation which was carried out to investigate various techniques in adapted hypermedia in education by comparing; adapted presentation, adapted navigation with a non adapted control hypermedia application. This experiment was also used to verify a static model (derived from the SaD static and dynamic) user model which was the result of initial analysis from a research project undertaken at Southampton University. Initial results from the evaluation look promising, with a shift in attitude of subjects towards the acceptance of adaptation in hypermedia applications with a high percentage of users preferring the adapted presentation application.

Keywords: Adaptive Hypermedia, User Modelling, Educational Hypermedia, Static User Modelling, Empirical Evaluation

1. Introduction

This paper will present the preliminary findings from an evaluation which was carried out to investigate various forms of adapted hypermedia. The user models used for this experiment were derived from an in-depth survey which was carried out to establish a simple user model for use in educational hypermedia applications. From the initial findings from this survey we suggested a SaD (static and dynamic) user model, section 2 will introduce the SaD model, section 3 and 4 will explain the design and preliminary results of the experiment and section 5 will present the direction which future research will take.

2. SaD (Static and Dynamic) User Modelling

A literature review revealed that over time most authors have shifted away from one level standard static user models which included information such as sex, age etc. (Totterdell et al, 1987) to more detailed two level models; a static user model with a dynamic user model containing user interaction information (Pohl et al, 1995) (Murphy et al, 1997) (Gutkauf et al, 1997) and (Beck et al, 1997). Although it is agreed that user models are required for adaptive hypermedia, ideas presented by individual authors vary in the level of detail required.
To gain a deeper understanding of user models a survey was carried out to construct a simple user model to be used in educational applications. Students ranging from computer science, arts and the science faculties from Southampton University took part in the study, all the students involved were using Microcosm (Davis et al, 1992) applications as part of their studies. They were required to answer questions on their previous computing and Microcosm experience, personality traits and background. The results were used to answer the following questions:

1. What variables should be modelled and to what level of detail?
2. How closely tailored to individual users does the model need to be?

Users were asked to evaluate themselves on their personality and computing experience. The analysis was carried out by correlating each variable with each of the other variables to establish if certain types of users interacted with or accepted a system in similar ways. It was evident from the analysis that some variables bared no relevance to the interaction process. For example, work, home environment, the number of people an individual may live with etc.

In general not all personality variables were found to be relevant to the interaction process however, traits such as shy, pessimistic and quiet did reveal some significance. Again the most relevant 'computing experience' attributes were quite clear i.e. users computing skill, windows, application experience proved to be extremely relevant for users when interacting with the application. More confident users required less help from the system. It was interesting to find that most users irrespective of their skill level wanted some level of information when errors were made. When analysing the results it became evident that a division in the type of relevant user model variables fell into two distinct categories which required the construction of two user models; a static user model which contained static user information and a dynamic user model which contained data which could be used by the system to dynamically adapt to the individual user. Hence, SaD (Static and Dynamic User Modelling), the basic SaD model can be seen in Figure 1.

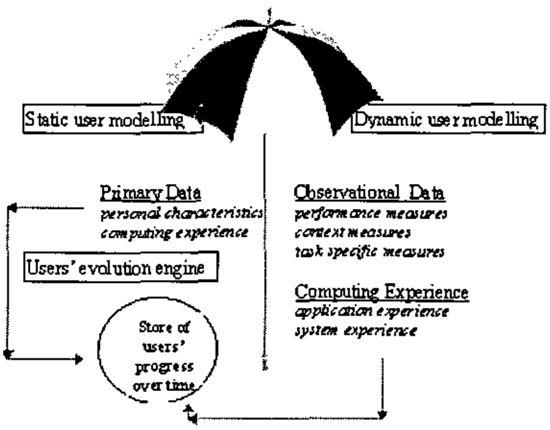


Figure 1 SaD (Static and Dynamic) User Model
A more detailed model can be found here.

3. Design of the Experiment

The experiment was devised to establish user's subjective preference when comparing different forms of adaptation (Brusilovsky,1997) and to verify the static user model. The experiment involved the implementation of three 'Archaeology Dating Techniques' applications, one which adapted the presentation of links (Application A), another which adapted the document content (adapted presentation) (Application B) and one as a control hypermedia application without any form of adaptation (Application C).

3.1 Adapted Hypermedia Applications

The applications were developed using Microcosm, an open hypermedia environment simply because it provides authors with the flexibility and freedom to explore different adaptive approaches. However, it is important to note here that the results from this experiment could be applicable to any hypermedia system as we are only considering adaptive hypermedia methods ie presentation issues and highlighted links.

Varying windows management experience between subjects presented a problem, therefore a software tool called SHEP (screen handler enabling process) was used to keep the interface consistent, whereby issues such as the overlapping or multiple opening of windows etc. could be addressed by manipulating settings in the SHEP files called Shepherds (Hall et al, 1997).

In all, four basic (static) stereotypical user models were devised based on system and content knowledge. Each of these models corresponded to a particular linkbase which held the links relevant to that particular user model group. Each subject was required to complete a pre-test questionnaire which was used to allocate that subject a user model. A matrix describing the codes used for each user model can be seen in Figure 2.

	Content skill *Novice*	**Content skill** *Expert*
System skill *Novice*	CNSN	CESN
System skill *Expert*	CNSE	CESE

Figure 2 Codes Used for Each User Model

For Application A, four linkbases (database of links in Microcosm) were used each corresponding to one of the four different user models so that the application displayed different links for the four types of users.
For Application B only two linkbases (CNSN and CNSE) were applied i.e. only the subject's system skill level was taken into account in the user model, because for this application we were only interested in the users' subjective view of the adapted presentation. In this application the entire information base and highlighted link structure was available to all users whatever their content skill level. However, more complicated and dense material which was included solely for the experts, was shaded out so that users could choose whether to read it or not, thus giving them a form of self adaptation.
Application C was used as a non-adapted control hypermedia application, which the subjects could use to compare the methods of adaptation. This application also utilised only one linkbase where only one user model was allocated to all types of users, however, No form of adaptation was used and all the subjects were presented with the same links, documents and hypermedia structure.

4. Empirical Evaluation For Adaptive Hypermedia

The results of the experiment were obtained using various evaluation techniques such as questionnaires, log of session data and observation. The experiment was set-up so that users were given pre-test questionnaires and depending on their subject and system knowledge they were allocated one out of four stereotypical static user models. The subjects were then allocated specific user names which corresponded to their user model, these user names were used to log into each application in turn.

The students were told about the main aims of the evaluation and what was required of them, these instructions were enforced by an instruction sheet which explained step by step exactly how to log onto the system. The instruction sheets also set out tasks for each application which the subjects were asked to complete in turn. Once each set of tasks had been completed they were asked to fill out a short questionnaire on that particular application type.

The evaluation was carried out over three separate sessions using a laboratory set up, the subjects came from Computer Science and Archaeology backgrounds. 7 females and 9 males with an age range from 18 to 35 took part in the study with experience ranging from 1st year undergraduates, PhD students to lecturers.

4.1 Preliminary Evaluation Results

4.1.1 Static User Model

The static user model did appear to match the correct user model to each subject. It was found that for total expert or total novice users in one particular area, i.e. in computing or content, it was straight forward to allocate correct user models as these groups work in a predictable way. However, for the average computer users and average content skill level groups (i.e. for users not on the extremes) it was more difficult to assess their knowledge unless they were asked to complete an in-depth questionnaire covering all the areas of the application. Therefore, a major problem revolved around users being unable to assess themselves on their computing skills. For this reason it is felt that the static user model requires more detail for users without computing background simply to determine what their real computer literacy levels are. Having said this it is important to point our that the pre-test questionnaire was designed to be very basic and force the users to select from very broad categories i.e. the users were given a choice of one out of two possible answers for most of the questions, for example expert/ novice.

A correlation analysis carried out on the data proved to be very useful. It clearly showed the variables which were significant to the way the subjects worked. In this experiment personality variables only showed some positive correlation. Having said this it is important to point out that the results from the first survey from which SaD was proposed did find certain personality variables extremely relevant in how well a user accepted a system. There is also a general acceptance within the user modelling community (Beck, 1997) that they are relevant to how a user interacts with a system. Therefore the lack of significance from the current analysis can be put down to the small population sample who took part in the study as opposed to the larger sample which took part in the first survey.

It was interesting to find that the number of hours a subject watched television and the number of hours a subject used a computer during a week was very significant in how a user accepted and worked with a system. The results showed that in general a high correlation between the subjects who spent more than 15 hours a week watching television and subjects with a low level of expertise in computing however, subjects who spent less then 15 hours a week watching television were more likely to be expert computer users. Also, the more windows a user claimed to work with at one time the more experienced they were on most of the packages listed in the questionnaire.

From the analysis it could be observed that some of the variables which were studied could be used to dynamically determine users' expertise. The time spent per document is a good indicator as the analysis showed, the CESE group spent less time on a document compared with the other two groups. Total session completion time per group was also significantly varied as the CESE expert group had a faster completion time and the CNSE group had the slowest session completion time. There was not much difference found in the number of nodes the different groups visited, however, the CESN group did visit more nodes within each application compared with the other two groups. The time spent on relevant documents proved to be more significant than the average time spent on each document, as the CNSE group spent more time on relevant documents than the CESN group and the CESE group overall spent the least time per relevant document.
During the analysis it was obvious that members of each user model worked in a similar manner as the number of links they followed, the time they spent on each document and the total session times etc were alike. This emphasised the relevance of allocating users stereotypical user models at the start of an interactive session.

4.1.2 Adaptive Techniques in Hypermedia

Adapted hypermedia was a new concept to most of the subjects who took part in the evaluation process, therefore a detailed explanation of adaptation and various methods was given in the pre and post-test questionnaire before asking the subject for his subjective preference on which method of adapted hypermedia they would feel most comfortable with. In the pre-test questionnaire most of the subjects chose the control hypermedia application, stating that they preferred to have access to all the material and all the links. However, after the evaluation session the results showed a significant shift in preference towards the acceptance of adapted hypermedia applications with a very low percentage liking the idea of the control hypermedia application but a very high percentage preferred the use of the adapted content application with the adapted links application coming a close second. These results can be seen in Figure 3 below.

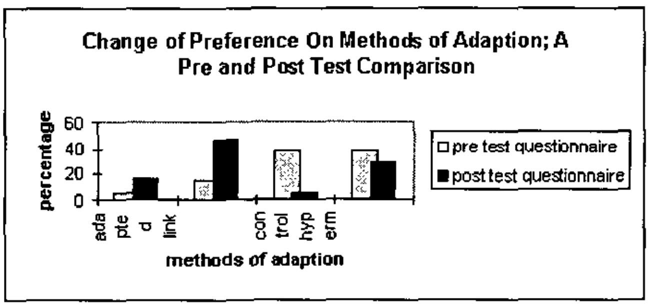


Figure 3 The Change of Preference Towards Adaptive Hypermedia (full size)

Some subjects even proposed the use of a combination of adaptation methods within one application. Again the reasons stated for this was that users would still have access to all the material but could avoid wasting time reading material which would be too complex or irrelevant for their particular task and that the information would be there in case they wanted to know more about a particular topic, other statements given included;

- *'help focus on relevant material instead of wasting time shifting through large amounts of text'*
- *'Avoid getting lost' and*
- *'it was easier to find relevant material'.*

These views were reflected in the answers given in the questionnaire where 78% of the subjects felt that all the material within an application should be made available to all types of users, just like a conventional book and 63% agreed that shading out chunks of irrelevant text reduced information overload. Figure 4 clearly shows that most of subjects spent the highest amount of time visiting nodes while using Application A. The subjects also spent on average a significantly greater amount of time visiting each node in Application C as opposed to Application B where the average time spent visiting each node was significantly less than with the other two applications. The results from this graph are significant in that although the subjects spent less time per node on average within Application B the percentage of correct answers was higher for this application in comparison to correct answers for the other two applications. This can be seen in Figure 5 below.

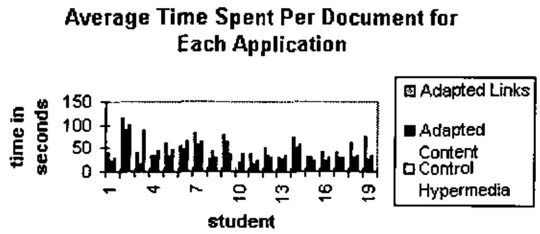


Figure 4 Average Time Spent On Each Document Within Each Application (full size)

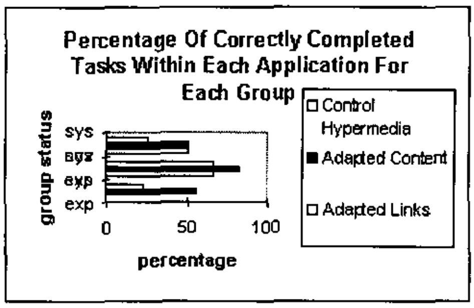


Figure 5 Correct Answers For Each Application (full size)

5. Conclusion and Future Work

The results from the experiment revealed that adaptive hypermedia was accepted by the subjects who had reservations about it at the start of the evaluation session. The subjects felt that adaptation was an important concept and that it could support the subject during the learning process when using educational applications. Most of the users felt comfortable with Application B but did agree that Application A played an important role in aiding navigation. There was a high emphasis placed on allowing users access to the entire information base within an educational hypermedia application.
This experiment also revealed some of the most important variables required for user modelling which included computing experience and subject studied etc. However, it was found that the static model requires more computing background information so the user has a better chance of assessing themselves correctly and the system has a more detailed model of the user to provide the correct level of adaptation.

The next stage of the research will involve gathering results from the evaluation described here design an educational adaptive hypermedia application which will aid navigation, reduce information overload and at the same time allow access to the whole information base within the application. The same experiment will be used to implement the refined version of the SaD model so that the adaptation will take place dynamically.

It is important to note here that the users cognitive model is one division of the user model and due to it's complexity this area was deliberately left out from this work.

References

Beck, J., Stern, M. and Woolf, B. P., (1997), Using The Student Model To Control Problem Difficulty in *User Modelling; Proceedings of the 6th International Conference* eds. A. Jameson, C. Paris and C. Tasso, Springer Wien.

Brusilovsky, P. (1997), Efficient techniques for adaptive hypermedia in Intelligent hypertext: Advanced techniques for the World Wide Web, *Lecture Notes in Computer Science*, vol.1326 eds. C. Nicholas, J. Mayfield, pp. 12-30. Springer-Verlag, Berlin.

Davis, H., Hall, W., Heath, I. and Hill, G., (1992), Towards an Integrated Information Environment with Open Hypermedia Systems in *Proceedings of the ACM conference on Hypertext* eds. D. Lucarella, J. Namard, M. Namard and P. Paolini, ACM Press.

Gutkaut, B., Thies, S. and Domik, G., (1997), A User-Adaptive Chart Editing System Based On User Modelling and Critiquing in *User Modelling; Proceedings of the 6th International Conference* eds. A. Jameson, C. Paris and C. Tasso, Springer Wien.

Hall, W., Weal, M., Heath, I., Wills, G.B., & Crowder, R.M., (1997), Flexible Interfaces in the Industrial Environment in *International Conference Managing Enterprises- Stakeholders, Engineering, Logistics and Achievement* (ME-SELA'97), pp453-460, Loughborough, UK.

Murphy, M. and McTear, M., (1997), Learner Modelling for Intelligent CALL in *User Modelling; Proceedings of the 6th International Conference* eds. A. Jameson, C. Paris and C. Tasso, Springer Wien.

Pohl, W., Kobsa, A. and Knutter, O., (1995), User Model Aquisition Heuristics Based on Dialogue Acts in *International Workshop on the Design of Cooperative Systems,* 471-486, France.

Totterdell, P. and Cooper, P., (1987), Design and Evaluation of the AID Adaptive Front End to Telecom Gold in *BCS Workshop Series Proceedings of the 3rd conference of BCS HCI Specialist People and Computers* eds. D. Diaper and R. Winder, Cambridge University Press.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Computed Web Links: The COOL Link Model

Michael Miller and L. Jay Wantz
Intelligent Automation, Inc.
Rockville, MD 20850 USA
{mm@i-a-i.com, jwantz@i-a-i.com}
phone: (301)590-3155, fax: (301)590-9414

Abstract: As the Web evolves it is likely to become more user-centered, where navigation and search are more personalized and individualized. Signs of this are already surfacing in various guises including filters and other censorship tools, the use of cookies as user profile repositories, and metadata-aided search. Whether the user-centered approach will become the predominant model is open to debate, but it does appear that it will continue to make inroads. With minimal adaptations the technology available today can better manage and deliver the vast amount of information on the Web by meeting the user's needs, goals, and preferences. This paper discuss one such adaptation: COOL links. These are multi-ended, computed hyperlinks, which foster a new model of Web navigation, applicable to a variety of disparate domains, including education and marketing.

Keywords: multi-ended hyperlink, computed hyperlink, user-centered navigation, individualization, user profiles

1. Introduction

Today's Web navigation features user manually selecting their own path through single-ended "hot" links. This *user-directed* approach, though successful and useful for many purposes, has a number of drawbacks, which, as more and more information becomes available on the Web, seem to loom larger. In particular, navigation is often punctuated by frequent and lengthy searches and dead-end paths through irrelevant, undesired, unsolicited or inappropriate information. The result is that the user is often unaware of relevant aspects of a collection of resources until he or she has navigated through much unwanted material. In addition, it is not unusual that clicking on a link leads to surprises, irrelevant paths, or a feeling of being lost in "cyberspace."

Traditional navigation ignores an individual's specific preferences, needs, or other abilities (e.g., reading level). This can be particularly problematic when the user is under time, or learning and comprehension constraints, as in educational situations. An alternative is automated guidance toward material that is of interest, contextually relevant, and appropriate for the user. In an educational setting it would be useful for a learner to be guided along a path of Web resources which meet his or her educational needs. In the commercial arena it would be useful to direct the Web user to those products and services in which he or she might be especially interested.

Notable examples of such non-traditional approaches are: Walden's Paths [FU97], WebWatcher [JO95] and Letizia [LI95]. These systems have shown some promise, especially in locating pages similar to those already found to be of interest to the user, but stop short of locating Web resources for specific users with specific educational needs and goals. With minimal adaptations and the technology available today, the vast amount of information on the Web can be even better managed and delivered to the meet the user's needs, goals, preferences and attributes. COOL links [WA97] are one such adaptation.

2. Not all Link Types are Created Equal

Most familiar to the Web community is the single-ended, uni-directional "hot" link, denoted by the familiar HTML *href* tag in an *anchor* block. Each link of this type is intended to refer to the address of exactly one resource, which itself is denoted statically in the HTML. Thus, one of the prime attributes and attractions of these links is that they are unambiguous. Another is that the HTML author knows exactly where a viewer of the page will "go" if a link is clicked.

Other link types, such as multi-ended links, do exist and have been implemented in a variety of other hypermedia environments (not the Web). At first it seems that a multi-ended link--one which "refers" to multiple resources--applied to the Web would be ambiguous. To which of the multiple resources is the user taken when the link is clicked?

COOL links are an example of multi-ended Web links that are *not* ambiguous, because a COOL Link is also a form of a computed link. This is accomplished by letting these links "refer" to a collection of resources only until click-time. The resources associated with a COOL link are unordered. There is no special significance given to those that appear closer to the start of the link description than those that appear nearer the end. Instead, when a COOL link is clicked the metadata of each component resource is evaluated and compared against a separate set of input parameters, for instance the user's profile, and the "best-fit" resource is returned.

Presumably, each distinct resource in a COOL link collection provides different benefits to different users (though each resource may contain information about the same subject or topic). The burden of choosing a link component from the collection is placed upon the browsing tool (or a plug-in) at runtime.

In summary, the four-part COOL link machinery contains: (i) The link itself (its implementation), (ii) metadata descriptions of the linked resources, (iii) external input features such as user profiles, and (iv) link computation algorithm(s).

2.1 Link Implementation

Our current implementation of COOL Links uses a Javascript function named *CoolLink* to create the computed nature of the link. The CoolLink function takes a variable number of arguments, where each argument represents the metadata of one of the destination resources in the multi-ended link. The metadata contains a pointer to the Web resource it describes, thus separating resource description and annotation from content authoring. In HTML, a COOL Link might look like the following:

```
<a href="default.html" OnMouseOver="this.href=
                       CoolLink('res1.rdf', 'res2.rdf', 'res3.rdf', 'res4
```

A single resource is selected from the collection by an evaluation scheme, based upon the user's profile. In this implementation, a user must first log onto the system before the COOL Links will function properly. The user's login name is stored locally in a cookie, but the profile information is stored in a server-side database for security reasons. If no user has logged onto the system, or if a link cannot be computed, the browser destination is set to the default location specified in the *href* attribute.

Our implementation of COOL Links will be further enhanced in the future. With the acceptance of the XML specification, the W3C will be supporting the notion of multi-ended links on the Web. Therefore, part of the COOL Link machinery will be supported inherently by the markup language. Secondly, we will allow for different types of computational algorithms to be associated with a given COOL Link. A COOL Link computation doesn't necessarily have to involve a comparison with the user's profile, it might be based upon other external input. Finally, in cases where the user's profile is required, we will be caching the user profile locally, and eliminating additional server hits.

2.2 Resource Metadata

Today's Web community (including the W3C, the major browser makers, and the creators of the search engines) has embraced the collection and use of metadata to characterize Web resources. Several notable metadata efforts have made an impact, directly or otherwise, on the Web. Among those efforts, and of interest to our current work, are the Dublin Core [WE97], the Gateway to Educational Materials (GEM) [SU98], the Warwick Framework [DE96], the Learning Objects Metadata Group (LOMG), and the Instructional Management System (IMS).

In our past implementations of COOL Links, we used the Courseware Description Language (CDL) [HA96tr, WA98] metadata dictionary to describe Web resources. The CDL dictionary was first developed in late 1995, independent of the metadata work in digital cataloging (Dublin Core) and online clearinghouses (GEM). Yet, as one might suspect, there are many similarities (e.g., most elements of the Dublin Core map directly to elements in CDL). The CDL is an extensive collection of more than 50 attributes for describing educational resources that are particularly appropriate for K-12 education applications. Descriptions may characterize a resource's applicability to specific grade levels, its support of various pedagogical models, its time requirements, its dependency on various teacher roles, and its relationship to other resources in a set, to name a few. One of the primary goals in the development of the CDL was to address the individual needs of students, including the special needs of students with disabilities. Through the use of specialized tools, such as COOL Links, we now can construct educational hypertext on the Web that contains dynamic paths. As an individual student clicks a link, his user profile is compared to the CDL descriptions of the associated resources, changing the path to meet his needs.

In our current implementations we are adopting the W3C's Resource Description Framework (RDF) syntax for representing metadata on the Web. We have embraced the Dublin Core and GEM metadata dictionaries as core components, while adding extensions from our own CDL dictionary, to create resource descriptions that follow the concepts of the Warwick Framework. The following is an example of how an eighth grade resource in biology might be described using RDF syntax and the three metadata dictionaries in combination:

```
<?xml:namespace name="http://purl.org/metadata/dublin_core#" as="DC"?>
<?xml:namespace name="http://www.geminfo.org/#" as="GEM"?>
<?xml:namespace name="http://www.i-a-i.com/services/cdl/#" as="CDL"?>
<?xml:namespace name="http://www.w3.org/TR/WD-rdf-syntax#" as="RDF"?>
<RDF:RDF>
   <RDF:Description RDF:HREF="http://www.i-a-i.com/LabExperiment.html">
      <DC:Creator>Olaf Kolzig</DC:Creator>
      <DC:Format>text/html</DC:Format>
      <DC:Date>1998-21-05</DC:Date>
      <DC:Title>A closer look at single-cell micro-organisms</DC:Title>
      <GEM:Grade>8</GEM:Grade>
      <GEM:Pedagogy>Self-directed groups</GEM:Pedagogy>
      <GEM:Duration>Two one-hour sessions</GEM:Duration>
      <CDL:UserRole>Discovery</CDL:UserRole>
      <CDL:UserMode>Multi-sensory</CDL:UserMode>
      <CDL:Equipment>Microscope, slides, eye dropper</CDL:Equipment>
   </RDF:Description>
</RDF:RDF>
```

2.3 User Profiles

A key concept here, and one that separates the current work from other Web-based metadata applications, is our employment of user profiling together with metadata to help locate Web resources. While traditional navigation ignores the specific traits of an individual, the user-centered approach relies on them. The characteristics of the user are compared with Web resources during search and navigation, helping to prune the information space and direct the user toward the most relevant resources.

Most of our current COOL link implementations are in the context of delivering educational material via the Web, and thus the class of profile traits is circumscribed by pedagogy. Clearly, however, this will need to be expanded for future applications. We do not see this as a problem, however. Extensive profiles, containing both domain dependent and independent traits may be maintained, though not all traits need factor in the comparisons done for a particular COOL link. (Exactly which traits are to be used are determined by context, and go into parameterizing a Selection Algorithm--see below.)

To date, we have used only manual profiling techniques, though we see this as insufficient. One problem with manual profiling is that the user is required to maintain the profile, making modifications as his or her characteristics change. For instance, as a student's reading abilities improve, the profile must be updated to reflect that progression. Otherwise, as a profile becomes out of date the user will be directed toward resources that are no longer relevant. Not only does this become obtrusive, but also there are traits and scales for which the user is likely unaware of precisely where he or she fits. Reading level, which may be based on a Flesch-Kincaid and/or Fog indices of readability, is an example.

In contrast, a (semi-)automated approach requires less direct interaction from the user in order to develop and maintain a comprehensive profile. There have been localized successes in the area of machine learning, where one or more preferences or characteristics can be inferred, but not a comprehensive user profile. The Web-based systems that have recently been built (including [JO95] and Letizia [LI95]) tend to focus on identifying those resources which will be of interest to a user, though not on identifying those which will meet the user's educational goals, given their capabilities and knowledge. We continue to investigate this fertile area of research.

2.4 Selection Algorithms

Resolving a COOL link, i.e., selecting a link from the collection of links contained in a COOL link, is the task of a selection algorithm contained in the browsing tool or a plug-in. This selection algorithm compares a resource's metadata characteristics with aspects of the user's profile, or other external features, such as time of day, domain name of the user's machine, etc., and outputs the resource found to be "most appropriate."

To facilitate this, metadata descriptions of each component of a COOL link must be fetched. In general, fetching is intended to be carried out at runtime, i.e., at the time that a COOL link is clicked, however, some optimization can be achieved (when a page contains a relatively small number COOL links) by doing all of the metadata fetches necessary for a page as a background process begun at page load time. Then when a COOL link is clicked, no fetch time is required.

Various criteria for choosing a component resource can be implemented by link selection algorithms. We have developed a generic link selection mechanism and incorporated it into our Web implementations (see, [WA97]), but other specialized algorithms, or parameterization of the generic version will be useful in future applications and contexts. We intend this selection component to help deal with some of the the domain dependence vs. domain independence issues mentioned above. Namely, the context of use of a page containing COOL links can help determine the parameters of a generic algorithm, or the specialized algorithm, which should be used for resolving the page's component COOL links. Thus, pages used in an educational setting could invoke a different selection algorithm than those used in a more commercial context. Note, however, that a single user profile could serve both, with profile traits being weighted in accordance with the domain.

3. Applications

COOL links are applicable in a variety of domains and in a variety of different ways, including:

Education: Students tend to progress at their own pace, learn differently, and have a diverse set of skills, even though they are following the same lesson. Ideally, an instructor takes a student's characteristics into account, and teaches accordingly. COOL links can serve as a tool to facilitate lesson individualization in an electronic learning environment. As students navigate Web-based lesson material templates, they can be guided on a path that is most appropriate to their learning goals and capabilities. A link selection mechanism determines a path dynamically as a student browses through courseware containing COOL links. An instructor or lesson developer can create a single lesson plan template which contains different material for use by students with differing needs. Attributes like the pedagogical "appropriateness," associated grade level, and readability index of a resource in relation to a student's profile may weigh heavily on the choices made by the selection mechanism.

User-centric Navigation: A great deal of research has been carried out on Web trails, paths and guided tours[TR88]. Trails tend to be contextually relevant and interesting to the user, but are not necessarily shaped by characteristics of the individual user himself. Taking these characteristics into account leads to user-centric navigation (UCN)--the automated guidance of an individual through the Web.[WA97] UCN is a paradigm shift from the more passive modes of hypertext navigation as it facilitates the development of dynamic trails based upon context, user interaction and user needs.

Advertising, Marketing and Sales: The color, size, cost, or style of a particular consumer item in relation to a consumer's preferences may affect his or her desire to purchase an item. Additionally, other attributes of the consumer like background, gender, age, career and income will affect the likelihood of a purchase. With advertising and sales a mainstay of the Web, sellers are looking for ways to direct those most interested in their products to their pages. COOL links, together with user profile data, are one way to accomplish some amount of successful marketing, and in the process eliminate large numbers of unwanted hits.

4. Conclusion

As the Web continues to grow, more and more information will become available. Experience has shown that the more information available, the harder it will be to find what we want unless some measures are taken to better tailor navigation and searching technologies. There will continue to be times that traditional user-directed Web navigation will be entertaining and informative, but there will be other times when it will be frustrating and wasteful. Multi-ended, computed COOL links offer a new way to view the Web, with minimal impact on the existing Web architecture. With the current emphasis placed on metadata and user profiling, and the mounting frustrations stemming from wasted time spent searching for resources, we hope that COOL links will continue to make inroads into the Web, especially as new applications highlight their benefit and exemplify their utility.

Acknowledgments

This research and development was supported (in part) by DARPA contract #N66001-95-C-8626, and US Department of Defense Education Activity (DoDEA) contract #N66001-97-M-C528.

References

[DE96] Dempsey, L., Weibel, S. "The Warwick Metadata Workshop: A Framework for the Deployment of Resource Description." *D-Lib Magazine*, July/August 1996.

[FU97] Furuta, R., Shipman, F., Marshall, C., Brenner, D., Hsieh, H., "Hypertext Paths and the World-Wide Web: Experiences with Walden's Paths," In *Hypertext '97: the Eighth ACM Conference on Hypertext* (Southampton, U.K., April 6-11 1997), pp. 167-176.

[HA96tr] Haynes, J., Wantz, L., Miller, M. "Courseware Description Language," *IAI Technical Report* (Rockville, MD), 1996.

[JO95] Joachims, T., Mitchell, T., Freitag, D., Armstrong R. "WebWatcher: Machine Learning and Hypertext," *Fachgruppentreffen Maschinelles Lernen* (Dortmund, Germany), August 1995.

[LI95] Lieberman, H. "Letizia: An Agent That Assists Web Browsing," *International Joint Conference on Artificial Intelligence* (Montreal), August 1995.

[MI98] Miller, M., Wantz, L. "COOL Links: Ride the Wave," In *Proceedings of the Seventh International World Wide Web Conference* (Brisbane, Australia), April 1998.

[SU98] Sutton, S. "Gateway to Educational Materials (GEM): metadata for networked information discovery and retrieval," In *Proceedings of the Seventh International World Wide Web Conference* (Brisbane, Australia), April 1998.

[TR88] Trigg, R.H. "Guided Tours and Tabletops: Tools for communicating in a hypertext environment." *ACM Trans. Off. Inf. Syst. 6, 4* (Oct. 1988), 398-414

[WA98] Wantz, L., Miller, M. "Educational Metadata In Use," In *Proceedings of the Seventh International World Wide Web Conference* (Brisbane, Australia), April 1998.

[WA97] Wantz, L., Miller, M. "Toward User-Centric Navigation of the Web: COOL Links using SPI," In *Proceedings of the Sixth International World Wide Web Conference* (Santa Clara, CA), April 1997.

[WE97] Weibel, S. et. al. "The 4th Dublin Core Metadata Workshop Report." *D-Lib Magazine*, June 1997

URL References

COOL Links:	*http://www.i-a-i.com/services/cool/*
Courseware Description Language (CDL):	*http://www.i-a-i.com/services/cdl/*
Dublin Core:	*http://purl.oclc.org/metadata/dublin_core/*
Gateway to Educational Materials (GEM):	*http://geminfo.org/*
Instructional Management System (IMS) project:	*http://www.imsproject.org/*
Letizia:	*http://lieber.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html*
NIST Learning Objects Metadata Group (LOMG):	*http://jetta.ncsl.nist.gov/metadata/*
RDF	*http://www.w3.org/Metadata/*
W3C:	*http://www.w3.org/*
Walden's Paths:	*http://www.csdl.tamu.edu/walden/*
Warwick Framework:	*http://www.ub2.lu.se/tk/warwick.html*
Web Watcher:	*http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-6/web-agent/www/project-home.html*
XML:	*http://www.w3.org/XML/*

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

User Adaptivity on WWW through CHEOPS

Alberto Negro Vittorio Scarano Rosaria Simari
Dipartimento di Informatica ed Applicazioni "R.M. Capocelli",
Università di Salerno
83025, Baronissi (Salerno), Italy
e-mail: {alberto | vitsca}@dia.unisa.it

Abstract: In this paper, we present some developments of CHEOPS , a package that can be used by WWW designers to provide their hyperdocument with adaptivity and navigational aids.
Previous version of CHEOPS , presented in (Ferrandino et al. 1997), is implemented by several CGI-BIN scripts that enables an HTTP server to interact with a user and adapt its responses to her behaviour. Solution is transparent to the user and to the client, and has a moderate impact on server performances.
What we present here is the extension of CHEOPS that takes into account user modeling (i.e. extends adaptivity to multi-session interactions) providing also annotations capability. The new version of CHEOPS , named 1.0, is developed in PERL and is therefore more portable.
We conclude by sketching future directions of CHEOPS and some possible usage of the knowledge model in on-line help and training systems.

Keywords: User Adaptivity on WWW, PERL, CGI-BIN

Introduction

The World Wide Web (WWW), developed in 1989 at CERN as a means of sharing information within the organization, is now an information retrieval system on the Internet that can be considered the first real global hypermedia network. The exponential growth of WWW has made it the most visible service offered by the Internet and, therefore, has attracted a variety of interests.

Among them, one of the most recognizable is the educational usage of WWW. In fact, several educational systems based on WWW have been developed in the recent past and many results are available on the educational use of the Web for distance learning [Dwyer et al. (1995),Hammond and Allison (1989),Ibrahim (1994),Ibrahim and Franklin (1995)]. In particular, in [Ibrahim (1994)] the need for a ``WWW work session" (using our terminology) in an educational setting was recognized and an application based on CGI-BIN programs was developed that adds a similar capability to the Web server.

In order to fully exploit WWW potentiality in the educational field, server response must be *adaptive*: it must, in fact be aware of user's behaviour so that it can take into account the level of knowledge and, as a consequence, provide the user with documents that she can read, given her background/capacity.

Adaptive hypermedia is a recent research area whose goal is to enhance the functionality of hypermedia by building a model of the user and adapting the response accordingly[Brusilovsky (1996),Brusilovsky (1997)].

What we propose here is a system that is based on an ``opaque" mechanism, that, based on the history of user's requests, can provide a different document that is likely to be understood and lightly challenging while not being intimidating. This can be useful also in other contexts, i.e., avoiding that the large amount of information available can scare a user off the site, since she can `` *have trouble in finding the information they need*" [Nielsen (1990)] and, therefore, can loose interest in the site itself.

CHEOPS System

CHEOPS is oriented toward a ``server-side", application-specific solution to the adaptivity requirement, with particular care in avoiding overloading the server with many tasks. Its architecture is designed to limit concerns on users' privacy: the session that CHEOPS keeps track of is strictly limited to the ``current" hyperdocument and is not extended on user's behaviour on the whole Web site. Moreover, CHEOPS , although specific to educational application, can be also generalized to other similar situations when presenting information at the right rate (for the user) is crucial.

CHEOPS is a design system that should make easy for a designer to add adaptivity to a hyperdocument in a modular way. Especially with the extensions described here and available in version 1.0, this goal looks quite reachable.

CHEOPS adopts the following strategy: each time a new user is presented to the systems, the server creates a user's profile, ``distributing" (virtually) a ``session card" *embedded within the hypertext* file that is sent back to the browser. Each successive requests to the server will be done with this session card and, therefore, recognized as belonging to a specific session. The system is also able to recognize (by a username-password pairs) users and recover the profile and the history for her.

CHEOPS system is non-obtrusive, widely usable, (one does not need specific browsers to use our system), has moderate impact on server load (putting all the ``memory" of the session where it belongs, i.e. on the disk) and is modular enough to be easily installed on a Web server for a specific hyperdocument.

CHEOPS is implemented through several CGI-BIN PERL scripts, used to build ``on-the-fly" documents that depend on user's previous inquiries at the same server. CGI-BIN scripts are program run on the server and triggered by input from a browser.

System Architecture

CHEOPS is a ``server-side" implementation of a session-based interaction model based on several specifications required by an educational, WWW-based, software.

The system, given a hyperdocument as HTML pages, is able to provide the requested page to the user according to her profile that is modified each time the user chooses a link. The system provides the page adding additional information like links to context-sensitive help, information about her current profile (given in terms of knowledge levels for each category, details follow in the next subsection), links to ``summaries" for each category, a form to add annotation to each visited page, link to the history of the visited documents and so on.

CHEOPS version 0.9 (presented in [Ferrandino et al. (1997)]) consisted of several Unix Shell scripts that provide the designer of a hyperdocument with an automatic mechanism that satisfies user's requests according to previous inquiries. The extension presented in this paper are mainly about the *user profiling* mechanism and the annotation capability that has been added to the system.

Analysis of requirements. Being focussed on developing a design tool for hyperdocument to be employed in the educational field, several requirements were identified during preliminary experiences:

- *Lecture-type Interaction:* The user should be able to interact with a WWW hyperdocument that makes her feel like having a teacher/instructor/field-expert *`` on the other side of the screen"*. To reach such a goal, it is necessary to have tools able to distinguish between requests of different users. It is also recommended the usage of context-sensitive help, where by context it is not only meant the part of hyperdocument currently visited but also the user knowledge and previous interactions with the help system.
- *Categories:* It is often the case that there is a ``natural" subdivision of the hyperdocument in categories that are well-suited to the argument. It is important that user is able to check her ``knowledge level" for each category and also able to choose to further explore a category that, willingly or not, was ignored at a previous step.
- *History Mechanism and Annotations:* User should be able to check the path followed during her interaction with the hyperdocument: being able to trace back previous interactions with the comments to each single visited page is extremely helpful in developing and reviewing the actions taken.

The Session Protocol. The session protocol that has been implemented is based on the following sequence of actions:

1. *Introduction.* A new user is introducing herself to the server through the choice of an ``introductory" link. At this time, the user can be recognized as an already known user and, in this case, CHEOPS is able to *recover*

the session previously interrupted (see step 5). If the user is a new one, the system is able to manage the request of a new user to enroll into the system by asking her a username-password. pair.

2. *Beginning of the Session.* When the user is known by CHEOPS , the server responds by sending the user a *Session Card*, that is a unique, opaque, identifier that will be used in the following interactions by the client thus identifying itself as ``known" for the system.
3. *Session.* An (unlimited) number of interactions user-server where each action is accounted for and used by the server in order to adapt its future responses to the same user.
4. *Interrupt the Session.* The user is simply chosing a different URL and leaves the system.
5. *Recovery of the Session.* The user is resuming the session and the system allows for recovery of information about the user.

Knowledge Model

CHEOPS is based on the knowledge model, already presented in [Ferrandino et al. (1997)]. Here we describe its motivations and characteristics.

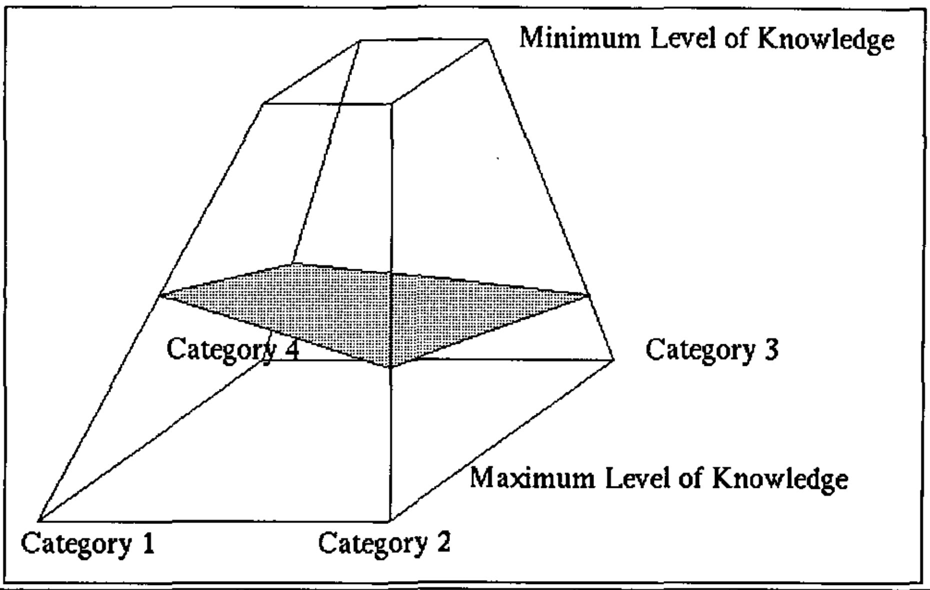
When a hyperdocument is designed for a broad audience, it is customary to face several problems in trying to make its *style* acceptable for novices, amateurs and experts at the same time. They all have different, some time opposite needs: a novice may need information that can be boring for an expert since it makes her loose time in finding the more advanced part of the hyperdocument. On the other side, a novice can be scared and confused by an excessive usage of technicality and leave the site before she could have the chance to access information that are suitable for her experience in the field. Then, great care must be taken to fulfill this important, we dare say paramount, requirement if one is asked to use at its best such an efficient communication media like the World Wide Web.

Another orthogonal need is to help the user (whichever level and experience they have) in navigating through the (sometimes huge) amount of information. This is often achieved by subdividing data in a limited number of *categories* that can be seen as providing an hypertext of a small degree of traditional linearity within the hypertext itself.

Putting all together, the view we propose as a model for dealing with the ``shapeless" amount of information is a geometric one: let us build a *Knowledge Pyramid*, having as bases two regular polygons of the same shape but of different size, the smallest on the top. Each vertex of the polygon represents one of the categories which the designers subdivided the whole amount in; the edges of the pyramid represents the experience level in each category.

The experience acquired by a user is defined as the surface obtained interpolating the intersection of the current experience levels on each category. The learning process can be seen as trying to get the surface as low as possible.

A possible variation of the model can be to have fewer category on the top than on the bottom. This corresponds to the natural mechanism to avoid mentioning some categories if the user may not be ready and actually may be disturbed by that.



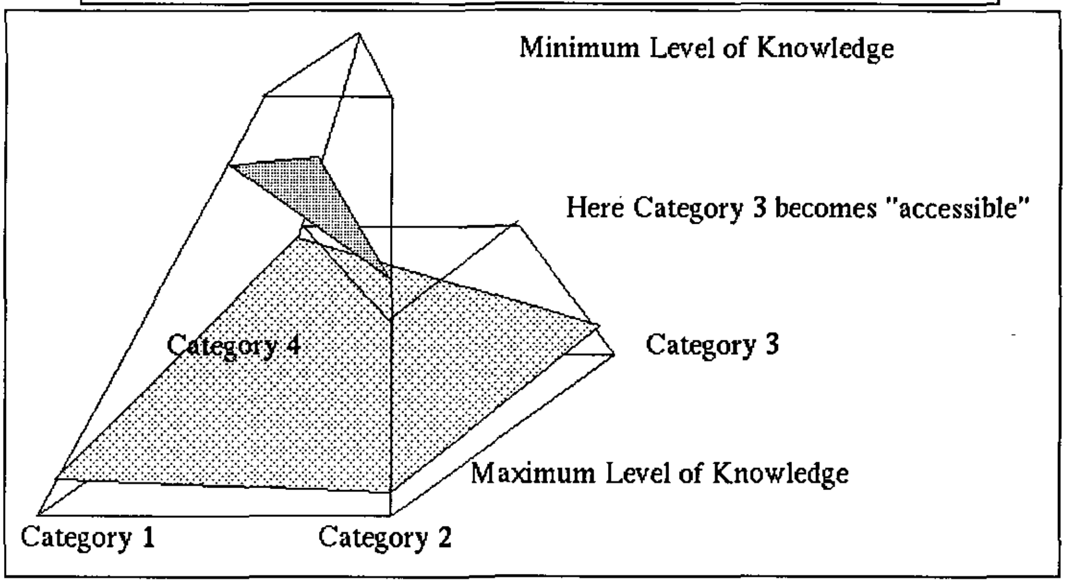


Figure 1: Top. The Knowledge Pyramid: corners represent categories and the surface is the ``current level" of knowledge of the subject.(full size) **Bottom.** The Model when Category 3 is accessible only given a certain knowledge of the subject.(full size)

CHEOPS uses the Knowledge Pyramid as an important navigational aid to the user: at each step, the user is able to see what her confidence for each category is (or is supposed to be by CHEOPS) and act as a consequence. She may choose to explore a part of the hyperdocument she left behind, or she may want to go deeper in a category or she may go on with her hypertextual visit of the document.

User Modeling through CHEOPS 1.0

Here, we briefly describe how to use and the new capabilities available in CHEOPS 1.0. More information about the package (and the package itself) will soon be available at the URL http://stromboli.dia.unisa.it/CHEOPS/.

The entire package has been moved from `csh` to a more portable and powerful language as PERL. The system, moreover, uses Relational Database System ([RDB]) for storing both users' profiles and history information.

How to use CHEOPS

CHEOPS is designed taking into account the needs of designers: not much additional work is required in order to make a hyperdocument adaptive to user's responses. Once, CHEOPS is installed into the CGI-BIN directory of the WWW server and the system is configured so that every tool needed is available, the remaining tasks are as follows:

- Design the WWW hyperdocument as usual: HTML files with hyperlinks pointing to other files within the whole hyperdocument.
- Categorize the files in categories with knowledge levels. Fix the visibility of some categories.
- Edit the default configuration file `general.dat` provided with the distribution. Insert all the information relevant to the hyperdocument being designed as its directory, home page, the first file to be shown, names of the categories and so on.
- Build the Category summaries that collect (in a domain-sepcific way) links to all the documents at the same level for a given category. Summaries are, usually, already been developed during the usual design of the hyperdocument.
- Provide an entry point to the document as a call to particular CGI-BIN script.

New Capabilities

In this subsection we briefly describe the major extensions provided in CHEOPS 1.0.

Annotations. CHEOPS 0.9 provided a history of the interactions to the user. Now, the system allows the user to add annotations to each visited page, by evaluating each page as ``Not interesting", ``Interesting" and ``Very Interesting" and by adding a one-line comment. At this moment, annotations are visible only to the user that wrote them. Future developments will allow to share comments on pages.

Adaptive and Adaptable. The system now fully allows the user to choose and modify the knowledge levels. CHEOPS is, therefore, not only adaptive but also adaptable, a characteristic that is very important for experienced users.

Categories. In this version, we have enhanced the way the designer can choose to adapt CHEOPS to its domain-specific application. In fact, while inCHEOPS 0.9 only a fixed number of categories were available, in the new version the designer can choose the number of categories as she likes. Moreover, and maybe even more important, the designer can decide to have a multi-faceted pyramid: a category can appear to the user only if a certain knowledge level on other categories has been reached.

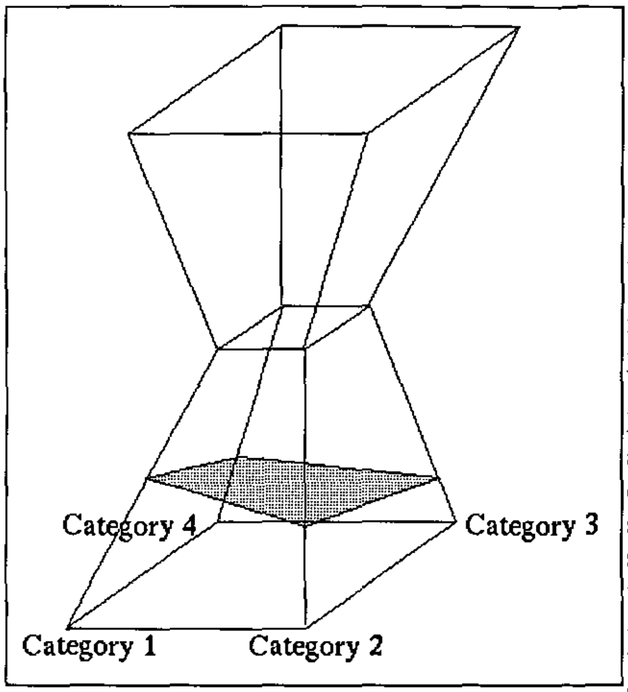
User modeling. The system is now able to keep the current profile of each user in a DataBase so that it can be easily retrieved in successive interactions. The system allows also (on a WWW server Apache [Apache]) to insert and change password for users through a fill-in form.

Removable Headers/Footers. CHEOPS inserts header and footer for each page. Now the designer can choose to provide or not headers/footers for each page while the user at run-time can choose to turn headers/footers on and off.

Conclusions

The design of CHEOPS has been highly influenced by a project of an hypertextual document on Theory of Music that, in fact, represents a useful paradygm since the argument can be accessed both by a novice and by an expert. Other experiences in using CHEOPS 1.0 have been done in the field of Medical Information System for Stomach Illnesses and have been deployed on a PC, using the portability of PERL.

We feel that the knowledge model introduce by CHEOPS can be very helpful in fields that are only marginally related to ``pure" educational purposes as in Computer Based Training (CBT) systems or on-line help and training systems.



In this case, in fact, the CHEOPS knowledge model can be extended to encompass two pyramids, a top one and a bottom one, connected as shown at the left. The motivation is that, at beginning, the trainee is in ``training" process and, therefore, needs a lot of information. When the trainee is sufficiently expert she does not need such a huge amount of information and therefore, her profile is now staying in the area between the bottom and the top pyramid. If necessary, when some particular problem requires extremely specific information, the trainee can go deeper and deeper and her profile moves down to the bottom pyramid.

Future extensions of CHEOPS are, therefore related to such applications. We plan to study the training process and develop different knowledge models by providing, at the same time, a package that allows to realize such adaptive systems on the World Wide Web. Relevant future work include dealing with multiple user annotations and providing adaptivity in a cooperative environment, where different trainees collaborate over the learning process and are seen by CHEOPS as a group of users that, as a whole, behave as a unique user still mantaining personal peculiarities. Moreover, research on authoring systems for CHEOPS is also planned.

References

Apache
``WWW Server Apache". `http://www.apache.org`.

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

Brusilovsky (1997)
P. Brusilovsky. ``Efficient techniques for adaptive hypermedia". In: C. Nicholas and J. Mayfield (eds.): Intelligent hypertext: Advanced techniques for the World Wide Web. Lecture Notes in Computer Science, Vol. 1326, Berlin: Springer-Verlag, pp. 12-30, 1997.

Dwyer et al. (1995)
D. Dwyer, K. Barbieri, H.M. Doerr. `` *Creating a Virtual Classroom for Interactive Education on the Web*". Proc. of WWW 95, Third Int. Conf. on World Wide Web.

Ferrandino et al. (1997)
S.Ferrandino, A.Negro, V.Scarano, ``CHEOPS : Adaptive Hypermedia on World Wide Web". Proceedings of the European Workshop on Interactive Distributed Multimedia Systems and Telecommunicazion Services (IDMS '97), 10-12 Sept. 1997. Ed. Springer-Verlag (Lecture Notes in Computer Science).

Boutell
T.Boutell, ``GD 1.2, *A graphics Library for fast GIF creation*". Quest Protein Database Center, Cold Spring Harbor Labs (US).

Hammond and Allison (1989)
N. Hammond, L. Allison. `` *Extending Hypertext for learning: An investigation of access and guidance tools*". In Sutcliffe,A. and Macaulay, L. (Eds.): *People and Computers*, Cambridge University Press, 1989.

Ibrahim (1994)
B. Ibrahim, `` *World-Wide Algorithm Animation*". *Computer networks and ISDN Systems*, Vol. 27 No.2, Nov. 1994, Special Issue of the First World Wide Web Conference, pp.255-265.

Ibrahim and Franklin (1995)
B. Ibrahim, S.D. Franklin. `` *Advanced Educational Uses of the World Wide Web*". Proc. of WWW95, 3rd International Conference on World Wide Web.

Nielsen (1990)
J. Nielsen. `` *Hypertext and Hypermedia*". Academic Press Ltd, 1990.

RDB ``RDB: Relational Database System". `http://www.rdb.com`

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Building a User Model for a Museum Exploration and Information-Providing Adaptive System

Marcello Sarini and Carlo Strapparava
Istituto per la Ricerca Scientifica e Tecnologica,
I-38050 Povo/Trento, Italy
e-mail: {sarini | strappa}@irst.itc.it

Abstract: Hyperaudio is a system able to organize the presentation of a museum contents taking into account the visitor's needs and the layout of the physical space. The system is able to integrate a physical space with a virtual space in order to build a more general notion of *augmented space*: not only can the system provide the visitor with information tailored on his own interests and interaction history, but it can also support the visitor in his own exploration of the physical space, helping him to find what he is looking for and suggesting new interesting physical locations.
In this paper we describe the developing of the User Model components for this type of system: we found helpful to separate distinct functionalities about user modelling. These components are useful in planning the content presentations to the visitors. A presentation could take into account what the user already knows (to avoid boring repetitions, to make new things easier to understand by making comparisons, etc...), what the user is interested to (to propose new concepts and/or location to go etc...) and the interaction way the user seems to prefer.

Keywords: Adaptive System, User Modelling, Augmented Space.

1. Introduction

The ideal guided visit to a ``cultural'' space (such as an exhibition, a museum, an open-air exposition, an archaeological site and so on) allows the visitors to organize the tour through the different areas according to their own interests or preferred criteria. On the other hand, it may happen that the default physical organization of the exhibition (chosen by an architect or by a museum curator according to a ``default'' perspective of information presentation) does not meet directly the visitor's expectations, possibly making it difficult to build a personal route. Virtual museums (implemented for example as hypertextual resources browsable from the World Wide Web) may offer a more flexible (dynamically computed) object display determined by the visitor's individual preferences. The visitors can play around a clickable representation of the museum rooms and objects, getting informative cards on what they are most interested in. However, with purely virtual spaces the visitor may perceive objects differently (e.g., different dimensions and colours) and miss the emotional involvement you get in experiencing with real objects (e.g., being in front of Mona Lisa at the Louvre is quite a different experience from looking at its reproduction on the Web).
Hyperaudio [*] is a research project for developing a museum exploration and information providing adaptive system, able to integrate a physical space with a virtual space in order to build a more general notion of *augmented space*: not only can the system provide the visitor with information tailored on his own interests and interaction history, but it can also support the visitor in his own exploration of the physical space, helping him to find what he is looking for and suggesting new interesting physical locations (see [Not *et al.* 1997]).
Each visitor is equipped with a palmtop computer endowed with headphones, on which an infrared receiver is mounted. Each meaningful physical location has a small (power-autonomous) infrared emitter, sending a code that uniquely identifies it (see Figure 1).
Exploiting the infrared signals, the system is able to identify when the visitor reaches a certain physical location and can activate a relevant portion of the information repository loaded on the palmtop. Meaningful information are selected and organized to be played as audio messages or displayed as follow-up links on the palmtop screen. Adaptive and dynamic hypertext technology (see for example [Knott *et al.* 1996]) can be exploited to tailor a presentation according to the visitor interests, the actual context of the visit and so on.
Fundamental in such a system are the user modelling components that take into account the user's knowledge, interests, preferences etc...In this paper we describe the peculiarities of User Model components for an adaptive system in an augmented space.

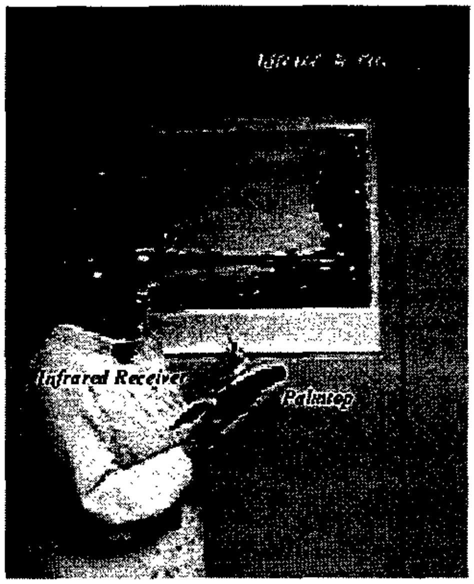


Figure 1: A visitor exploring an augmented room in a museum.

2. Different Functionalities in User Modeling

In an adaptive system, the user model represents the system's assimilation of the interaction and contains information about the user and the current context that can increase the system's ability to exhibit *pragmatically correct* behavior and, more generally, to engage in effective communication [Stock *et al.* 1995].
Ideally, presentations provided by a museum exploration system should contain the information that will be most helpful to the user. But since not all users are alike, achieving such behavior requires that the system has a model of the particular user with whom it is currently interacting. This model could include information about the user's knowledge, beliefs, abilities, attitudes, preferences and possibly goals and plans for achieving these goals.
Taking into account our experience in building multimodal natural language dialogue systems [Stock and Team1993], [Stock *et al.* 1995], it could be useful to separate two distinct functionalities about user modelling (what the user has been exposed to, linguistically or through images, or is assumed to know, and what the user seems to be interested in). Moreover, a third functionality could take into account user preferences with respect to presentation and interaction style. This is particularly useful in a system able to generate many alternative multimodal presentations.
This kind of information is represented in three modules (see Figure 2) and in a user profile.
The user's knowledge model, or *UK*, is based on an initialization (see section 4) and on a modelling of what the user has become aware of so far. User knowledge could be implemented as a set of *facts*, without involving advanced inference capabilities (at least for the first prototypes).
The user's interest model, or *UI*, provides a model of the potential interest of the user and could consist on an activation/inhibition weighted network whose nodes (the *interest areas*) are associated with ordered sets of entities. Each time a certain entity is ``taken into consideration" by the user (by clicking a link while navigating in the hypertext) or is presented by the system, the areas to which that entity is associated receive an activation impulse. Impulses are propagated through the network, decreasing in intensity according to the weights of the links traversed. The interest model evolves and becomes more and more focussed during the interaction between the user and the system. The status of the interest model provides a criterion for the computation of the relevance of the exhibits with respect to the current interaction context.
The user's preferences about presentations and interaction style could be also implemented as an activation/inhibition weighted network like the user's interest model in the case it is important to have an evolving model of user's preferences. Otherwise it could be modelled as a static characteristic of the user.
These components are useful in planning the content presentations to the visitors. A presentation could take into account what the user already knows (to avoid boring repetitions, to make new things easier to understand by making comparisons [Milosavljevic1997], etc...), what the user is interested to (to propose new concepts and/or location to go etc...) and the interaction way the user seems to prefer.
A vital statistics user profile (age, education level etc...) is useful, for example, to plan more use of technical language (in the case of an expert) in the content of the presentations.

3. User Model for Museum Explorations

In the literature, there have been many empirical studies on how visitors typically behave in museums. In many cases, the substantial amount of collected results has allowed curators to identify visitors' expectations and preferences, therefore allowing the improvement of museum exhibitions (in terms of quality of layout, lighting, labels, information services, etc...).
In particular, psychological studies suggest some general statements (for a complete list see `http://ecate.itc.it:1024/cgi-bin/zanca-cgi/print-bo-s.cgi`) that could be useful to characterize a system for museum explorations. Some statements help identifying parameters and features that influence how the User Model evolve and could be organized.

Examples of these statements are:

- Visitors to any exhibition rarely visit all exhibit components. The message from a museum exhibition is seldom exploited *in toto* (in a museum there is not a single message) [Korn1995].
  This implies modeling how visitors choose objects in the exhibition.

- Visitors to museums, despite their heterogeneity, may behave in reasonably predictable patterns [Falk1985].
  This could justify the introduction of a concept of *community* for grouping similar single user models.

Starting from the statements provided by psychologists and from the findings emerging from questionnaires, workshops and observations about users, a set of parameters and dynamic features can be identified as key elements for modelling users' behaviour.

Two major classes of parameters/features can be distinguished:

- parameters related to the exhibits, modelling aspects of the exhibition that most concur to the visitors' satisfaction and learning.
- parameters/features related to the visitor, modelling how visitors typically experience the museum.

These parameters tend to be more effective in an augmented space than in a purely virtual space. It could also be possible to take into account some ``physical" aspects like evolving time, tiredness of the visitors, and in general it is more sensible to give value to the emotional involvement in experiencing with real objects.

3.1 Parameters related to the Exhibits

Many museum studies highlight the fact that there are many ways in which layout, lighting, object dimensions, physical structure of the rooms, ... may affect the visitor's desire to approach and study an exhibit (see [Boisvert and Slez1995] for an overview on this topic). When building a portable information system for a museum (from simple tape-based audio guides, up to more sophisticated adaptive electronic guides), developers have to keep in mind that information has to be provided in the context of the exhibition ([Serrell1996]): this means that all the messages to be conveyed to the user need to take into account how the objects are placed, how each of them contributes to the overall topic of the exhibition, how they attract and keep visitors' attention. This because all the aspects of the museum experience (perceptual and cognitive) should contribute synergically to an increased enjoyment, understanding and learning on the part of the visitor. Attracting and holding power should be considered in modelling the parameters related to the objects in an exhibition ([Boisvert et al., 1995]):

- *Object attracting power.* Probability that the visitor stops and observes the exhibit.
  In museum studies, attraction is usually computed as the percentage of visitors who stop and observe an exhibit for at least a given amount of time [Boisvert and Slez1995]. Attraction is strictly dependent on the physical characteristics of the object (e.g., dimension, color, originality of the topic) and ``scenic" strategies (e.g., location of the object, lighting, type of labels). A portable guide could help increase the attraction of an object by stressing to the visitor the importance to go and see it.

- *Object holding power.* Average time spent by visitors observing the exhibit.
  It has been demonstrated that this index increases appreciably when the exhibit displays concrete concepts and allows the visitor to interact with it. A portable guide could help increase the holding power of an object

by improving the interaction and the explanation of the concepts exemplified by the object [Falk1982].

The model of the exhibition is therefore a very important piece of knowledge when setting up an electronic guide. More importantly, an adaptive information system can also exploit the model of the exhibition to interpret visitors' movements to dynamically update the user model. For example, if the visitor does not stop in front of an object with a high attracting power, the system could assume that he is not interested in the particular item and could update the user model accordingly.

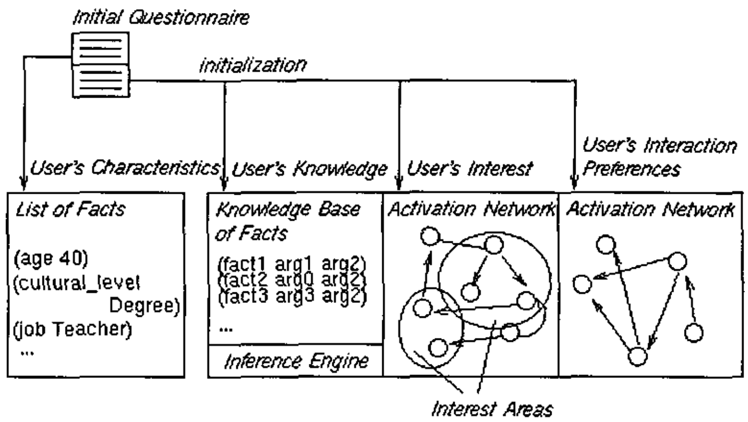


Figure 2: User Model Components

3.2 Parameters/Features related to the Visitor

Similarly, we try to find a significant set of parameters/features more related to the visitors.
Obviously, more features might emerge from the literature (either in the field of psychology and of user modelling) or from workshops with users and the analysis of questionnaires.

- *Attention*. From a psychological point of view, the *attention* is a selective functionality related with the subjects activation level, which depends both on ``internal" status (such as tiredness, fatigue, etc...) and on any stimulating information or event from the exhibits [Canestrari1994].
  From a computational point of view we distinguish between two kinds of attention: physical and cognitive attention. Psychological studies show that purely physiological attention decreases during a visit; while a measure of the cognitive attention can be given by a structural matching of the personal knowledge of the visitor and the ``relevant facts" related to exhibition objects (these are, for example, what the exhibition curator would like to communicate to the visitors).
  So a measure of the visitors attention in front of a particular object is given by a function of their physical attention and their cognitive attention.

- *Visitor's Engagement*. Another parameter that models the visitors is their engagement. It is defined as how much the visitor gets involved with an object. To have a measure of such a parameter, a `direct' observation of their experiencing should be necessary: their feelings, their desires to communicate to other visitors their impressions etc...[Boisvert and Slez1995].
  In our context it is not always possible to ``see" the visitors and quantify their involvement.
  However, an observable behaviour could be how the visitor ``navigates" in the informational space proposed by the system.

The attention is figured as soon as the visitor comes in front of the exhibit, while engagement is evaluated during and/or after the visitor is enjoying the object.

4. An example of evolving UM

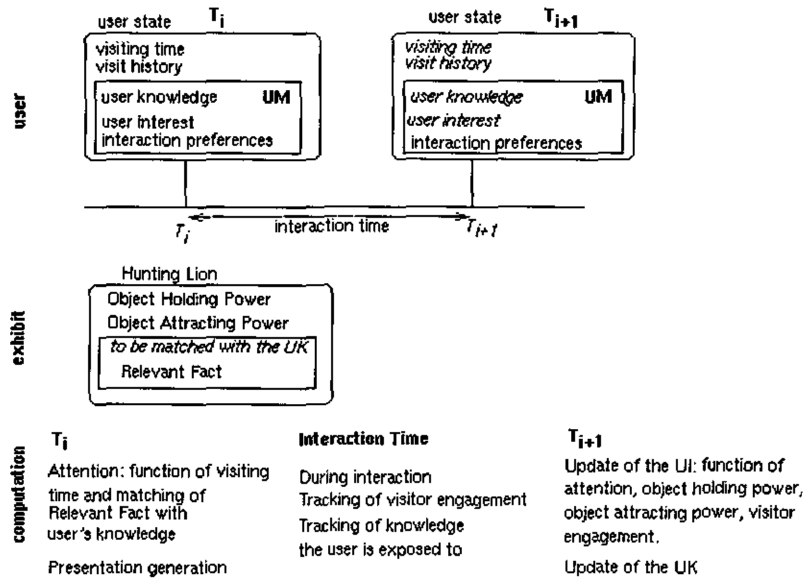


Figure 3: Example of UM evolution

By means of a questionnaire the user model is initialized in its profile part (e.g. visitor's age, handicaps, etc...), which refers to user's characteristics. The questionnaire could also contain a set of questions (e.g. about competence in some sectors, particular interest in some topics etc...) that gives initialization to the three user model components (e.g. ``default" facts in the user's knowledge, initial impulses to the interest model, initial interaction preferences). During the visit and taking note of the user behaviour, the system will refine these initializations that could be not accurate for some aspects.

As an example of how the components of User Model evolve during a visit, consider Figure 3.

After the visitor has spent some time (now ***Ti***) in the exhibition (eventually increasing his knowledge about displayed objects and showing interest in some of them) the visitor comes to the ``Hunting Lion" showcase. The exhibit has a particular *holding power* and *attractive power* and has associated a ``relevant fact". This fact is, for example, what the exhibition curator would like to communicate to the visitors.

The system proposes to the visitor a user's tailored presentation that contains a sequence of audio files which will be played on the headphones, a set of anchors for further elaborations which will be depicted on the display, and possibly a properly oriented map with the visitor's current position.

During the interaction time, the visitor is focussed on the exhibit enjoying the audio files and the navigation in the information space. The system detects the *visitor engagement* considering how much the user ``navigates" to get more explanations about the object. For the moment this is the only possible way to check the engagement. More sophisticated verification could be realized providing, for example, an eye-movement tracking or other behaviour recognition systems.

Another capability the system has to provide, during the interaction time, is the tracking of the knowledge acquired by the visitor hearing the audio files. This knowledge is *a priori* associated to the information the user is exposed to.

At the end of the interaction (at ***Ti+1*** time) the user interest model is updated by the system considering through an ``impulse" function of attention, variance on object holding power and object attracting power, and visitor engagement.

The update of user knowledge is made by adding the knowledge recorded during the interaction time. (this is obviously a simplification: this doesn't guarantee that the user really understood the ``new" knowledge)

5. Conclusion

We have described the developing of User Model for a museum exploration and information-providing adaptive system.

We found helpful to have separate modules that model different aspects about user modelling: what the user has been exposed to, linguistically or through images, or is assumed to know, what the user seems to be interested in and user preferences with respect to presentation and interaction style.
These modules are useful in planning the content presentations to the visitors to avoid boring repetition, to propose new concepts and/or location to go and considering the interaction way the user seems to prefer.
As future development, we are investigating the possibility to introduce the concept of visitor communities, built from dynamically clustering single visitor's user models. A community groups visitors with similar characteristics during their museum explorations. Information about communities could be used for planning group tailored educative strategies, for sending messages to a community about museum events etc...

References

[Boisvert and Slez1995] Dorothy Lozowski Boisvert and Brenda Jochums Slez. The relationship between exhibit characteristics and learning-associated behaviors in a science museum discovery space. *Science Education*, 79(5), 1995.

[Canestrari1994] R. Canestrari. *Psicologia Generale e dello Sviluppo*, volume 1. CLUEB, 1994.

[Falk1982] John H. Falk. Time and behaviour as predictors of learning. *Science Education*, 67, 1982.

[Falk1985] John H. Falk. Predicting visitors behaviour. *Curator*, 28(4), 1985.

[HIP] WWW home page for the HIPS project:
http://www.ing.unisi.it/lab_tel/hips/hips.html.

[Knott *et al.* 1996] Alistair Knott, Chris Mellish, Jon Oberlander, and Mick O'Donnell. Sources of flexibility in dynamic hypertext generation. In *Procof the 8th International Workshop on Natural Language Generation*, Herstmonceux Castle, UK, 13-15 June 1996.

[Korn1995] Randy Korn. An analysis of differences between visitors at natural history museums and science centers. *Curator*, 38(3), 1995.

[Milosavljevic1997] Maria Milosavljevic. Content selection in comparison generation. In *Proc. of the 6th European Workshop on Natural Language Generation*, Duisburg, Germany, 1997.

[Not *et al.* 1997] E. Not, D. Petrelli, O. Stock, C. Strapparava, and M. Zancanaro. Person-oriented guided visits in a physical museum. In *Proc. of the 4th International Conference on Hypermedia and Interactivity in Museums*, Louvre, Paris, September 1997.

[Serrell1996] Beverly Serrell. *Exhibit Labels. An Interpretative Approach.* AltaMira Press, 1996.

[Stock and Team1993] O. Stock and The ALFRESCO Project Team. ALFRESCO: Enjoying the combination of NLP and hypermedia for information exploration. In M.T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAIPress/MIT Press, Menlo Park CA/Cambridge MA, 1993.

[Stock *et al.* 1995] O. Stock, C. Strapparava, and M. Zancanaro. Explorations in a natural language multimodal information access environment. In *Workshop on Intelligent Multimedia Information Retrieval*, Montreal, 1995. IJCAI 95.

...Hyperaudio

Hyperaudio is a project developed at IRST in collaboration with Civic Museum of Natural Science in Rovereto (Italy). Hyperaudio will contribute to more advanced research and scenarios inside HIPS European Project [HIP] whose consortium includes: University of Siena(coordinating partner), CB&J (France), GMD (Germany), IRST (Italy), SIETTE-Alcatel (Italy), SINTEF (Norway), University of Dublin and University of Edinburgh.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Personalizing Access to Web Sites: The SiteIF Project

Anna Stefani [*] and Carlo Strapparava
Istituto per la Ricerca Scientifica e Tecnologica,
I-38050 Povo/Trento, Italy
e-mail: {stefani | strappa}@irst.itc.it

Abstract: The growing size and complexity of WWW made evident the need to provide more flexible mechanisms for delivering personalized information to the user. On the other side, knowledge of customers interests could be a real advantage for companies that work using Internet and want to develop personalized marketing applications. This paper gives an overview of the SiteIF system. SiteIF takes into account the user's browsing behavior and tries to anticipate what documents in the web site could be interesting for the user. The system dynamically learns the user's areas of interest generating/updating a user model. The architecture of the system consists of many components. The paper focuses on the agents that model the user interest and generate personal documents as entry points in the site.

Keywords: Internet, Information Filtering, User Modeling, Intelligent Agents, Personalized Marketing.

1. Introduction

In the last years we have seen a continued growing of the information available on Internet. The expansion of this net, of the local nets and of the documents and resources contained by them, has been stressed sometimes with enthusiastic words, sometimes in alarmist ways, by many authors that tried to quantify its dimensions. In the analysis described in [Bray, 1996], written in the November 1995, more than eleven millions of documents have been found. In [Etzioni & Weld, 1995] the authors wrote that in the May 1995, every day more than 30 millions of people used Internet, and every person is a potential producer of information.

More information becomes available, more difficulties have the users to control and effectively manage the potentially endless flow of information: it is not easy to find what you are looking for unless you know exactly where to get it from and how to do it.

Information filtering systems can help users eliminate useless documents and bring to their attention only the relevant information. This implies that the system has to be able to recognize the users and to maintain a model for their interests.

Several tools have been proposed in literature to search and retrieve relevant documents ([Lieberman, 1995]; [Armstrong et al., 1995]; [Kamba & Sakagami, 1997]; [Minio & Tasso, 1996]). Anyway all these systems share two basic limitations: the technique used to represent a user's profile is based on simple lists of keywords (and single words are often not enough to describe someone's interests) and the learning method requires the users' conscious and active involvement filling a form of keywords (topics) for their interests or adding a score to each visited document.

This paper describes SiteIF system. SiteIF acts as a housekeeper for a web site. It works with an ordinary web browser that supports Java, tracks down the user's browsing behavior (e.g. following links) and tries to anticipate what documents in the site could be interesting for the user.

From a marketing point of view, knowledge of customers' profiles is a resource to reach a good one-to-one relation, allowing the development of personalized applications following the needs of every user.

Section 2 gives an overview of SiteIF functionality and structure. Examples of the user interaction with SiteIF system are presented in section 3. Section 4 presents some directions for a further development.

2. The SiteIF System

SiteIF is a personal agent that follows the users from page to page as they browse the web site, "watching over the user's shoulder". It learns user's interests from the requested pages that are analysed to generate or update a model of the user.

This model is represented using a semantic net developed similarly to IFTool system [Minio & Tasso, 1996]. However, unlike from IFTool, SiteIF avoids involving the user in its learning process (it does not ask the user for any keywords or opinions about pages) and only takes into account the addresses of the visited pages.

In this way it is possible to give advices about pages and documents of the web site that SiteIF supposes could be interesting for the user. The whole system is implemented in approximately 3100 lines of Java code and 290 lines of C code.

2.1 The Architecture

Figure 1 shows the SiteIF architecture which includes the following modules:

- **SITEIF INTERFACE AGENT**: it controls the graphic interface and manages the interaction operations with the user.
- **SITEIF AGENT**: it yields the function of writing and generating personal documents based on the user's interests.
- **WUP AGENT**: it implements the main functions of the system: it helps retrieve and select the documents useful for the user, inside the web site.
- **BROWSER**: it controls the interaction operations of the user about the normal navigation on the Internet and shows documents and results.

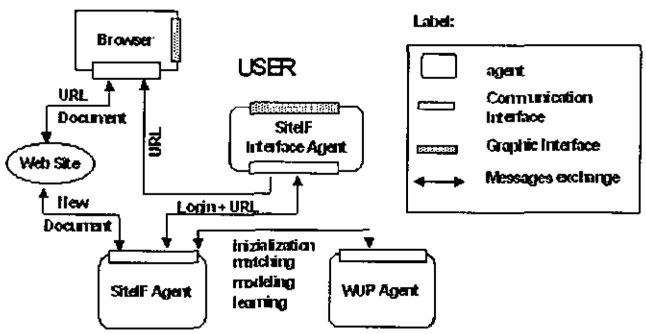


Figure 1 - Functional architecture of SiteIF (full-size)

User can interact through two graphic interfaces: the first is controlled by the SiteIF Interface Agent, the second is the browser itself. The SiteIF Agent is called by the SiteIF Interface Agent that sends a request of identification/authentication and, once verified, it allows the user to enter the web site. The SiteIF Interface Agent follows and monitors the actions of the user inside the site. Every time he/she follows a link, the selected URL is sent to the SiteIF Agent while the Netscape window displays the requested document. The SiteIF Agent records all the browsed documents in a log file.

The log file is sent to the WUP Agent that initializes or updates the user model. After the modelling phase, the WUP agent filters the documents of the site according to the user model built before and sends back the results to the SiteIF Agent.

Now we briefly describe features and functionalities of the implemented agents. Every agent has a quite complex architecture that can be divided in other sub-agents or modules.

2.2 The SiteIF Agent

In Figure 2 it is shown the functional architecture of SiteIF Agent. This agent manages different functions: it logs the pages visited by the user (in a log file that will be sent to the WUP Agent) and creates "on the fly" personal documents based on user's interests. For the moment these personal documents are simple lists of results. A natural language generation module, that creates personalized HTML pages, is under development.

SiteIF Agent is made up of the following elements:

- *ACCESS CONTROL MODULE* : it authenticates the user (login, password, authorization of collecting personal information) and adds new users in the site users database.
- *DOCUMENT REPRESENTATION MODULE* : it analyzes the new site incoming documents and produces an internal representation, constituted by information about their contents. In particular, this is made through standard techniques (such as segmentation, stop list deletion, stemming and weighting) [Salton & McGill, 1983] and a specific algorithm which is devoted to identify the best terms to represent the content of a document (compression) [Asnicar & Tasso, 1997].
- *DATA MANAGER MODULE* : it manages and organizes system data structures.
- *DOCUMENT GENERATOR MODULE* : it produces HTML pages. The documents are personalized following information about user's interests.

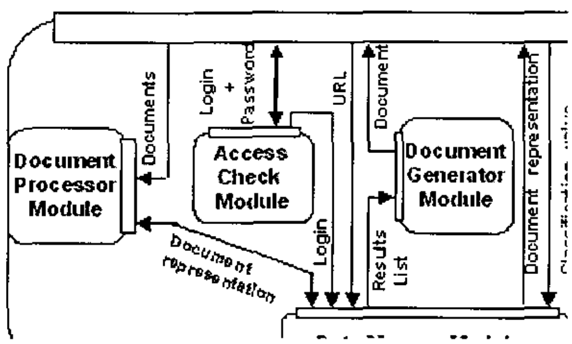


Figure 2 - Functional architecture of SiteIF Agent (full-size)

2.3 The WUP Agent

The WUP (Web User Profiling) Agent implements the following steps: the user modelling, the comparison of the internal representation of the document with the user model and, on the basis of the obtained results, the classification of the document (i.e. interesting or not interesting).

The WUP Agent yields the user model as a semantic net. Every node is a word (or an interesting concept) and the arcs between nodes are the co-occurrence relation of two words; every node and every arc has a weight (that represents a different level of interest for the user).

The weights are periodically reconsidered and possibly lowered (depending on the time passed from the last update). Also no more useful nodes and arcs may be removed from the net. So it is possible to consider changes of the user's interests and to avoid that uninteresting concepts remain in the user model.

During the filtering phase, the matching module receives as input the internal representation of a document and the current user model. It produces as output a classification of the document (i.e. it is worth or not the user's attention).

In literature ([Stevens, 1992], [Hoeffer et al., 1995], [Baclace, 1992]) the most used matching technique is the standard keyword matching: a simple count of the terms which are simultaneously present in the document representation and in the user model. This technique has some problems for the synonymy and plural meanings of some words [Foltz & Dumais, 1992]. A lot of words describe different concepts if used in different contents. For example the words "system", "expert" and "operative": the first and the second word can occur in a document about expert systems, while the first and the third can be found in operative system pages. So the "system" word can have more than one meanings, depending on the context in which it is used.

The idea behind SiteIF algorithm consists of checking, for every word in the representation of the document, whether the context in which it occurs has been already found in previously visited documents and already stored in the semantic net. This context is represented by the co-occurrence relationship, that is by the couples of terms included in the document which have already co-occurred before in other documents (information represented by arcs of the semantic net).

3. An Example of Interaction

In this section, we present a sample session, run with a user model partially filled in previous sessions. The user types his access codes in the fields LOGIN and PASSWORD and clicks on the CONNECT button (Figure 3).

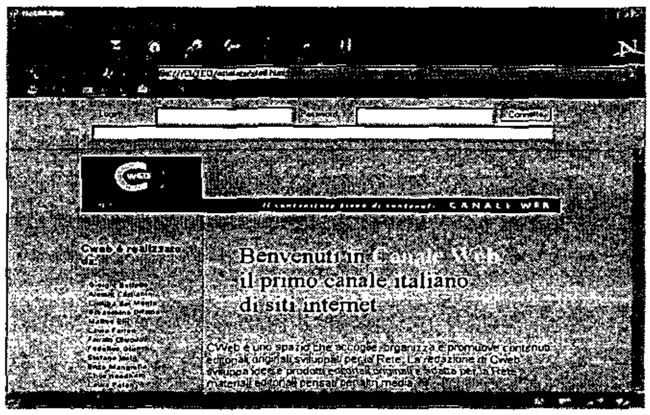


Figure 3 - WWW page to access the SiteIF System. (full-size)

Then, the SiteIF System Interface (Figure 4), a document generated on the fly by the SiteIF Interface Agent, is displayed to the user. It contains a range of all the possible documents that the system "thinks" could be useful or interesting for the user (the list box on the left, contains all the classes in which the site having at least an interesting document; the list box on the right shows the documents of the class selected in the left list, ordered by a decreasing way following the value of interest).

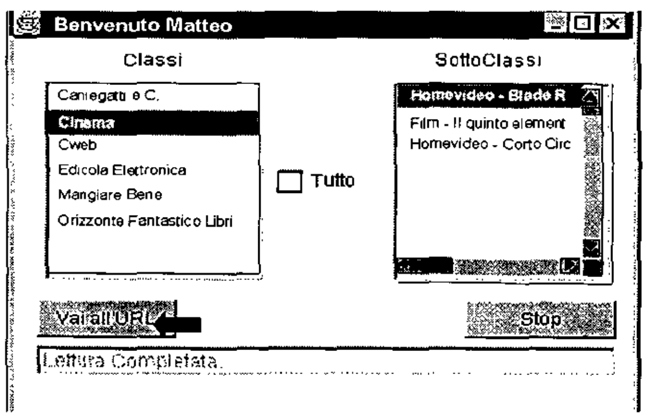


Figure 4 - User interface of SiteIF(full-size)

Every time a user clicks on a link of the list, the applet takes into account the URL and opens that URL it in the Netscape window (Figure 5).

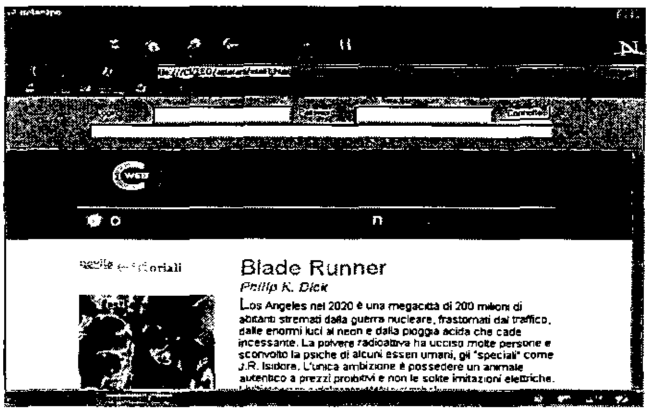


Figure 5 - Document related to the URL chosen by the user, shown in the Netscape window.(full-size)

If the user is not interested in the suggested documents, he can select the control box TUTTO and go on with the normal browsing of the site. Even in this way, the applet records and logs all the choices of the user and the URLs of the followed links.

4. Conclusions and Future Work

In this paper we have described the SiteIF system. SiteIF takes into account the user's browsing behavior and tries to anticipate what documents in the web site could be interesting for the user. The system interactively and incrementally learns about the user's areas of interest generating/updating a user's model.

This paper describes a work in progress. At this point we don't have final evaluations of the effectiveness of the system, although preliminary tests are encouraging. A complete evaluation test is planned in collaboration with Telecom Italia Network, one of the more important Italian service provider.

There are many possible future developments:

- privacy of the user models is a rising issue in this kind of systems. Appropriate precautions must be taken to ensure that the user will keep a complete control of the model.
- a system of authentication of users that log on the web site, based on one time password ways (i.e. smart card) that protect users of a system from not authorized interceptions. The use of a one time password system is necessary for a firm or an organization that needs a good level of security.
- a natural language generator module able to create personalized HTML pages.

Using systems such as SiteIF will be fundamental for companies that work using Internet and that want to develop personalized marketing applications following the needs of every customer. At the moment not all the companies have customers that use Internet, but it is only a matter of time. If the activity will grow at an international level, this is certainly the way to follow. Who starts today will be in a better position tomorrow, when Internet and WWW will become tools for everybody.

References

[Armstrong et al., 1995] Armstrong, R.; Freitag, D.; Joachims, T.; Mitchell, T. *WebWatcher: A Learning Apprentice for the World Wide Web,* AAAI 1995 Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, Stanford, March 1995.

[Asnicar & Tasso, 1997] Asnicar, F.; Tasso, C. *ifWeb: a Prototype of user models based intelligent agent for document filtering and navigation in the world wide web,* Sixth International Conference on User Modeling, Chia Laguna, Sardinia, Italy, 2-5 June 1997. Rif.: http://www.dimi.uniud.it/~ift/positionp.html

[Baclace, 1992] Baclace, M.C. *Competitive agents for information filtering,* Communications of the ACM, 35(12), pp. 50, 1992.

[Bray, 1996] Bray, T. *Measuring the Web* , Proceedings of the Fifth International World Wide Web Conference, Paris, France, May 1996, in Computer Networks and ISDN Systems 28(7-11), pp. 993-1005, 1996.

[Etzioni & Weld, 1995] Etzioni , O.; Weld, D. S. *Intelligent Agents on the Internet: Fact, Fiction, and Forecast,* in IEEE Expert, August 1995.

[Foltz & Dumais, 1992] Foltz, P.W.; Dumais, S.T. *Personalized information delivery: an analysis of information filtering methods,* in Communication of the ACM, 35(12), pp. 51-60, 1992.

[Hoeffer et al., 1995] Hoeffer, M.; Knaus, B.; Winiwarter, W. *An Evolutionary Approach to Cognitive Information Filtering,* SIGIR95, 1995.

[Kamba & Sakagami, 1997] Kamba, T.; Sakagami, H. *Learning Personal Preferences on online Newspaper articles from user behaviors,* Sixth International World Wide Web Conference Proceedings, 1997. Rif.: http://proceedings.www6conf.org

[Lieberman, 1995] Lieberman, H. *Letizia: An Agent that assists Web browsing.* Proceedings of the 1995 International Joint Conference on Artificial Intelligent, Montreal, Canada, August 1995. Rif.: http://agents.www.media.mit.edu/groups/agents/papers/

[Minio & Tasso, 1996] Minio, M.; Tasso, C. *User Modeling for Information Filtering on INTERNET Services: Exploiting an Extended Version of the UMT Shell,* Workshop on User Modeling for Information Filtering on the World Wide Web, in Proceedings of the Fifth International Conference on User Modeling, Kailia-Kuna Hawaii, January 1996. Rif.: http://www.cs.ju.oz.au/bob/um96-workshop.html

[Salton & McGill, 1983] Salton, G.; McGill, M.H. *Introduction to modern information retrieval,* McGraw-Hill, New York, 1983.

[Stevens, 1992] Stevens, C. *Automating the creation of information filters,* in Communications of the ACM, 35(12), p. 48, 1992.

[*]

Most part of the work was made when the first author was at University of Trento, Economics Department, during her degree thesis.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Web Engineering and Flexible Hypermedia

Michael Bieber, Roberto Galnares and Qiang Lu
Computer and Information Science Department
New Jersey Institute of Technology
University Heights, Newark, NJ 07102 USA
bieber@njit.edu, galnares@homer.njit.edu, qianglu@homer.njit.edu
http://megahertz.njit.edu/~bieber

Abstract: We take a two-stage approach to engineering applications for the World Wide Web. First the software engineer performs a *Relationship-Navigation Analysis*, analyzing an existing or new application specifically in terms of its intra- and inter-relationships. Second, a dynamic hypermedia engine (DHymE), automatically generates links for each of these relationships and metaknowledge items at run-time, as well as sophisticated navigation techniques not often found on the Web. Because the number of links generated can be overwhelming, we need to filter them based on user task and preference.

Keywords: hypermedia design, automatic link generation, relationship management, Web development

Background

We're currently working on an approach to Web Engineering, which We'll describe in some more detail below. In short, We're taking a 2-step approach towards moving computational applications to the Web: a relationship-navigation analysis to determine where the links should be, and a hypermedia engine that dynamically generates these links at run-time. By computational applications, we really mean any application that generates its content in real time, which includes most analytical applications. Because their documents do not exist beforehand, all hypertext must be generated dynamically after the documents are created.

Three more notes before turning to flexibility issues. First, we view hypermedia primarily as *supplemental* support for applications. Adding hypertext links, annotation and navigational tools enhances an application with intuitive ways to access the information in which the user is interested. Second, nothing about this approach is really Web-specific, though "Web" is the buzzword that seems to have finally attracted people's interest in my research. In fact, our first two prototypes were on the Macintosh and the IBM PC. The current engine prototype is being written in JAVA with a Web user interface module. Third, this approach works equally well for adding hypertext to new applications as to existing applications. Migrating an application to the World Wide Web provides an excellent opportunity to reengineer it for supplemental hypermedia support.

Flexibility

FLEXIBILITY ISSUE 1: My approach could easily generate an overwhelming number of links for an anchor. How can the system produce a reasonable number of links, directed at the user and his or her current task, and rank ordered by relevance?

FLEXIBILITY ISSUE 2: One type of generic relationship (link) type the Relationship-Navigation Analysis finds is a "schema" or "design document relationship." When a user selects an item of information on the screen, the engine will determine automatically where in the schema or other design documents that item is. Then from the schema/design, the engine will determine what interesting things are related to the selected item. How can the system figure out which related things are *interesting* and of these, which are *relevant*?

One approach is through machine learning; by monitoring the user, then over time the system should be able to "learn" about the user's tasks and preferences. Over time we will probably add such learning to the system. But we see this more as fine tuning, assuming the user is not so overwhelmed that he or she will stick with the system for the long term.

So for now We're leaning towards asking the user what his or her current task and preferences are. The mapping rules have a "condition" parameter which can be used to block or activate that rule. The first step of the Relationship-Navigation Analysis is to determine the possible stakeholders for the application. The relationships and metainformation found during the analysis can be marked for specific stakeholders and tasks, and thus coded in the mapping rules directly. We also are building a user preference module to serve all the engine's registered applications. Again, preferences could be coded in the mapping rules directly. All this is planned, but not designed or implemented yet. We hope to incorporate these later this year. How should we go about this?

Question and Workshop Goal:

We hope that our team will design and implement task and preference filtering this fall, as well as schema relationships. But we have not thought about it carefully yet, nor caught up on the literature about this. How should we do it? What should we take into account?

Relationship-Navigation Analysis

The Relationship-Navigation Analysis (RNA) technique has 5 steps:

(1) stakeholder analysis
(2) element analysis
(3) relationship and metaknowledge analysis
(4) navigation analysis
(5) relationship and metaknowledge implementation analysis

RNA has two major purposes. On its own, a relationship analysis will help the developer form a deeper comprehension of the application. This occurs primarily in steps 1-4. The developer then must decide which of these relationships actually to implement. Some may provide only marginal benefit. Others may be very costly or difficult to implement. These decisions take place in the last step.

While quite useful in its current form, we intend to develop the RNA technique further by producing specific guidelines for each step and by reducing the range of options that the developer must consider within steps 2 and 3 of the analysis. These refinements should make the analysis more systematic and easier to conduct, while allowing it to remain necessarily open-ended.

Step 1: Stakeholder Analysis

The purpose of the stakeholder analysis is to identify the application's audience. Knowing who will be interested in an application helps the developer broadly determine the entire range of important elements and relationships, and then to focus on these specifically. Especially those applications with public Web access have a much broader range of stakeholders than many imagine. Many developers, in fact, find this the most enlightening part of the RNA. The developer also should identify and understand the tasks each type of user will want to perform within the application. These will help the developer focus on specific areas during the RNA steps that follow.

Step 2: Element Analysis

Here the developer lists all the potential elements of interest in the application. At one level these include all types of items displayed in any on-line display (information screens, forms, documents, and any other type of display), as well as the screens, forms and documents themselves. The easiest way to start is to examine each screen (or mock-up) and identify each value and label it contains. Note that developer should identify kinds or classes of elements, not individual instances. The relationship types we discuss in step 3 all are for specific kinds of elements. In the decision analysis domain, for example, these include "model" and "data value" as opposed to specific models or data values.

Step 3: Relationship Analysis

Relationship analysis concerns inter-relationships, intra- relationships and metaknowledge. The developer should consider each element of interest identified in the prior step in terms of each of the following general kinds of relationships, for each group of stakeholders. Certain relationships will be useful to only certain stakeholders, and the hypermedia engine will filter these. Relationships can lead to information inside and outside the application. Developers should not feel constrained by real-world considerations of availability or implementation cost and effort. In this step they should exercise their creativity as fully as possible. Only in step 5 should they consider how to implement the relationships and metaknowledge found.

RNA currently helps developers identify the following types of relationships and metaknowledge within an application: schema, process, operation, structural, descriptive, parametric, statistical, collaborative and ordering relationships. [Bi98] gives more details for each. Bieber and Vitali [BV97] shows how several of these general relationship types can supplement an on-line sales invoice. [Bi98] shows how they can supplement a mathematical modeling decision support system.

Step 4: Navigation Analysis

Once we identify the relationships, we can think of how the user might access them. The most straightforward implementation would make each relationship a link, and then provide simple traversal (users selecting an anchor and link, and the system displaying the link destination). But certain relationship types lend themselves to more sophisticated navigation. The concept of hypermedia includes many other navigation features based on relationships or links. These include guided tours and trails, overviews and structural query [BVA97]. In this step of RNA, the developer should decide which navigation features might best serve stakeholders' needs.

Step 5: Relationship and Navigation Implementation Analysis

Clearly step 3 can generate a lot of relationships and step 4 can generate a lot of possible navigational opportunities. In this step, the developer must decide which actually to implement. This step is not the actual implementation, simply the logical decision of which relationships to implement. Designers should consider the costs and benefits (actual and marginal) of both implementing and displaying each. We separate this step from steps 3 and 4 so the designer can exercise all of his or her creative talents there without constraint by real world considerations. The designer then writes a mapping rule (using our specified format) for each of the relationships to be implemented. Mapping rules specify the commands or algorithms for finding the endpoint of each relationship.

DHymE (Dynamic Hypermedia Engine)

The DHymE hypermedia engine executes separately from the target application. We write a wrapper program for each application to integrate it into our engine architecture. Applications or their wrappers then connect to DHymE through a Web proxy server. DHymE intercepts all messages passing between the application and the user interface, and uses the mapping rules specified above to map each appropriate element of the message to a hypermedia *node* or *anchor*. Our Web browser wrapper integrates these anchors into the document being displayed and passes it through the proxy server to the user's Web browser. When the user selects an anchor, the browser wrapper passes it to DHymE, which returns a list of possible links (one for each appropriate relationship as determined by the mapping rules). If the user selects a normal application command (mapped to an operation link), DHymE passes the command on to the application for processing. If the user selects a hypermedia engine link (e.g., to create an annotation), DHymE processes it entirely. If the user selects a supplemental schema, process, operation, structural, descriptive, information or occurrence relationship, DHymE infers the appropriate application commands, meta-application operations (e.g., at the operating systems level or schema level) or hypermedia engine operations that will produce the desired information. If the user selects a user-created annotation, DHymE retrieves it. Thus DHymE automatically provides all hypermedia linking (as well as navigation) to applications, which remain hypermedia-unaware and in fact often entirely unchanged. We currently are integrating several applications with DHymE, automatically giving each a Web interface or supplementing its existing Web interface: a personnel requisition tracking system, a relational database management system, a mathematical model management system, a transportation spreadsheet analysis system and a multi-criteria decision support analysis tool. [Bi98] describes these ideas and an older, non-Web prototype of DHymE in more detail.

Conclusion

We hope that our most enduring contribution is successfully convincing developers of Web applications (both new and transported from other computer environments) to take full advantage of linking in their applications. Time and time again designers have told us that RNA has shown them links they never imagined in their own applications. Identifying these is the necessary first step towards implementation. Implemented thoughtfully, the Web's linking and navigation facilities can go a long way to reducing the complexity of applications for users. RNA gives developers a tool for identifying opportunities for supplemental linking within applications. The DHymE hypermedia engine automatically generates these links, with little or no change to the application.

Acknowledgments

We gratefully acknowledge funding for this research by the NASA JOVE faculty fellowship program, by the New Jersey Center for Multimedia Research, by the National Center for Transportation and Industrial Productivity at the New Jersey Institute of Technology (NJIT), by the New Jersey Department of Transportation, by the New Jersey Commission of Science and Technology, as well as grants from the Sloane Foundation and the AT&T Foundation, and the NJIT SBR program.

References

[Bi97] BIEBER, M., Hypertext Engine: Support for Computational Applications, Technical Note, 1997.

[Bi98] BIEBER, M., Supplementing Applications with Hypermedia, under submission, 1998.

[BK95] BIEBER, M. AND KACMAR, C. Designing hypertext support for computational applications. Communications of the ACM 38(8), 1995, 99-107.

[BV97] BIEBER, M. and VITALI, F. (1997). Toward support for hypermedia on the World Wide Web. IEEE Computer 30(1), 1997, 62-70.

[BVA97] BIEBER, M., VITALI, F., ASHMAN, H., BALASUBRAMANIAN V., and OINAS- KUKKONEN, H. Fourth generation hypermedia: some missing links for the World Wide Web. International Journal of Human Computer Studies 47, 1997, 31-65.

[CB97] CHIU, C. AND BIEBER, M., A generic dynamic-mapping wrapper for open hypertext system support of analytical applications, Hypertext'97 Proceedings, ACM Press, New York, NY, April 1997, 218-219.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Automated Generation of Hypermedia Presentations from Pre-existing, Tagged Media Objects

Lynda Hardman, Lloyd Rutledge, and Dick Bulterman
INS, CWI, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands
http://www.cwi.nl/~lynda/
Lynda.Hardman@cwi.nl
tel +31-20-592 4127
fax +31-20-592 4199

A standard reference model for the automated creation of multimedia presentations has been developed [BMFR98]. This is a high level framework for describing the division of processes and the communication among processes required for the automated creation of multimedia presentations.

The work is based on several operational projects for creating presentations with multiple media. The underlying document model assumed in the reference model lacked, however, a true time dimension and hypertext links. Also, it assumes sufficient knowledge of the domain and sufficient processing power to enable the constituent media items in the final presentation to be generated from scratch. Our approach is different in two respects. First, in the more complex underlying document model and, second, that the media items to be used in the presentation already exist. In addition, we are also investigating the utilisation of existing tools and standards for implementing some of the generation processes, as described in the SRM.

Our work on integrating a more complex document model is based on the Amsterdam Hypermedia Model (AHM), and has resulted in [HaWB98]. This describes how the different processes in [BFMR98] can be seen as ``filling in" different parts of the AHM document model. These include the media items to be used, their temporal and spatial layout, style information and composition information. Relevant to this is the process of deciding on navigation structures among the various scenes of the presentation, in the line of HDM [GaPS93]. Work is currently being carried out in this area by a masters student.

We have also carried out our own system design for the generation of hypermedia presentations, based on the availability of tagged media items stored in a database [WBHT97]. This work is being continued with the implementation of a specialist tagging tool built on top of an existing image database. Video will also be incorporated. Two undergraduate students and a graduate student are working on the database and video aspects of this project.

We have recently started work on using style sheets, including structural transformations, for implementing the processes specified in the SRM. In particular as a vehicule for communication among the content, design, realization and presentation display layers. This has resulted in the submission of [RHOB98].

In broad terms, we see the process of generating a hypermedia presentation as follows.

A database contains a collection of media items. Each of these is tagged with one or more terms from a domain-specific thesaurus. The name of the source of the tagging is also included, since a single media item may be tagged with terms from multiple thesaurus.

As the end-user explores the information presented on the screen they are at some point in ``conceptual space". This can be described by one or more terms from the domain-specific thesaurus. When the user requests further information, the system uses known information about the user (such as what has already been seen) and searches the accessible databases for relevant information. The search is based on the terms associated with the media items in the database(s).

Once a number of items have been found, they are grouped together into scenes which can be presented on the screen. Not all items can be presented at once, and choices have to be made on the temporal, spatial and linking relationships used to present the information in the best (for the end-user) possible way. Heuristics for grouping the information, and for temporal and spatial alignment can be expressed using a style sheet language such as DSSSL [ISO96].

Our work is in progress and we are approaching the problem from a number of different directions.

References

[BFMR98] M. Bordegoni, G. Faconti, M.T. Maybury, T. Rist, S. Ruggieri, P. Trahanias and M. Wilson (1998). A Standard Reference Model for Intelligent Multimedia Presentation Systems. Computer Standards and Interfaces.

[GaPS93] F. Garzotto, P. Paolini and D. Schwabe (1993). HDM -- A Model-Based Approach to Hypertext Application Design. TOIS 11(1) 1-26.

[HaWB98] L. Hardman, M. Worring and D.C.A. Bulterman (1998). Integrating the Amsterdam Hypermedia Model with the Standard Reference Model for Intelligent Presentation Systems. Computer Standards and Interfaces. http://www.cwi.nl/ftp/mmpapers/AHM-SRM.ps.gz

[ISO96] International Standards Organization. Document Style Semantics and Specification Language (DSSSL). ISO/IEC IS 10179:1996, 1996.

[RHOB98] L. Rutledge, L. Hardman, J. van Ossenbruggen and D.C.A. Bulterman (1998). Implementing Adaptability in the Standard Reference Model. Submitted to Multimedia Modelling 1998.

[WBHT97] M. Worring, C. van den Berg, L. Hardman and A. Tam (1997). System Design for Structured Hypermedia Generation. Lecture Notes in Computer Science 1306, Visual Information Systems, ed. C. Leung. http://www.cwi.nl/ftp/mmpapers/LNCS-visual.ps.gz

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Adaptable and Adaptive Information Access for All Users, Including Disabled and Elderly People

Alfred Kobsa
GMD FIT, 53754 St. Augustin, Germany
Alfred.Kobsa@gmd.de
http://zeus.gmd.de/hci/alfred.kobsa.html

Constantine Stephanidis
ICS-FORTH, 71110 Heraklion, Crete, Greece
cs@csi.forth.gr
http://www.ics.forth.gr/proj/at-hci/cs.html

Abstract: One of the main objectives of the AVANTI project is to extend the scope of current research on adaptability and adaptivity, in that the needs of disabled and elderly users are also taken into account. Existing user modeling tools can be fruitfully employed in this endeavor.

Introduction

Research on adaptable and adaptive interactive software systems aims at rendering these systems tailorable to the needs of different user groups. Those end-user groups who have been considered so far were nearly always comprised of users with "average" physical and cognitive abilities. People with special needs (including disabled people, and to some extent also elderly people) should, however, also be given the opportunity to access computers, since they are indispensable at many workplaces and increasingly are becoming a medium through which important services at home and in public places can be accessed. For many categories of disabled people, computers may even allow for (partial) compensation of their handicaps.

Computer access for the handicapped has been a research issue for many years. Considerable effort has been put into making software systems accessible by user categories other than the ones they were -originally- designed for (e.g. for visually or motor impaired users) and into developing databases with information for disabled people that supplements already available data (e.g. information on wheelchair accessibility of public transportation, or verbal descriptions of paintings in major museums). These solutions mostly address a small number of disabled users and are, therefore, usually fairly expensive due to the restricted customer base. It seems, however, that techniques from the area of adaptable and adaptive interactive systems can be extended in such a way that they permit tailoring generic interactive software systems to all users, including the disabled and the elderly. This approach is not only theoretically more satisfactory, but may also be economically more viable than isolated dedicated solutions.

Application domain and user needs

The AVANTI prototype [1, 4, 6, 8] is a distributed system that provides hypermedia information on places of interest, public transportation and public buildings in a city. The system is accessible from users' homes and from public information booths. The intended users are tourists and citizens with different aims, interests, experience and abilities (including restricted sensory and motoric abilities).

These different users have varying needs with respect to information content as well as information presentation. For example, information for lay-persons should not be very technical and detailed but instead be augmented by explanations and by visual material [2]. Information for motor-impaired users should be supplemented, e.g., by data on wheelchair accessibility. Information for blind users should be complemented by data on services available for blind users. Moreover, such material cannot be presented visually, but only acoustically or via special output devices for the blind.

Technical approach

Information in AVANTI is presented in hypermedia form on the World Wide Web. The description of each hypermedia page contains information options (like optional information on wheelchair accessibility) as well as information alternatives (like technical vs. non-technical descriptions, or a picture of a painting vs. its verbal description). Additionally, the objects with which the user interface is constructed can be presented differently to the user (e.g., a pull-down menu can be presented visually or acoustically [5, 6]).

The selection of options and the preference of alternatives over the default is controlled by the user. If permitted by him or her, the system can, in most cases, choose options and pre-select alternatives automatically (either can be overridden by the user). The selections are made based on a central user model. Information about users is gathered with initial interviews and by monitoring users' interaction with the system. Pending results on user acceptability, smart cards will also be considered as an input device. Secondary information is derived with inference rules and stereotypes. Methodologies and tools for developing a user modeling server [3] and for producing adaptable and adaptive interfaces [7, 9] are being employed in the project.

The iterative evaluation of the AVANTI prototype is now nearing its completion. In particular, adaptability and adaptivity in the user interface and the information content levels are the subject of a separate, specifically designed part of the evaluation procedure, which is currently under way [10].

Acknowledgements

The ACTS AC042 AVANTI project "AdaptiVe and Adaptable INteractions to Multimedia Telecommunications Applications", is partially funded by the European Commission (DG XIII). The AVANTI consortium comprises: ALCATEL Siette (Italy) - Prime contractor; CNR-IROE (Italy); ICS-FORTH (Greece); GMD (Germany); University of Sienna (Italy); MA Systems (UK); MATHEMA (Italy); VTT (Finland); ECG (Italy); University of Linz (Austria); TELECOM ITALIA (Italy); EUROGICIEL (France).

References

[1] Fink, J., A. Kobsa and J. Schreck (1997): Personalized Hypermedia Information through Adaptive and Adaptable System Features: User Modeling, Privacy and Security Issues. In: A. Mullery, M. Besson, M. Campolargo, R. Gobbi and R. Reed, eds.: Intelligence in Services and Networks: Technology for Cooperative Competition. Berlin Heidelberg: Springer, 459-467.

[2] Kobsa, A., A. Nill and J. Fink (1997): Hypertext and Hypermedia Clients of the User Modeling System BGP-MS. In M. Maybury, ed.: Intelligent Multimedia Information Retrieval. Boston, MA: MIT Press.

[3] A. Kobsa, W. Pohl (1995): The User Modeling Shell System BGP-MS. User Modeling and User-Adaptive Interaction 4(2), 59-106.

[4] A. Nill (1997): Providing Useable and Useful Information by Adaptivity and Adaptability. Proceedings of the Flexible Hypertext Workshop, 8th ACM International Hypertext Conference (Hypertext'97), Southhampton, England. http://zeus.gmd.de/~nill/flexht97.html

[5] C. Stephanidis, Y. Mitsopoulos (1995): INTERACT: An Interface Builder Facilitating Access to Users with Disabilities. Proc. HCI International '95, Tokyo, Vol. 2, 923-928.

[6] C. Stephanidis, A. Paramythis, M. Sfyrakis, A. Stergiou, N. Maou, A. Leventis, G. Paparoulis, C. Karagianidis (1998): "Adaptable and Adaptive User Interfaces for Disabled Users in AVANTI Project", 5th International Conference on Intelligence in Services and Networks (IS&N '98), "Technology for Ubiquituous Telecom Services", Antwerp, Belgium.

[7] Stephanidis, C., Savidis, A., & Akoumianakis, D. (1997). Unified interface development: Tools for constructing accessible and usable user interfaces: Tutorial No. 13 in HCI International í97 Conference. Available at: http://www.ics.forth.gr/proj/at-hci/html/ publications.html

[8] A. Bini, P.L.Emiliani (1997): Information about mobility issues: The ACTS AVANTI project. 4th European Conference for Advancement of Assistive Technology (AAATE '97), Porto Carras, Greece, 29 September - 2 October 1997.

[9] A. Savidis, A. Paramythis, D. Akoumianakis, C. Stephanidis, "Designing user-adapted interfaces: the unified design method for transformable interactions", ACM Conference on Designing Interactive System: Processes, Methods and Techniques (DIS '97), Amsterdam, The Netherlands, 18-20 August 1997, 323-334.

[10] AVANTI Deliverable DE030 "Global evaluation of the experiments", forthcoming public deliverable of the AVANTI consortium, August 1998.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Issues in Authoring Adaptive Hypertext on the Web

Karen A. Lemone
Computer Science Department, WPI
100 Institute Road, Worcester, MA 01609 USA
http://www.cs.wpi.edu/
kal@cs.wpi.edu

Abstract: In this paper, we examine some problems that authors face in creating adaptive hypertext, and why systems may need to provide more than one mechanism to authors. We discuss two such mechanisms as they have been implemented in ReCourse, a course management system, http://www.webrecourse.com.

Keywords: adaptive hypertext, dynamic hypertext, authoring, web-based course management systems

1 Introduction

WebReCourse, the Retargetable Course Generator enables creation and reuse of Web courses. It is a secure software system for online course management, allowing instructors to increase the accessibility of online course material and to create a convenient means of communication between instructors and students.

ReCourse allows an author to write a single document which can dynamically choose its content when a reader selects a particular topic.

We call this *Retargeting,* because the information is retargeted to different readers and different audiences. Such retargeting is a staple among course instructors who teach similar courses to different audiences, or teach the same course multiple times under different circumstances.

In this paper, we presume that the instructor/author knows the criteria which have segmented the readers, so that readers need only register for the correct group and the pages will be chosen dynamically for them. This is currently done statically, but future versions of ReCourse will address the (difficult) issue of discovering the appropriate information about the reader. However this is done, the author still has to decide what information is appropriate for what collection of reader attributes.

We consider *retargeting* one of the integral features of adaptive hypertext from the author's perspective.

2 Author Characteristics

In Spring of 1998, we tested ReCourse with a small set of instructors from the WPI community. We purposely chose instructors from a number of disciplines and of varying computer skills. ReCourse provides a number of tools for both instructors and students - bulletin boards, automatic quiz generation, grading etc.

Although we had preliminary versions of the tools for web page creation and retargeting implemented, we didn't make them available because we wanted to see how authors actually created the pages they used within ReCourse, and to test out some hypotheses we had developed.

We had hypothesized that some instructors would wish to author their own pages using HTML (and other languages such as java, cgi etc.), but that less computer-literate instructors would require a point and click tool. These two groups did exist. (There was a third group that used the campus's page creation service, but even they professed a wish to be able to retarget the material once it was created for them.)

3 ReCourse Mechanisms

ReCourse's two mechanisms for authors reflect the needs of the two groups: the HTML- group and the point-and-click group.

Authoring for the HTML-group

We designed special HTML-like tags for this group. When authoring documents for use within ReCourse, writers can use special tags such as <omit> </omit>, <precondition *name* >, <postcondition *name* >

The <omit> </omit> tags are inserted into documents when the text between them is not to be shown when a user matches a particular profile. The <precondition *name* >, <postcondition *name* > tags force inclusion of particular material - when the precondition name matches a postcondition name, both pieces of information must be included. We are currently investigating other tags.

Authoring for the HTML-group

A tool currently called *Site Composer* allows both pages creation and retargeting. Page creation is accomplished by selecting graphics, dragging them to the desired position in the page, writing and placing text, links and other hypermedia objects. Retargeting is accomplished by selecting various objects on the page and then saving them to a group name.

4 Summary

We will be testing these two mechanisms in the fall, again, with a select group of WPI faculty members. While we won't request that the point-and-click group use the html approach, we will request the other group to try both mechanisms and tell us which they prefer (we will know by checking their pages).

Adaptive hypertext is desirable in web applications to allow readers to see what is appropriate for them. But such features require a set of tools "adaptive" to the various needs of authors.

References

1. Lemone, K. (1996). Retargetable Course Generation: a Method for Reusability in Distance Education In *Workshop on Architectures for Intelligent Tutoring Systems*, Montreal, Canada.
2. Lemone, K. (1997). Assessment of Tools for Virtual Teaching In *Proceedings of Enable '97*, Espoo, Finland.
3. Lemone, K. (1997). Experiences in Virtual Teaching In *Proceedings of WebNet'97*, Toronto, Canada.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Modelling Personalizable Hyperlink-Based Interaction

James Ohene-Djan, Alvaro A.A. Fernandes
Department of Mathematical and Computing Sciences
Goldsmiths College, University of London
New Cross, London SE14 6NW, UK
{ j.ohene-djan | fernandes }@gold.ac.uk

Abstract : This paper describes the development of an abstract model for hyperlink-based interaction in which personalization can be studied with greater conceptual clarity than is possible by technology-driven experimentation. The model characterizes a rich set of abstract user-initiated tailoring actions, which enable individual users to come closer to satisfying their specific, and often dynamic, information retrieval goals. The model forms a foundation for our current work, a systematic investigation of the nature, scope and effects of system-initiated tailoring actions on *hyperlink-based systems*(HLBSs).

Keywords: Hypermedia Design, WWW Personalization, Hypermedia Modelling.

Motivation

Research into *Personalization & adaptation(P&A)* actions in HLBSs is motivated by a great interest (both scientific and commercial) in increasing the effectiveness of HLBSs as a platform for information retrieval tasks in which different users have different information goals and different histories. In HLBSs that lack P&A actions most of the interaction a user might experience with a hyperdocument is determined by the design decisions that shaped the hyperdocument in terms of its content, its rendering aspects and the navigation possibilities it offers the user. As a consequence the designer *owns* the hyperdocument. Thus, in HLBSs that lack P&A actions, users can navigate through a hyperdocument using links, but they are prevented from enforcing their individual preferences as to content, rendering and navigation possibilities.

Our approach to overcoming this impediment is to extend HLBSs with P&A actions that effect a transfer of ownership from the original designers of the hyperdocument to each of its users, thereby enabling the latter to redesign the former according to their specific information goals and histories.

We also draw some of our motivation from the fact that there seems to be no discernible consensus among researchers in adaptive hypermedia with respect to following questions:

1. Which are the emergent properties of HLBSs? Equivalently, what is the scope of P&A in HLBSs?
2. Which P&A actions *could* be made available to users? Equivalently, what *descriptive stance* should be taken with respect to P&A actions in HLBSs?
3. Which P&A actions *should* be made available to users? Equivalently, what *prescriptive* stance should be taken with respect to P&A actions in HLBSs?

We would argue for a precise, abstract characterization of what emergent properties can be assigned to HLBSs, so that issues relating to P&A in the technologies of which an open HLBSs tends to be a client (e.g., in database and in user-interface technologies) are not confounded with P&A issues in HLBSs. In other words, we believe it is important to characterize, at an informative level of abstraction, what is unique to HLBSs rather than inherited (or shared) with server technologies. Such a characterization would open the way to a principled exploration of specific P&A issues in HLBSs.

Our Approach

The main ideas guiding our approach can be phrased as follows:

1. The model is an abstract model, as many steps removed from concrete implementations as necessary to allow as systematic, exhaustive investigation of P&A issues in HLBSs.
2. The model is an open model, insofar as we view HLBSs as clients of a variety of servers, an in particular of data and user-interface servers.
3. Personalization involves a transfer of ownership of the process of interaction with a hyperdocument, from designers to users.
4. To ensure that the set of personalization actions is consistent, its elements are induced from the formal definition of the hyperdocuments they act upon.
5. All design decisions are, in principle, in scope for personalization actions.
6. Our model of hyperlink-based personalization can express most, if not all, personalization actions proposed in the literature (see[1] for a comprehensive review).
7. Our model describes which personalization actions can be made available. In order to prescribe which should be made available, empirical studies are needed.

THE Goldsmiths Hyperlink Model

To model adaptive, personalizable hyperlink-based interaction we propose a model core hyperlink behaviour by partitioning it into three regions. Non-adaptive, non-personalizable HLBSs are modelled by the functions provided by what we refer to as the H-region. Personalizable HLBSs require the addition to the H-region of the functions provided by what we refer to as the P-region. This causes no disruption whatsoever and requires no changes at all to the H-region. Adaptive HLBSs require the addition to the P- and H-regions of the functions provided by what we refer to as the A-region.

The H-Region

The **H-region** is formalized as a *composer from specifications*, i.e., what a designer writes is not a document but rather a specification of how to build the document upon request. A formal language for writing such specifications has been defined along with a formal abstract machine to execute them thereby yielding renderable documents. At the most basic level, the functionality provided by the H-region is to process specifications of hyperdocuments into renderable texts. In this approach, a designer writes specifications as to where the contents can be found, what rendering is the document to have and how to compose content and presentation features into a renderable text that a user sees. The basic dynamics of the H-region is the following. The user requests a page to be rendered, as usual. Such a page exists, and is fetched, as a specification of where to find its content and how to render it. One then proceeds to fetch the contents by client-server querying designer-specified sources. The retrieved content is then composed, as specified by the designer, into the text to be rendered. Finally, the core responds to the request with the text thus composed.

The P-Region

The P-region comprises a group of functions that are non-disruptively added to the H-region in order to model personalizable hyperlink-based interaction. The P-region provides two basic processes: the personalization of hyperpages by annotation and rewriting, and the enforcement over a renderable text of previously expressed preferences (in the form of notes on a hyperpage). When superimposing the P-region onto the H-region, users can not only request a hyperpage, but also annotate or rewrite it, thereby creating their own version of it. Thus design of that hyperpage can therefore be overridden by user and this event characterizes ownership transfer.

The kinds of personalization actions that we model are based on annotating and rewriting the hyperpage specifications. Annotation pairs a hyperpage specification with notes of interest to the user, and by doing so, presumes that versioning takes place. Such notes take one of the following forms. Firstly, a note can assign user-specific values to user-generic attributes of interest (e.g., that the level of difficulty of a given page is high, or that `planets' is a keyword of relevance to a given segment of a page). Secondly, a note can specify a rewriting action over the renderable text after it has been composed by the H-region, i.e., after content has been fetched and made ready for display (e.g., to map American into British spelling forms). This form of post-composition rewriting can also be conditional on the environment (e.g., replace images with captions if the display unit is text-only). A formal language for tailoring the hyperdocument specified by the designer into a personalized version has been defined along with a formal abstract machine to generate them. For details, see[3]. The existence of annotations on hyperpages allows for:

1. *personalization* of a specified hyperpage;
2. the specification of *alternatives* to a specified hyperpage;
3. the specification of *comparable hyperpages* to a specified one; and
4. the recording of *information* about a hyperpage (i.e., what are the current values of attributes set by previous annotations).

The A-Region

The A-region comprises a group of functions that are non-disruptively added to the P-region to model Adaptivity. We view adaptation as system-initiated personalization. The immediate import of this is that our conception of adaptation requires a HLBS, at least, to employ a model of each user as the basis of a prescriptive theory of what personalization action might the user have taken, and to initiate that action on behalf of the user. Furthermore in our view, adaptation is, in principle, as expressive as personalization and requires no other technologies than those involved in user modelling and in decision making from a user model. The A-region enables users and designers to define strategies as to when the system should take the initiative and actively tailor the interaction to a user in the light of that user's information goals and history of use.

Conclusions and current work

The approach we are currently exploring to add, adaptive capabilities centers on an adaptation function. This function implements an inference engine over a decision theory (i.e., a theory as to which actions are more likely to yield the most benefits given some accumulated knowledge of past interactions). The accumulated knowledge are the information goals and the history of each user, while the actions which the inference engine is in charge of suggesting are personalization actions as defined by the P-Region.[2] Besides greater conceptual clarity regarding the scope for P&A in HLBSs. The model summarized in this paper we hope represents a contribution towards a better foundation to underlie the exploration of P&A[1] issues in HLBSs.

References

[1] Peter Brusilovsky. Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction, 6:87--129, 1996.

[2] A.A.A. Fernandes, M.H. Williams, and N.W. Paton. A Logic-Based Integration of Active and Deductive Databases. New Generation Computing, 15(2):205--244, 1997.

[3] James Ohene-Djan and Alvaro A.A. Fernandes. Personalizable hyperlink-based interaction. Technical report, Department of Mathematical and Computing Sciences, Goldsmiths College, University of London, 1998.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Position Statement: Adaptive navigation tools for thesaurus-based retrieval in cultural heritage applications

Douglas Tudhope
Editor: The New Review of Hypermedia and Multimedia
DSTUDHOPE@GLAMORGAN.AC.UK
Computer Studies Department
University of Glamorgan
Pontypridd, Wales, UK
tel 01443-482271
fax 01443-482715
http://www.comp.glam.ac.uk/pages/staff/dstudhope

Research on intelligent hypermedia systems has been carried out at the University of Glamorgan (UG) for about five years. Research has investigated how semantic structure underlying information can be used to enhance browsing and search tools via computed links. There have been several publications in the hypermedia, information retrieval and cultural heritage areas. Tudhope has recently (1997) taken up the editorship of the refereed annual review journal New Review of Hypermedia and Multimedia (previously Hypermedia - one of the oldest specialist hypermedia journals). Cultural heritage and adaptive hypermedia are the two special themes for the 1998 issue, and evaluation the special theme for 1997.

Six years ago the University was commissioned to build a hypermedia museum exhibit on local history from the photographic archives of the Pontypridd Historical and Cultural Centre. This inspired a University research assistantship in collaboration with PHCC that resulted in Carl Taylor's PhD work on semantic modelling and navigation in museum hypermedia systems. As part of this work, a number of research prototypes were built investigating a hypermedia architecture with a semantic index space separate from the document space (Beynon-Davies et al 1994, Taylor et al 1994). A variety of conventional hypermedia navigation techniques were implemented with this architecture (Tudhope et al 1994). Primary access routes were time, space and as subject index the Social History and Industrial Classification (SHIC). Rather than using fixed embedded links, navigation was based on queries over an underlying semantic index space, with results post-processed for expression in a particular navigation tool. Queries to the database can be simple or complex. Conventional hypermedia navigation techniques, including both local and global browsers, guided tours, and Boolean queries can be implemented by relatively simple underlying queries. More complex queries return partial matches using measures of semantic closeness between terms in a semantic index space; advanced navigation options included query expansion when a query fails to return results and navigation via similarity to current item (Cunliffe, Taylor, Tudhope 1997). Most existing commercial museum access systems using thesauri rely on interactive approaches or limited query expansion techniques. A classification system, or thesaurus, embodies a semantic network of relationships between terms, the three main thesaurus relationships being hierarchical, associative and equivalence. Thus there is some inherent notion of distance between terms, their 'semantic closeness'. Distance measurements can be exploited to provide more advanced navigation tools. A distance measurement between terms (and between sets of terms) offers the opportunity for imprecise information requests. Semantic term expansion lies at the heart of the measures of closeness between terms in automated term expansion and similarity measures in retrieval tools.

The intention in future research is to build on the principles underlying the semantic index space, extending the semantic model to a full set of thesaurus relationships, and investigate the potential of intelligent navigation tools in a major museum which can facilitate evaluation – current collaborators include the Science Museum, London. One issue is tailoring the navigation tool to dynamically adapt to different classes of user. This can affect the visualisation of the thesaurus structure for browsing and the parameters controlling the semantic closeness algorithms and query expansion.

Relevant publications

Beynon-Davies P., Tudhope D., Taylor C., Jones C. 1994. A Semantic Database Approach to Knowledge-based Hypermedia Systems, Information and Software Technology, 36(6), 323-329.

Cunliffe D., Taylor C., Tudhope D. 1997. Query-based navigation in semantically indexed hypermedia. Proceedings 8th ACM Conference on Hypertext (Hypertext'97), Southampton. April, pp 87-95.

Jones C., Taylor C., Tudhope D., Beynon-Davies P. 1996. Conceptual, Spatial and Temporal Referencing of Multimedia objects. Advances in GIS Research II, Proceedings 7th International Symposium on Spatial Data Handling, (eds. Kraak and Molenaar) , August, Delf, pp 2.13-2.26.

Taylor C., Tudhope D., & Beynon-Davies P. 1994. Representation and Manipulation of Conceptual, Temporal and Geographical Knowledge in a Museum Hypermedia System. Proc. ACM European Conference on Hypermedia Technology (ECHT'94), Edinburgh, 239-244.

Tudhope D., Beynon-Davies P., Taylor C., Jones C. 1994. Virtual Architecture based on a Binary Relational Model: A Museum Hypermedia Application, Hypermedia, 6(3), 174-192.

Tudhope D., Taylor C., Beynon-Davies P. 1995. Taxonomic Distance: Classification and Navigation. Proceedings 3rd International Conference on Hypermedia and Interactivity in Museums: Multimedia Computing and Museums (ed. D. Bearman), San Diego, October, pp 322-334.

Jones C., Beynon-Davies P., Taylor C., Tudhope D. 1995. GIS, hypermedia, and historical information access. Proceedings 7th International Conference of the Museum Documentation Association, Edinburgh, Nov, pp 109-113.

Tudhope D., Taylor C. 1996. Flexible Access to Multimedia Museum Collections. Proceedings Electronic Imaging and the Visual Arts (EVA'96), London, July, pp 7.1- 7.11

Tudhope D., Taylor C. 1996. A unified similarity coefficient for navigating through multi-dimensional information. Proc. 59th Conference of the American Society for Information Science (ASIS'96), 67-70.

Tudhope D., Taylor C. 1997. Navigation via Similarity: automatic linking based on semantic closeness. Information Processing and Management, 33(2), 233-242.

Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia,
HYPERTEXT'98, Pittsburgh, USA, June 20-24, 1998

Designing Dynamic Hypertext

Michael White
CoGenTex, Inc.
The Village Green, 840 Hanshaw Road, Ithaca, NY 14850 USA
http://www.cogentex.com/
mike@cogentex.com

Abstract: To support the development of adaptive, dynamic hypertext systems at CoGenTex, we have been evolving *Exemplars,* a rule-based object-oriented framework for dynamic hypertext generation. In this paper, we sketch our approach to designing dynamic hypertext systems using the *Exemplars* framework, and discuss how it fits with the tasks of content authoring and information design.

Keywords: dynamic hypertext, natural language generation, authoring, information design

1 Introduction

As various researchers have suggested [e.g., 3, 4], automatic text generation techniques can be used to produce dynamic hypertext tailored to an individual user's needs. At CoGenTex, we have developed several web-based dynamic hypertext systems over the past few years [e.g., 1, 2, 5], one of which, Project Reporter [2], is entering the beta-release stage of product development. While our emphasis to date with these systems has been on providing easy access to up-to-date information, we have included some capabilities to tailor the dynamically generated hypertext to explicit user preferences selected from an options menu, and envision making use of more sophisticated means of user tailoring in future systems.

To support the development of these dynamic hypertext systems at CoGenTex, we have been evolving *Exemplars,* a rule-based object-oriented framework for dynamic hypertext generation [7]. In this paper, we sketch our approach to designing dynamic hypertext systems using the *Exemplars* framework, and discuss the challenges we have faced in integrating our approach with the tasks of content authoring and information design.

2 The *Exemplars* Framework

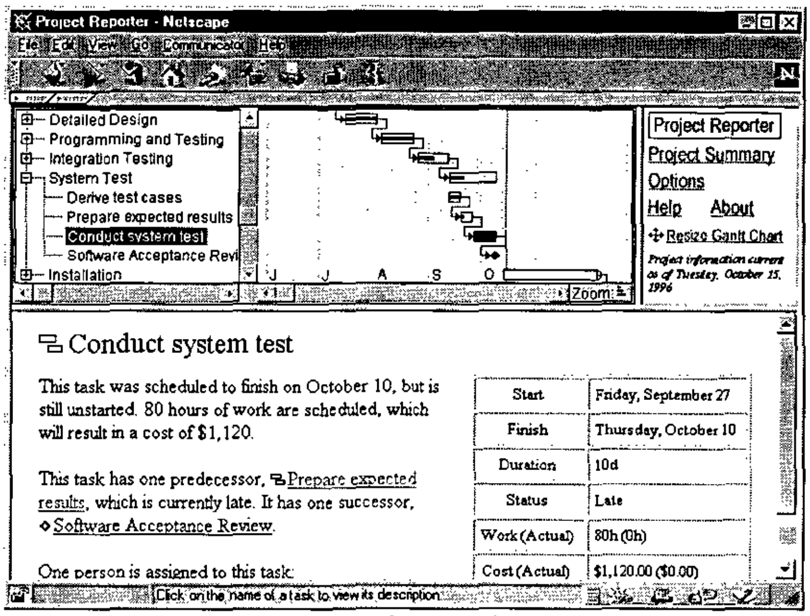
The main idea behind the *Exemplars* framework is to enable the designer to determine the behavior of a dynamic hypertext system by writing a set of object-oriented text planning rules and arranging them into a specialization hierarchy, where more specialized rules can augment or override the more general ones they specialize. By text planning rules, we mean rules that determine the content and form of the generated hypertext. Each such rule has a condition and an action: the condition defines the applicability of the rule in terms of tests on the input data, the discourse context, and the user model, whereas the action defines what hypertext to add to the current output and how to update the discourse context and user model.

The text planning rules are known as *exemplars,* as they are meant to capture an exemplary way of achieving a communicative goal in a given communicative context. The name also reflections our conception of designing practical dynamic hypertext systems as essentially an expert-systems task, though one where specialization plays a key role. We will briefly describe by way of example the role specialization plays in our approach in the next section.

Following Reiter and Mellish [6], we view exemplar selection, at the heart of our approach, as a process of rule classification. Perhaps not surprisingly, their classification-based approach to hypertext generation turns out to be quite similar to ours, though each was developed independently; where we consider our approach to improve upon theirs is in its extensibility, HTML/SGML support, and Java-based implementation. For further details on the framework and its novel features, see [7].

3 Example: Project Reporter

Project Reporter is an innovative web-based tool for monitoring the status of a project. Using information obtained from a project management database, Project Reporter automatically generates fluent natural-language reports describing task progress, staffing, labor expenditures, and costs for a project. It also displays project data in tables and in Gantt chart form, providing a complete multimodal view of the project's status, as shown in the following screenshot:



(click to view full size)

A slightly simplified exemplar from Project Reporter is shown below. When applicable, this exemplar adds to the page-in-progress a one-sentence paragraph such as *This task was scheduled to finish last Friday, May 22, three days ahead of the baseline schedule, but is currently only 75% complete.*

```
exemplar DescribeLateTaskStatus(Task task) extends DescribeTaskStatus
{
  boolean evalConstraints() { return task.completionStatus.isLate(); }

  void apply()
  {
    <<+
      <p>
        This task was scheduled to finish {{ IdentifyDate(task.finishDate
        {{ AddBaselineStatusModifier(task) }},
        but is currently only { task.percentComplete }% complete.
    +>>
  }
}
```

To briefly illustrate the role specialization plays in our approach (without needing to get into the details of the domain object model), let us consider how Project Reporter contextually identifies dates here. (If viewing on-line, see the annotated UML diagram.) The DescribeLateTaskStatus exemplar shown above calls the IdentifyDate exemplar with the given task's finish date. This call is mediated by the text planner component of the framework, which automatically selects and invokes the most specialized applicable exemplar. It does so by traversing the specialization tree below IdentifyDate, evaluating conditions as it goes, passing through IdentifyDateOutOfFocus, IdentifyDateWithinADay (which fails), IdentifyDateWithinAWeek, and finally selecting IdentifyDateLastWeek; for the example at hand, the IdentifyDateLastWeek's action is to add *last* and *Friday* to the output of the default exemplar, which is *May 22*. Of course, depending on the input date, today's date, and the last date mentioned, the resulting text could be quite different.

4 Discussion

With *Exemplars*, dynamic content is generated using object-oriented text planning rules. This approach provides great flexibility, though at the expense of added overhead on content authoring. In our view, the *Exemplars* framework becomes useful when the desired content is primarily dynamic rather primarily static. When there is a great deal of primarily static content to be included, our experience with CogentHelp [1] (esp. feedback from SIGDOC-97) has taught us that it makes sense to develop a separate interface for content authors, leaving the exemplar writing to information designers, possibly together with application programmers.

References

1. Caldwell, D. E. and White, M. (1997). CogentHelp: A tool for authoring dynamically generated help for Java GUIs. In *Proceedings of the 15th Annual International Conference on Computer Documentation (SIGDOC-97)*, Salt Lake City, Utah.
2. CoGenTex, Inc. (1997). Text Generation Technology for Advanced Software Engineering Environments. Final Report, Rome Laboratory Contract No. F30602-92-C-0163.
3. Knott, A., Mellish, C., Oberlander, J. & O'Donnell, M. (1996). Sources of flexibility in dynamic hypertext generation. In *Proceedings of the Eighth International Natural Language Generation Workshop (INLG-96)*, Herstmonceux Castle, Sussex, UK, 151-160.
4. Milosavljevic, M., Tulloch, A., and Dale, R. (1996). Text generation in a dynamic hypertext environment. In *Proceedings of the 19th Australasian Computer Science Conference*, pages 229-238, Melbourne, Australia.
5. McCullough, D., Korelsky, T., and White, M. (1998). Information Management for Release-based Software Evolution Using EMMA. To appear in the *Proceedings of the Tenth International Conference on Software Engineering and Knowledge Engineering (SEKE-98)*, San Francisco Bay, USA.
6. Reiter, E., and Mellish, C. (1992). Using classification to generate text. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL-92)*, Newark, Delaware, 265-272.
7. White, M. and Caldwell, D. E. (1998). Exemplars: A Practical, Extensible Framework for Dynamic Text Generation. To appear in *Proceedings of the Ninth International Natural Language Generation Workshop (INLG-98)*, Niagara-on-the-Lake, Ontario, Canada.

Computing Science Reports **Department of Mathematics and Computing Science**
Eindhoven University of Technology

In this series appeared:

96/01	M. Voorhoeve and T. Basten	Process Algebra with Autonomous Actions, p. 12.
96/02	P. de Bra and A. Aerts	Multi-User Publishing in the Web: DreSS, A Document Repository Service Station, p. 12
96/03	W.M.P. van der Aalst	Parallel Computation of Reachable Dead States in a Free-choice Petri Net, p. 26.
96/04	S. Mauw	Example specifications in phi-SDL.
96/05	T. Basten and W.M.P. v.d. Aalst	A Process-Algebraic Approach to Life-Cycle Inheritance Inheritance = Encapsulation + Abstraction, p. 15.
96/06	W.M.P. van der Aalst and T. Basten	Life-Cycle Inheritance A Petri-Net-Based Approach, p. 18.
96/07	M. Voorhoeve	Structural Petri Net Equivalence, p. 16.
96/08	A.T.M. Aerts, P.M.E. De Bra, J.T. de Munk	OODB Support for WWW Applications: Disclosing the internal structure of Hyperdocuments, p. 14.
96/09	F. Dignum, H. Weigand, E. Verharen	A Formal Specification of Deadlines using Dynamic Deontic Logic, p. 18.
96/10	R. Bloo, H. Geuvers	Explicit Substitution: on the Edge of Strong Normalisation, p. 13.
96/11	T. Laan	AUTOMATH and Pure Type Systems, p. 30.
96/12	F. Kamareddine and T. Laan	A Correspondence between Nuprl and the Ramified Theory of Types, p. 12.
96/13	T. Borghuis	Priorean Tense Logics in Modal Pure Type Systems, p. 61
96/14	S.H.J. Bos and M.A. Reniers	The I^2C-bus in Discrete-Time Process Algebra, p. 25.
96/15	M.A. Reniers and J.J. Vereijken	Completeness in Discrete-Time Process Algebra, p. 139.
96/17	E. Boiten and P. Hoogendijk	Nested collections and polytypism, p. 11.
96/18	P.D.V. van der Stok	Real-Time Distributed Concurrency Control Algorithms with mixed time constraints, p. 71.
96/19	M.A. Reniers	Static Semantics of Message Sequence Charts, p. 71
96/20	L. Feijs	Algebraic Specification and Simulation of Lazy Functional Programs in a concurrent Environment, p. 27.
96/21	L. Bijlsma and R. Nederpelt	Predicate calculus: concepts and misconceptions, p. 26.
96/22	M.C.A. van de Graaf and G.J. Houben	Designing Effective Workflow Management Processes, p. 22.
96/23	W.M.P. van der Aalst	Structural Characterizations of sound workflow nets, p. 22.
96/24	M. Voorhoeve and W. van der Aalst	Conservative Adaption of Workflow, p.22
96/25	M. Vaccari and R.C. Backhouse	Deriving a systolic regular language recognizer, p. 28
97/01	B. Knaack and R. Gerth	A Discretisation Method for Asynchronous Timed Systems.
97/02	J. Hooman and O. v. Roosmalen	A Programming-Language Extension for Distributed Real-Time Systems, p. 50.
97/03	J. Blanco and A. v. Deursen	Basic Conditional Process Algebra, p. 20.
97/04	J.C.M. Baeten and J.A. Bergstra	Discrete Time Process Algebra: Absolute Time, Relative Time and Parametric Time, p. 26.
97/05	J.C.M. Baeten and J.J. Vereijken	Discrete-Time Process Algebra with Empty Process, p. 51.
97/06	M. Franssen	Tools for the Construction of Correct Programs: an Overview, p. 33.
97/07	J.C.M. Baeten and J.A. Bergstra	Bounded Stacks, Bags and Queues, p. 15.

97/08	P. Hoogendijk and R.C. Backhouse	When do datatypes commute? p. 35.
97/09	Proceedings of the Second International Workshop on Communication Modeling, Veldhoven, The Netherlands, 9-10 June, 1997.	Communication Modeling- The Language/Action Perspective, p. 147.
97/10	P.C.N. v. Gorp, E.J. Luit, D.K. Hammer E.H.L. Aarts	Distributed real-time systems: a survey of applications and a general design model, p. 31.
97/11	A. Engels, S. Mauw and M.A. Reniers	A Hierarchy of Communication Models for Message Sequence Charts, p. 30.
97/12	D. Hauschildt, E. Verbeek and W. van der Aalst	WOFLAN: A Petri-net-based Workflow Analyzer, p. 30.
97/13	W.M.P. van der Aalst	Exploring the Process Dimension of Workflow Management, p. 56.
97/14	J.F. Groote, F. Monin and J. Springintveld	A computer checked algebraic verification of a distributed summation algorithm, p. 28
97/15	M. Franssen	λP-: A Pure Type System for First Order Loginc with Automated Theorem Proving, p.35.
97/16	W.M.P. van der Aalst	On the verification of Inter-organizational workflows, p. 23
97/17	M. Vaccari and R.C. Backhouse	Calculating a Round-Robin Scheduler, p. 23.
97/18	Werkgemeenschap Informatiewetenschap redactie: P.M.E. De Bra	Informatiewetenschap 1997 Wetenschappelijke bijdragen aan de Vijfde Interdisciplinaire Conferentie Informatiewetenschap, p. 60.
98/01	W. Van der Aalst	Formalization and Verification of Event-driven Process Chains, p. 26.
98/02	M. Voorhoeve	State / Event Net Equivalence, p. 25
98/03	J.C.M. Baeten and J.A. Bergstra	Deadlock Behaviour in Split and ST Bisimulation Semantics, p. 15.
98/04	R.C. Backhouse	Pair Algebras and Galois Connections, p. 14
98/05	D. Dams	Flat Fragments of CTL and CTL*: Separating the Expressive and Distinguishing Powers. P. 22.
98/06	G. v.d. Bergen, A. Kaldewaij V.J. Dielissen	Maintenance of the Union of Intervals on a Line Revisited, p. 10.
98/07	Proceedings of the workhop on Workflow Management: Net-based Concepts, Models, Techniques and Tools (WFM'98) June 22, 1998 Lisbon, Portugal	edited by W. v.d. Aalst, p. 209
98/08	Informal proceedings of the Workshop on User Interfaces for Theorem Provers. Eindhoven University of Technology ,13-15 July 1998	edited by R.C. Backhouse, p. 180
98/09	K.M. van Hee and H.A. Reijers	An analytical method for assessing business processes, p. 29.
98/10	T. Basten and J. Hooman	Process Algebra in PVS
98/11	J. Zwanenburg	The Proof-assistemt Yarrow, p. 15