Teaching Hypertext and Hypermedia through the Web

Paul M. E. De Bra
(Eindhoven University of Technology
debra@win.tue.nl)

Abstract: Since early 1994 the introductory course 2L670, "Hypermedia Structures and Systems", has been available on World Wide Web, and is an optional part of the curriculum in computing science at the Eindhoven University of Technology. The course has since been completed by more than 200 students from three different universities, two in the Netherlands and one in Belgium. Since January 1st 1996 the course is also offered at two more Dutch universities and at the Dutch Open University.
In order to participate in this course the student only needs a World Wide Web browser. There is no need for separate e-mail, netnews, bulletin boards or ftp software.
In this paper we present the technical environment for the latest edition of the course, which features automatic evaluation of small assignments, a repository for assignment work, a discussion system, and complete monitoring of each student's progress. We also reflect on the previous architecture, used for the first 200 students, and draw lessons from that experience.
Key Words: hypertext, courseware, document repository, discussion groups, monitoring progress

1 Introduction

In the spring of 1993 the Department of Computing Science of the Eindhoven University of Technology agreed to add an optional course on hypertext and hypermedia to the curriculum [De Bra 94]. In order to give the students some hands-on experience with hypertext while studying this course we decided to offer the course text in hypertext form, using World Wide Web technology. We also decided to offer the text as a real hyperdocument, not having a linear or strictly hierarchical structure. The URL (or Web address) for the course is http://www.win.tue.nl/2L670/. The course text is written in HTML-3.2, a standard version of the HyperText Markup Language used on World Wide Web. Only features supported by most current WWW browsers are used.

The course places a focus on principles and practice, not on fancy features and multimedia applications. The aim of the hypermedia course is to teach students how to create hyperdocuments that are easy to use, how to develop hyperdocuments from information items and semantic relationships, and how to build hypermedia systems that offer a rich set of useful navigation aids. World Wide Web is the enabling technology, but not the subject matter of the course. Hypermedia functionality not (yet) offered by the Web plays an equally important role as the features offered by popular browsers and servers.

In this paper we focus on the use of World Wide Web technology to make such a course possible, and on the tools we developed and/or used to help both the teacher and the student throughout the course. These tools include tracking student progress, automatic evaluation of intermediate and final knowledge tests, an upload and grading facility for assignment work, and a Web-based discussion system. For remote students without permanent Internet connection (like the students of the Open University) most of the course text also remains available in a static form. Thus, students can download the text and read it off-line to avoid a high phone bill. (In Europe even local phone calls are charged per time unit.)

2 Initial Experience with On-Line Hypertext Courseware

At first the hypermedia course was just an "ordinary" course, taught in a classroom. For such classes, students typically get a book, a reader, or some other kind of hand-out, and they take notes during the lectures. With the hypermedia course the "book" was only available on-line, and the classrooms were not equipped with computers on which the students could make annotations to the pages that were being discussed. In order to give students the opportunity to study the course with a computer, the actual teaching in class was cancelled after four semesters (two in Eindhoven, The Netherlands, and two in Antwerp, Belgium). This immediately made it a lot easier to offer this course to students from other universities. Since January 1996 the course is available to students from six different universities in two countries.

Like many courses at the Eindhoven University this course consists of two parts: the students have to read a substantial amount of information, and they have to complete an assignment to show that they understand and master the principles taught in this course. This second aspect is of course more important than the first.

The hypermedia course deals with the following subjects (chapters):

- introduction
- definition of hypertext and hypermedia
- history of hypertext and hypermedia systems
- reference models for hypermedia system architecture
- navigation (or browsing) in hypertext
- information retrieval in hypertext
- authoring and converting text to hypertext
- distribution and concurrency issues
- future evolutions

From this list it is clear that there is quite a bit of overlap between subjects. Concepts are illustrated by pointing to the features of systems in the historic overview. Converting text to hypertext involves navigational aspects, and writing hypertext is often done in a distributed way, by means of collaborating authors working concurrently. Because of this (and a lot more) overlap there are many cross-references between subjects, making the course text a real hyperdocument, which is difficult to reduce to a linear (book) structure.

Students have sometimes complained about the lack of a printed version of the course text. For other courses it would probably be better to offer such version, but for the hypermedia course we considered it more important to give the students some hands-on experience with a hyperdocument.

In order to determine whether students have read most or all of the course text, a multiple-choice test is included. The students have to answer 20 questions, by selecting *one* out of four or five possible answers, or for some questions *one or more* out of four or five possible answers. To make it more difficult to solve this quiz by guessing, the student has to answer all 20 questions before getting a score. The only information given to students is the number of correct answers. Every time the student completes all questions, and submits them for automatic evaluation, the score is logged. These logfiles can be used to verify whether a student makes a very large number of attempts to guess the right answers, and also to check whether a student doesn't simply copy the right answers from a friend, and answers all questions correctly on the first or second attempt. The questions are sufficiently difficult and tricky to make it very unlikely that anyone will answer all of them correctly the first time around. In [4. Using Multiple-Choice Tests] we give some suggestions on how to make better use of multiple-choice tests.

The students' grades depend only on the assignment work. Answering all questions of the multiple choice test correctly is just a prerequisite for being allowed to turn in the assignment. This, and the fact that the test is the same for all students, does not encourage them to put a lot of effort in trying to find the right answers in the course text. About 40% of the students copy the answers from a friend (and answer all questions correctly on the first or second attempt). Another 20% is very sloppy and require 50 attempts or more. It appears that these students start from the quiz, and search through the course text for information to get more and more correct answers, instead of reading the complete text before making the first attempt at the multiple-choice test. So far no disciplinary actions have been taken against these students, but the evaluation quiz will be changed as described in [4 Using Multiple-Choice Tests] to make such abuse less likely.

The assignment for this course is the conversion of a linear document, the *State of the Art Review on Hypermedia Issues And Applications* [Balasubramanian 93] to hypertext. This means dividing the document into fragments of a reasonable size and creating a hierarchical link structure as well as a reasonable number of cross-reference links. (Advice on how to do this can be found throughout the course text.) In order to perform well on this assignment the students need to understand hypertext concepts well, as described in the course text, and they also need to study the document they have to convert to hypertext. (For the assignment a hypertext review is more useful than a document on any random topic.) The students have to encode the hypertext structure by means of HTML, so their hyperdocument can be read using a WWW browser.

In order to grade the assignment, a locally developed tool *www2dot* is used. This tool extracts the link structure from set of HTML documents, and produces a graph, which is visualized by means of the *dot* program from Bell Labs, which is freely available to universities (after signing a license agreement with AT&T) [Koutsofios and North 91] [North and Koutsofios 94]. This graph structure is used as the main basis for the evaluation. It clearly shows the hierarchical structure of the generated document, and the cross reference links (or the lack thereof).

The possibilities for dividing the source document into pieces, and for marking it up using HTML, are almost unlimited. Some students also add special links to indicate the original linear structure, and to point to section and chapter index pages. Although such additions are helpful for the reader of their document, they make it more difficult for the teacher to grasp the real link structure of the hyperdocument the students have generated.

Only one case of fraud has been discovered so far. Two students did most of the work together, resulting in identical structure and HTML markup. They later added some differences, mainly in these extra links to index pages, but not enough differences to not get caught. In [5 Automatic Distribution and Evaluation of Assignment Work] we suggest a way to make it easier to detect cases of fraud.

Although this first edition of the course has proven to be usable, and the effort to put the course on line has paid off, mainly because the final grading of a student project can be done in under 15 minutes, a number of problems and suggestions for solutions have emerged over the past two years. In the subsequent sections we discuss these improvements, not only for this course, but for hypertext courseware in general.

3 Tracking Students' Progress

The non-linear, non-hierarchical structure of the course text makes it difficult for students to keep track of their progress, i.e. to decide how many, and which pages they still have to read. The anonymous nature of World Wide Web access implies that this information is not present in the server log files either. The teacher cannot even deduce from the server log how many students are taking the course. Each student completing the assignment is a surprise to the teacher. (Strictly speaking the teacher can already find out about students when they make their first attempt at the multiple-choice test because these attempts are logged separately, with the students' names.)

Another problem students are faced with is distinguishing pages that belong to the course from other documents on World Wide Web. Throughout the course there are links to other relevant information that is not really a *part of* the course. A student has to watch the URL of the document as displayed by the WWW browser to notice leaving the course text.

We have developed a system to track each student's progress. Students now have to register at the beginning of the course. All pages of the course are delivered by a CGI-script, a program that runs on the WWW server, and that keeps track of a student's actions. The student is tracked by supplying pages with the student's identification embedded in the URL (name) of each page. A list of pages that were read, and one of pages still to be read, is part of a standard banner, added to every page of the course. A student can thus easily find out which pages still have to be read, and the generated list contains active links to these pages. Also, the absence of the banner is a visual indication that the student has followed a link to a page outside the course text. We show an example of the standard banner below [Fig. 1].

Figure 1: *Banner of each page of the hypermedia course.*

Note that the banner has a link to the home page of the university, the home page of the teacher, which has all address and email information, and to the discussion list, described in [6 A Discussion System on World Wide Web]. The banner indicates how many pages there are in the course, and how many have been read.

Another advantage of keeping track of the student's progress is that it becomes possible to provide context-dependent information. When following a link to a page the student is not yet ready to understand, the system could give advice on which pages to read first, or it could add some extra explanation of terms for which the student has not yet read the definition. Also, superfluous links to pages with definitions which the student has already read can be removed from other pages before presenting them. (The current courseware does not yet take advantage of this possibility.)

4 Using Multiple-Choice Tests

The multiple-choice test in the hypermedia course serves only one purpose: to verify whether a student has read, understood and remembered the course text. The level of understanding that is needed to complete the test successfully may come as a surprise to the student. Therefore, small tests at the end of each chapter are added, to test the student's knowledge throughout the course, and to guide the student to additional information or explanations that may be needed.

To use multiple-choice tests as an aid while studying the course text, it is necessary to offer explanations in case the student gives a wrong answer. (This explanation may contain links to course text pages.) Below is an example of a multiple-choice question [Fig. 2], and a possible reply from the courseware [Fig. 3]. The "final" test with 20 questions, at the end of the course, does not provide such informative feedback.

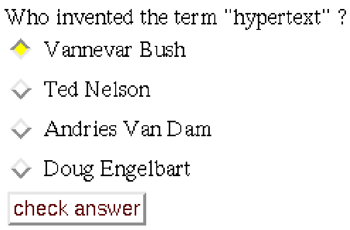

Figure 2: *Example of a multiple-choice question.*

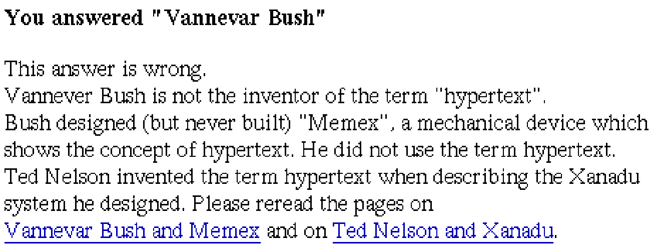

Figure 3: *A possible reply to a wrong answer.*

For each test the teacher can provide more questions and answers than are presented to the student. The software randomly (based on the student id) picks some questions and answers, and presents them in random order. This makes it harder for students to communicate the correct solutions. In the new version of the courseware the final test with 20 questions is taken from a set of 50 questions.

A final factor in the use of these tests is their influence on the student's grade. Experience with some other courses has shown that students tend to put more effort into tests that count towards their final grade than into tests or assignments that do not count. For the final test the number of attempts and the number of correct answers in each try are counted. For the intermediate tests only the correct answers on the first try are taken into account.

5 Automatic Distribution and Evaluation of Assignment Work

The distribution of the assignment, the actual project work, and turning in the completed assignment all have proved to be problematic:

- The document to be converted to hypertext, and the HTML markup language used for the hyperdocument to be generated offer many possibilities for variation, but do not rule out the possibility of students either producing exactly the same result accidentally, or on purpose by copying. By means of a random generator slight variations can be introduced that are easy to find in the completed assignment. Such variations make it easier to detect fraud (by examining the generated document), without having to keep all projects of all students for many years (for comparison with new project work).
- The project work requires many skills, related to authoring hypertext, such as selecting an appropriate page size, creating an appropriate structure of hierarchical and cross-reference links, adding links to the list of bibliographical references, etc. The most common problem with the assignment is that about 15% of the students "forget" to create cross-reference links, thus generating a purely hierarchical link structure. By means of small assignments, after each chapter, the required skills can be learned more or less separately, before starting the final project. Such small assignments, like generating a graph simulating a hyperdocument's structure, can be easily graded automatically. A small assignment can teach students that for this course a purely hierarchical structure is not acceptable. They can then realize they have to create cross-reference links before completing the final project.
- Handing in the project has been the biggest problem in the past. The students in Eindhoven and Antwerp have accounts on a Unix machine. The assignment work can be easily copied from their home directory. Unfortunately, many students fail to set the file permissions right so the teacher can access their work while their fellow students cannot. (Many students block access to everyone, including the teacher.) For both the smaller assignments and the final project, the DReSS system can now be used [De Bra and Aerts 95]. This "Document Repository Service Station" makes it possible to upload documents or even complete directories to the department's Web server by means of a form that is part of the course text.

DReSS is similar to the well known BSCW system [Bentley et al. 95]. (BSCW stands for "Basic Support for Collaborative Work".) DReSS is much simpler than BSCW. It only provides upload and modification facilities, with full authorization facilities, but no version control or workspace overviews. Some newer browsers support file upload. DReSS uses a different upload method, with a small helper application that can be downloaded from the course text, and that is very easy to install. With DReSS students can upload a whole directory at once. (With BSCW or the browser-supported file upload this would not be possible.) The advantage of using DReSS instead of direct file access to the students' directories is that, after uploading an assignment, the work can be checked using CGI-scripts, because it becomes part of the Web server's document space.

6 A Discussion System on World Wide Web

Students from different universities almost never meet in person. A discussion platform enables them to exchange ideas and remarks on the course. Instead of using a mailing list, which would require the students to get acquainted with e-mail, and mailing list archives, we have selected the WWW-based Futplex system [Holtman 95], developed at the Eindhoven University. Futplex has been used in several projects and Web sites already, including the Digital City of Amsterdam.

Futplex lets students talk publicly by posting messages and replying to others' messages, and they can create new pages, representing new lines of discussion. Each time a student enters the Futplex system, an index page to all recently added messages is presented. Futplex offers most of the functionality of Internet Netnews, but does so through the Web. It allows special "chat" pages, for discussions of a very temporary nature, and it also allows "shouting" to quickly get in contact with another student.

7 Conclusions, Advice and Future Work

7.1 Conclusions and Advice

World Wide Web offers interesting possibilities for creating on-line courses. Our experience with four semesters of teaching the course 2L670 on "Hypermedia Structures and Systems" for students from three universities, and with adapting the course for students from two other universities and the Open University, leads to the following conclusions and remarks:

- Not having a paper version of the static portion of the course text is a disadvantage. For most courses it should be possible to offer a printout. (For the hypertext course we explicitly wanted a hypertext document.)
- The anonymous nature of World Wide Web makes it difficult (for the teacher and the students) to find out what each student has read. Our new tracking software solves this problem by generating personalized logging information.
- On World Wide Web everything is connected. We have provided a uniform banner to distinguish pages of the course text from other Web documents.
- Students can be helped by intermediate tests and assignments, in addition to a final test and project. Small projects can easily be evaluated automatically.
- Multiple-choice tests should be varied automatically to prevent copying.
- The DReSS system enables students to hand in their assignment work through the WWW browser, and allows verification of the work by means of CGI-scripts on the WWW server.
- The Futplex system lets students from different universities discuss the course with each other, and with the teacher. It only uses the WWW browser and server, so it does not require setting up email software and mailing lists.

7.2 Future Work

In the future we will use DReSS and Futplex to enable students from different universities to not only discuss the course but work on the project together.

We also plan to upgrade the course text to full HTML-3.2, making use of stylesheets to make the presentation more attractive, and of client-side imagemaps to improve the efficiency and reduce the server load. We also plan to include dynamic illustrations using Java. We wish to keep most of the interaction server-based however, in order to remain able to do logging of each student's progress.

From the list of course topics, given in [2 Initial Experience with On-Line Hypertext Courseware] it is clear that the first three subjects have an introductory nature while the others are more advanced. The hypertext courseware offers links to all of them. There is nothing that prevents students from jumping straight to an advanced chapter, not understanding what is explained, and subsequently getting lost. A new version of the courseware will dynamically adapt the link structure to the student's knowledge state. Throughout the course text links may conditially appear or disappear while the student is gaining knowledge. Also, the content of nodes can be modified, to avoid or use technical terms depending on whether the student has studied their definition or not. An initial experiment with this dynamic version of the hypermedia course is being carried out at the University of Antwerp (URL: http://win-www.uia.ac.be/u/debra/INF706/).

More universities and other institutes for higher education have expressed interest in offering this course to their students. Further automation of the evaluation and grading of assignment work will be needed to cope with the increasing number of students taking the course.

References

[Balasubramanian 93] Balasubramanian, V.: "State of the Art Review on Hypermedia Issues And Applications."; Independent Research Report part of PH.D. Program, Graduate School of Management, Rutgers University, Newark, NJ. Converted to HTML by Denys Duchier. (http://www.isg.sfu.ca/~duchier/misc/hypertext_review/), (1993).

[Bentley et al. 95] Bentley, R., Horstmann, T., Sikkel, K., Trevor, J.: "Supporting collaborative information sharing with the World Wide Web: The BSCW Shared Workspace system"; Fourth International WWW Conference, Boston, WWW Journal, 1 (1995), 63-74. (http://www.w3.org/pub/WWW/Journal/1/bentley.151/paper/151.html)

[De Bra 94] De Bra, P.: "Hypermedia Structures and Systems"; Course 2L670, Eindhoven University of Technology, (1994). (http://www.win.tue.nl/2L670/)

[De Bra and Aerts 95] De Bra, P., Aerts, A.: "Multi-User Publishing in the Web: DReSS, A Document Repository Service Station"; NLUUG Autumn Conference, "Publishing on the World Wide Web", Ede, (1995), 13-26. (http://www.win.tue.nl/~debra/dress/dress-paper.html)

[Holtman 95] Holtman, K.: "The Futplex System"; ERCIM Workshop on CSCW and the Web, GMD, Sankt Augustin, (1995), 25-30.
(http://orgwis.gmd.de/W4G/proceedings/futplex.html)

[Koutsofios and North 91] Koutsofios, E., North, S.C.: "Drawing graphs with dot"; AT&T Bell Labs, Murray Hill, NJ, 59113-910904-08TM, (1991).
(ftp://ftp.research.att.com/dist/drawdag/dotdoc.ps.Z)

[North and Koutsofios 94] North, S.C., Koutsofios, E.: "Applications of Graph Visualization." AT&T Bell Labs, Murray Hill, NJ, (1994).
(ftp://ftp.research.att.com/dist/drawdag/koutsofios-north-GI94.ps.gz)