Teaching through adaptive hypertext on the WWW

De Bra, P.M.E.

Published in:
International Journal of Educational Telecommunications

Published: 01/01/1997

Document Version
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

- A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 07. Dec. 2018
Teaching through Adaptive Hypertext on the Web

Paul M.E. De Bra
Department of Mathematics and Computing Science
Eindhoven University of Technology
PO Box 513, 5600 MB Eindhoven
The Netherlands
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 six different universities, five in the Netherlands and one in Belgium.

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 (although these additional tools could be useful for communicating with the teacher and with other students, and for handing in the final assignment).

In this paper we present the evolution of this course, from a static hypertext document (used in 1994) to a fully adaptive hypertext courseware (used since January 1997). The current edition of the course features automatic evaluation of small tests, a repository for assignment work, a discussion system, complete monitoring of each student's progress, and adaptive content and link structure. We also reflect on the problems students have experienced with each version, and the solutions that have lead to the current courseware and that will lead to future developments.

Keywords:
adaptive hypertext, courseware, document repository, discussion groups, monitoring progress

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, 1994). 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 latest edition of the course is http://wwwis.win.tue.nl/~debra//ijet/journal.htm. The course text is written in HTML-3.2, a standard version of the HyperText Markup Language used on World Wide Web. All current WWW browsers understand most features of this version of HTML.
(Features not yet supported by the latest version of Netscape Navigator (3.01 at the time of writing) are not used. Most noteworthy is the lack of support for style sheets.)

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.

http://wwwis.win.tue.nl/~debra//ijet/journal.html
25-4-2013
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, a Web-based discussion system, and a document and link-structure generator based on adaptive hypertext principles. 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.)

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 a better incentive to study the course via the (hyper) course text, 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 and study 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):

- 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.

Assessment and evaluation

Throughout the course there are small multiple-choice tests to enable students to verify whether they have studied a topic in sufficient detail. For each wrong answer some additional explanation is offered, and one or more links to relevant pages from the course text. At the end of the course
students are presented a larger multiple-choice test, consisting of 15 questions (randomly selected from a larger set). Some of these tests involve selecting one out of four or five possible answers, while others have 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 15 questions before getting a score, and no feedback is provided on the correctness of individual questions. 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.

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. Earlier versions of the course text (up to 1996) used a single multiple-choice test consisting of 20 questions. This made it very easy for students to copy the correct answers from their colleagues. Statistics show that about 40% of the students copied the answers from a friend (or found some other way to obtain the correct answers, for instance by spying on the teacher's source code). Another 20% was very sloppy and required 50 attempts or more. It appears that these students started from the quiz, and searched 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. The new, randomized quiz, described in Section [Using Multiple-Choice Tests], combined with the adaptive hypertext described in Section [Using Adaptive Hypertext on the Web] seem to reduce cheating.

**Course assignment**

The assignment for this course is the conversion of a linear document, the *State of the Art Review on Hypermedia Issues And Applications* (Balasubramanian, 1993) 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 & North, 1991; North & Koutsofios 1994). 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). We plan to offer www2dot and dot to students, as well as some other structure-analysis tools in the future, so they can visualize and analyze link structures before handing in the assignment.

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 Section [Automatic
Distribution and Evaluation of Assignment Work] we suggest a way to make it easier to detect cases of fraud.

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

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 have read and how many and which 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 were faced with in the first course text was 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 had to watch the URL of the document as displayed by the WWW browser to notice leaving the course text.

In order to deal with both these problems, we have developed a system to track each student's progress. Since 1996 students have to register at the beginning of the course. All pages of the course are now delivered by a CGI-script (a program that runs on the WWW server) 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].

---

![Eindhoven University of Technology](http://wwwis.win.tue.nl/~debra/iJet/journal.html)

Paul De Bra read 15/163 nodes. (these read, these still to do.)

**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 Section [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 latest (1997) edition of the courseware takes advantage of this possibility. It is described in Section [Using Adaptive Hypertext on the Web].

In a limited way the individual logfiles can be used to monitor how much time each student spends reading pages from the course text. However, only the time a page is requested is recorded. The time of the next page access does not indicate how long the student has read the page in the case where the student has visited other pages (not belonging to the course text), when the browser session is ended, or when the student simply performs other tasks (e.g. get coffee) while the page is displayed. It turns out to be possible to use a Java applet to record the time stamp when the student leaves a page to visit another document. However, this still only solves part of the problem, because the Java applet which is activated upon exiting a page is not activated by the browser when the student exits the browser or shuts down her computer.

Using Multiple-Choice Tests

The final 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 15 questions, at the end of the course, does not provide such informative feedback.

---

Test for the History of Hypertext and Hypermedia

1. Who invented the term "hypertext"?
   - Vannevar Bush
   - Ted Nelson
   - Andries Van Dam
   - Doug Engelbart

Submit answers

Figure 2: Example of a multiple-choice question.

---

Test for the History of Hypertext and Hypermedia

1. Who invented the term "hypertext"?

   Your answer: Vannevar Bush

   This answer is incorrect.
Additional comment:
Vannevar Bush is not the inventor of the term "hypertext". Bush designed (but never built) "Memex", a mechanical device which shows the concept of hypertext. He did not use the term hypertext. Ted Nelson invented the term hypertext when describing the Xanadu system he designed. Please reread the pages on Vannevar Bush and Memex and on Ted Nelson and Xanadu.

Your score: 0%

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.

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, because the comments on wrong answers contain the correct answer.
(Currently the results for the tests do not count for the student's grade. Only the assignment work counts.)

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 learnt 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. We are currently developing a tool for visualizing and analyzing Web structures created by students. This tool will automatically remove nodes with large numbers of incoming or outgoing links in order to avoid cluttering the graphical representation.
- 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.) Students from other universities (Delft, Tilburg, Twente and the Open University) can upload their work using ftp, or send it as an email.

http://wwwis.win.tue.nl/~debra//ijet/journal.html
For both the smaller assignments and the final project, the DReSS system can be used (De Bra & Aerts, 1995). 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, Horstmann, Sikkel & Trevor, 1995). (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. DReSS is currently available for Unix clients only, but a Windows client is being developed.

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. Until now a mailing list has been used for communication with and between the students. Experience has shown that few students ever use this communication medium, although they are used to email for communication in general.

We have selected the WWW-based Futplex system (Holtman, 1995), developed at the Eindhoven University, as a replacement for the mailing list. 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.

Using Adaptive Hypertext on the Web

A linear document strongly suggest a single (sequential) reading order. Hypertext suggests many different reading orders. This makes it difficult for the author to ensure that each reading order which is made possible by the link structure actually makes sense. Some common problems with hypertext, especially when used for a course text, are:

- A document may contain technical terms, for which a "plain" word can be used in informal introductions, but should ultimately be replaced by the proper term, once the student has read the definition of that term. As an example, in the hypertext course the word page can be used to denote the concept of a node, but once the student has read the definition of nodes the word page should no longer be used. However, different navigation paths may lead to the same page, some of which include a visit to the definition page and some not.
- Some links to pages should be prominently present as long as the student has not read these pages, but once they have been read links to these pages may be omitted. Again, these conditions are difficult to predict when authoring the hypertext.
- Some pages serve as a table of contents or index, leading to parts of the (hyper)document some of which the student may not be ready to read. The welcome page of the hypertext
course is a prime example. Although no strict reading order is needed, links to advanced topics should remain inaccessible until most of the introductory topics have been studied.

Figure 4 below shows (part of) the static welcome page of the hypertext course. It advises the student to first read an instructions page, and then continue with the different chapters, but does not enforce this order. (The banner is omitted in the figure.)

Hypermedia structures and systems

Welcome to course 2L670 at the Eindhoven University of Technology.

If you are just beginning to browse through this course, you should first read the instructions. These will explain how to use this course text, together with a graphical World Wide Web browser such as the Netscape Navigator, Microsoft Internet Explorer, or NCSA Mosaic. In order to get to the instructions you must click (the left or only mouse button) on the phrase "the instructions".

This course contains the following (not necessarily disjoint) parts:

- Introduction (it is advised to read this before the other items)
- Definition of hypertext and hypermedia
- The history of hypertext and hypermedia
- The architecture of hypertext systems
- Navigation (and browsing semantics) in hypertext
- Information Retrieval using hypertext
- Writing hypertext
- Distribution and Concurrency issues
- The Future of Hypertext and Hypermedia
- Assignment for this course

Figure 4. Static welcome page of the hypermedia course.

In 1996 we introduced an adaptive version of the course text, which only enabled the link to the instructions page first, and then the links to the first three chapters. Only after reading some of each of these chapters the remaining topics became accessible. Starting 1997 we have added adaptive content to the conditional link structure. The first time a student visits the welcome page it looks like Figure 5:

Hypermedia structures and systems

Welcome to course 2L670 at the Eindhoven University of Technology.

Since you are just beginning to browse through this course, you should first read the instructions. These will explain how to use this course text, together with a graphical World Wide Web browser such as the Netscape Navigator, Microsoft Internet Explorer, or NCSA Mosaic. In order to get to the instructions you must click (the left or only mouse button) on the phrase "the instructions".

⚠️ You cannot start reading the dynamic course text until you have read these instructions.

The items below indicate (not necessarily disjoint) parts of the course text, which will become accessible after you have read the instructions.

http://wwwis.win.tue.nl/~debra//ijet/journal.html
After reading the instructions the advise to read the instructions first is no longer needed. The welcome page then becomes like Figure 6 below:

Hypermedia structures and systems

Welcome to course 2L670 at the Eindhoven University of Technology.

This course contains the following (not necessarily disjoint) parts:

- **Introduction** (it is advised to read this before the other items)
- **Definition of hypertext and hypermedia**
- **The history of hypertext and hypermedia**
- The architecture of hypertext systems
- Navigation (and browsing semantics) in hypertext
- Information Retrieval using hypertext
- Writing hypertext
- Distribution and Concurrency issues
- The Future of Hypertext and Hypermedia
- Assignment for this course

**Figure 6.** Welcome page after reading the instructions.

Finally, after reading through the first three chapters the remainder of the course text becomes available (see Figure 7 below):

Hypermedia structures and systems

Welcome to course 2L670 at the Eindhoven University of Technology.

This course contains the following (not necessarily disjoint) parts:

- **Introduction** (it is advised to read this before the other items)
Adaptive hypertext is being used in educational environments and in World Wide Web at other institutes as well. A renown example is the ELM-ART (Lisp) tutoring system developed at the University of Trier ([Brusilovsky, Scharz & Weber, 1996a), and the subsequent InterBook tool ([Brusilovsky, Schwarz & Weber, 1996b) from CMU. ELM-ART and Interbook use link annotation to advise students to follow or avoid certain links. This has the advantage that the textual contents of each page remains always the same, creating a "stable" image for the students. However, as the ELM-ART course shows this can easily lead to a suggested linear reading order, which means that browsing through the document in any other order can only be done by going against the advice offered by the system.

**Conclusions, Advice and Future Work**

**Conclusions and Advice**

World Wide Web offers interesting possibilities for creating on-line courses. Our experience with six semesters of teaching the course 2L670 on "Hypermedia Structures and Systems" for students from six universities 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.
- Students often fail to hand in assignments using non-WWW technology (like nfs, email or ftp). 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.
- Students do not easily start group-conversations through a mailing list. 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. Different projects have already demonstrated that group-discussions are generated spontaneously through Futplex and similar systems.

**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 assignment-project together.

We also plan to upgrade the course text to full-HTML-3.2, to make the presentation more attractive by means of style sheets and Java applets.

- One of the interesting hypertext concepts still missing from the course text is the ability to mark links differently depending on their semantics. We hope to be able to use different link colors to indicate different types of links.
- Another interesting possibility is to provide a fish-eye view of the course text's structure by means of a Java applet. Students in Antwerp are currently developing such applets.
- We will also move some functionality to the browser by means of client-side imagemaps, to reduce the server load.

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, V., (1993). 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/)


