Interactive, Animated Hypertextbooks for the Web: Ongoing Work[*]

[Extended Abstract]

Rockford J. Ross
Computer Science Department
Montana State University
Bozeman, Montana, USA
ross@cs.montana.edu

ABSTRACT

A *hypertextbook* is a teaching and learning resource designed for computer presentation that is intended to augment or supplant a traditional textbook as the primary teaching and learning resource for a course. This extended abstract describes ongoing investigations by the author into the feasibility and effectiveness of hypertextbooks for the web.

Beyond the static text and graphics presentations found in traditional textbooks, hypertextbooks include

- different paths through the material for different learning needs by way of hyperlinks
- active learning, animated visualizations of concepts—incorporated as programmed applets
- integrated audio and video

Arguably, the most important of these three enhancements is the active learning component. Because a hypertextbook runs on a computer, programmed visualizations can be included that encourage active learning on the part of the learner.

One of the biggest problems addressed by hypertextbooks is the widely recognized lack of use of visualization systems in education. If a hypertextbook is adopted as the main teaching and learning resource for a class, use of seamlessly interwoven active learning visualizations will occur as a matter of course.

1. INTRODUCTION

A hypertextbook project has been underway in the Webworks Laboratory at Montana State University for a number of years now. This project arose as a natural extension of work already being done by the project director in animated visualization systems for computer science education. The idea of integrating educational visualization systems into a hypertextbook that would augment, or indeed supplant, traditional textbooks presented itself as an appealing possible solution to one of the most perplexing problems facing educational visualization researchers: the lack of actual use of visualization systems in the classroom. A convincing argument was made that if a comprehensive, seamless teaching and learning resource could be constructed (i.e., a hypertextbook) that became the primary teaching and learning resource for a particular class, students and instructors alike would quite naturally use all aspects of the resource, including any embedded visualizations.

This paper, as an extended abstract, outlines some of the key points learned during our work on two hypertextbooks for the web, one on the theory of computation and one on biofilm engineering. Although online courses and electronic books that incorporate some visualizations have been around for a while, no other efforts that we know of are as comprehensive as the work we are doing, particularly with respect to active learning visualization systems. In the remainder of this paper, we outline some of the major issues facing hypertextbook authors. We have reported on some of these issues elsewhere, but the presentation for PVW-2002 will include live demonstrations of our most recent work, presentations of preliminary evaluations of this work, and a discussion of the problems and solutions we have encountered in our quest to construct hypertextbooks. For this extended abstract we leave out all references, but will provide a comprehensive list at the workshop.

2. HYPERTEXTBOOK DESIGN ISSUES

In this section we briefly highlight the major issues we have encountered in our work on hypertextbooks that feature active learning visualizations. The intent is not to discuss these issues in detail here, but to provide a framework for future discussions and a presentation at PVW-2002.

[*]This work is being supported in part by two grants from the National Science Foundation, numbers 0089397 and 0088728.

2.1 Platform

Which platforms should the hypertextbook run on? We chose to design our hypertextbooks for the web. This has the appealing advantage of platform independence. Of course, platform independence is a somewhat elusive objective, especially when one is working with the latest versions of the Java Virtual Machine (JVM), which not all browsers for all platforms implement at the same time. One is also restricted to features (e.g., HTML) that the standard browsers can process. Finally, the wars between Microsoft and the rest of the world conspire to impede progress on some important platform independence issues.

2.2 Java Applets for Active Learning Visualization Modules

Once the web is chosen as the platform for delivery of a hypertextbook, one can incorporate active learning visualization models into the hypertextbook in the form of Java applets. For our hypertextbook projects we currently have the following active learning visualization models for use as standalone applications or for seamless integration as applets in computer science hypertextbooks:

- A program animator (which runs Pascal programs in visual fashion)
- A finite state automaton animator
- A regular expression animator
- A context free grammar animator
- An LL(1) table construction animator

For use in hypertextbooks relating to biofilm engineering, we have:

- A fugacity model animation
- A biofilm accumulation model animation

These applets all have a number of features that encourage active learning on the part of users. For example, the finite state automaton applet mentioned above allows students to construct their own automaton from scratch (or to modify an existing one), run that automaton on arbitrary inputs, and then compare their result with a hidden, correct automaton. Feedback is provided to aid the learner in correcting detected mistakes. Active learning features like these contribute to the success of the applets as learning tools. A number of these applets will be discussed in detail at the workshop.

2.3 Seamless Integration of Visualization Components

One of the primary issues we have been forced to confront in the design of visualizations for use in a hypertextbook is the necessity of integrating visualization applets seamlessly into the fabric of the hypertextbook. We have discovered that in order for visualizations to be useful and used in a hypertextbook, they must appear as a naturally integrated part of the whole. It is distracting, confusing, and counterproductive to the goal of encouraging students and instructors to use visualizations if these visualizations must be opened in windows outside the normal flow of the text. It isnt clear in such situations, for example, when to close a newly opened window or how to return to the main textbook presentation. Furthermore, if the presentation of visualizations becomes too disjointed through the opening and closing of auxiliary windows, the visualizations might simply be ignored.

We have tackled this issue by constructing the visualization applets so that they can be embedded directly within the text of the hypertextbook at arbitrary locations as preconfigured examples for students to follow or as preconfigured exercises that students must solve. These same applets can also be included as standalone models in an appendix for arbitrary student and instructor use.

2.4 Cooperation Between Visualizations

Not only must visualizations be interwoven seamlessly into the fabric of a hypertextbook, but they often must also be constructed to interact with each other. For example, in the hypertextbook on the theory of computing that we are building, the finite state automaton applet, the regular expression applet, and the context free grammar applet must all be made to work with each other, so that, for example, animations can be included that demonstrate conversion algorithms from one form to another (e.g., from finite state automata to regular expressions).

2.5 Applet Authoring Tools

It is important to provide a prospective hypertextbook author with tools for configuring and incorporating applets into a hypertextbook as desired. For example, if an exercise that requires a learner to construct a finite state automaton is to be included in a hypertextbook at a certain point, the author must be able to configure the finite state automaton applet to be invoked at that point with

- a description of the exercise
- a blank working area
- a correct finite state automaton that is hidden from the student (for comparison with the version constructed by the student)

In other words, an author must not be required to be cognizant of the implementation details of any particular applet in order to embed that applet in appropriate ways and places in a hypertextbook.

2.6 Representation and Presentation Independence of Visualizations

One key issue that took us a while to uncover is that to be most effective and adaptable, visualizations designed for the web must have two components that are independent of each other. The first component is the *internal representation* of the object being visualized, and the second is the *visualization* of that object. In other words, the internal representation of the object should not dictate how the object is to be visualized. For example, a description of a finite state automaton should be given in a form that provides all of the components necessary to define the automaton in generic fashion. The visualization applet, then, must use this generic definition to present the automaton in an appropriate visual form (e.g., as a directed labeled graph, or as a table). This independence of representation from presentation provides great flexibility in the forms a graphical display of an object can take. It also makes cooperation between different visualization applets much easier to accomplish (e.g., in demonstrating the conversion of a regular expression to a finite state automaton) for obvious reasons.

We are taking this idea of independence of representation from presentation one step further by requiring that all of the internal representations of objects for visualization have a standard definition in the eXtensible Markup Language (XML). This provides the added bonus of having an internal description for visualization objects that is in a standard form for the web.

2.7 Other Considerations

There are a number of issues other than the incorporation of visualization applets into a hypertextbook of which a prospective author needs to be aware. A few are listed below:

- The hypertextbook should have an attractive, inviting portal (cover).
- The hypertextbook should appear to the learner as a single, integrated component, not a collection of disorganized pieces.
- The hypertextbook should have a polished, professional look and feel that gives the learner confidence in its use. A consistent design theme and a consistent appearance to all of the pages and the linking structure are important.
- The use of distracting bells and whistles should be avoided.
- An organization of the material by way of hyperlinks that leads learners through the hypertextbook in different ways based on different learning levels or styles should be provided.
- Audio and video should be incorporated where it would be effective.

In other words, a hypertextbook should have a professional look and should make use of the unique capabilities offered through hyperlinking to reach out to students with varying learning needs.

3. PROBLEMS

No endeavor this large is without problems. One large hurdle is the amount of work needed to write a hypertextbook. The effort entailed in writing a traditional textbook is plenty already. A hypertextbook entails the additional work of providing an organization that caters to different learning needs, of (possibly) recording and integrating audio and video clips appropriately, and of designing and including effective active learning applets. This additional work is substantial. Consider just the development of visualization applets. Even passive learning applets, in which the learner participates only as a viewer, are time-consuming to design. Consider the effort involved when such applets must be constructed to support active learning, to be adaptable to various situations, to be integrated into a hypertextbook, to interact with different but related applets, and to come supplied with authoring tools.

4. EVALUATION

Do hypertextbooks work? Phrased this way, the question is perhaps too broad. More specifically, we want to know the answers to some of the following questions:

- Do students learn better in traditional learning environments with hypertextbooks that incorporate active learning modules?
- Do students learn faster in traditional learning environments with such hypertextbooks?
- Can students learn better and/or faster in non-traditional learning environments (e.g., in distance learning situations) when hypertextbooks that incorporate active learning modules are available for the subject?
- Do students perceive that their opportunities to learn are enhanced when such hypertextbooks are used?
- Do students enjoy learning more when such hypertextbooks are used as opposed to traditional textbooks?

We would be happy to determine, for example, that students learned at least as well with hypertextbooks as with traditional textbooks if at the same time they were more satisfied with the learning process and therefore that more underrepresented classes of students (e.g., women) were encouraged to complete a degree in computer science.

Our hypertextbook projects have just begun to reach a point where they can undergo preliminary evaluation, some of which is underway at the time of this writing.

5. SUMMARY

In this extended abstract, we have presented an outline of our investigations into the feasibility and effectiveness of hypertextbooks, particularly those that incorporate active learning visualization applets. This outline will serve as the basis for a presentation and discussion at PVW-2002. At that time live demonstrations of the hypertextbook project will be presented, the results of our current evaluations discussed, and references provided.