11. Marshall, C., and Novick, D., Conversational effectiveness in multimedia communications, *Information Technology & People* **8** (in press).

funds provided by NASA. U S WEST provided ATM support and conducted the videotaping of student collaboration used in evaluating the course.

References

1. CERN (European Laboratory for Particle Physics), “World-Wide Web Home”, URL: *http://info.cern.ch/* (undated).

2. NCSA (National Center for Supercomputing Applications), “A Beginner's Guide to HTML,” URL: *http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html* (undated).

3. NCSA (National Center for Supercomputing Applications), “NCSA Mosaic Home Page,” URL: *http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/NCSAMosaicHome.html* (undated).

4. Jones, T., and Chiappetta, L., “You too can learn Italian!”, URL: *http://www.willamette.edu/~tjones/languages/Italian/Italian-lesson.html* (undated).

5. Marriott, A., “Welcome to X Windows On-Line,” URL: *http://www.cs.curtin.edu.au/units/cg252-502/src/notes/html* (November 12, 1994).

6. Globewide Network Academy, “Welcome to the Globewide Network Academy,” URL: *http://uu-gna.mit.edu:8001/uu-gna/index.html* (undated).

7. Fickas, S., “Highlights of a Web-based Course on Software Engineering,” *http://www.cs.uoregon.edu/~fickas/nero-course/nero-wrapup.html* (undated).

8. Norman, D., Cognitive engineering, in Norman, D., and Draper, S. (Eds.), *User-centered system design*, Lawrence Erlbaum Associates, Hillsdale, NJ (1986).

9. Wharton, C., Bradford, J., Jeffries, R., and Franzke, M., Applying cognitive walkthroughs to more complex user interfaces: experiences, issues, and recommendations, *CHI'92*, 381-388 (1992).

10. Kirwan, B., and Ainsworth, L., *A guide to task analysis*. Taylor & Francis, London (1992).

issues—the interim resolution has proven to be a pragmatic one. In practice, the modules are being reused for new courses only by their original authors and most of the modules are freely accessible via the Web. However, these interim solutions beg the difficult question of making available resources for developing new Web-based courses: If, as our experience suggests, the cost of developing a networked hypermedia course is a significant multiple of the cost of a "flat" course, the developers will have to establish ownership rights and access restrictions in order to recoup their investment. Moreover, universities should not expect that faculty will be able to prepare hypermedia courses at the same level of effort as a regular course. Substantial time and effort are required even to convert existing notes into HTML; and to go beyond flat text takes much longer.

Despite these uncertainties and costs, this project created a prototype cross-state hypermedia course in software engineering for software engineering. In doing so, we broke trail for distance educators by discovering a number of problems and techniques. We have identified unresolved problems so that they might be avoided in the future, and have described technical solutions, including forms-based templates for exercise, that we developed for problems that we were able to resolve. Our newer courses, now under development, have the benefit of this experience and of a catalogue of course components that we produced for the course. We explored collaboration tools for both course development and course delivery.

Acknowledgments—The authors thank their collaborators on this project, Warren Harrison of Portland State University and Bruce D'Ambrosio of Oregon State University. Graduate students participating in the project included Kartik Mithal, Ken Maupin, Ira Smith, David Tiktin, and Ted Kirkpatrick. This research was supported by the Oregon Joint Graduate Schools of Engineering and the Network for Engineering Research in Oregon (NERO) with

reasonable amount of detail. Consequently, the audiovisual protocols for the usability-testing module had to be presented with our own “slide show” technique, which was itself unsatisfactory for other reasons.

Other technical obstacles related more directly to the hardware. The interface cards for the ATM network turned out to remain in Beta release through the life of the project, thus attenuating our use of the NERO network. Internet response time for most of the course materials was satisfactory, but we believe that access to the ATM network would have improved usability of the video and audio materials. As researchers in collaborative multimedia, we had become accustomed to working with marginally available or unstable hardware and software; after all, these were the usually the very objects of our research. But in this project we learned that, as educators, we should count on neither (a) hardware in Beta or about to be released, nor (b) software “everybody's working on.” Consequently, the least adventurous modules turned out to be the most immediately reusable.

4. Conclusion

With the benefit of these lessons, participants in this project have gone onto other courses for Web-based delivery. Courses in preparation include C++ programming (at PSU) and object-oriented programming (at OGI). Both new courses will be more conservative in their use of modalities, relying entirely on hypertext and forms rather than video and audio.

Our experience with this course also raised a number of organizational issues. Who owns and controls access to material on the Web? How should faculty be compensated for materials used later or by others? Certainly, release forms should be provided to students so that course projects can be added to the common fund of software. For course materials prepared by faculty—in the absence of clear law, interinstitutional agreements or faculty contracts on these

(1) Our prototype was a good vehicle for exploring UI evaluation issues. The Web is simple enough to allow students to make UI modifications quickly, and complex enough to raise a rich set of UI issues.

(2) Students were able to move routinely between prototype and rationale.

(3) The course material could be studied successfully by two students using commercial teleconferencing tools (InPerson in this case, and ShowMe in a subsequent session).

A post-session questionnaire given to the students indicated high satisfaction with both the course material and the collaboration made available through teleconferencing.

3.2 Technical obstacles

The technical obstacles proved significant for us. As early adopters of NMC technology, we found that means of delivery turned out not to be available as anticipated. Some of these problems have since been resolved through the release of newer browsers for the World-Wide Web, such as NetScape and later versions of Mosaic. Other problems remain. Most of these problems arose in part because of heterogeneity among delivery platforms. We needed to use hypermedia forms for both the prototype registration program and the exercise package. But only the Unix version of Mosaic supported forms; the Macintosh version did not. We also needed to deliver video to the users' desktops in order to show (a) design sessions and (b) usability protocols. As we began our project, only the Macintosh, with QuickTime, supported synchronized audio and video. We anticipated, with the rapid development of the Web, that an X-based synchronized audiovisual viewer would soon become available; this proved not to be the case. Eventually an X-based viewer for QuickTime was released but we found that QuickTime's resolution was inadequate for images of computer screens with any

Web makes accessing NMC documents appear relatively seamless (especially with high-speed networks), available collaboration tools such as Mbone, email and the Web itself were not adequate for distributed collaboration on such a technically innovative enterprise. This failure also arose in part from the heterogeneous computing environments at the four participating schools, which had a changing mix of Sun, HP and Macintosh workstations, each with different collaboration tools. Development of the course required interaction covering everything from brainstorming organization and design details of the course to resolving technical obstacles. Consequently, the press of obligations to deliver the course resulted in too-frequent travel to from Portland and Eugene to Corvallis for meetings.

The delivery of the course as a hypermedia product contributed to the need for collaboration. We learned that preparation of hypermedia course involves significant extra effort as compared with televised courses or traditional CAI. We judge the effort in producing a hypermedia-based self-study course to be comparable to the effort required in writing a textbook. However, we note that the result of developing courses in hypermedia is a set of teaching materials that are much more flexible in delivery than traditional telecourse or CAI materials. The relative effectiveness of collaborative hypermedia remains an open issue that is the subject of ongoing research (see, e.g., [11]).

3.1 Course evaluation

We evaluated the course materials by simultaneously videotaping two students going through the course material at a distance using a teleconferencing system. From our analysis of these videotapes, we concluded that:

was the source of the transcript. An extended hypermedia review of the IPA module is available at *http://www.cs.uoregon.edu/~fickas/nero-course/nero-wrap-proto.html.*

The UI-evaluation module was developed at the Oregon Graduate Institute. This module included eleven HTML documents covering the role of testing in the design method, an introduction to usability (following, e.g., [8]), analytical evaluation methodologies such as interface walkthrough (following [9]), and empirical evaluation methodologies such as protocol analysis [10]. These materials, presented in hypertext, contained about 50 audio annotations that enabled the user, in effect, to create a customized lecture. Three exercises were developed using forms in HTML. The exercises covered interface walkthroughs and protocol-based usability testing. The exercises, with a common introduction, were produced in ten linked HTML documents. The basic course materials and the exercises were supported by six video clips formatted for the “slide show” utility.

One early problem we faced was that the four contributing schools developed modules that lacked a consistent look and feel. This heterogeneity was inherent in our project’s objective of trying different approaches to multimedia-based distance learning. To cope with the diversity of our components, we produced a homogeneous set of Web pages that served as “wrappers” to tie the modules together. The links in the syllabus pointed to the modules' wrappers, which provided a more consistent look and feel to the course. The wrappers, in turn, contained links to the specific content of the modules. The wrappers also facilitated easier reuse of the modules by enforcing a fairly simple single entry point for each topic.

3. Lessons Learned

As a result of this project, we learned that distributed collaborative delivery of a course is significantly easier than similarly situated development of the course. While the World-Wide

course materials themselves were distributed across multiple sites—thus testing the NERO network and our collaboration tools.

Students accessed the course through a central HTML document at the University of Oregon. This document contained links to a general description of the course and a syllabus, which in turn contained links to modules across the state. The course was delivered for initial use in the fall of 1994.

Our approach enabled, and in fact led to, exploration of a variety of styles of educational hypermedia, including text, forms, video and audio. The modules at Portland State University that introduced use and authoring of HTML documents were produced in relatively straightforward hypertext. Modules at the University of Oregon included QuickTime video clips of design sessions, plus complex forms that implemented a working prototype system for course registration. Modules at Oregon Graduate Institute included extensive use of audio annotation, a forms-based package of exercises, and an experimental "slide show" viewer that presented sequences of images with a loosely synchronized soundtrack.

The course modules dealing with requirements engineering were developed at the University of Oregon. QuickTime clips were edited using Adobe Premiere. Text transcripts were also made of the entire video tape and put onto HTML pages. The video clips were indexed in two ways. First, we developed IPA (Issue/Position/Argument) Web pages around design issues that arose. We used these pages to link a position (i.e., a design choice) with arguments (i.e., video clips). With this style, the user could move from an IPA page "forward" to the actual implementation (by clicking on an HTML link) or move "backward" from the implementation (by clicking on hypertext links we embedded in the application) to the IPA page. Second, we added HTML links to the text transcript to enable the user to see the actual video session that

associated course material to be mastered by students, such as requirements acquisition, prototyping on the Web, and UI evaluation.

We found that self-study courses can be a mixed bag, often dull and plodding. We aimed to counter this trend in our proposed NMC self-study course by:

(1) Capturing the course lectures in text, audio and video.

(2) Directing a set of student "study projects" during the course. These projects would be purpose-built to show off some feature of NERO. Student design meetings would be captured in text, audio and video.

(3) Integrating lectures with projects. In particular, we would use a hypertext framework to allow students to come at a topic from either theory or practice, and move back and forth between each with ease.

(4) Using a hypermedia framework to tie project software to (a) audio and video transcripts of design meetings and (b) audio and video transcripts of lectures.

We planned to package all of the above in a hypermedia-based course-delivery system that used NERO-capable workstations: students would have the course delivered to them at their own pace and at their own station. The projects of the course would be freely available as platforms on which students can build. Especially, we expected that the course materials would be an evolving entity, with new projects being added as new students "take" the course.

As it was developed collaboratively at four geographically separated universities, we designed the course for delivery via high-speed networked hypermedia. The authors and their collaborators at Portland State University and Oregon State University built the course as a set of distributed modules. That is, not only was the course delivered to different sites but the

involved capture and design of a course to take advantage of the NERO network to train a new generation of NMC researchers.

Contemporaneous efforts by other investigators and educators addressed complementary efforts but had not as yet produced a distributed, hypermedia-based for a comprehensive course available completely over the Web. Some of the materials available now include individual lessons for beginning Italian [4] and course notes for computer graphics programming [5]. The Global Network Academy [6] provides pointers to Web-based courses offered by a loose consortium of institutions. An extensive list of Web-based educational course materials can be found at *http://wwwhost.cc.utexas.edu/world/instruction/index.html.*

2. The Course

The principal content of our course [7] was an integration of requirements engineering with user interface (UI) evaluation. The educational objectives of our course grew from elements that we believe would lead to effective software engineering for network-based multimedia systems. These elements included understanding and adoption of user-center software development methods for requirements analysis, prototyping and testing; systematic linking of system features to the design rationale; basic skills in construction of Web documents with HTML; and distributed team-based collaboration.

In particular, the methods taught in the course should enable users doing UI evaluation to view the rationale for a specific UI component by viewing the relevant system requirements. To achieve this, our students had to complete four course tasks: (1) capture system requirements in the form of video interviews and scans of hardcopy material, (2) build a prototype system and UI from this information, (3) link the prototype with rationale (the video and graphic material), and (4) run UI evaluation experiments with the prototype+rationale. Each of these four tasks had

that combines documents prepared in Hypertext Markup Language (HTML) [2] with an extensible set of multimedia resources. We would build a useful course in software engineering in hypermedia—using hypermedia—by pooling Oregon's talent, writing the course once for recurrent delivery and providing practical training in hypermedia development.

A corollary goal was to “break trail” for educators wishing to provide distance education in Oregon. Thus our project planned to implement and evaluate different styles of instruction over the Web, to find technical roadblocks, and to explore modes for assigning and evaluating exercises. As a result of this effort, we anticipated building a catalogue of course components that could be reused as units in future courses.

Finally, we planned to explore collaboration tools. In part, the use of collaboration tools arose out of anticipated work by students taking the course. But more fundamentally, the very development of the course would be a test of distance collaboration: The downstate schools were approximately a hundred miles away from the Portland-area schools. The schools were linked by the Internet and with new service provided by NERO, which is an ATM-based network that links the University of Oregon, Oregon State University, Portland State University, Oregon Graduate Institute and Oregon Health Sciences University.

Our interests centered on capture and delivery of NMC expertise; we wanted to broaden the set of researchers working on NERO-based NMC projects in Oregon. For this reason, we chose not to spend our resources implementing yet another hypermedia system. Accordingly, we planned to use a freely available, off-the-shelf system such as Mosaic [3], then the most popular browser for the WWW. Mosaic is a cross-platform program developed and distributed by NCSA, running in X-based Unix, Macintosh and PC-Windows environments, which by the fall of 1993 had the basic capabilities to deliver hypermedia to the desktop. So our major effort

networked multimedia communications (NMC): a set of faculty with expertise in NMC, an untapped set of graduate students with interests in NMC, and a new high-capacity, high-performance network called the Network for Education and Research in Oregon (NERO) that links faculty and students among the four schools. We faced the question of how best to marshal these three resources to obtain sustainable results in networked multimedia communications. In formulating an answer, we identified three goals:

(1) To tap into the enormous potential of directing graduate students towards NMC, NERO-based projects and theses. This would give us sustained research well into the future.

(2) To use faculty time effectively. If we wish to harness graduate student power, we needed to provide educational foundations in NMC. Traditionally, this would be done through a course taught every year. However, it is unrealistic to expect that the relevant faculty in the state can be brought together every year to teach the necessary foundations course, even using broadcast technology.

(3) To provide direction for development of the NERO network as early as possible. Prompt use of NERO for a heterogeneous set of collaborative tasks would disclose the system's strengths and weaknesses.

Accordingly, we set out to achieve these goals by designing and teaching an NMC course *once*, bringing all relevant faculty together for the event. We hoped at the same time, to *capture* the course through multimedia but, as we will discuss below, this was only partly successful. The main idea was to package the course as a hypermedia, NERO-based resource available to any and all faculty and students on the NERO network for self-study from their workstation via the World-Wide Web [1], a network-based standard for hypermedia documents

Collaborative networked hypermedia education: Lessons from the NERO project

David G. Novick
Oregon Graduate Institute

Stephen Fickas
University of Oregon

ABSTRACT

This paper describes the genesis, development and delivery of a course in software engineering for hypermedia. The course was developed collaboratively at four geographically separated universities for delivery via high-speed networked hypermedia. The authors and their collaborators at Portland State University and Oregon State University built the course as a set of distributed modules. This enabled (1) exploration of a variety of styles of educational hypermedia, including text, forms and audio modalities, and (2) rapid reuse. Novel course materials included forms-based prototypes, forms-based exercises, audio-annotated notes, and a simple audiovisual "slide show" utility for delivery of protocol data for use by students. As a result of this project, we learned that distributed collaborative delivery of a course is significantly easier than similarly situated development of the course. The preparation of a hypermedia course involves significantly extra effort—comparable to writing a textbook—but produces teaching materials that are much more flexibly deliverable than traditional CAI materials.

1. INTRODUCTION

A major focus of interest in science and engineering is tool support for collaborative research. Emerging from this field is the notion that multimedia communications will have a large role to play in future collaboration systems. By the fall of 1993, the graduate departments of computer science in Oregon (Oregon Graduate Institute, University of Oregon, Oregon State University, and Portland State University) had three resources crucial to the success of

Collaborative Networked Hypermedia Education: Lessons from the NERO Project

David G. Novick
Oregon Graduate Institute

Stephen Fickas
University of Oregon

Contact information:

David G. Novick (author for correspondence)
Department of Computer Science and Engineering
Oregon Graduate Institute of Science & Technology
P.O. Box 91000 Portland, OR 97291-1000
voice: (503) 690-1156
fax: (503) 690-1334
email: novick@cse.ogi.edu

Stephen Fickas
Department of Computer and Information Science
University of Oregon Eugene, OR 97405
voice: (503) 346-3964
fax: (503) 346-5373
email: fickas@cs.uoregon.edu