Building an Electronic Learning Community: From Design to Implementation

Anne Rose*, Wei Ding, Gary Marchionini*, Josephus Beale, Jr., Victor Nolet

Digital Library Research Group
Human-Computer Interaction Laboratory*
University of Maryland, College Park, MD 20742
http://www.learn.umd.edu

{arose, weid, march, melc, vnolet}@learn.umd.edu

ABSTRACT

The University of Maryland at College Park in cooperation with Baltimore City Public Schools and several partners is working to build an electronic learning community that provides teachers with multimedia resources that are linked to outcome-oriented curriculum guidelines. The initial resource library contains over 1000 videos, texts, images, web sites, and instructional modules. Using the current system, teachers can explore and search the resource library, create and present instructional modules in their classrooms, and communicate with other teachers in the community. This paper discusses the iterative design process and the results of informal usability testing. Lessons learned are also presented for developers.

Keywords

digital libraries, learning communities, dynamic query, java, video

INTRODUCTION

Educators have become increasingly interested in understanding the role technology can play in the classroom. One of the goals of the U.S. Department of Education's Technology Challenge Grant program is to demonstrate how public school-university-business consortia can cooperate to improve educational outcomes through enhanced technology. One of the projects funded, the Baltimore Learning Community (BLC), is a collaborative effort among the University of Maryland at College Park (UMCP), Johns Hopkins University (JHU), and the Baltimore City Public Schools (BCPS). The goal of this five year demonstration project is to create a learning community through the use of high quality educational resources and high-speed networking. The JHU component (The Baltimore Initiative) focuses on interactive video conferencing. The goal of the UMCP component (Maryland Electronic Learning Community) is to provide teachers with multimedia resources and link their use to outcome-oriented curriculum guidelines [2][3]. This paper focuses on the technology component of the Maryland Electronic Learning Community.

For the past two years, the University of Maryland has been working with a team of middle school science and social studies teachers from three Baltimore City Public Schools and several partners including Discovery Communications, Inc., the U.S. Archives, the Space Telescope Institute and Apple Computer to build an electronic learning community. From a technology standpoint, this has thus far involved installing four or more computers in each of 12 classrooms; linking those machines and teachers at home to the Internet; creating a digital library of more than 1000 multimedia educational resources; and developing a prototype system that allows teachers to explore and search the resource library, build and present instructional modules, and exchange ideas with other teachers in the community (via email and video conferencing).

Classroom Setup

Apple Computer provided 40 Macintosh PowerPCs for use in the classrooms. Each classroom is equipped with 3-4 Macintosh 5200 computers capable of displaying MPEG video and a teacher machine, a Macintosh 5400 computer with a 4gb hard drive. Each classroom also has two 27 inch "s-video" monitors that can be used for classroom presentation. Currently, the machines in each school are networked via a ethernet LAN with a dial-up connection to an Internet service provider. High-speed networking solutions, such as fiber or hybrid wired/wireless solutions, are being investigated for the full implementation years of the project. Since teachers do much of their planning at home, they have also been provided with Internet access at home and when necessary, with modems and computers.

Digital Library

The resource library is a collection of videos, texts, images, web sites, and instructional modules that cover a wide range of social studies and science topics. These resources have been collected from several sources including Discovery Communications, Inc. which is providing up to 100 hours of digitized video, the U.S. Archives, and the Space Telescope Institute. The selection of which resources to include in the community has been driven primarily by the teachers and their curricula. Teachers specifically requested several hundred images from the U.S. Archives and provided input to Discovery on what video programming to include. Discovery clears digitization rights and provides MPEG files based on segmentation done by UMCP staff. As of the Fall 1997, six hours of video were included in the database and eleven additional hours were in the segmentation/indexing process. Selected websites are also indexed after they are checked for student suitability and publisher stability. At that time, the library consisted of approximately 1000 resources that had been digitized, indexed, and added to the database.

Indexing involves mapping a resource to one or more topic-outcome pairs and there are more than 2500 pairs mapped to the 1000 resources. We chose to index by topic-outcome pairs because teachers currently select and organize materials by topic (e.g., space, ecology) but they organize their lesson plans according to the Maryland Statewide Performance Assessment Program (MSPAP) outcomes and a specific goal of the project is to help teachers develop more outcome-oriented lessons. We are using more general outcomes proposed by science and social studies teacher professional organizations because this allows the resources to be shared with other states in the future.

Prototype System

For the past two years we have been working toward our goal of building an integrated system that supports resource exploration, module construction and presentation, and community exchange. Using the current prototype, teachers have created over 100 modules on topics ranging from “Being a Space Explorer” to “Interpreting the Boston Massacre” to “The Bountiful Chesapeake”. The rest of the paper discusses the iterative design process and the results of informal usability testing. Lessons learned are also presented for developers.

DESIGN PROCESS

Because the project is a longitudinal, demonstration project that aims to influence human behavior, a user-centered, iterative design process was adopted to "grow" a system according to the needs of an evolving community of users. We knew that we wanted to provide rich system support for a variety of resources and communication venues, but could not overload busy teachers who have a wide range of previous computer experience with a full-featured system. Moreover, we knew that over the five years of the project, the technological capabilities and costs (especially with respect to high-speed networking) would change dramatically. Thus, multiple phases were defined and this briefing focuses on interface developments in the first phase of the overall project. The first two years of work have included three major iterations, which are detailed below.

Iteration 1: Concept Prototype

By the end of the project, the resource library is expected to grow to over 50,000 objects so the first prototype focused on the problem of how teachers could explore and search the available resources. In addition to the traditional keyword search strategy, we chose to use the dynamic query approach [1] because we thought it would encourage teachers to explore the library, not just search for specific resources. Dynamic query (DQ) applications support fast and easy exploration of data by allowing users to make queries by adjusting sliders and selecting buttons while the search results are continuously updated in a visual display (e.g., x/y scatterplot, map, etc.).

The prototype was developed using Visual Basic, which allowed us to demonstrate our ideas to teachers very quickly. First, users selected a subject (e.g., science, social studies) and topic (e.g., chemistry, geography). Then the resulting resources were shown on a x/y display where the x axis showed outcomes and the y axis showed subtopics (Figure 1).

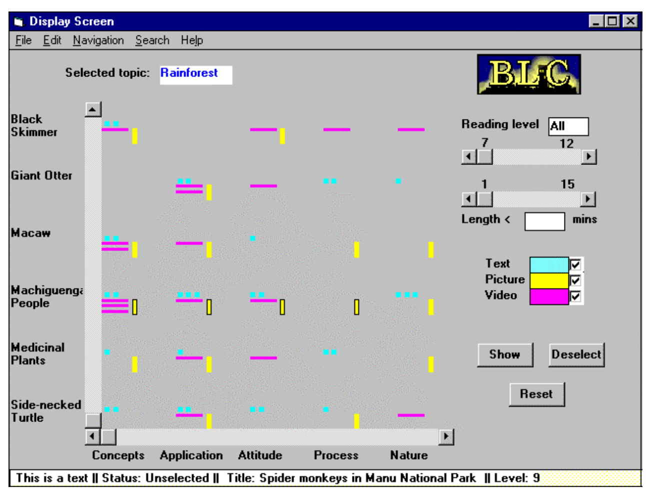

Figure 1. Visual Basic prototype of Explorer Component

The axes were chosen to help teachers map their current practice of searching for materials by topic to outcomes of student learning which are the bases of assessment. Because of the axes chosen, several resources were mapped to the same point so in the next iteration we needed to design a strategy for handling this overlap. Query controls illustrated how users could filter the display by reading level (for text), resource type, and length. Selecting a specific topic-outcome region displayed more detailed bibliographic information for those objects.

Since the aim of the concept prototype was to provide a tangible platform for discussing pedagogical functionality and introducing the dynamic query interface to teachers, the PC-based Visual Basic design served the project well. The prototype served as a foil for arguing requirements and specifications and teachers' initial reactions were positive so we proceeded to build a WWW based implementation prototype.

Iteration 2: Web Prototype

Whereas the first iteration focused on the search function, the second iteration considered three main functions: planning instructional modules (module construction component), searching for resources for these modules (explorer component), and using the completed modules in class (presentation component). Thus, we created the first prototypes of the module construction and presentation components, and continued the development of the explorer component. Other components such as teleconferencing will be added in coming years. The module construction prototype used a simple HTML-based, form fill-in style interface with five main sections: a general heading section, a before instruction section, a during instruction section, an after instruction section, and a reflections section (Figure 2). A direct link to the MSPAP outcomes was provided so they could easily be copied and pasted into teacher's modules. This approach would allow other states to plug in their outcomes very easily and with minimal change.

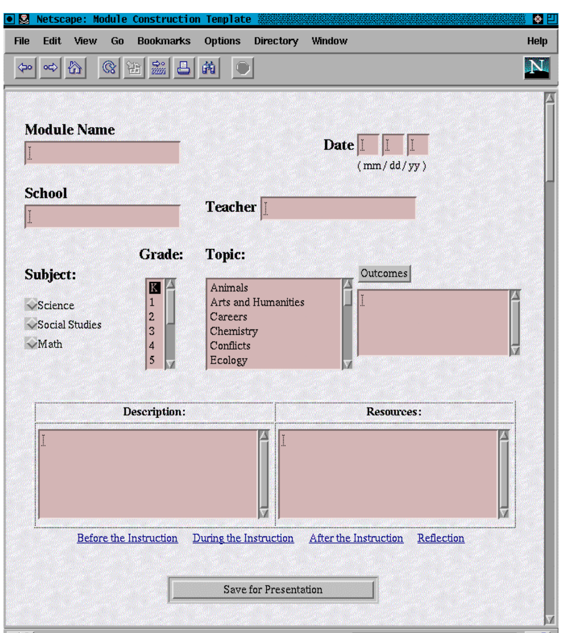

Figure 2. First Module Construction prototype

The first presentation prototype displayed a simple read-only summary of a module that allowed teachers to display resources by clicking on their URL links. Teachers could also use the presentation component to print out a hard copy of their modules. After using the presentation component, teachers requested the addition of a simple slide show function that would make it easier to display the resources during class.

In continuing the explorer design, we moved from the Visual Basic prototype to a Code Warrior prototype that could run on Apple Macintoshs (Figure 3). Major design changes included displaying overlapping resources as bar charts color coded by type, using real resource data, and providing image and video previews (four static key frames). Even though the display was very sparse, it allowed us to visualize which topics and outcomes had more resources and the quantity of various resource types (e.g., images, videos, etc.). When users selected a bar chart, a popup window appeared with a list of titles and a preview specific to the given object (see Figure 4 for an example of a video object).

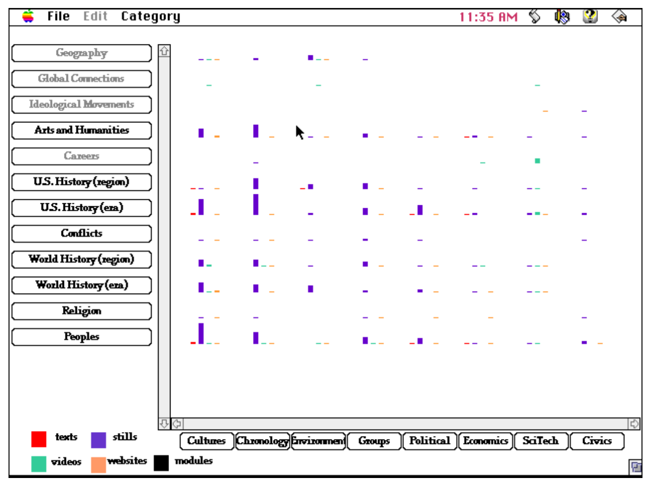

Figure 3. Code Warrior prototype of Explorer component

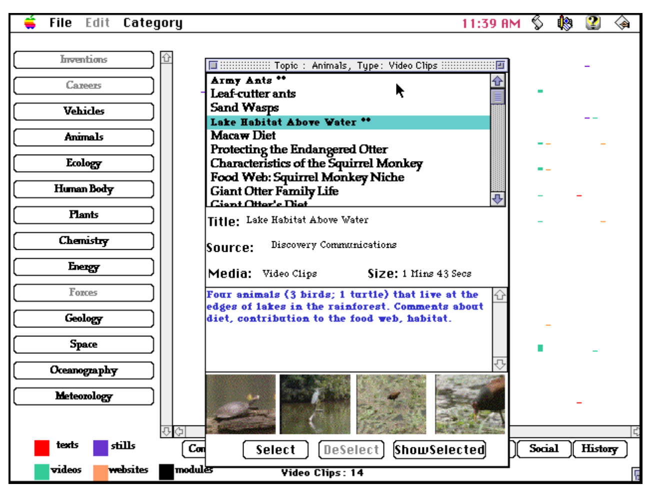

Figure 4. Static key frame video preview

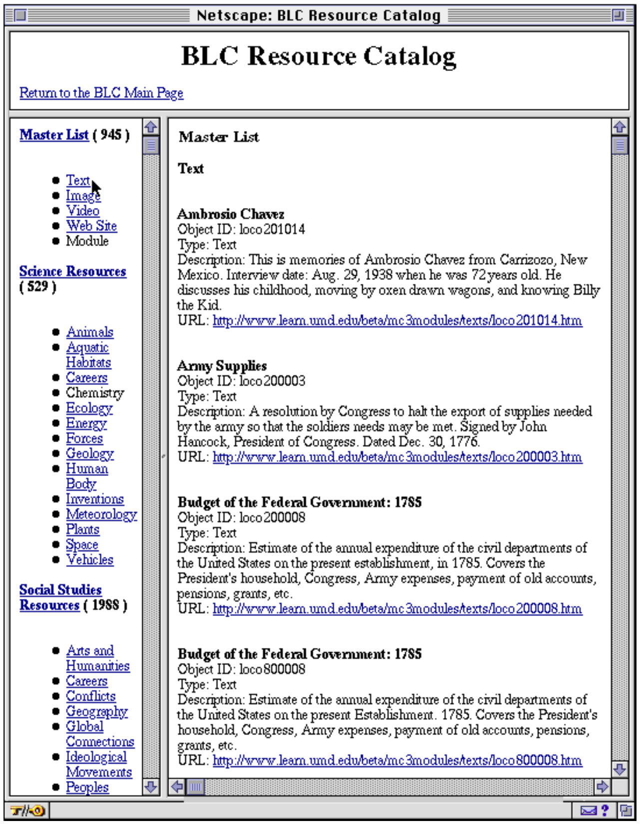

Figure 5. Resource Catalog

Even though the explorer component was not integrated with the module construction component yet, teachers were able to begin creating and using their own modules. Resources were added by typing in the appropriate URL. As an alternative to the paper resource catalogs given to teachers for finding resources, we created a simple HTML resource catalog that allowed uses to search, print, and view resources (Figure 5). This simple, interim solution enhanced the usability of the current system for many of the teachers during the 1996-97 academic year.

Iteration 3: Beta System

In the beta system, the distinct components were integrated to work from the main web page (BLC gateway) and re-implemented in Java. Our original intention was to create a Netscape plug-in from the Code Warrior explorer but given the rapid development of Java tools and the difficulties of creating plug-ins for Netscape in a project where components would change each year, we decided to re-implement all the components in Java. This allowed us to build an integrated suite of tools available on the WWW with the added advantage that the tools would be platform independent (for the most part). This opportunity also allowed us to incorporate many significant design changes recommended by the teachers into the revised components. Overall, we tried to minimize the amount of scrolling and reduce popup windows whenever possible. However, screen real estate was also an issue since teacher machines could only display 640x480 pixels.

In the module construction component, we added tabs for quick access to sections, multiple mechanisms for adding resources (e.g., explorer, resource catalog, personal resources (not in resource library)), and a section for controlling access level (read/write permissions) (Figure 6). This version not only added new functionality for the teachers but also greatly compacted the design and integrated the explorer component into the more general module construction component.

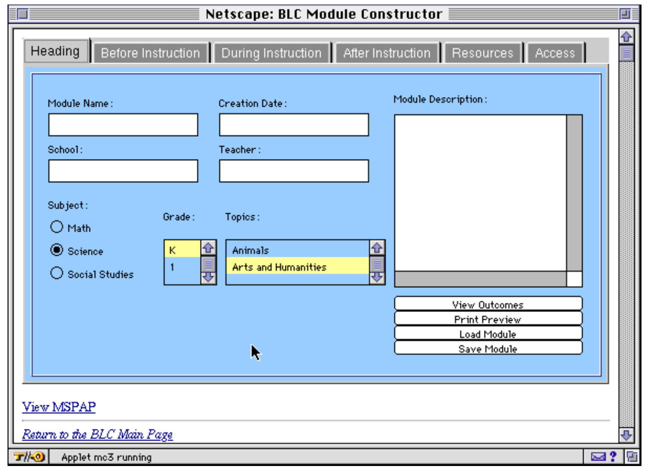

Figure 6. Java version of Module Construction

New functionality was also added to the explorer component (Figure 7) such as color coding by different attributes (e.g., type, source, etc.), changing axis attributes (e.g., topic, outcome, type, source, etc.), and "slide show" video previews that do not limit the number of key frames (Figure 8). In addition to displaying bar charts, the explorer now supports jittered starfield displays (another alternative to the overlap problem) where all the resources mapped to the same point are shown by colored dots circling the point. When there are too many resources at the same point for the jittered display (>50), a bar chart is displayed.

Controls are provided for querying by subject, type, source, and keyword (Figure 9). Initially the explorer supported dynamic querying (as a user manipulated a query control the display results on the starfield were updated simultaneously) but because of slow network speed users are now required to submit their queries (by pressing a button) before the results are displayed.

The major changes to the presentation component included providing quick access to module sections and adding "slide show" controls to allow teachers to present their modules (Figure 10).

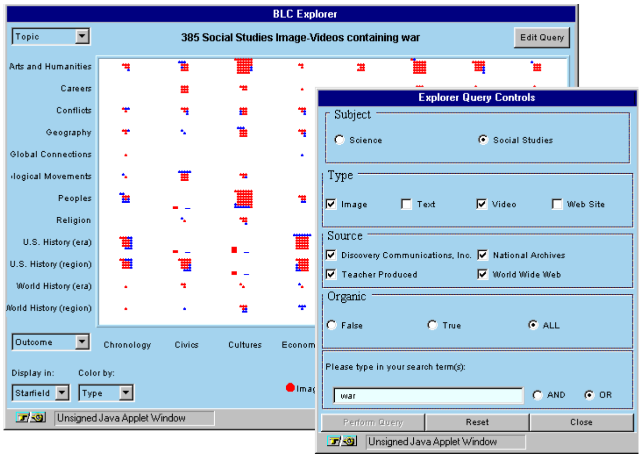

Figure 9. Query controls

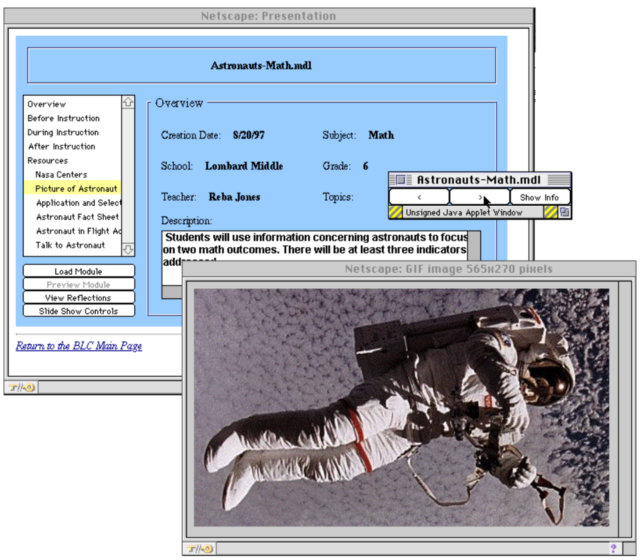

Figure 10. Java version of Presentation component

INFORMAL USABILITY TESTING

In addition to after school in-service meetings and periodic visits to the classrooms during the school year, the project brings all the teachers together each summer for a three day summer institute. At the 1996 institute, teachers used the web prototype components to construct sample modules and made suggestions for what became iteration 3. Since the explorer component was not functional, they used paper catalogs and web search engines to locate objects. During the 1996-97 school year, teachers mainly used web resources and some images in their modules while we worked on adding video to the library and developed the beta system.

Before the 1997 summer institute, teachers were asked to create a module using the old system (iteration 2). During the institute, the fourteen teachers who participated were introduced to the new system. They used the beta version to enhance their modules by adding video and to construct new modules. For the 1997-98 school year, videos will be loaded from CD-ROM or local hard disk while we install video serving capabilities on the server and high-speed connectivity in the schools. The concept of a module varied from teacher to teacher. Some modules contained slide shows of several images or video clips, while others integrated non-computer tasks and were meant to span several days. The project leader for the teachers demonstrated how the system could be used to create a module very quickly. In 8 minutes, he created a module for discussing the patterns evident in a series of cities (images) by using the explorer to select resources. After he presented his module, we noticed more teachers trying the explorer rather than depending only on the resource catalog.

Overall, teachers thought the new components were more compact, intuitive, and easier to use. On a scale from 1 to 10 where 10 is worse, teachers rated planning time (9.4) and class time (8.5) as their biggest problems in implementing this project in their classrooms. The complexity of the system (1.9) and the software (3.3) were not considered obstacles. After using the new components for one day, teachers had the lowest comfort level with the explorer (3.9 out of 5) which was not surprising considering they had seen and reacted to prototypes but had never actually used a dynamic query application and were used to using the resource catalog.

One teacher noted that the improved module construction component (iteration 3) was "easier to use because you don't have to scroll up and down so much." Teachers also liked that the explorer was "compact, color coded" and the ability to explore/search by topic. Another teacher commented: "The dittered dots allowed me to know the general idea of how the resources are distributed." However, the explorer is very slow to load so some teachers preferred to copy and paste resource information from the resource catalog. Teachers also "surf the web" to find resources since the resource library is not yet very large. The slide show feature was definitely the most popular improvement to the presentation component.

During a discussion of when and how the modules will be used in September and October, one teacher noted: "I think the BLC project has taught me a great deal about how to use the technology in the classroom as a tool and not as a play toy for my students. They really get involved with learning." Another interesting observation echoed by several project staff was how teachers went to other teachers for help rather than always depending on a staff member -- a sign of an emerging community.

It was not surprising that teacher's reactions were very positive since they had been involved in the design of the system from the beginning. However, several usability issues did emerge during the institute that will be incorporated into future versions. Some of these are simple oversights or features not completed in time for the summer institute, but others require new functionality.

1. Allow users to edit personal resources - Currently users can only add and delete personal resources so if they need to make a change, like changing a URL that is out of date, they need to delete the resource then add it again with the new information. Teachers were constantly frustrated when they wanted to make minor adjustments to the personal resources that they included in their modules.

2. Move the save/load buttons in the module construction component to the bottom - Since the save/load buttons are only visible in the heading section, several teachers lost work because they forgot they needed to save their modules. Messages to warn users of any unsaved changes will also be added in the next iteration.

3. Add functionality to print hard copy of module - Teachers are required to turn in lesson plans that take a significant amount of time to prepare. We hope to agree on a printed module format that would be accepted by their administration.

4. Add a field to indicate the intended length of a module - Now that more modules are being created and teachers are sharing their creations with others, it would be useful to know the length of instruction, an hour or a few weeks.

5. Implement word wrap – Word wrap is not implemented in the module construction component and only primitively in the presentation component. This seemingly minor functionality causes the teachers a fair amount of frustration since they format their text in the module construction component then see it displayed very differently in the presentation component (due to inconsistent implementation and different sized text boxes). We realized this would be an issue before summer institute but did not have the time to add and coordinate word wrap before the institute.

6. Load the resource library more quickly in the explorer – One of the chief complaints when using the explorer is that it is too slow to load. Currently, the resource library is loaded from a flat file stored on the server. Some of the alternatives we are investigating include using more efficient data structures or a real database engine. As the resource library grows, the problem will only worsen.

LESSONS LEARNED

After three iterations, we learned several important (and sometimes surprising) lessons that can benefit future developers.

Provide alternative search strategies - Providing different alternatives (e.g., explorer, resource catalog, keyword search) for teachers to search for resources enhanced the overall usability of the system by allowing users to choose the tool that best suits their work style, their computing environment, and the task at hand. The explorer's dynamic query interface provides an overview of all the available resources and allows users to search and explore using a variety of controls but its current implementation requires a fairly fast CPU with sufficient memory. Teachers working at home on slower machines through modem connections would more likely prefer to use the resource catalog or a simple keyword search. Using the resource catalog or keyword search is sufficient when searching for a specific resource. Some teachers may even prefer to work from a hard copy of the resource catalog that they can read at their leisure.

Provide "quick fixes" until functionality is available – There is always a tradeoff between waiting for a fully functional system and implementing interim solutions. Some developers may view interim solutions as a waste of time since they will be discarded when the final system is developed but in our case we were not only building a system but we were also building a community. Communities are defined by personal involvement so it is important for that users start using the system early and contribute to its design on an ongoing basis. In the coming year, we plan on installing video streaming software on the server. In the mean time, the system has been configured to play videos from CD-ROMs given to the schools. The resource catalog was another interim solution that was built while the integrated explorer component was being developed. User reaction to the resource catalog was so positive that we decided to keep it in the current version even though the explorer was functional. These interim solutions have allowed teachers to use the system while development continues and in some cases have altered the system design.

1000 objects is very small – Collecting, cataloging, and indexing the resources is a very time consuming process. Currently, it is being done by a few UMD graduate students. While 1000 objects is a good start, it is still too small for teachers to find the majority of resources they need. For some topics, like space, there is a lot of material but for others there is very little. One benefit of using the explorer is that teachers can see how the resources are distributed. For example, there are currently more social studies resources than science resources and most of the videos are on science topics. Discussions regarding the process for contributing resources to the community library have begun.

Seemingly minor functionality can become a major usability roadblock – Most of the suggested changes were things we had planned on fixing in the future but it was still surprising how much frustration they caused. Some of the problems were present in previous versions but had only really come to the foreground in the beta prototype. This is most likely because teachers were now presented with an integrated system that they could really start using in their classrooms so they were more focused on usability.

Building a community takes time – It is no surprise that in many systems like ours the biggest challenges are social and political, not technical. While there are several technical challenges we still need to tackle, our ultimate goal is to build an electronic learning community. Early on we provided teachers with instructional tools and email in an effort to help "grow" our community. As the community grows and teachers become more comfortable with the technology, the system will continue to evolve and teachers will become even more active participants in the design process. By working with the early adopters, we hope to hasten the adoption of the system by others teachers.

CONCLUSION

In three iterations we have gone from a concept prototype that illustrated ideas to a beta system that teachers use to create instructional modules for the classroom. Involving teachers in the design process early on enhanced the usability of the system and has helped "grow" the community. As designers we have learned the value of providing alternative strategies for performing tasks, the benefit of interim solutions, how time consuming it is to build a library, how minor functionality can decrease usability, and that it takes time to build a community. As teachers start using their modules more extensively in the classrooms, the system design will continue to change to satisfy evolving needs. Our current efforts are focused on collecting additional material for the resource library, implementing video streaming on the server, responding to teacher feedback, and assessing the impact of the project on teaching and learning. The system will continue to grow along with the community.

ACKNOWLEDGMENTS

We acknowledge the support of the U.S. Department of Education Technology Grant (#R303A50051) to the Baltimore City Public Schools. Thanks are due to the BCPS teachers and UMCP faculty and students who form the BLC community.

REFERENCES

1. Ahlberg, C., Williamson, C., Shneiderman, B. Dynamic queries for information exploration: An implementation and evaluation. *Proceedings of CHI '92*, (Monterey, CA, May 3-7, 1992), ACM Press, 619-626.

2. Marchionini, G., Nolet, V., Williams, H., Ding, W., Beale, J., Rose, A., Gordon, A., Enomoto, E., Harbinson, L. Content + Connectivity => Community: Digital Resources for a Learning Community. *Proceedings of Digital Libraries '97*, (Philadelphia, PA, July 23-26, 1997), ACM Press, 212-220.

3. Nolet, V., Marchionini, G., Enomoto, E. Full Motion Video and Dynamic Query in a Networked Learning Community. *Proceedings of EDMEDIA '97*, (Calgary, June 14-19, 1997), AACE, 777-78