ED 436 186   IR 019 811

AUTHOR   Andrews, Sandra; DiGangi, Samuel A.; Winograd, David;
              Jannasch-Pennell, Angel; Yu, Chong-Ho
TITLE   Designing On-Line Instruction Using Multidisciplinary
              Approaches.
PUB DATE   1999-02-00
NOTE   9p.; In: Proceedings of Selected Research and Development
              Papers Presented at the National Convention of the
              Association for Educational Communications and Technology
              [AECT] (21st, Houston, TX, February 10-14, 1999); see IR 019
              753.
PUB TYPE   Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE   MF01/PC01 Plus Postage.
DESCRIPTORS   Computer Assisted Instruction; Computer Software;
              *Educational Technology; Guidelines; Higher Education;
              Information Technology; *Instructional Design;
              *Instructional Development; *Interdisciplinary Approach;
              Laboratories; Teacher Developed Materials; Teamwork; *World
              Wide Web
IDENTIFIERS   Arizona State University; *Course Development; Instructional
              Support Services; *Technology Integration

ABSTRACT
              Instruction Support (IS) at Arizona State University is a
multidisciplinary group integrating academia with information technology.
Unique features having to do with the status of computing at the university
have led to the establishment of the IS Lab, which has produced a series of
successful initiatives regarding World Wide Web courses. The four key
features that have led to successful Web courses, initiatives, and students
are: (1) integration of academia and computing; (2) a university-wide policy
of decentralization with regard to computing resources; (3) a lab mission
that includes many levels of support as well as research and development; and
(4) an openness in the lab itself to real-world methods. In particular, a
close relationship with the College of Education allows the IS Lab to offer a
high level of assistance in Web course design and technology integration to
faculty members. The IS Lab, having tested a variety of approaches to the
creation of Web courses, has chosen two. Software packages are made
available, with training provided, and these packages may be enhanced if
desired by individually developed Web sites. A multidisciplinary, real-world
team approach, modified to include instructional designers, is a second
method. A set of standards for Web courses has also been developed. (Contains
25 references.) (MES)

DESIGNING ON-LINE INSTRUCTION USING MULTIDISCIPLINARY APPROACHES

Sandra Andrews
Samuel A. DiGangi, PhD
David Winograd
Angel Jannasch-Pennell, PhD
Chong-Ho Yu, Ph.D.
Arizona State University

Instruction Support at Arizona State University is a multidisciplinary group integrating academia with Information Technology. Unique features having to do with the status of computing at the University have led to the establishment of the Instruction Support Lab, which has produced a series of successful initiatives regarding web courses. In particular, a close relationship with the College of Education allows the Instruction Support Lab to offer a high level of assistance in web course design and technology integration to faculty members. The IS Lab, having tested a variety of approaches to the creation of web courses, has chosen two. Software packages are made available, with training provided, and these packages may be enhanced if desired by individually developed websites. A Multidisciplinary, real-world team approach, modified to include instructional designers, is a second method. There are pros and cons to both approaches; lessons learned have been documented. Web course examples of principles discussed are archived on the corresponding presentation website, at http://is.asu.edu/aect/multi/.

Introduction

It surprised many when Nicholas Negroponte, who in the past has spoken of the web as creating "a totally new, global social fabric"(1995), announced in *Wired* (1998) the end of the digital revolution. The implication was not that we would now cease to use computers, but that computers, and the changes they bring with them, have become commonplaces of our existence. One could argue with Negroponte's statement, yet in its very publication this remark signals the end of an era: for one hallmark of this revolution has been that a thinker in tune with the times could create new realities with a word (Davis, 1998; Rheingold, 1993a). Logos has had incredible power in these last years: writers, thinkers, educators could formulate a goal and then, thanks to electronic communication, quickly find a group of like minds with which to plan and implement means of reaching that goal. With this in mind, it is an appropriate time to detail the history of Instruction Support at Arizona State University, sharing methods that might be useful to others, and sharing lessons learned in the first two and one-half years of the Instruction Support Lab.

The Instruction Support Lab was founded in the summer of 1996, as an open door for faculty to the Instruction Support group in central Information Technology at ASU. Instruction Support itself came into being as part of a deliberate restructuring of central computing that brought an academic influence into Information Technology, in particular from the College of Education. This restructuring began with the creation of an academic position, the Vice Provost for Information Technology, filled by a professor from the Department of Computer Science and Engineering, and continued with the creation of the Instruction and Research department. A professor from the College of Education was brought in to head Instruction Support, and he was joined by two more researchers from the College of Education.

Much of the Lab's development has centered around the evolution of web courses and web supported courses. Very quickly, users proliferated. In the summer of 1997, before the Lab was even one year old, the Lab staff began searching for ways to streamline development work. Caught between the need to develop interactive websites and the need to serve faculty members new to the web, the Lab staff began designing templates and interfaces so that faculty members could set up web pages, quizzes and bulletin boards without knowing HTML. As other solutions arrived on the market they were tested and offered to professors who were willing to try them out. As there was no one solution that provided all the features we identified as important in a web course, IS continued to develop our own applications. We also kept in constant touch with what our colleagues in the commercial world were thinking and doing by following discussions on professional mailing lists. We had seen from the beginning that real-world development is always the product of a multimedia team (Nielsen, 1995), and we began refining this concept for academic purposes.

Major Arguments

That the lab has not remained static, but has been able to adjust support for technology integration in the face of a technology that changes daily, is the real hallmark of success. Three areas in particular in which we can point to specific successes are first, new areas of support in Information Technology that have developed from lab initiatives; second, students who have "graduated" from the lab; and third, courses and other projects developed in Instruction Support. Web courses and web-supported courses have been a central area in terms of faculty requests for

support, with over forty courses affected by the Lab. A number of Lab projects having to do with improving web course development have spun off into other areas. Each of these projects, having grown too large for the Lab, is now being developed/offered to faculty elsewhere in Instruction Support and Information Technology. For instance, support for those using web course creation software packages such as WebCT has moved out of the lab to Instruction Support's Education area. Support for collaboration software, pioneered with faculty members in the Lab, has also moved out to a production environment. Our initial efforts in creating scripts for interactivity resulted in the funding of an IS graduate student to complete a set of interfaces by which online quizzes could be created and administered. Our efforts to develop a web course creation package resulted in the project being funded. Student successes include three students who are now full-time staff members at ASU, a student who became one of the three-member webmaster team at ASU, and students who received excellent job offers upon graduation or earlier.

Key features contributing to the Instruction Support Lab

Four key features that have led to these successful web courses, initiatives, and students are, first, the previously-mentioned integration at ASU of academia and computing; second, a university-wide policy of decentralization with regard to computing resources; third, a lab mission that includes many levels of support as well as research and development; and fourth, an openness in the lab itself to real-world methods. The previously-mentioned change in the nature of Information Technology is the first key feature affecting the Lab. At its inception the IS Lab replaced a modest lab where faculty came for tasks such as printing. The new academic focus made it possible for representatives of Information Technology who understood the needs of faculty to reach out to the Colleges; improved access for faculty members to IT; and allowed IS Lab staff members to advise faculty on good instructional design. In the past, no web course support had been offered other than to one law professor who had come to the faculty lab to learn HTML. Now that same professor was offered instructional design support; help with graphics that were carefully crafted to enhance instruction while simultaneously following the rules of good web design; and interactivity. An invitation to collaborate in research could also be extended since Instruction Support now had researchers to evaluate major projects and present the results at academic conferences, or publish them in scholarly journals.

The second key feature affecting the Lab's success is that, as ASU Provost Milton Glick (1999) has stated, Arizona State University is a decentralized campus with regard to computing resources. We in Instruction Support feel it is inappropriate to mandate technology solutions across the board where instruction is concerned, just as it would be inappropriate to mandate that all professors use the same instructional strategies regardless of academic discipline. In 1999, solutions for instructional problems are still being developed; no one solution can be relied on to take care of all needs, whether it is a web course management system, an instructional strategy, or a particular platform. In keeping with this position, ASU has allowed differing support structures to arise. Currently there are three primary support structures for web courses. ASU professors desiring support can work with Information Technology's Instruction Support, with the College of Extended Education, or with a new entity, the Center for Learning and Teaching Excellence. Each has a different focus; faculty members can use more than one if they desire. Instruction Support offers an individualized approach that allows/requires the faculty member to participate in development. Extended Education is our distance education arm at ASU; faculty using their services may supply a course already developed by the professor, supply only the content and have the graphics developed, or experiment with "canned" content. The Center for Learning and Teaching Excellence is expected to serve as a bridge between the two, focusing on faculty development but familiar with the changing offerings of other support areas.

The stated mission of the Lab is the third key feature influencing the success of the Lab. Its tenets derive from the atmosphere of academic freedom that arises where a campus is decentralized with regard to technology, and where Information Technology has an academic flavor. The primary mission of the IS Lab is to help integrate technology into instruction. A faculty member might still ask for help with printing, but others have asked for help in planning the simultaneous premiere of a new work of classical music on the web from nine locations; in finding the way for a department to rearrange itself into a team in order to publish an online as well as a print version of a scholarly journal; and in rewriting a piece of software so that a professor of communication can create a web course without with maximum opportunities for communication. Beyond this, the Lab constantly conducts its own investigations into new technologies, so as to be prepared with advice when faculty approach us with instructional problems as well as to help drive the evolution of these technologies in the directions we as academics need them to evolve.

Finally, a fourth key feature contributing to success is the use of real-world development methods, always tempered and enriched by the academic viewpoint. Like labs on many other campuses, we have experienced the heady feeling of having the best of two worlds, academia and the commercial "real" world. In the academic tradition we have the freedom to conduct research and the luxury of supporting worthwhile projects. In common with the commercial world we have the realization that we are fortunate to live in a time in which our shared intelligence can find solutions and new directions almost as often as we need them. Developers who are in the forefront of web

design, digital artists, and many of the best instructional design minds all share their insights with us through online conversations and collaboration.

Real-world methods

The most important real-world method in use in the IS Lab is the concept of the multidisciplinary /multimedia web development team. No single person can create any kind of multimedia or web instruction alone if that instruction is to compete visually and professionally with commercial products. It would be impossible for the individual faculty member to simultaneously be a professional in all of the areas needed in development. Thus the faculty member coming into the lab to develop a web supported course becomes one member of a development team, in this case the content expert as well as a project co-manager. A second project manager, equal in authority, represents the Lab. As needed, a graphics expert, a videographer, a programmer, and an instructional designer may be assigned to the project. Each team member is a student, and all bring the latest information from their respective fields. The lab is a crucible in which these students from different disciplines must learn to work together. The graphics designer learns quickly that small file sizes are important to the web designer; the programmer learns to appreciate that it takes as much dedication and as many late-night hours to create visual perfection as it does to create the perfect program. As student team members learn to respect one another they realize the invaluable experience they are gaining: they will be prepared for the development teams that exist in their future places of employment (Crandall & Levich, 1998; Ullman, 1997.) The supervisor becomes the bridge among students from the various disciplines. In all, the Instruction Support Lab has a staff of twelve student workers, from as many of the University's Colleges as possible. While the majority are students in Education, Engineering, Fine Arts, and Architecture, students from all disciplines have something to offer.

A second real-world method is the concept of the "flat" hierarchy. This concept is prevalent in what Pearce (1997) calls "virtual" companies: small, low-overhead companies that can expand and contract as needed. Downes and Mui (1998) report the concept as a successful strategy in development, quoting a CEO as asking, "How do you change a culture from one of hierarchy with the normal pyramid to an open, flat culture where the 25-year-old can say to me, 'You're crazy. That won't work. Where did you get that dumb idea?'" A flat hierarchy does not mean that the CEO is no longer the CEO, but that a system of revolving authority can allow the expert in a field to make the decisions involving that field. The expert must be willing to share information, and the other team members must be willing to learn, so that responsibility for decision-making stays with the expert only so long as the decisions involve the expert's field. In practice, this implies that the supervisor is constantly being taught by the employees, so that the supervisor can participate in decision-making regarding the employees from an informed standpoint; and it gives employees a chance to be committed and to do their best.

A third real-world method is that of allowing staff members to manage their time. Student workers in the IS Lab are told upon being hired that they will be expected to spend approximately half their time on lab tasks shared with other Lab staff: working with drop-in clients, scanning, documenting, even dusting. The other half of their time should be spent in activities which will help them develop skills in their area of studies and interests. There may be times when as much time as possible will be devoted to a project deadline that demands that everyone contribute. However, this will be followed by time for the student to learn a new software program or to work on a project of interest.

Fourth, the Lab uses a system of networking and internships to find qualified/competent student staff. Because our work environment is unique on the ASU campus, the Lab has no trouble attracting applicants, yet we do most of our hiring through networking with former employees often recommending new students. These new students usually apply first as unpaid interns, willing give their time for a chance to learn the skills of working in a multidisciplinary team and creating professional work. Such interns are given the responsibilities of full Lab staff members. Interns are required to keep a journal and to produce, or work as a team member on, one large project that becomes a part of their portfolio and is useful to the lab. In addition to interns, the Lab has a mentoree program, in which young adults who would not otherwise have access to technology are given the status of interns in the lab. Of our four mentorees thus far, one was a learning-disabled young man; two came to us through a technology mentoring program for the formerly homeless; and one is deaf.

Fifth, the Lab relies on collaborative procedures involving electronic and face-to-face meetings, for the iterative nature of the developmental process requires collaboration (Yourdan, 1998). All students are given accounts on the Lab servers; after a recent survey of internal files on our main server it was estimated that almost one quarter were uploaded for collaborative purposes. Students working on a project may not be in the Lab at overlapping times, so collaborate by uploading their work to a shared directory on the server, then notifying the other students via e-mail of the project status. A particularly important collaborative procedure is the "studio model" adopted from the College of Architecture. Under this model, students develop in a lab environment, and a team member is encouraged to request feedback from peers at any time in the development process, even if they are not on the same team. An extension of this is the concept of the small usability study, in which an interface is tested by a group of seven to

4

thirty users. In addition, we are currently using collaboration software in order to keep documentation and student workload information easily accessible.

Finally, as far as possible, the Lab uses real-world planning and documentation procedures. These can be difficult to implement, as neither professors nor students may see the use. However, the use of a multidisciplinary team that does not always meet face to face requires the use also of planning documents such as timelines and storyboards. Such methods contribute to the success of commercial companies, as they break projects down into smaller tasks and issues (Kristof, 199); thus they can only improve our instruction. While developers agree that a project management tool that meets all needs has not yet appeared (England & Finney, 1999), as better software for project management is developed, documentation will become an expected part of project development for all team members.

Streamlining procedures

Streamlining goes hand in hand with technological sophistication (Porter, 1997) and is necessary if the level of sophistication is to be maintained. One means of allowing the Lab to remain dynamic in the face of changing technologies, numbers of users, and levels of user ability, has been the Lab's relationship to other areas within Instruction Support. Large projects may require support from staff members in production areas. Similarly, Lab staff members initiate, evaluate, and test new ideas in a development environment; successful initiatives and ideas are then moved to a production environment as projects grow. This streamlining procedure ensures that Lab staff are then free to begin new initiatives. With web course development we focus on finding ways to streamline procedures while maintaining high standards in terms of interface design, programming issues, web tenets, instructional design, and learning theory, while allowing maximum individualization for/ input from professors. This balance ensures that we keep current with information and with demand. A far as possible, web course development falls within a set of overarching ideas, philosophical and instructional. This set of standards frees staff developers to experiment and bring in new ideas so long as the ideas remain within this set of guiding parameters. For hyperlinked examples, see the website at http://is.asu.edu/aect/multi/ . The guidelines are as follows:

1. The professor is an integral part of the design team, not just the subject matter expert.
2. Development should provide for maximum reusability of procedures, templates and tools developed for web courses. Any such products developed for a course should be designed so as to be easily configured for other courses.
3. We attempt to apply principles from communication, learning theory, commercial multimedia/video production, and distance learning.
4. We value content first, using technology only where it is appropriate.
5. We seek to design for all users. We value the accessibility standards developed by the World Wide Web Consortium (1998). The electronic medium can provide disabled students with a chance to use their abilities and creativity (Barnette, 1994) if developers adhere to those standards. Whenever possible, we use multi-platform solutions; we keep file sizes and screen sizes to a level that as many users as possible can view; we use few passwords, preferring to share information when possible; and we attempt to use good HTML so as to be viewable everywhere.
6. Those using the lab resources are asked to give back to the Lab in some way. Often a professor places a graduate assistant or student worker in the lab.
7. Whenever possible, standard multimedia documentation procedures are used. We encourage professors to make plans with us for both formative and summative evaluation.
8. We pay particular attention to visual communication in interface design. This is an aspect of web course development that many course designers have been slow to take into account. This may be because interface design is not seen as being particularly important by professors who are used to considering only content when designing a face-to-face course. Many web courses are consequently still content-only, or use standardized icons for navigation that are the same across courses. It is our belief that visual communication interface design does indeed have a counterpart in the face-to-face world. The classroom itself is, after all, an interface (Jonassen, Davidson, Collins, Campbell & Haag, 1995). Students learn in an environment with which they interface using all senses. One course may take place in a large mediated classroom with projection screens; another may take place in a small room with rows of desks. Ignoring these aspects of the environment produces difficulties for those attempting to learn or teach via the web (Fleming, 1998; Mandel, 1997). Sensory cues are bound up with their learning and help students remember what they have learned.
9. We also pay particular attention to navigation issues. Confusing interfaces slow down the student, who will have to adjust to the interface before learning can begin (Fowler, 1995; Galitz, 1997). Commercial websites and television may be perceived of as noise-free by viewers, since research is used to produce the optimum visual environment. Students have become accustomed to receiving information in a particular format, and information given in another, less professional mode may be difficult to remember. This is not to say that educators need to become entertainers. Educational television uses the design principles of commercial television, and thus successfully transmits content. Web courses need similarly to investigate how information is transmitted in the commercial world, in order to avoid distracting users.

Methods of producing webcourses in Instruction Support

Professors and instructors requesting support with the creation of web courses are offered a choice of two major methods for creating those courses: first, they can choose to use a web course creation software tool, and second, they can choose to become a part of an individualized multimedia team. The two methods can also be combined as desired. The professor who wishes to create a web course is asked to attend an initial meeting with

5

representatives of the Lab staff. Here, levels of possible development are discussed. The professor may choose (a) to create a basic level web supported course, with syllabus and class schedule; (b) to create an interactive course using a software tool; or (c) to create an interactive course that will be designed from the ground up. If the faculty member wishes to use a software solution, the course is moved to the IS Education area, where training is provided. If the faculty member wishes instead to develop with a multidisciplinary team of Lab staff members, then the faculty member becomes project co-manager and usually provides one or more graduate students for the team. Faculty members may elect to work in both ways; a professor might decide to develop a website in the Lab that ties together and individualizes two courses developed, with a software tool, in the Education area. Should the faculty member decide to develop an interactive course from the ground up, then the multidisciplinary team may include not only IS Lab students, but also IS staff members such as programmers and videographers from production areas.

The multidisciplinary team approach to course development

If development is team-based, the professor is asked to attend at least four meetings, so as to be able to provide direction; this also ensures that communication is efficient. After this, a graduate student provided by the faculty member may attend as co-manager in place of the faculty member. Documentation paperwork, beginning with the first meeting, guides the team through a standard set of planning documents including proposal, content outline, flow chart, and detailed storyboards. An instructional designer from the College of Education is responsible for helping team members create these necessary documents; the documents will save valuable production time later. IS Lab graphics experts work with the instructor to create a series of graphics that communicate visually what the course is designed to do. Typically, a student from Fine Arts or Architecture creates a series of three sketches, and the professor then chooses one line of development. An programmer, usually from Computing Science and Engineering, works with the artist to create the website. Interactive elements are brought in as needed. All graphics must work in harmony; motifs are chosen to reflect the personality of the instructor as well as the nature of the course, for the course is in some sense a creative work, and the professor creating the course stands in the relationship of artist or writer to those who access the course (Maddox, 1996).

We view these initial meetings as an opportunity to learn the answers to questions that we ask in order to improve our work. The early meetings also provide an opportunity to help the professor understand the nature of the procedures used. We take the time to explain professional development methods (such as timelines and storyboards) to the professor. We ask where professors heard about the Lab; this enables us to plan for future information dissemination about Instruction Support. We discuss evaluation procedures, suggest collaboration in research, and explain backup procedures. We also ask the professor for the primary reason for creating a web course or web supported course. The answer to this question helps us direct development and enables us to predict what sorts of courses we need to prepare to develop. For example, a number of faculty members have come to us because they teach very large classes, and feel that they need new means to make information accessible; for this group, we may suggest posting presentation slides, lecture notes, and syllabi. A second group of faculty members may wish to create a course for faculty development or funding reasons. A third group is interested in the concept of web instruction and wishes to try out new ideas; for this group, we can investigate the pros and cons of those ideas and ways of implementing them. Other faculty members have been particularly interested in experimenting with interactivity, so that small groups can collaborate or so that members of a small course can work closely together.

The software method and the multidisciplinary team approach each have strengths and weaknesses that must be taken into account when choosing a method for a particular course. The advantages of a multidisciplinary team are that this method (a) provides for maximum individualization, (b) provides for maximum configurability, (c) can lead to a course that is professionally developed and of the highest quality in every area, (d) consists of well-integrated modules connected via good navigation design, (e) allows easy access to the visually impaired and others who use a text reader and/or speech software, and (e) can contain any element desired, as long as it is possible and there is sufficient time. Disadvantages are that (a) this method is time-consuming for all, including the professor; (b) if the professor is not a committed member of the team, expectations may be unreasonable; (c) if one team member leaves or graduates, it may slow down production, as the replacement will need time to become a member of the team.

The software tool approach to course development

For those who choose the software solution, the Instruction Support Education area currently offers a choice of three tools. We do not believe at this time that it is necessary or wise to offer only one, as any software tool has its advantages and disadvantages. While the standard packages have improved greatly, none really offers all of the features one would desire. An intensive look at the tools and packages being offered reveals that there are a number of considerations when deciding on which tool to use or to test. We have compiled a list of questions that faculty and staff should ask themselves before making this decision.

6

1. First, was the tool developed originally for the corporate world or the academic world? If for the corporate world, will its use conflict with sound instructional theory? If you are not sure, ask someone in the College of Education who is familiar with computer-based instruction or web-based instruction.
2. Is the interface intuitive and professional for both students and professors? Can you, as instructor, use your own graphics? If not, there is a danger that all courses will look the same from year to year, so that students forget where they learned important content. If possible, ask a graphics design professional to look at the interface and give an opinion on design and usability.
3. Which interactive features does it offer; which do you see a need for? You may want a forum, online quizzes, a chat room. Sometimes the programming that is necessary to build in interactivity slows the speed with which your users can access your course. If you elect to incorporate interactivity, do these features interfere with speed? You may want to ask an expert from Engineering or Computer Science.
4. Does the package provide for data collection and user tracking? Depending on the software tool, you may be able to give quizzes, add quiz scores to a grade book automatically, check on which pages have been accessed by which students, and keep logs of chat room activity. Again, sometimes the programming that is necessary to build in such data collection slows the speed with which your users can access your course. You may want to check with an expert from Engineering or Computer Science to see if these features unacceptably interfere with speed of access.
5. What is the pricing? How does this compare to other packages?
6. Does it offer accessibility for the visually impaired? Graphical interfaces can present difficulties for the visually impaired (Coombs, 1995). If a disabled student is in your course, you will need to offer all information in a manner that the student can access. It is relatively easy to write web pages so that they can be read either by a text reader or built-in speech, but not all web course tools offer this ability.
7. How many users and classes will it support? Software may allow many courses to be created, but break down in actual use if a relatively small number of students attempts to login at once. You may have to test the software on the server you will be using in order to know how many simultaneous logins the software can support.
8. Finally, does it work? If it is buggy, will students believe that they are not being treated fairly? Again, ask an expert from Computer Science to test with you.

To summarize, there are both advantages and disadvantages to the use of web course creation software tools. Advantages are that (a) they are relatively easy to use, as designer does not need to learn HTML but only the program that creates the HTML; (b) they provide interactivity with no extra effort on the part of the instructor; (c) they provide data collection to varying degrees; and (d) built-in password protection allows a sense of protected community. Disadvantages to take into account are that (a) courses developed with such a program tend to look alike, since they use an interface created by the program; (b) courses may look old-fashioned quickly, a problem that could be alleviated if the creators of the software used new templates each year; (c) interactivity and data collection are done via programming that may slow down the speed with which the pages are accessed; (d) if there are software bugs, they may not be easily remedied.

Discussion

During two and one half years of development the road has not always been smooth. We have had to adjust and change policies innumerable times, and from a bird's-eye view of that road we can now offer a number of "lessons learned" that we can share with others. To begin with many difficulties lie in the fact that there is too much information to manage (Stoll, 1995). It has been suggested (Applehans and Laugero, 199) that it is beneficial to think in terms of knowledge management rather than information management, where knowledge refers to documented, interpreted information that, once interpreted, becomes useful. Knowledge management then refers to defining job functions, identifying the knowledge needed for each job, and ensuring that workers receive that knowledge. Applying this idea to the Lab, we see that along with programmers and creative artists, a third important job function is the administrative/clerical area. People who enjoy detail-oriented organizational tasks, and can relate to these tasks in a creative and technological environment, are as necessary to functioning as a lead programmer. We have learned that unless we make the three roles equal in status and responsibility, development will not proceed smoothly.

Documentation and data collection are a high priority when developing in an environment in which team members may be collaborating electronically, when new paradigms are being tested, when pedagogies are being extended to other media, and when students are relying on our expertise to help faculty members offer web courses. While all this adds information to manage, it is important for predicting where we are going and identifying issues that need to be addressed. Giving high priority to the administrative/clerical side of the Lab has helped, as has an increase in supervisory staff. In 1996, the Lab had only one supervisor. This rapidly proved to be a difficult situation as one person cannot efficiently ensure that all policies and procedures are followed, guide the development of all projects, supervise all employees, be a bridge to production areas, and successfully conduct documentation. The addition of an assistant supervisor as well as a graduate student who can act as assistant supervisor has helped immensely. However, ensuring good communication and free information sharing among supervisors becomes of paramount importance so that policies and procedures remain standard.

The Instruction Support lab remains a drop-in lab for faculty requiring immediate help as well as being a development lab for those creating web courses and other projects. This has its advantages and disadvantages: we gather new ideas from all parts of the University by thus keeping our doors open. Moreover, we have a better idea of

the level of familiarity with technology among faculty than we would were we solely a production environment. On the other hand, it can be difficult to meet a deadline for a meeting with a faculty member when team members must take the time to help drop-in clients as well. Having other support areas within Instruction Support that we can call upon if necessary is one way in which we have been able to deal with this problem. A quiet space where students could retreat to work uninterruptedly would be helpful also. Further, while keeping our door open to drop-in faculty members helps us judge the level of support needed, it does not help with those faculty members who are afraid to enter the doors. Having the Lab supervisor go directly to a College to support technology integration there has been helpful in bringing to light the technology problems of those who are just beginning to use it.

In a University setting such as ours it is common for some staff to be student workers. We found that while the majority are committed and dedicated, a few cannot handle the freedom of a professional development environment in which they must make many of their own decisions and manage their own time. We have found it useful to place the student under the direct supervision of a student who is doing well; the second student then reports to the supervisor on progress. In this way the student who is having trouble can learn from a student who has adjusted to the professional model. Sometimes a student having difficulty with time management will fail to accomplish tasks; other students will must learn to avoid burning out. Similarly, some students find it difficult to adjust to the idea that they will be working in a team. Here, it is useful to seek and hire students who understand the value of having experience in a multidisciplinary team, and who are eager to learn to work with their peers from other Colleges.

No discussion of lessons learned would be complete without a reminder to back up data. The early status of the Lab as a developmental, experimental venue worked well when faculty members who co-developed were among the early adopters of technology. As the users of Lab services grew in numbers, many were not expert. In time we realized that we would need to treat any project developed for a course as "production" rather than "developmental" since not all faculty would understand the difference. Any course element being developed in the Lab must now be moved onto a production server located in a secure area before being offered to students.

Works Cited

Applehans, W. Globe, A. & Laugero, G. (1998). *Managing knowledge: a practical web-based approach.* Reading, MA: Addison-Wesley

Banks, R., & Coombs, N. (1995). *Your Onramp to the Internet: The Power of Electronic Mail.* Paper presented at the meeting of the American Association for Higher Education, Washington, D.C.

Barnette, R. (1994, Spring). Reflections on education on the information superhighway. *Online Chronicle of Distance Education & Communication.* Available: http://www.fcae.nova.edu/disted/fall94/article.html#barnette [1999, February 1].

Crandall, R. & Levich, M. (1998). *A network orange: logic and responsibility in the computer age.* Springer-Verlag: New York.

Davis, E. (1998). *Techgnosis: myth, magic and mysticism in the age of information.* Harmony: New York.

Downes, L. & Mui, C. (1998). *Unleashing the killer app.* Harvard Business School Press: Boston.

England, E. & Finney, A. (1999). *Managing multimedia: project management for interactive media.* (2 ed.). Addison Wesley: Sussex, England.

Fleming, J. (1998). *Web navigation: designing the user experience.* O'Reilly: Sebastopol.

Fowler, S. & Stanwick, V. (1995). *The GUI style guide.* Academic Press: Orlando.

Galitz, W. (1997). *Essential guide to user interface design.* Wiley: New York.

Glick, M. (1999, January). *The future of computing at Arizona State University.* Presentation at the meeting of Information Technology/Computer Consultants, Tempe, Arizona.

Jonassen, D., Davidson, M., Collins, M., Campbell, J., & Haag, B. B.(1995). Constructivism and computer-mediated-communication in distance education. *The American Journal of Distance Education*, 9(2), 7-26.

Kristof, R. & Satran, A. (1995). *Interactivity by design: creating & communicating with new media.* Adobe Press: Mountain View.

Maddox, J. (1996, Spring). Teaching on the net. *Online Chronicle of Distance Education & Communication.* Available: http://www.fcae.nova.edu/disted/spring96/articles.html [1998, March 14].

8

Mandel, T. (1997). *The elements of user interface design.* Wiley: New York.

Negroponte, N. (1995). *Being digital.* New York: Alfred A. Knopf, Inc.

Negroponte, N. (1998, December). Beyond digital. *Wired.* Available: http://www.wired.com/wired/archive/6.12/negroponte.html [1999, February 1].

Nielsen, J. (1995, October) . Who should you hire to design your web site? *Alertbox.* Available: http://www.useit.com/alertbox/9510.html [1999, February 1]

Pearce, C. *The interactive book: a guide to the interactive revolution.* (1997). Macmillan: Indianapolis.

Porter, L. (1997). *Creating the virtual classroom: distance learning with the internet.* Wiley: New York.

Rheingold, H. (1993a). A slice of life in my virtual community. In L. M. Harisim (Ed.), *Global networks* (pp. 57-80). Cambridge: MIT Press.

Stoll, C. (1995). Stoll, C. (1995). Silicon snake oil. (1 ed.). New York: Doubleday (1 ed.). New York: Doubleday

Ullman, E. (1997). *Close to the machine: technophilia and its discontents.* City Lights: San Francisco.

World Wide Web Consortium (1998). *Web accessibility initiative.* Available: http://www.w3.org/WAI/ [1999, January 25].

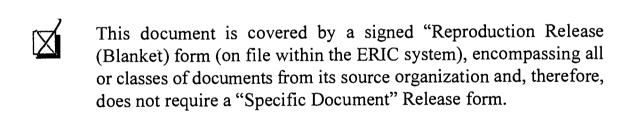Yourdon, E. (1998). *Rise & resurrection of the American programmer.* Prentice Hall: Upper Saddle River, N.J.