Dear WWW: Interacting with the Public through the World Wide Web

Kenneth Weiss
krweiss@ucdavis.edu
University of California, Davis
Davis, California

Steve Faith
faith@apple.com
(formerly of the University of California, Davis)
Apple Computer
Elk Grove, California

Abstract: For many educational institutions the World Wide Web has become a major avenue of communication with its constituencies. Students, potential students, faculty, staff, people at other institutions and the general public are all directing their questions to the email address www@institution.edu. The personnel that receive this mail are often ill-prepared for the quantity and variety of questions that arrive. This paper examines the evolution of this problem over the first four years of UC Davis’ use of the Web, and presents some of the strategies that have proven useful for managing this onslaught.
Introduction

The World Wide Web (WWW) rose to prominence in 1994. It quickly supplanted older technologies such as Gopher and FTP as a tool for educational institutions to communicate with the public. The widespread adoption of this technology, both within the educational community as information providers and with the general public as clients, led to an unprecedented volume of incoming electronic mail. Almost every WWW site includes some means by which readers can send comments back to the Web site administrators. That address is most often in the form www@institution.edu (www@ucdavis.edu, for example). This email address was originally intended as a way to let the server administrator know about broken links and other technical or administrative problems with the web site. However, the address quickly became one of the most popular ways for people to send questions on any topic to the institution.

Most WWW servers were established by faculty or students in technical departments (Computer Science, Electrical Engineering, etc.) or by technical staff. These people are generally unaccustomed to handling the full range of inquiries that arrive at a major research-oriented educational institution.

The closest pre-existing analog to the problem faced by the recipients of mail to www@somewhere.edu can probably be found in the challenges faced by research librarians. On any given day, anyone might walk into the library and ask for information on just about any topic under the sun. Some of the strategies developed by the library community to manage this type of unstructured service can be applied to the ‘Dear WWW’ problem.

At its core, this is a problem of communication and expectation. At several points in the process there are disconnects — places where the expectations of the two parties involved in the dialog have diverged completely. Some of these disconnects arise out of mutual ignorance. Others are an inseparable result of the nature of the World Wide Web and its evolution as a communications tool.

The Content/Technical Dilemma

There are two dimensions to a WWW server: the technical problems of establishing and maintaining a reliable and robust service, and the
informational processes of creating and managing useful and timely content. At best, these two dimensions tend to have few points of intersection. More often, there are mutually exclusive or antagonistic aspects to the efforts associated with each process.

**Overall goals can be in conflict**

One important technical goal is the rapid and deep deployment of new technology. Given the pace of change in this arena, anything less guarantees immediate obsolescence. In the university environment, one effective strategy for rapid deployment is decentralization. A centralized Web server tends to lead to slower adoption of the technology. Faculty and staff in departments are often resistant to the idea of turning their content over to a central administrative unit for publication and maintenance. The content that is turned over to the central administrators tends to be less volatile, and of limited quantities. Funding a large central Web server at a level adequate to provide good service to the people that create the content is problematic. A larger number of small Web servers located in the departments can encourage more rapid integration of the new technology into the business and communication processes of the campus.

An equally important administrative goal is the creation of a unified image and voice for the institution. A hodgepodge of Web sites with different names, different graphic styles, and widely varying levels of overall quality is a public communications nightmare. Multiple sources of information may be in disagreement with each other. Ideally an institution wants to have complete consistency throughout all of its web servers.

Most universities find significant benefits in developing technical skills deep in the organization. The overall cost of technology is reduced, and the institution is more flexible and able to adapt to a rapidly changing environment. Those institutions that are more dependent on a centralized technology model find themselves reacting to technological change instead of driving the process.

Web servers are integral tools for the business of most universities today. That means the servers must be reliable.

Cost is always an issue. Everyone wants to minimize their total investment in the World Wide Web. These costs include the obvious ones like server hardware and software, and the staff to run it. There are other
less apparent costs like the staff to convert existing documents and applications to the WWW, provision of “7/24” (7 days a week, 24 hours a day) reliability, data backups, capacity planning, providing adequate network bandwidth, and a host of other issues that are often completely invisible to someone just getting started with the Web.

In one way or another, almost every one of these overall goals conflicts with one or more of the others. Rapid development of large and timely body of information is most easily achieved with a decentralized architecture. This goal conflicts directly with the institution’s desire to present a unified and coherent public face. Reliability is essential, but it is also very expensive. Technical skills should run deep into an organization, but pushing technical responsibility out into departments will inevitably result in reduced reliability and consistency.

In many ways the goals of the people responsible for the informational content of a campus Web are in conflict with the goals of the people responsible for the technical success of the infrastructure. Even more problematic is the fact that at many institutions these two groups of people may have virtually no daily interaction. Until the Web threw them together they may have been only dimly aware of each other’s existence.

**Who determines the technical environment?**

The individual web administrator may have very little to say about the technical environment of the institution. That environment can have some very direct effects on the way that web administrator goes about her job. The technical environment can also have an impact on the way content is developed and published.

The technical environment is both physical and cultural. Are fast network connections widely available? Does the campus have a high quality connection to the Internet? What kinds of hardware and operating systems are commonly used at the campus? How strong is the central information technology organization? Is technical expertise available in the departments?

At most institutions, there’s no clear answer to these questions, no clear individual or office responsible for the technical environment, and the whole thing is a moving target anyway. However, at most institutions it’s possible to say that they lie towards one end or the other of a continuum from highly centralized and mainframe oriented to highly decentralized and client/server oriented. Neither approach is inherently good or evil. (I
had to say that. We all know that mainframes are inherently evil. Different approaches to deploying WWW servers are suited to different environments. What’s really interesting, though, is the potential this technology has shown for directly influencing the development of the technical environment. Initially it can be difficult to force adoption of a decentralized WWW architecture in a highly centralized organization. After a year or so, though, those decentralized servers can be found driving a whole new approach to the delivery of applications and information.

Who picks the architecture?

At many campuses, the WWW architecture developed by default. The architecture was chosen by a few students and faculty that brought up servers back in late 1993 and early 1994. Once those servers were established and the good hostnames (www.institution.edu, info.institution.edu) were taken, decentralization was almost a fait accompli.

A few campuses stayed out of the fray long enough to develop a deliberate strategy to moving into the Web. Very few. If you are from such a campus, drop me a line (krweiss@ucdavis.edu). I’d be curious to hear how you came to be in that enviable position, and how you handled the problem of devising a workable Web architecture.

The most interesting developments today seem to be at those institutions with a strong tradition of central services. Faced with anarchy, they are developing some very interesting strategies to bring the strays back into the herd. One approach provides a massive central server, with full staffing and support. Various locally-developed tools allow users to create and manage content without actually logging on to the central server. Often the tools themselves are Web-based. This allows faculty and other authorized users to create web pages that are served centrally, on a well configured 7/24 system. Another common approach is based on policy and procedure. Renegade web servers are simply declared illegal.

Who determines content?

The role of the content provider is at least partially determined by the architecture. In a decentralized Web, there is generally a decentralized model for building content as well. However, even in a decentralized
architecture it’s common for content development to be handled by someone other than the server administrator.

Net Result — Guaranteed Disconnections

The outcome of all these various inherent conflicts between the administrative and technical aspects of creating and running a Web service is simple. It’s a certainty that the people running the servers will not be fully cognizant of the information that those servers provide to the general public. Technical people run the servers. Administrative people create the content. Both technical development and content development are largely ad hoc processes at most institutions. The final point that makes this whole thing nearly perfect as a tool for poor customer service is the fact that in almost every case it is the technical staff that receive electronic mail directed to ‘www@institution.edu.’

Case Study: WWW at UC Davis

UC Davis has a very highly decentralized Web. There are over 300 separate servers, and more than 200 individuals that function as Web server administrators. Almost all of these people and systems are out in the departments, not associated with the central information technology unit.

This architecture came about as a result of some very deliberate decisions made very early in the process. In early 1994 there were only two production Web servers on campus. One was in the Computer Science Department, and one in Electrical and Computer Engineering. None of the good hostnames like www.ucdavis.edu had been taken yet.

The central IT organization at UC Davis has a unit called Distributed Computing Analysis and Support (DCAS). Among other things, DCAS is charged with identifying emerging technologies that will have an effect on the way the campus does business. DCAS is also supposed to prototype those technologies and act as an incubator/advocate until they achieve widespread use. Two individuals at DCAS were working on the World Wide Web. In March of 1994 they registered the hostname www.ucdavis.edu and put up a server with some basic information on computing at the campus, and pointers to the other two UC Davis servers.
This scenario is very typical. For quite some time the content of most academic web sites was dominated by technical support information, created and maintained by the central IT organization. What set Davis apart to some degree was a decision made by those two early webmasters. They decided that the central host (www.ucdavis.edu) would never be a primary source of content. Instead, it would serve as a clearinghouse for links to other servers on campus, where the actual content would reside.

This decision was based in large part on previous experience with Gopher services. Gopher was run on a central server. Information providers would forward their files to the server administrators, who would then put the material on line. It quickly became clear maintenance of content implied ownership of the content in every meaningful sense. The tasks associated with keeping the Gopher data current and accurate became quite onerous. Information providers weren’t happy with the turnaround they were getting on updates. The general public wasn’t happy with the fact that the Gopher administrators were not able to answer questions about the content. And the administrators were caught in the middle.

This simple decision to act as a clearinghouse of links instead of a primary source of content had far-reaching implications. Instead of devoting central IT staff time to maintaining Web content, that time was spent assisting departments in bringing up their own servers. A year after this policy was initiated there were almost 100 Web servers at UC Davis. Two years later there were over 200 servers. Today there are over 350 servers, and the growth is tapering off. However, the total amount of content provided by those servers continues to grow rapidly. There are now over 152,000 documents in the central campus index of Web content.

This growth was achieved at remarkably low cost, at least from the perspective of the central IT unit. Two individuals devoting about 25% time each to the project were able to achieve this. However, there are some darker sides to this aggressively decentralized approach. Some faculty expressed dismay at the suggestion that they, or their department, should establish their own servers. They wanted to have someone else take care of all those details. To accommodate this, and to help departments that wanted to run their own server but didn’t yet have the technical expertise or network connectivity, a single centrally managed server was established. This system was called pubweb.ucdavis.edu.

The hardware and management model for pubweb were deliberately chosen to be just a little bit inconvenient for the end user. This was done so that people would not lose their motivation to establish their own
servers. A Macintosh 6100 was set up in the Center for Advanced Information Technology, a joint IT/Library facility located in Shields Library. This server ran MacHTTP, but the incoming FTP service was not activated. People had to schedule time with the CAIT to come in and move their data into pubweb. They had to go to the library and sit at the system console to do any maintenance to their content.

Although unpopular and a bit controversial, the decision to slightly under-deliver services on pubweb worked out as planned. Almost everyone used the system as an interim solution, and eventually set up production servers in their own departments. No one was ever denied service. No one was told that there was no place for them within the UC Davis Web. Everyone was encouraged to participate, and encouraged to place the server as close to the source of the content as possible.

**Reality Butts In**

By mid-1995 the UC Davis webmasters were feeling pretty good about themselves. The campus had gone from nowhere to being one of the best-connected educational institutions around, in just one year. At least, that’s what some industry surveys were telling us. But, as the amount and variety of information available on the UC Davis Web grew, the webmasters began receiving more and more mail, of a stranger and stranger nature.

Davis has one of the premier veterinary schools in the world. Davis also has a very well-known program in Viticulture and Enology, a full medical school and associated teaching hospital, many dozens of major specimen collections in the life sciences, and more information on Integrated Pest Management than anyplace else on the Web. Users of the campus web site were totally oblivious to the fact that the webmasters knew how to run a web server, but knew nothing at all about pulmonary disease in ferrets. Users of the campus web were also generally oblivious to the fact that within two clicks they were off www.ucdavis.edu, and onto a departmental web server with which the central webmasters had no involvement. When they had questions, they sent them to www@ucdavis.edu and waited for a reply.

This problem is less severe where the Web service is centralized. The server administrators tend to know more about the content on their own system. Where the content is centralized, knowledge about that content is also pretty well centralized. At Davis, the totally decentralized
architecture for the Web led to a total lack of central information on what content was available and where it might be found.

The mail to www@ucdavis.edu pretty quickly settled into a pattern, and has remained consistent ever since. There are genuine server administration issues, requests for admissions information, requests for phone numbers and email addresses, requests for general technical support and requests for administrative services such as transcripts. The remainder are all over the map. People write and ask for jobs, they write to tell us that our home page is ugly, they write to ask if we know where their roommate from 1982 is living today. [In our live presentation at CAUSE 97 we will present some of the more amazing and funny letters we’ve received.]

In all, about 80 messages per week are sent to www@ucdavis.edu. In a very real sense, the webmasters are the front door to the campus. For many members of the public, www@ucdavis.edu will be their only contact with the campus. When one of the webmasters remarked on this in a presentation made to an audience of about 50 people, the Associate Vice Chancellor for Information Technology was heard to gasp in a horrified voice, “Oh my God!” She simply had never realized the degree to which the reputation and public image of our campus was dependent on the actions of two of her staff.

**Solutions at UC Davis**

The UC Davis webmasters use several different techniques to manage the Dear WWW mail. About half the volume of mail can be handled with a simple form letter. This works for admissions inquires, requests from staff, faculty and students for changes to their directory records, and requests for general technical assistance.

All of the most popular questions have been answered in a Frequently Asked Questions document. Following the excellent lead of Kent Wada at UCLA, a link to the FAQ was placed on the home page in the location normally occupied by the direct mailto: link to the webmasters. This way, a reader has to go through the FAQ before they get to the mailto: link. This simple idea cut the volume of email to www@ucdavis.edu by about 25%.

UC Davis was one of the early adopters of local search engines. We began with Harvest, from the University of Colorado. Harvest was a research project headed by Mike Schwartz. For quite some time it was the only
search engine available that came with its own robot/spider. The robot or spider is the software that goes out into a decentralized web and indexes the content of multiple servers. Harvest never turned into a commercial product, and once the research project was completed support for the software ceased. By then there were a few commercial alternatives that were suited to the UC Davis environment. We purchased the AltaVista search engine, and are evaluating InfoSeek. The search engine serves two important purposes. First, it eliminates many questions at the source. People can find answers for themselves. Second, it allows the webmasters to find content and respond to questions quickly and accurately. A good example was a recent request for information on HYPP. The webmasters had no idea what HYPP might be. But, after about 20 seconds with the search engine, an answer was on its way. HYPP stands for Hyperkalemic Periodic Paralysis, a genetic disease that afflicts quarterhorses. This disease is of particular interest because all known cases can be traced back to a single bloodline originating with a horse named *Impressive*.

The webmasters built a network of contacts in different departments to whom queries can be referred. There are people in the Office of the Registrar, the School of Veterinary Medicine, the School of Medicine, Viticulture and Enology, and a number of other places that have come to expect these out-of-the-blue requests for information from the webmasters. Having this network of contacts in place has greatly eased the problem of finding answers.

**Plans for the Future**

At UC Davis we have a few plans for continuing the development of our tools for managing inquiries from the public, and for our web in general. Our first priority is to start routing Dear WWW mail directly to the right server administrator. More and more of the questions relate to servers out in the departments. We are compiling a database of all the servers and the names and email addresses of the server administrators. We will add a link to that table to our FAQ page, so that people can send their mail directly to the right place.

We are formalizing our referral network so that departments are fully aware of the need to have people available to field the email that comes into the campus.

Our biggest plan for the near term is to move into a service that will fully separate technical server management from content management, and allow us to run a central web service with distributed content. We have
begun serving our HTML documents from a distributed file system called AFS. During this year we will prototype a service for faculty that would give them directories on an AFS volume that they can access from their Macintosh or Windows desktop. A centrally managed HTTP server will provide public access to the documents in those distributed directories. While this won’t do anything to ease the public’s misconceptions about the role of the server administrators, it should go a long way towards fixing some of the internal disconnections between content providers and technical managers.