WORLD WIDE WEB DEVELOPMENT AT NOTRE DAME: IT'S EASIER TO ASK FORGIVENESS THAN PERMISSION

Mike W. Miller
Consultant/Analyst
G038 Computing Center
Notre Dame, IN 46556
Phone: (219) 631-8024
Fax: (219) 631-8201
E-Mail: mike.w.miller@nd.edu

ABSTRACT

The Office of University Computing at Notre Dame has introduced several successful enterprise-wide services in recent years. A World Wide Web (WWW) server was started at ND in May 1993 as proof of concept, and sat for nine months with virtually no development. I discovered Mosaic and the Macintosh WWW server software in February 1994 and over several late nights, a server called the Orange Room was born. The following week, much of the Orange Room material was converted to more official documents which became an improved Notre Dame server.

To tackle some of the implementation issues, four teams and a steering committee were formed and a broad development strategy was outlined and presented to the directors, who were concerned. "Too many teams, too many staff cycles" they responded, and the project was temporarily iced.

The core team regrouped and developed a "leaner, meaner" approach. Discarding traditional bureaucratic development channels, the team proposed tackling this roll-out in a much more aggressive, more individual-work-focused manner. Starting as a "renegade" project and having to switch development strategies helped refine our development processes in this time of decreasing product cycles. The paper will conclude with a current statement of the good and bad aspects of our development journey, as well as outlining the future of Notre Dame's WWW service.

OVERVIEW

The University of Notre Dame is a private Catholic university serving 7500 undergraduate and 2500 graduate students in eight colleges and schools. Notre Dame has a strongly centralized computing organization which serves the student body as well as 5000 faculty and staff in all academic and administrative departments.

The Office of University Computing (OUC) has introduced several successful enterprise-wide services in recent years. Our roll-out of electronic mail was very well received, and currently 90% of students and networked faculty and staff are using some form of electronic mail. Our Gopher server is actively used by many academic and administrative departments. A number of unique services are offered in our computer clusters and we are currently wiring our residence halls for computing and planning the services and support which will be delivered in the residence halls.

THE BEGINNINGS OF WWW AT NOTRE DAME

In May 1993, a member of the Networking Services group of the OUC installed a World Wide Web (WWW) server as proof of concept and to learn about the software. Consisting of a single, anemic page, this server sat idle for nine months with no development.

Serendipity struck in January 1994 when, while browsing the FTP archives at the University of Michigan, I discovered NCSA's WWW client *Mosaic for Macintosh.* At the same time, I found Chuck Shotton's WWW server software for the Macintosh, *MacHTTP.* Playing with Mosaic, I was immediately ensnared in the Web, and I tried my own hand a putting up a server. Over several late nights, a light-hearted server called the Orange Room (*http://orange-room.cc.nd.edu*) was born. In the following week, much of the Orange Room's content was converted to more official documents which were installed in a new and improved Notre Dame WWW server (*http://www.nd.edu*).

Over the next six weeks, the semi-official ND server and the "underground" Orange Room server both grew and became quite popular, on campus and off. Sensing the start of something big, I enlisted another staff member who was very interested in the Windows side of things, and together we presented two informational demonstrations to OUC staff and selected others which introduced World Wide Web, the Mosaic family of clients, and gave an overview of constructing HTML documents.

The first steps had been taken, but as Notre Dame's fledgling WWW service grew, it became clear that a number of issues had to be addressed in order to pull off a coherent, organized campus-wide roll-out. The implementation team for Gopher had done a good job with most aspects of

© 1994 ACM 0-89791-656-5/94/459940 $3.50

introducing that service to the campus community, and so I worked with the person responsible for the Gopher introduction to start exploring how the OUC should approach this new service.

FORMALIZING THE SERVICE, ROUND ONE

A small group of seven people met initially to brainstorm the issues which faced us in a WWW roll-out and what teams might be needed to address these issues. It became to seem that the same team structure that tackled Gopher, with a few small modifications, would work for WWW. The Gopher roll-out effort involved four teams:

- a **Client Team**, responsible for issues of selecting and deploying appropriate client software as well as training and documentation for that software
- a **Server Team**, responsible for selecting, installing, and maintaining the server software and developing any new code necessary for our campus service
- an **Information Provider Team**, charged with developing training and guidelines for how various members of the campus community would publish information via Gopher
- an **Information Architecture Team**, responsible for structuring the data in Gopher and deciding exactly which types of information would be presented and how. These four team charters were adopted for use in the WWW roll-out.

In the Gopher development phase, the four teams worked well independently, but at times there was not much coordination. To address that, we also added a fifth team, a Steering Committee made of two members of each of the other teams. The purpose of the Steering Committee was to serve as the coordinating body, making sure that the four teams worked in concert and that there were no areas of overlap or neglect in the broad development effort.

We were satisfied with the proposed development direction and made a formal presentation to the directors of the OUC to officially ask for the staff cycles necessary for the teams.

The reaction from the directors was not exactly what we expected. They were concerned about the number of people involved in this project. They were not especially familiar with WWW or the explosive communication potential it represented. And to some degree, I believe they slightly resented being asked to endorse a large project which did not come from themselves. Whatever the reasons, the directors responded negatively, saying that the OUC had other, higher priorities and that WWW development needed to be rethought. Simply put, the project was officially iced.

ROUND TWO: LEAN AND MEAN

Disappointed but not discouraged, the core team of seven regrouped to lick our wounds and figure out what had happened. We talked about the speed with which new services are evolving and how old-style product development cycles cannot work in the current environment. We also discussed some realities of most work groups: that of the eight people on a typical team, one or two tend to do all the work and the rest attend the meetings, smile, and nod agreement with whatever has been proposed or done. We also realized that what had been done so far in an entirely ad hoc manner was of very high quality, and perhaps the individual-driven development mode could be adapted to allow WWW growth to happen quietly and with a much smaller staff footprint.

The results of our brainstorming were a much "leaner, meaner" approach to service development. We would eschew traditional hierarchical development models and proceed in an empowered, individual-work-focused development strategy where the core team would each develop in their area of expertise, checking in with the others as needed to ensure coherence in the overall process. In many ways, it was the same as what would have happened with all the staff involved in teams, but without the overhead of having to take every step back to a committee for their smiling, nodding approval.

One significant change in strategy that the smaller implementation team settled on was to split the development into two phases. First, we decided we would concentrate on developing the infrastructure and let users, by and large, fend for themselves during this phase. For the pioneering users who wanted to begin content development immediately, they were encouraged to do so but with the understanding that official documentation and support would be minimal. The client software was distributed, but users were cautioned that the service was in an experimental stage and support would be very limited.

In the second phase, we would go back and fill in the framework by actively supporting users and content development. Training and documentation on the client software and on developing content for WWW publishing would be developed. We would begin helping users with the migration from Gopher to WWW as well as encouraging new content providers to begin publishing via WWW.

IMPLEMENTATION ISSUES

There were, of course, policy and procedural issues to be addressed. One of the successful aspects of our Gopher service was that we included an automatic expiring mechanism so that all documents had to include an expiration date, after which the document would be automatically pulled from Gopherspace and returned via E-Mail to the author. This served to ensure that old data would not be on the Gopher server for eternity. That worked well since all Gopher data lived in one central directory structure, but part of the beauty of WWW is that the data served through our server can be scattered throughout our distributed file system.

We've developed a modified version of this expiration mechanism whereby documents are requested to contain an expiration data in the meta-information in the header of the document. (Those that do not contain an explicit expiration tag have an implied expiration date of thirty days after the creation date..Since this is too short for most users, they tend to include proper expire dates in their headers.) An in-house developed "web-bot" is set to recursively scan every night the entire tree of documents served through our server and, as it finds out-of-date documents, move them to a *www-expired* directory in the author's root directory. The removed document will be replaced with a new document (including the author's e-mail address stripped from the expired WWW document) stating that the information previously there has expired and that interested parties should contact the author for more information.

An example of another issue which faced the core team was how to accommodate student publishing of personal pages. We realized that if we ignored the desire of students to publish, they would find other ways by installing server software on cluster or private machines and running multiple WWW servers all over campus. Since the economies of having just one WWW server on the campus are quite clear, we realized we must somehow facilitate and support student content. Notre Dame being a private religious school with a conservative administration, we were concerned with properly distancing student publishing from official efforts, while at the same time making it easy for students to explore the fun and benefits of personal publishing. We decided to allow them to serve personal documents through the official server, but to keep all "non-official" publishing in one area of the server with clear disclaimers and delineation from the "official" content. It remains to be seen how well this strategy will hold up.

THE SERVICE: INNOVATIONS AND STANDARD FARE

In addition to the standard informational pages, Notre Dame's server currently supports some innovative features. We are developing a mechanism to serve custom home pages for users, generated on the fly and based on their corporate data. The Notre Dame home page contains a form which allows users to enter their file system ID and password. That information is verified, and then based on the user's identity and data about them obtained from the corporate mainframe, a multi-part custom page is generated on the fly and returned to them. The top of the page has their personal information including, if available, their picture. Any University-wide notices or announcements are stripped in next, and then a section contains special links to their particular college if they're a student or faculty notes if they're on faculty, links to their residence hall's home page, and so forth. Finally, the server checks for a *homepage* file in the user's root directory, and if it finds one, it strips in whatever HTML the user wishes to appear on their personal home page.

We are exploring a number of forms-based services for implementation. One that has been developed is the Student Government used book database. Students will be able to advertise their own and search for offered used books through a forms interface in WWW. The Registrar is exploring forms-based course registration and the OUC is also exploring allowing campus users to register for computer training via WWW.

With student publishing as an explicitly supported part of our WWW charter, students are developing some radical and imaginative WWW pages. But even more exciting is the faculty development of innovative WWW-based courseware. One faculty member allows his students to validate their identity and have their up-to-the-minute class grades E-Mailed back to themselves. Others are posting links to on-line libraries of information in addition to their own content and course notes. Even things as simple as posting pictures of teaching and research assistants for students to see have been tremendously popular.

REFLECTIONS AND GROWTH

Starting as a "renegade" project, attempting and failing to formalize according to traditional hierarchical development strategies, and having to switch modes back to a "leaner, meaner" team forced us to reexamine our development processes. In this time of decreasing product cycles, it is clear the old models for service development and introduction must be rethought and retooled to move as quickly as the technology changes.

In our implementation of WWW at Notre Dame, we have been able to use only the necessary, highly motivated and properly skilled people to bring the service from test mode into a production-quality service. We have done this largely without teams or committees and without wasting the time of people who did not need or want to be directly involved in the development process. And, by cutting out the administrative overhead of requiring all proposed actions approved and reviewed by teams and committees, the development was able to move much faster, stopping only at necessary checkpoints to ensure that all pieces were developing according to the same plan.

By not being an officially recognized project, however, much of the development was relegated to our so-called "spare time," and often was suspended when other, more pressing needs arose. While this is unavoidable to some degree in a constantly hectic work environment such as a computing center, it did prevent us from doing as much development as quickly as we would have liked.

There are also some concerns about the ability of older machines to be able to handle all the various media types delivered by Mosaic. While we have an aggressive upgrade strategy for faculty and public workstations, there will continue to be older model, non-color, low memory machines on our network for years to come. We must keep these machines in mind as we develop services on WWW.

Where to from here? The future of WWW looks very bright, and we hope to continue to find innovative ways to use the WWW vehicle and medium. Forms-based services, coupled with robust and secure authentication services could allow much university paperwork to be done online. University publications which are distributed to all students or faculty and staff could be published online instead, allowing constant access to the most up-to-date information, without most of the costs of traditional publishing and distribution. Already some schools have received government approval to distribute required information electronically. As client and server software both improve, we look forward to implementing these and still undreamed-of services for both our campus users and the global Internet community.

POSTSCRIPT

Given the rapid pace of development and change examined in this paper, it is likely that this very paper, having been written four months before it was presented at SIGUCCS 1994 User Services Conference, is out of date as you read this. However, an up-to-date online version of this paper and the accompanying graphics will always be available on WWW at *http://orange-room.cc.nd.edu/AboutMWM/Publications.html*.