Approved and recommended for acceptance as a thesis in partial fulfillment of the requirements for the degree of Master of Computer Science.

Special committee directing the thesis work of Daniel R. Cosley.

Thesis Advisor — Date

Member

Member

Department Head — Date

Received by the Graduate School Office

Date

MaSH:
Making Serendipity Happen

Daniel R. Cosley

A thesis submitted to the Graduate Faculty of

JAMES MADISON UNIVERSITY

in

Partial Fulfillment of the requirements

for the degree of

Master of Science

Department of Computer Science

May, 1999

Acknowledgements

Although it often feels like a thesis is an individual effort of Herculean proportions, it would not have been completed without the help of the following:

- **Mark Lattanzi**, thesis advisor, who read the thesis several times, and who actually got me to start writing the thesis—very important step, that. He made nice comments, too, that kept me going at the writing process.
- **Chris Fox** and **Bob Tucker**, who read the thesis more than once and made constructive suggestions and nice comments as well.
- **Sue Cosley**, who read it twice but only fell asleep reading it once. Thanks, hon, you are a fine (if sleepy) proofreader.
- **Susan Hastings**, who reassured me that the library would actually accept the thesis.
- **All the CS138 students**, without whom the data would have been rather thin.
- **Everyone else** who had to listen to me complain about the entire thesis process.
- **Anyone** who peruses this thesis after it hits the shelves. You must be desperate, and I salute you. I hope it proves useful.

Table of Contents

Lists of figures and tables

Figures

Tables

Abstract

The size and scope of the World Wide Web is both blessing and curse. Locating information is difficult: search engines are hard to use, and the Web is organized by author rather than by topic. Finding **good** information is even harder. One of the primary ways people win on the Web is by following the recommendations of others. Another is through the accidental or fortuitous discovery of resources, also known as serendipity.

MaSH, or Making Serendipity Happen, is a system that allows groups of users to share their useful discoveries in a Yahoo!-like topic directory. Unlike Yahoo!, however, users can add, change, delete, rank, categorize, and annotate links. This system is intended for online communities to build collective resources. In particular, interest groups, clubs, university classes, research teams, and other relatively small groups with shared interests can benefit from the software. The goals are to build communities and topic resources appropriate to those communities, to provide a tool for educators, and to make serendipity happen—or at least to help it along.

MaSH exists at the junction of several threads of research: online bookmarks, Web directories, collaborative filtering, and C*omputer Mediated Communication* (CMC). These ideas helped to drive the construction of the first version of the software. This software was used in a case study, supporting a freshman-level university course in computer science. The system was well-used (about 55,000 accesses by 140 students) and provided several concrete benefits to teaching the course. In particular, it allowed for rapid and simple dissemination and collection of information. Students used the system above and beyond the course requirements in support of their interests.

Usage trends and incidents in the case study supported conclusions from the research about both the difficulty of getting contributions and the potential to build communities. The study also suggested that MaSH can achieve its goals and pointed to directions for further work. A public release of an improved version of the software is needed to demonstrate its wider usefulness. Creating quality

metrics for the resources built and comparing them against other Web resources like search engines and hand-made directories would be a fertile area for research as well.

1 Introduction

1.1 Description and goals

This thesis is concerned with MaSH, a software tool designed to Make Serendipity Happen. The ideas behind it are simple.

- Finding things on the Web is hard.
- Finding good information is even harder.
- People win big on the Web through serendipity.
- Web users could share their discoveries.
- Online communities are growing.
- The Web has great promise for education.

Why not help things along?

MaSH allows groups of users to explore the Web and share their discoveries through an open, hierarchical topic directory. Users can add sites to the directory, and rank and comment on Web sites that have been added. The goals of the software are several: to help people locate resources on the Web, to support virtual communities based on shared interests, to allow these communities to create Web resources tailored to their interests, to identify particularly good or useful resources, and to provide educators with a tool to support both traditional classrooms and distance learning. The goals of this thesis are a little more modest: to document the practical considerations and the research that led to the idea for the software; to build the first version of the software; and to demonstrate its potential benefits. To this end, the software was used in a large case study, supporting a freshman-level university course in computer science.

The rest of this introduction will flesh out the list of ideas above. After the introduction, relevant works will be explored and academic debts paid. The discussion of the research will develop a picture of what the system should do. A short section on the system itself comes next, where its design, architecture, and limitations are discussed. A much longer section detailing the setup and results of the case study follows, along with a discussion and analysis of these results. Particular attention will be given to whether the results of the case study agree with the research and whether the system has the potential to achieve its goals. Finally, conclusions will be drawn and directions for further work explored. A glossary follows the body of the paper. Readers who take a moment to scan it before continuing may benefit from the additional context it provides.

The first task is to develop the background and rationale for the thesis. Three real-world problems on the World Wide Web played major roles in shaping the ideas behind the software.

1.2 The jungle of information

Finding relevant, high quality information on the Web is not a novel problem. Strategies from personal bookmarks to keyword search engines, from directories to links pages, have been used since the early days of the Web to clear paths through the jungle of broken links, vanity pages, and out-of-date or incorrect information. Surfing the Web is a popular pastime, and the Web is well suited to browsing and the serendipitous discovery of resources. A student searching for information on Henry Ford could go via one link to a discussion of the forces behind the Industrial Revolution—which might in turn become a prelude to the music of the late 19th century.

Lost in this journey of discovery is the original purpose: finding information about Henry Ford. Part of the hypothetical problem above is a lack of focus on the researcher’s part. But the primary problem is structural: information on the Web is organized by page and by Web site, rather than by topic. The strategies listed above attempt to address this structural reality, and all are useful in their way. Bookmarks allow individual users to tame the site-oriented Web by organizing unrelated links into hierarchical folders of topics. Search engines gain their power by drawing upon vast numbers of pages, though their judgements are often suspect (Leighton & Srivastava, 1997). Web directories such as Yahoo! and the Open Directory Project combine the first two approaches, using the knowledge and judgement of many people (ODP's motto: "HUMANS do it better") to find and organize useful sites. However, these directories generally index far fewer sites than the keyword search engines. All of these strategies attempt to address a fundamental problem: the amount of information available is vast, but the amount of relevant and valuable information is much smaller.

1.3 People power

The Web directory approach has several limitations. First, the content is filtered through a small group of reviewers. This can lead to bottlenecks in adding sites to the directory (Oakes, 1998); it also means that directories generally index far fewer documents than a typical search engine. The second problem is that directories, especially editorially-controlled ones like Yahoo!, suffer from a "Hotel California" phenomenon: documents check in, but they can never leave. With few editors and many sites to add, there is little incentive to revisit links already added. Sites and pages disappear, or their content changes, leaving broken or miscategorized links in their wake. Finally, these directories do not generally rate the quality of sites; links are typically presented in alphabetical order.

The major problem when using directories and search engines to find quality information is that the notion of "quality" is missing. A strategy for using the superior evaluation ability of humans has been developed. Instead of passing documents through an editorial board or indexing program, document evaluations can be collected from readers, a process called *collaborative filtering*. This can be done automatically or manually and can range from evaluations as simple as "yes or no" to full-text annotations. These collaborative filtering systems, also known as *recommender systems*, combine user input with clever algorithms to recommend, retrieve, rank, or reject other documents. The beauty of this idea is that users already have opinions on what they read. Resnick et al. put it nicely when building GroupLens, a Usenet recommender system. "Right now, people read news articles and react to them, but those reactions are wasted. GroupLens is a first step toward mining this hidden resource" (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994, p. 186). The trick, of course, is in the mining.

Researchers have developed several such recommender systems whose strengths and weaknesses will be discussed later. Collecting readers' reactions is a promising strategy for finding useful information.

1.4 The knowledge community

Another consideration is the direction in which information flows over the Web. There is no doubt that the Web can be more user-friendly than traditional media. Hypertext done well allows the reader to choose where to go, to organize information according to her needs, to blaze her own trails. The Web's combination of hypertext empowerment with the extra zip of multimedia (pictures, sounds, movies, etc.) is a true communications revolution. The reader's mind and needs now take precedence over the old order's linear presentation of static, preordered data.

But this mighty revolution is not enough. The communication still flows—mostly—in one direction, from author(s) to reader. Interactive technologies such as CGI, Java, Javascript, DHTML, ActiveX, and their ilk try to fill the void. However, even when Web pages are generated or chosen based on user input, the communication is still one-sided. The author provides content and the reader partakes. Not that this is necessarily bad, of course. Gutenberg and that crowd have been in the one-way communication business for a long time, with mostly positive results.

One-way is not the only way, though. Educators have researched how computers can enhance education, from online courseware to individualized study (Hiltz & Wellman, 1997). Much of this research takes place under the rubric of *Computer Mediated Communication* (CMC), and proposes that using the computer to merely present information is inferior to using computers to facilitate active and group learning. J. Hill (1997) claims that "most course-based or learning sites simply post course materials. Use of the Web as merely an 'electronic book' falls far short of the potential the medium affords" (p. 75).

If Hill's claim is true, then using a Web site to post copies of lecture notes and syllabi is a fundamentally deficient use of the Web for education. Instead, online education should focus on creating interactive spaces: e-mail lists, bulletin boards, newsgroups, chat rooms, group notebooks, and the like. These spaces provide an environment where students can solve problems and work together. Many systems provide “courseware,” a full-scale environment that can replace the traditional classroom. One in particular, the Spinalot system, contains a subsystem that is much like the one developed for this thesis (Smith, 1998). Other tools are more focused, concentrating on one task or another that supports a traditional classroom. One thing both types of tool have in common is a belief in the potential for using computers to improve education.

1.5 Making Serendipity Happen

MaSH develops a new way to organize content on the Web, offering solutions to problems and incorporating the ideas discussed above. A group of users shares a Web page that offers a space for collecting information about useful sites. The sites are organized using the familiar bookmark metaphor. All users can add, delete, and rank links; create topics; and add comments on any site in the directory. Simple keyword searches select potentially useful topics and links in the system.

This system has a number of uses. One is for educators. Teachers can use the system as a gathering place for students to share the results of their Web explorations. This use can support and strengthen communication and provide a venue for group research and learning. Another use is building detailed, annotated, and useful topic-centered directories. Hopefully, the quality of these resources will stack up well against search engines and “traditional” Web directories. By encouraging the sharing of useful information, the system can help to support and strengthen virtual communities. When members of a community contribute to a common cause, the community is stronger. Each community can have its own directory, with information relevant to and ranked by the specific interests of that community. The individuals benefit, too. Everyone in the group can make unexpected, useful discoveries through the experiences of others. In other words, the system works to make serendipity happen. The name of the software, MaSH, is derived from this vision.

MaSH draws on the three themes outlined in this introduction. It serves as a dynamic, accessible, and flexible repository for the quality links found by its users. Browsing and search interfaces help new users to locate (and old users to re-locate!) the goods, and allow users to benefit from the discoveries of others. By allowing users to rank and comment on links, MaSH can help the better information rise to the top. By being so open, it encourages users to contribute. Users can add, move, change, and delete items in the directory. This should help with the problem of stale or out-of-place links. This openness, combined with the ability to contribute toward links others add, allows users to collaborate—strengthening communities and supporting educational uses.

2 Background and research

Like any other work, MaSH stands on the shoulders of those who have come before. This section discusses ideas and work that contributed to this thesis. Current technologies for finding information on the Web are discussed, including search engines, directories, and bookmarks kept online. The idea of collaborative filtering are discussed in some depth, followed by a brief discussion how computers can support education. Most sections include a brief comment on how the idea ties in to MaSH. Be forewarned that this section cites many online examples, which change regularly and rapidly.

2.1 Keyword search engines

Part of the motivation for developing MaSH was the difficulty of locating information on the Web. Keyword search engines such as AltaVista are one of the primary vehicles Web surfers use to find information. These engines typically use automated programs, called Web spiders (also “crawlers”, “bots”, “robots”, and “agents”) to discover new Web pages, recursively following links they encounter. The pages are sent to an indexer, which analyzes each page and adds information about it to the search database. Finally, users submit queries to the search engine. The search engine assembles the results, and if the query is well-formulated, a few pages, or perhaps a few million, are presented to the user. If the user can then visit the link (as opposed to getting the too-familiar “404 File Not Found” error), and the page contains the relevant information, the search succeeds.

Perhaps that description is a bit unfair, but it is hard to argue that keyword searches constitute an efficient way to locate resources, especially for a novice. Using these systems requires a number of skills, including the ability to construct good queries and some knowledge of the subject (Brandt, 1997). This causes a bootstrapping problem—locating information about a topic requires some knowledge of the topic, but finding that knowledge is the whole point. Having domain knowledge does not ensure success, either. Queries can produce thousands (or millions, for very broad queries) of hits, most of which are probably not what the searcher wants. A number of these will be irrelevant, and an additional number will not even be reachable because the server or page no longer exists.

Of course, most searches will end with a modicum of success at a Web site that has some relevant information. This is often not enough, especially for research. Finding several good sources for a topic can be very difficult. "Several" is a problem because the Web is organized by site (essentially, by author), rather than by topic. Even when a site with relevant information is found, there are no guarantees that it will lead to additional information. "Good" is even harder, because the idea of quality is missing from this process. The best that a good search engine with a good query can do is strive for relevance—that the results returned pertain to the words in the query. Quality is another matter entirely. Different searchers need different things—many pages or one? introductory or advanced material? specific facts or general overviews? This makes it hard for one quality metric to make a claim to be "best." MaSH can help with this many-faced aspect of quality by allowing small groups to evolve their own definitions as they create their own resources.

Berghel (1997) puts it well: "Search engines as they now exist represent a primitive, first cut at efficient information access on the Internet" (p. 20). Other strategies exist.

2.2 Directories

Some Web authors distill their knowledge of resources on the Web into links pages organized by topic. Such a page can be a searching gold mine, leveraging the time and knowledge invested by others. These resources exist on several scales, from useful (if not complete) links pages maintained by devotees of a topic, to the comprehensive and extensive catalogues of resources on a topic commonly called clearinghouses. Topic-based collection and organization of Web resources reaches its zenith in directories like Yahoo!. These behemoths collect Web pages across the spectrum of human knowledge and organize them into hierarchical topic trees. These trees provide a framework for Web surfers and can guide them in their quest for information. Most such directories are centrally controlled and managed by editors. An alternative organization is provided by the Open Directory Project, which allows anyone to become a category editor.

2.2.1 Editorially-controlled directories

Yahoo! is the canonical example of a Web directory (“Yahoo!”, 1998). The premise is simple: a group of Web users employed by the company (called “Surfing Yahoos”) wanders the Web, seeking out useful and interesting sites. These sites are added to the directory. Users can interact with the directory in three ways: browsing the topic hierarchy, searching the directory with keywords, and suggesting sites to add. The interface is distinctive and has been “borrowed” by a number of Yahoo! wannabes who recognize that Yahoo! is a standard and believe that Web users will be more likely to use their service if it has a familiar interface. MaSH is proud to share in this tradition, with minor modifications to support MaSH’s additional features.

There are problems with a centrally controlled, Yahoo!-style approach. One is that links are not really rated or annotated, except with very brief descriptions and an optional “cool” identifier. Again, the notion of quality is missing. Another problem is that large swaths of the Web are omitted. Lawrence and Giles (1998) reported a “lower bound” of 320 million publicly accessible pages as of December 1997. Yahoo! employs about 80 Surfing Yahoos, indexing over 750,000 sites—or well less than one percent of the total pages (Sullivan, 1998). Even though this comparison conflates pages with sites, and there are advantages to indexing by site, this information suggests that Yahoo!’s Web coverage is far less than complete, a trait it shares with the keyword search engines (Lawrence & Giles, 1998). With so few reviewers and so many links, sites already added are not likely to be manually reviewed, leading to out-of-date or inaccurate information.

These criticisms are not absolute condemnations, of course. An editorial directory could increase its capacity by adding more resources. More employees, faster connections, and automated helper tools would mitigate the problems of sparse Web coverage and re-reviewing sites. Covering just a portion of the Web is not necessarily bad, either. Sturgeon’s Law says that 90 percent of everything is crud, and when it comes to the Web, that may be a conservative estimate. Still, many sites—potentially useful sites—are not listed in the directory, which makes them relatively invisible. Yahoo! does encourage users to submit sites for its employees to review, but this is no panacea: there have been a number of complaints about lethargic processing (Oakes, 1998).

Sites can also be excluded from directories because of editorial slant, either through explicit editorial policy or an implicit shared mindset among employees. Looksmart has such a policy, explicitly rejecting sites that contain "pornography or offensive material" ("Looksmart", 1997). This sounds a bit draconian, but it is completely appropriate to exclude certain types of material when building a resource specific to a community, a resource that should reflect its norms and needs. MaSH attempts to do this. However, putting the power to determine community mores in the hands of one or a few administrators has a darker side. Centralizing this decision-making is probably not appropriate for a general-purpose, large-scale directory of the Web, especially for some definitions of "offensive material." Stories of how AOL chat room censors and software hinder discussion of topics like "breast cancer survivors" (Quittner, 1995) illustrate the danger of standards applied from on high. MaSH puts the responsibility for determining community standards where it belongs—in the community itself.

2.2.2 Open directories

Another strategy is to construct a Web-sized directory, but put power in the hands of its users. The Open Directory Project, (formerly named GnuHoo, then NewHoo), a directory like Yahoo! but with a more open style, went online in June 1998. Founded by Rich Skrenta and Bob Truel, the site's motto is "HUMANS do it better." This simple slogan captures their basic premise: "the open editorial model is the only indexing method which can possibly scale to the size of the Web" (Skrenta & Truel, 1998). Users hold the keys to the castle in this system. Instead of just browsing the directory and suggesting links to add, anyone can apply to be a category editor. A category can have multiple editors; editors can add, delete, and change URLs in their categories. This expansion of manpower has the potential to address the limitations noted above for Yahoo!. The Open Directory Project is in the spirit of what this thesis hopes to achieve. The directory grew rapidly, with over 50,000 sites added in the first month. Netscape saw merit in this approach, acquiring NewHoo in November 1998. The directory contains about 285,000 sites as of January 1999 ("Open Directory Project", 1998).

There are significant differences between the Open Directory Project and this thesis. In particular, the Open Directory Project has no provisions for link ranking or annotations, except for a link description and an optional "cool site" tag. Being an editor also implies a sense of ownership, which could be problematic if a category has several editors who have different views on how to handle the directory. Such ownership feelings, and the accompanying problems when editors disagree, is unacceptable for MaSH, which is about building small groups who work together. To forestall strong ownership feelings, users of MaSH have access to the entire directory. Of course, this access does not address the problem of editors with divergent views. Since MaSH is targeted at small, relatively like-minded communities, this problem should be minimized.

This discussion is not by any means a complete list of directories on the Web. Those mentioned, however, are strongly representative and made large contributions to MaSH. For more examples, a keyword search on "directories" in Yahoo! yields literally thousands of others.

2.3 Bookmarks online

One standard tool for Web surfers is the bookmark manager that comes with Web browsers. These have some limitations: bookmark formats differ from browser to browser, and bookmarks on a computer at home are not very useful at school or work. Responding to these deficiencies, and to the possibility of making some cash, a spate of online bookmark managers has appeared on the Web. Two early, solid, representative examples are itList (Frankovitz, 1999) and MURL ("MURL.COM", 1998). Both allow users to add, edit, and otherwise manipulate bookmarks, not unlike the built-in bookmark managers in a Web browser. The primary advantage of such a service is that it makes a user's bookmarks easily available and editable from anywhere on the Web. The primary disadvantage is the omnipresent clickable banner ad. Some newer systems have value-added features like file storage to go along with their ad values. Both itList and MURL have password-protected user accounts to address privacy concerns. itList also allows users to make their bookmark lists public.

The systems appear to be a combination of small business and hobby, single-owner enterprises that provide the service in order to learn, for fun, and/or to sell advertising. itList, "The Web's *First* Online Bookmark Manager," was written by Jason Frankovitz. In his own words, "[it] irritated me that I couldn't use the same bookmarks on different computers. I was beginning to learn the CGI specification and Perl at the same time, so making it real was a useful way to build my skills" (J. Frankovitz, e-mail, July 23, 1998).

It is fair to ask whether MaSH needed to be written—after all, multiple users could share a user account on one of these systems, and MURL even has nested folders, important for creating a topic directory. These online bookmark systems also have an interface that is well-suited to their functionality. Both have convenient ways to add links while surfing, eliminating the problem of requiring two open browser windows. But they lack a number of the features incorporated into MaSH, including multiple annotations, easy editing of links once added, and data collection. Getting permission to use the code and modifying it was judged to be less educational, more error-prone, and less likely to be successful than designing and building a system from scratch.

Several new online bookmark services have appeared even since the first draft of this section a few months ago. The feature bar is being raised: MURL can export to Netscape's bookmark format; Webfavorites allows for online file storage (up to five megs) but requires a browser plug-in ("About Webfavorites", no date); Clickmarks filters a user's least recently used bookmarks but collects personal information to target its advertisements ("Help/FAQ", 1999). Everyone has a business model, and it should be interesting to see how these online bookmark managers sort themselves out.

2.4 Collaborative filtering and recommender systems

The directories discussed above have little provision for link ranking and quality; the bookmark managers are not really about link sharing. Since one of the goals of MaSH is to evaluate, locate, and share the best Web pages, it needs something more. A number of systems have been developed that take user input and ratings for a set of documents (Web pages, newsgroup articles, movies, etc.) and use those ratings to filter, rank, or otherwise recommend some documents over others. Tapestry, held to be the first such system by Resnick and Varian (1997) in their overview of the topic, called this activity "collaborative filtering." Resnick and Varian prefer the term "recommender system", as they consider recommending to be a more general and appropriate term for this sort of activity.

2.4.1 Tapestry, the granddaddy

Tapestry (Goldberg, Nichols, Oki, & Terry, 1992) was developed at Xerox PARC in response to the problem of "users being inundated by a huge stream of incoming documents" (p. 61), primarily e-mail. The plan was to create an archive of electronic documents. Users of the system could query the database with standard keyword, date, and author searches—and also based on annotations that other users had given. This uses the input of others to helpdetermine whether an article was "good" or "bad" or "interesting"—or any other criterion supported by the annotations. The system itself was rather complicated, and in the authors' own words, "missing a few important pieces" (p. 69). These included a practical way to browse the results, and Tapestry seems to have not escaped the lab.

Tapestry's primary idea of using human input, however, started a small cottage industry around the idea of collaborative filtering. Goldberg et al. (1992) also foresaw one of the major issues for these systems, drawing on their observations of posters on Usenet:

> We envision two types of readers for various classes of documents. Eager readers will read all the documents in the class in order to get immediate access. More casual readers will wait for the eager readers to annotate, and read documents based on their reviews. Experience with NetNews suggests that there will not be a lack of readers willing to be 'eager' annotators. (p. 69)

2.4.2 Under-contribution

This rather glib statement hides an important question: what is the payoff for users to contribute ratings? Goldberg et al. thought this would not be a problem. Systems that followed Tapestry, however, cast doubt on their prediction of "eager annotators." Resnick et al. (1994) built GroupLens at the University of Minnesota in order to apply collaborative filtering to Usenet articles. Users can assign each article a rating on a scale of one to five. Others' ratings are used to recommend articles. A nice feature of the system is that instead of automatically throwing away articles, GroupLens just reports the recommendations. Newsreader programs can use the GroupLens API to receive ratings, and then each client can determine how to use them. Articles could be sorted by rating, low-rated articles can be marked as read, etc.

However, even with an interface that required nothing more than typing a digit to rate an article, the GroupLens team found that gathering enough ratings was a problem in a larger-scale study (Konstan et al., 1997). Part of their difficulty was ascribed to technical reasons (users only read a small portion of high-volume newsgroups), but "informal feedback suggests that users are 'lazy' in that they would prefer not to even think about the appropriate rating" (p. 84). This laziness persists despite the fact that GroupLens can make more accurate predictions for an individual if it has a sample of ratings by that person. Avery and Zeckhauser (1997) use game theory to argue that enough evaluations will *never* be produced without some external reward (i.e. cash). Their argument models the costs and benefits of ratings between two users as a sort of prisoner's dilemma, and it is not clear that this model is appropriate. Another point they raise is much better: evaluations will be performed not by a representative sample of the group, but by people who enjoy rating things. Yet another caution for collecting ratings comes from Resnick and Varian (1997), who warn against content providers who might spam for their content and against competitors. MaSH does not attempt to solve these problems directly, though a technical solution for the spam problem is discussed along with the design of the software. Administrators can, with some effort, use contributor names to measure each user's input and create incentives.

2.4.3 Implicit ratings

An approach suggested by Konstan et al. (1997) that addresses the above concerns is to capture "implicit ratings" (p. 84). In other words, the time spent on an article, whether the reader replied, and other criteria that could be automatically observed could be used as indicators of like or dislike. Alexa, founded in 1996 by Brewster Kahle and Bruce Gilliat, does this in an interesting way, by tracking the paths that users follow through the World Wide Web and using this information to establish connections between sites ("Alexa", 1998). The Alexa software works alongside a Web browser. Information about which links a user follows (the "paths") is sent back to the Alexa server. When a user asks for a suggestion, the server can respond with the most popular next destinations.

This is not necessarily a good strategy; guesses as to which links to follow are just that—guesses, and often wrong. But the idea has merit: consider an analogy between these paths through the Web and well-worn paths through a dense forest. Dinosaurs no doubt followed well-trod paths to the tarpits as well, but even if a researcher decides that following the paths of others is not fruitful, Alexa captures other potentially useful information. This information includes time spent on Web pages, site update frequency, and number of visits to a site. Like Alexa, MaSH captures the number of visits for each link in the system, and this information could be used toward ranking links.

Another system that uses a form of implicit ratings is PHOAKS (Hill & Terveen, 1996). Here, the posters to an individual newsgroup are regarded as a "community of experts" (p. 111). URLs are often posted as part of a message, and although many of these are not useful (almost half are included in signature files), a certain number are recommendations to visit certain Web sites. If the chaff can be filtered out (and according to Hill and Terveen, it can), these recommendations could be captured and turned into human-selected, topic-centered Web directories. These can be accessed by newsgroup name or through a keyword search at the PHOAKS Web site, though the search facility was broken when this was written ("PHOAKS", 1996).

2.4.4 Other concerns

A common theme runs through all of these systems: simplicity of contributing ratings. GroupLens requires the pressing of one key to generate a rating; PHOAKS and Alexa require no active input at all. WiseWire, developed by Ken Lang at Carnegie Mellon before going corporate, asks users whether they liked a page. The possible answers are “yes”, “slightly”, and “no” (“WiseWire”, 1998). Two factors dictate this attention to ease of rating. One is the problem of getting ratings in the first place, as discussed above. Another is that making use of more complex ratings like text annotations is problematic. A program will have an easier time calculating with “5”, “3”, and “1” than it would with “This article was great! Wow!”, “Not too bad, if you’re a Green Bay fan”, and “I hated the materialist dialectic that dominated this discussion.” Alexa provides more information than a rating, but it collects this information automatically. The moral is that ease of contributing is important. MaSH offers a simple rating scheme by allowing users to move links up and down a list. It also solicits the more involved text annotations to support its goal of facilitating team research.

Collaborative filtering systems share two other, related concerns. One is whether to make general recommendations or to try to tailor recommendations to individual users. For example, Imana, Inc.’s SiteSeer takes a user’s bookmark file, and uses the folders it contains to find “similar” folders from those it has seen before. These similar folders are then used to generate user-specific recommendations (Rucker & Polanco, 1997). GroupLens can do it either way. It can generate predictions based on community averages. However, it can also generate predictions for a user from the predictions of others who have agreed with the user in the past. Konstan et al. (1997) find that user-specific predictions are more accurate in the sense that the predicted rating correlates more closely with the actual rating a user gives an article than predicted ratings based on a group average. They hold that their data suggest that “approaches that simply model users uniformly across domains are diluting their predictive power” (p. 84).

On the other hand, neither PHOAKS nor Alexa attempt to tailor their recommendations based on user profiles. In PHOAKS’s case, the system does not even collect information about who accesses the data. Alexa could at least theoretically be written to take the similarity of users into account when generating its related sites list. However, matching individual users is both more complex and more computationally intensive. Konstan et al. (1997) discuss ways to scale GroupLens up from its trial size of 10,000 users and 20 newsgroups to cover all of Usenet; all of these require extra complexity, power, or both. WiseWire, which supports personally tailored wires, also has a complex architecture ("WiseWire", 1998). The decision whether to make personalized recommendations should take into account whether this will add significant utility, and whether the utility is worth the additional system complexity. MaSH does not make personalized rankings. Doing this would require collecting much additional information and does not fit well with the metaphor of a shared bookmark list.

The other related concern is user privacy. Suppose Alexa is recording one's Web path wending through a number of "adult" sites. Or maybe when rating documents in GroupLens, a user always downgrades a colleague's posts. Neither of the above users would probably like it if their name were attached to these activities. Even PHOAKS, harvesting information from public newsgroups, was concerned enough about privacy to decide not to recommend mailto URLs (Hill & Terveen, 1996). Most collaborative filtering sites have a strongly-worded privacy policy; see Alexa for an example ("Alexa", 1998). MaSH respects the privacy of its users by not requiring usernames or personal information to access the directory. MaSH does allow contributors to enter a name, but there is no identity-checking, and the entered name can be a real name, an alias, or the default "Anonymous."

2.4.5 Comparison of filtering systems

Although the systems address somewhat different areas, a summary comparison of relevant features of the recommender systems mentioned above would be useful. Table 1, below, has such a comparison. It lists the systems mentioned in this section, along with the type and source of documents they handle, their rating mechanisms, and how the ratings are presented. Resnick and Varian (1997) have similar, more detailed, tables for the systems they discuss in their overview.

System	Documents		Ratings			
	Domain	Source	Votelike	Implicit	Other	Display
Alexa	Web	Spider, user input	—	time spent, hit count, link paths	—	Next destination; link info option
Grouplens	Usenet	Rated articles	1, 2, 3, 4, 5	—	—	Varies by newsreader
MaSH	Web	User input	up, down	Hit count	Text	Ranked Yahoo!-style directory
PHOAKS	Web	Usenet mentions	—	Mention in articles	—	Searchable index, list
Siteseer	Web	Bookmark files	—	Organization of bookmarks	—	List
Tapestry	Mail lists	List subscription	—	—	Text	?
WiseWire	Web	Spider, subscriptions	yes, no, slightly	—	—	Ranked list

Table 1. Features of collaborative filtering systems

2.4.6 Special-topic recommendations

In addition to the general systems discussed above, many special-purpose systems exist for making recommendations in limited domains such as music, movies, and books. Some work like the systems described above. MovieLens, an outgrowth of the GroupLens project, applies a GroupLens-style voting system (one to five stars) to making movie recommendations. As with GroupLens, users build up profiles based on ratings they give, and are then matched up with other users of similar taste in order to generate more accurate predictions (Herlocker, 1998).

Amazon.com has multiple recommending systems. One, BookMatcher, is like MovieLens; readers can submit ratings and receive personalized recommendations. Another lists items that have been purchased by people who bought an item that a customer is considering. A third allows customers to read and enter text reviews (along with five-star ratings) for items that interest them (“Amazon.com”, 1999). Some titles have hundreds of reviews—Ayn Rand’s *Atlas Shrugged*, for example, has over 450 at the time this was written. This rating facility is much like the comment facility that MaSH provides for users to annotate links.

2.5 Computer Mediated Communication

One of MaSH's primary purposes is the creation and support of user communities. Computers have been used to enhance communication for some time (Hiltz & Wellman, 1997), but the field has really taken off with the growth of the Internet and especially the Web. *Computer Mediated Communication* (CMC) goes beyond computer conferencing and refers to "any form of organized interaction between people, utilizing computers or computer networks as the medium of communication" (Romiszowski, 1997, pp. 32-33). This is a broad definition, including e-mail, discussion lists, newsgroups, the Web, videoconferencing, MUDs, IRC, BBSes, "groupware" like Lotus Notes, and any other special-purpose tool created that makes the computer an accessory to a conversation. MaSH is one such tool.

2.5.1 Community building

One of the features of CMC is that it can be used to foster communities. Hiltz and Wellman (1997) have an excellent discussion of how CMC-supported communities resemble and differ from traditional communities. Two points stand out for the purpose of this thesis. The first is that these virtual communities are better suited for exchanging factual information than for argument or emotional content. The second is that, unlike traditional communities which are built on geographical or family ties, these virtual communities are built on shared interests. This point makes sense: picture cyberspace as a sort of virtual landscape where ideas shape the terrain. The parallel between geographic closeness and closeness of interests follows easily. MaSH creates little spaces for people to congregate around their shared interests, and by focusing on the exchange of information, avoids the need to support problematic emotional communication of the type described above. Its spaces are more library carrels than coffeehouses.

2.5.2 Learning goes online

These spaces are also not much like classrooms, although—not unlike library carrels—they can be used to support a classroom. The asynchronous nature of CMC makes it suited to educational uses. Hiltz and Wellman (1997) use the term *Asynchronous Learning Network* (ALN) to refer to virtual classrooms where CMC is used to foster a community of teachers and learners. Khan (1997) gives a wide-ranging overview of the specific potential benefits online learning offers, including virtual communities and the potential for collaborative learning. Another excellent overview of CMC in education is given by Hartley et al. (1997). Of particular interest to this thesis is their claim that CMC can be used either to replace the traditional classroom (e.g. distance learning) or to enhance it by "[adding] new dimensions to the typical class in a variety of ways" (p. 221).

Much of the research on educational tools focuses on replacing the traditional classroom with a distance learning environment. Colleges and universities, in particular, are interested in using this technology to support non-traditional students, increase enrollments, use new technologies to improve education, and extend their reach. Penn State's program offers this vision:

> Distance Education is a University-wide function that enhances Penn State's ability to serve students in all parts of the world, increasing student flexibility regarding the time, place, and pace of study and creating a highly interactive, learner-centered environment that is marked by increased access to faculty expertise and increased access to information resources. Distance education is integral to the research and service elements of the University's mission; it helps the University reach out to a broader community and, at the same time, to bring worldwide expertise to Penn State campuses. Distance education is not simply the addition of technology to instruction; instead, it uses technology to make possible new approaches to the teaching/learning process ("Definition", 1997).

A number of these "courseware" systems exist. Interest in these systems seems to be greater in countries such as Canada and Australia where barriers to communication like climate and distance make it hard for students to meet in a classroom. Landon (1999) maintains a Web site with detailed reviews and links to sites for many such tools. One system not listed by Landon, but that should be mentioned here, is the Spinalot system built by Tom Smith. It is designed to help communities "collaborate, organize, create, discuss and publish in a personalized media-rich environment" (Smith, 1998). The system is in a state described as "early beta," and a demo version is online. Its features include a general mechanism for adding content, including Web pages, discussions, media clips, personal profiles, and annotated bookmarks. These bookmarks could be organized in a topic hierarchy and work much as MaSH does.

However, Spinalot requires logging in, maintaining personal information, managing security, and learning umpteen features. These characteristics are all bad in the context of MaSH's design goal of simplicity, and point out the disadvantages of a full-scale courseware tool. The need for technical support, the time and skill required for online course development, and the learning curve for both students and instructors must all be weighed when deciding whether to use a courseware product. For teachers who just want a supplement to their classroom, using a courseware tool would be akin to killing a flea with a sledgehammer. MaSH is a much lighter tool, squarely positioned in Hartley et al.'s second category of enhancing the traditional classroom. In deciding how to provide online education, researchers have analyzed courses and broken the activity of running a course down into a number of distinct tasks (Cordani & Tucker, 1998; Hartley et al., 1997). MaSH can support or enhance a number of these tasks, including increasing (and possibly measuring) student participation, allowing submission of online assignments, facilitating basic online discussions, and supporting collaborative research and learning.

3 MaSH: design, features, and limitations

Berghel (1997), who calls keyword search engines “a primitive, first cut” at the problem of finding information (p. 20), saw commercial information providers as one possible answer. These providers would “grade, rank, review, append, annotate, transfix, collect, and re-package Internet resources” (p. 23). MaSH does not approach this lofty goal; instead, it is a tool designed to help groups perform these functions for themselves. This section outlines the features MaSH supports today, and those it would like to support someday. A short discussion of major decisions that drove the design is next. Finally, limitations and concerns with the current software are acknowledged.

3.1 Features

MaSH exists at the junction point of concepts described during the literature review: a searchable, online topic directory that resembles bookmarks but with additional facilities for collaboration, meant to help advance education, build communities, and provide users with another tool to bring order to the Web. Like the bookmark managers built into Web browsers, it allows the creation of hierarchical topics and the addition and sorting of links. MaSH provides two additional features. First, any user is able to access and edit the bookmark list by adding, changing, and deleting links. Second, users can easily annotate and rank bookmarks so that more useful links can quickly percolate to the top of a category. These features combine to create a dynamic information resource where users can benefit from the experiences and judgement of everyone who uses the system.

Other features of the system include:

- a simple internal search engine;
- simple verification that entered URLs have a valid format;

- checks to reduce the possibility of duplicating links within a topic; and
- collection of statistics, including link hit counts by MaSH users and activity logging.

Features that would be useful for future versions include:

- automatic link verification for new sites and as a regular check on sites in the database;
- capturing last modified dates during link verification;
- a utility for importing and exporting bookmarks from browser bookmark formats;
- a utility for using search engines to quickly populate topics; and
- cross-listing of links in multiple categories.

3.2 Software-specific issues

A brief discussion of the high-level design considerations involved in building MaSH is included here. The primary design goal was that potential users actually adopt the system. This was judged most likely to happen if the system installs easily, looks familiar, and requires a minimum of effort to use and maintain. As discussed earlier, the research on collaborative filtering suggests that even small amounts of effort can determine whether users make contributions. In light of the claim that "schools don't want software, they want curriculum" (Soloway, 1998, p. 13), a standalone tool such as MaSH had better require a minimum of learning and maintenance. Familiarity, simplicity, and ease of use drove most design decisions.

3.2.1 Yahoo!-like user interface

MaSH is conceptually like a shared bookmark list; links are grouped into a set of nested folders, and are ordered within a folder by moving them up and down. Although bookmarks should be a familiar concept for most Web users, even the term "nested folder" implies some sort of weird complexity. An even more familiar idea is that of browsing through a directory—and more specifically, browsing through Yahoo!. So, for maximum familiarity and ease of use, MaSH borrows heavily from the popular and familiar Yahoo! interface, with minor modifications. The primary interface is shown in Figure 1.

Figure 1. The primary MaSH interface

Simplicity is served by making common functions easily accessible. Visiting or ranking a link is a one-click operation. More complex functions like adding or reclassifying items require more effort. Updates take effect immediately, supporting the bookmark metaphor. This is in contrast to the WiseWire voting system, where ratings are processed through a series of filters in an effort to make better decisions about what the ratings mean (“WiseWire”, 1998). Voting is also a familiar metaphor, but WiseWire votes have no immediate effect on how links are ranked in a "wire" (category). The immediacy of seeing the link move is more appropriate for MaSH's bookmark metaphor.

3.2.2 Interface for contributions

Since Yahoo! has no interface for adding or changing link information, or for contributing comments, MaSH had to come up with its own. Ranking links is simple; the up and down arrows in Figure 1 tell the system to move that link up or down one spot in the list. This could be tedious for users who wanted to move a link many spots, and a better interface would help. The interface for adding comments is straightforward (if a bit awkward on screen), asking for a comment title, an optional name, and the comment text. The interface for adding and changing links is also straightforward; see Figure 2.

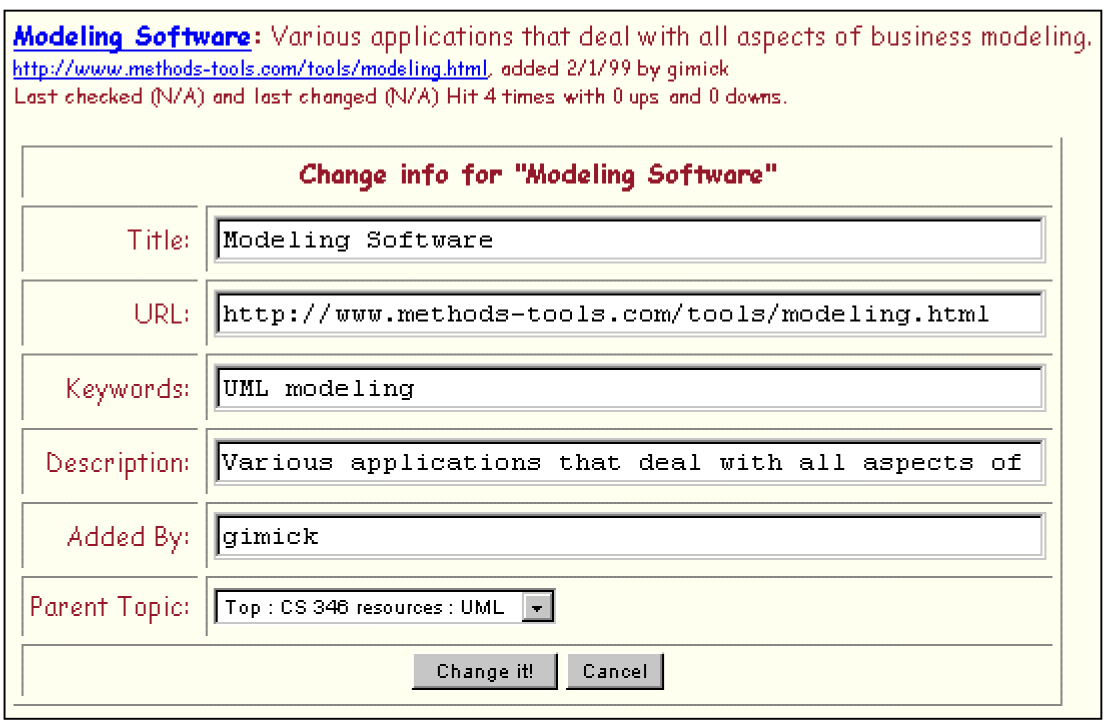

Figure 2. Changing (or adding) a link in MaSH

This interface is not perfect. For one thing, the form should include an easy way to add an initial comment. Another concern is that adding links is not as convenient as it should be, requiring two open browser windows. A smaller version should probably be created.

3.2.3 Administration and installation

Administering a MaSH system involves the same sort of activity that any other user might perform: adding and deleting links, moving them between categories, and rating or commenting on a link. So there is no specialized administrator interface. Administrators do, however, have control over how the system looks and behaves through a well-commented settings file. An except from the settings file is shown below in Figure 3.

```
# $headerFile and $footerFile should include valid HTML that you would like
# to have appended to the top and bottom of the pages MaSH generates.
# If you need a generic HTML wrapper, set $needWrapper to 1.  If not (e.g.
# your header and footer files include the <HTML>, <HEAD>, and <BODY> tags),
# then set it to 0.
#
  $needWrapper = 1;
  $headerFile = "header.txt";
  $footerFile = "footer.txt";
#
# If you want to make your own custom entry page, you can do it in one of
# two ways.  A full-blown HTML page can be specified in $welcomePage; a
# file that should be wrapped up with the standard headers and footers that
# MaSH would put out for any other page should be specified in $welcomeFile.
# We use one or the other, Page first...
#
  $welcomePage = "";
  $welcomeFile = "welcome.txt";
```

Figure 3. A portion of Settings.pl

This design allows administrators to learn only parts of the system they care about, while giving them the ability to take more control if they wish. Installation should also be simple; the system consists of a number of files that are installed with the administrator’s favorite archiving program.

Quick-start instructions and reasonable defaults get the system up and running in short order with a minimal amount of effort. Copious documentation is included in the distribution.

3.2.4 CGI architecture

Simplicity determined that CGI would power the system. One alternative would be to provide a program to download. This could be a browser plug-in or a program like Alexa that works alongside the browser. Such a program could offer great flexibility and power; however, it would introduce additional complexity and impose a burden on users. An add-on program is also not appropriate in situations where users cannot install software, like many school computer labs. Another option was to use Java or Javascript to build a more powerful and interactive interface that would act more like Netscape's built-in bookmark manager than is possible with forms-based CGI. This option was rejected, mostly because Java and Javascript are not available in all places. This is especially true for one of the prime target audiences, educators, as schools often have equipment that is on the trailing edge. Even when they are available, there are frustrating incompatibilities with how different Web browsers implement these languages.

3.2.5 Perl implementation

Perl was the natural choice for the server side of MaSH. The server's tasks are to maintain a database of links, parse requests, and spew out Web pages. Perl comes with facilities that make working with small-scale databases very simple. These facilities might not scale well to a million-hit site, but are more than adequate to work with a database of several thousand links. Perl is also an excellent choice for text processing, including generating HTML documents. It comes with a powerful regular expression engine, built-in sorting, and good string manipulation. A handy built-in array data type supports applying operations to all elements, insertion and deletion at arbitrary indices, and other useful enhancements that simplify working with collections (for example, the links contained within a topic). All of these features lend themselves to simple and easy-to-write code for the task at hand. The language is available for nearly every common platform and comes bundled with many.

3.2.6 Function-based, modular design

Although non-object-oriented designs are retro in computer science, a functional decomposition was appropriate. Each of the main tasks MaSH must complete (main CGI server, data manager, display and layout, and storing settings/error messages) acted like a function, and each conveniently broke down into sub-functions. Perl does offer object-oriented syntax and facilities, and one part of the system (the Item data type, which stores information on links, topics, and comments) was implemented as an object for convenience. However, Perl's object oriented syntax and model are a bit strained. Since the program most naturally felt like a collection of functions, that is how it was designed. The functions broke down into several modules, allowing for straightforward replacement of individual pieces. It could come in handy, say, if MaSH were to be bought (along with what sometimes seems like every other interesting small-scale project on the Web) by some large portal site, and the database needed to be upgraded to handle a larger load.

3.3 Limitations and concerns

Several system limitations and concerns should be addressed. Number one on the list is "is this really useful?" In the movie *Field of Dreams*, the title character is told "If you build it, they will come" (Kinsella & Robinson, 1989). Just as plausible is a slight paraphrase of the Vietnam-era slogan: "What if they had a [Web directory] and nobody came?" Of course, educators have an advantage in this area: they can require students to use the system, though this is not necessarily harmonious with the intended goal of being a shared resource.

Another interesting and important problem is whether a group of MaSH users would come to a consensus on whether a link is "good." The rank ordering of links within a category is determined by the joint action of all users. It is possible that users will not agree on which links in a topic are better, especially when it comes to topics that involve matters of taste (Konstan et al., 1997). In such a situation, links could yo-yo up and down the rankings as their champions and critics battled. The differing needs of users might also contribute to this. Even in a community of shared interests, different members might have different views on what is useful. The problem did not happen in the case study—if anything, people did not spend enough time ranking links—but it is a potential concern.

Human nature is also a concern—some people are just plain jerks. Users could add off-topic links, or delete links out of spite. They could also make offensive comments—and since names are not validated by a password, could make them in the name of another contributor. For public MaSH directories, people or companies could add their “favorite” sites to a number of topics. A company, for example, could add an advertisement to a large number of unrelated topics, in the time-honored tradition of the infamous Green Card incident on Usenet (Campbell, 1994). This has happened to the Open Directory Project. Another potential problem is ratings inflation, which Resnick and Varian (1997) call the “vote early and often” phenomenon (p. 57). These problems should be mitigated in the small-group settings that MaSH targets, but they are not entirely eliminated.

A number of tactics could be used to combat the problems described above. Even a simple measure like adding a five- to 10-second delay after every add or delete operation would tend to inhibit all but the dedicated miscreant. A related plan would be to monitor activity by IP address, and after a small threshold of destructive activity, imposing delays just on that IP for a while. Another simple hack would be to checkpoint the database by making copies every *n* hours or accesses. Somewhat more complex would be adding a rollback mechanism (the Open Directory Project has this). On the other hand, MaSH’s extremely open system allows for the quick repair of certain types of sabotage, such as the addition of offensive information or forged contributions. An example of such self-repair happened in the case study. Moderation could be used to help with the problem of multiple additions; links that users add to the Open Directory Project sit in an “unreviewed sites” holding bin in each category until they are blessed by an editor. However, this is against the open spirit of MaSH.

One other strategy for handling user problems would be to add security. Spinalot, for example, has user accounts and logins, as well as permissions for items that are added. These tools could be used to prevent deletions and unauthorized accesses. Adding security measures, however, increases the complexity and decreases the usability of the system. While implementation complexity is not a big concern, the last thing most people need is to keep track of one more username and password. Even a small cost is liable to discourage use of the system, unless the value gained by users is worth the hassle. (This implies that for a committed community, or one that is required to use the system, logins and passwords would be more tolerable.) More importantly, granting ownership of certain links and topics in the system works against the goal of an open, shared resource. Finally, the security risks are small: some useful links might be deleted. Why put more effort into security than the system is worth? These reasons combine to make the tradeoff between security and usability squarely on the side of usability. This could be changed if security proves to be a problem in practice, though it was not found to be so in the case study.

Building a directory of Web sites, combined with the ability to make annotations, could lead to legal concerns. Suppose, for instance, that “Helpful User” comments on a site he adds by quoting extensively from the site. This is a potential copyright violation, but who is liable? Helpful User, or the administrator of the MaSH directory? A number of factors are used when determining copyright violations: the amount and nature of the use, the zealousness of the copyright owner, the extent of the fiscal loss, the skill of the lawyers, and perhaps the phase of the moon. This is not a very reassuring answer. Then there are libel and slander, and all the other concerns that accrue to information publishers. These are not MaSH-specific issues, but potential users should be aware of them.

The ethical implications are interesting as well. The administrator could go through the logs MaSH keeps and find out the IP address (and often username) of system users. Suppose Helpful User, being a bit of a crank, adds a reprehensible topic such as “Pro-Capitalist Paeans” and populates it with sites, all under the name “Anonymous.” This is, of course, grounds for failure—even academic excommunication—at some universities. Since information on Helpful User is logged, the MaSH system administrator might be able to discover her identity. Of course, this is no different than the responsibility that accrues to the administrator of any system. Administrators are responsible for acting ethically. But it needs to be made clear to users that "anonymous" use of the system is, perhaps, not. A "warning label" for the system might be appropriate.

4 Practical MaSH: the case study

The goals of the study were to see if MaSH performed in ways consistent with expectations created by the research and to show that MaSH can deliver on its promises. A secondary goal was to identify future areas of improvement and study. Two kinds of data are presented here. Summary views of system activity, generated from the log files, give overall usage trends. A series of cases, detailing tasks MaSH was used for and how it fared on each, provides finer details. Both summary data and case descriptions become grist for the mill of analysis which follows.

4.1 Description

This section presents the results of the first major test of MaSH, a case study of how the system can support and enhance a college course. The course, "Being Productive With Computers," is an introductory computer science course taught at James Madison University. It is targeted at freshmen who plan to become computer science majors but do not have enough experience using computers to go directly into a programming course. The range of assignments for the course is quite broad, including a generous helping of Internet use, research, and exploration. This allowed MaSH to be used in a variety of contexts, including building a topic directory, sharing research results, sharing "fun" sites, and conducting discussions. The course was taught in four sections with a total enrollment of about 140 students, almost all of whom were first-semester freshmen.

4.2 Summary statistics

The data below were generated by counting entries in the system log and are meant to provide an overall picture of system usage. The log was edited in the following ways to remove erroneous data.

- A bug caused entries to be logged twice for the first few days. Duplicates were removed.
- One student ran a Web spider that kept visiting MaSH, generating 2,700 hits (five percent of the total system activity). The spider also visited links indiscriminately, including the links for moving items up and down and for requesting to add and change information. This threw off the relative proportions of actions, particularly those of "active" actions like ranking or changing an existing site. All hits from that IP address were removed. This cost a small amount of good data, but was judged to be much less disruptive than keeping those hits.
- Four malformed lines were manually removed.
- Only entries from September 2 (when the system was introduced in class) to December 18 (the last day of the semester) were kept. A small amount of activity, around 170 hits, happened after the end of the semester. This activity appeared to be from former students using the directory for their own ends. Since this activity was not strictly part of the course, it was not counted.
- The log was added to the system in a hurry, which showed when it came time to do these analyses. Preparing summary statistics was much harder than it should have been. Improving the logging facility is a prime area for improving the system—especially since, in retrospect, MaSH could probably have used the standard server log formats and pre-existing tools for processing such logs as the basis for a much stronger facility.

4.2.1 Breakdown by activity

A fundamental way of looking at system usage was to look at how often each activity occurred. Table 2 provides the details.

Row	Action	Count	Pct.	Description
1	View Topic	38284	69.71%	Navigate around the directory
2	Visit Link	12394	22.57%	Visit a site
3	Add	1259	2.29%	Add an Item
4	Request Add	1074	1.96%	Request to add a link or topic
5	Request Change	864	1.57%	Request to change Item
6	Change	416	0.76%	Change an already-entered Item
7	Move Item Up/Down	240	0.44%	Move links within a topic
8	Request Delete	212	0.39%	Request to remove an Item
9	Delete Item	128	0.23%	Remove a link or topic
10	Welcome	35	0.06%	Introductory screen
11	Help	10	0.02%	Help page (not implemented)
12	Total	54916	100.00%	Overall system accesses

Table 2. Frequency counts for each activity in the MaSH

The columns in Table 2 give each action, how many times it happened, the percentage of total system activity, and a short description of what each action means to the system. Rows one through 11 give the count for each action in the system, while Row 12 gives the total activity. Rows one through 11 are sorted in descending order of frequency. These activities can be broken down into three major categories: passive use, active use, and administrative use. Passive actions were uses of the system to access resources without making a contribution and include the View Topic and Visit Link actions. Active actions were those that required user contributions, including adding, changing, or deleting (sometimes one adds by subtracting!) items, and moving or commenting on a link. The rest of the activities were administrative in nature, including requests to add, change, and delete links, as well as retrieving the welcome and (nonexistent) help files. These were lumped together into the "administrative" category. A summary of actions by category is included below in Table 3.

Row	Category	Count	Pct.	Description
1	Passive	50678	92.28%	Viewing topics; loading links
2	Active	2043	3.72%	Changes to the directory
3	Administrative	2195	4.00%	Requests for changes; welcome
4	Total	54916	100.00%	Total system accesses

Table 3. Counts for the three broad categories of action in MaSH

The columns in Table 3 give each category, how many actions happened in each, the percentage of total system activity, and a short description of what actions went into each category. Rows one through three give the count for each category, while Row four gives the total activity.

4.2.2 Breakdown by popularity

Looking at the topics and links that were in highest demand indicates that use of the system clustered around popular areas. Table 4 shows which links and topics were accessed most often.

		All 663 Visited Links				Top 50 Links			
Row	**Category**	**Hits**	**Percent of Total**	**Count**	**Hits/ Link**	**Hits**	**Percent of Total**	**Count**	**Hits/ Link**
1	**Homepage**	7125	57.51%	189	37.70	2467	61.68%	35	70.49
2	**Neat Site**	1987	16.04%	236	8.42	196	4.90%	3	65.33
3	**Projects**	1009	8.14%	53	19.04	62	1.55%	1	62.00
4	**HTML**	899	7.26%	14	64.21	579	14.48%	4	144.75
5	**Numbers**	321	2.59%	5	64.20	189	4.73%	2	94.50
6	**Areas**	316	2.55%	130	2.43	0	0.00%	0	0.00
7	**History**	258	2.08%	2	129.00	258	6.45%	2	129.00
8	**Discussions**	185	1.49%	5	37.00	74	1.85%	1	74.00
9	**Sources**	106	0.86%	2	53.00	83	2.08%	1	83.00
10	**Final**	92	0.74%	1	92.00	92	2.30%	1	92.00
11	**Position**	78	0.63%	23	3.39	0	0.00%	0	0.00
12	**Other**	13	0.10%	3	4.33	0	0.00%	0	0.00
13	**Total**	12389	100.00%	702	17.65	4000	100.00%	50	80.00

Table 4. Hit frequencies for links by topic, for all links and for the top 50

Table 4 gives a summary view of how many links and hits were recorded for each assignment. Rows one through 12 present data for each category. These categories correspond to major uses of MaSH in the course, and are sorted in descending order of total hits for each category. Links were assigned to categories based on the date they were added to the system, which normally corresponded to a particular assignment. These automatic assignments were then hand-checked to minimize categorization errors. The thirteenth row presents totals, for reference. The nine columns are a bit more involved. The first is the category name. The next four columns present data for all the links in the system. These include the number of hits for all links in a category, what percent of total hits that comprised, the number of links in the category, and the number of hits per link for the category. The last four columns do the same job as the previous four, except they only treat the top 50 links. Note that a small portion (7.5 percent) of the links received a disproportionate share (32.3 percent) of total activity. Fifty is a somewhat arbitrary number, but it is a natural breaking point in the data. The next 100 most popular links were almost all in the homepages category.

4.2.3 Breakdown by date

The amount of use the system received varied from day to day. Table 5 presents a few summary statistics about how usage changed on school days and days when assignments were due.

Row	Category	School Days	Off Days	Assign Days	Non-Assign	Totals
1	**Number of Days**	74	34	16	92	108
2	**Total Accesses**	44311	10605	22378	32538	54916
3	**Accesses/Day**	598.80	311.91	1398.63	353.67	508.48
4	**Passive Accesses**	40914	9764	20455	30223	50678
5	**Active Accesses**	1661	382	915	1128	2043
6	**Passive-to-Active**	24.63	25.56	22.36	26.79	24.81

Table 5. Usage summary for school days and days with MaSH-related assignments due

Table 5 makes two distinctions: whether school was in session, and whether an assignment that required MaSH was imminent. “School Days” were Mondays through Fridays during the semester, except for school holidays; “Off Days” were the non-school days. “Assign Days” are defined to be the day before and the day that an assignment requiring MaSH was due, with other days relegated to the “Non-Assign” category. The Totals column is given for reference. Rows one and two represent the number of days in each category and the total number of accesses on those days. Row three is derived, and is the mean number of accesses per day. Rows four and five present passive and active usage of the system. Row six is derived from rows four and five, and gives the ratio of passive uses to active ones. Passive and active are taken to mean what they do in Table 3: passive uses include viewing topics and links, while active uses cause the directory to change.

4.2.4 Breakdown by time of day

One other way to view the data is by the time of day it was accessed, as shown in Figure 4.

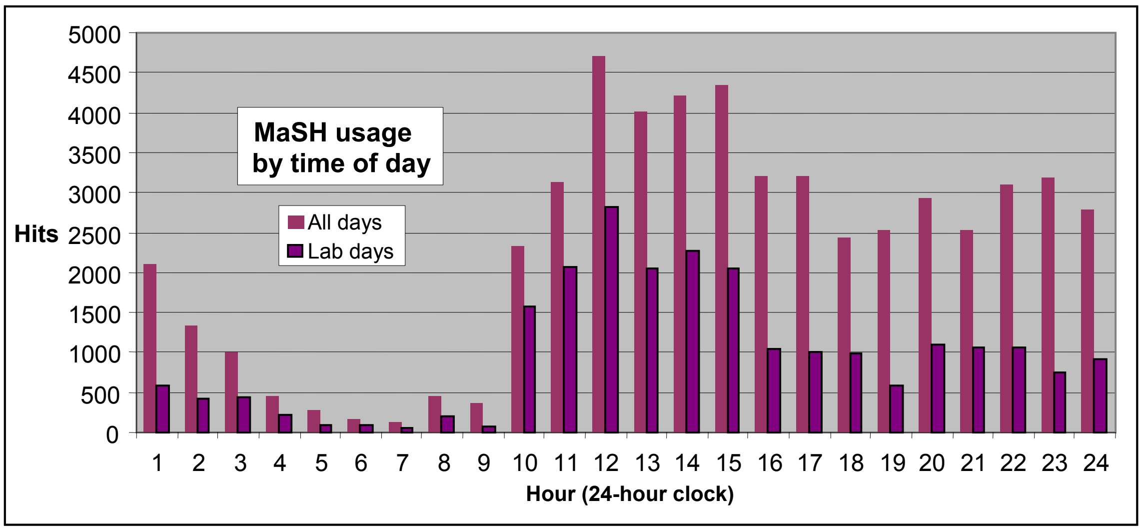

Figure 4. MaSH access broken down by time, in hour increments

Figure 4 is a chart that plots number of hits on the vertical axis and time of day on the horizontal, broken down by hour. The “all days” data series represents accesses for every day of the week. The most busy times were between 10am and 5pm, with steady levels of use from 5pm to 1am and a drastic reduction in use in the wee hours of the morning. The second data series, “lab days,” shows access by hour only for Tuesday and Thursday. These were in-lab days where computers were readily available and class duties were most likely to come to mind. Labs met from 9:30am to 3:15pm these days, and the data clearly show that the system was most active during these times.

4.3 Course uses

MaSH was used to support a number of activities during the semester: learning the system, finding interesting information, researching topics within computer science, holding course-related discussions, supplementing lecture notes, and allowing students to publicize their homepages and work. Students also used the system in several unanticipated ways, which will be discussed below.

4.3.1 Neat sites

One of the introductory units asked students to discover a “neat” site and add it to the MaSH system under a Neat Sites category. Students also had to visit at least two of the sites and optionally submit comments on them. The primary goals were to expose students to the system and to see if they could figure it out. Lesser goals included seeing if students would use and create categories, whether they would comment in-depth if they were interested in the material, and whether they would use the system beyond the extent required. The system was briefly introduced in a lab, but students were expected to learn to use it on their own (self-teaching being another goal of the course).

The primary goal was achieved. All but three students—some of whom had not used the Internet much in the past—were able to complete the basic tasks. This suggests that the Yahoo!-like “look” helped orient users, and that the functions of adding, ranking, and commenting on links were not difficult to use. However, the meanings of some of the icons, in particular the icon for making comments, were not obvious to a number of students. These icons should be redone by someone with graphic design skills, a criterion that unfortunately excludes the author. This experience led to the addition of a legend that explains the icons.

Students’ use of the features of MaSH was mixed. Almost all students entered reasonable titles and descriptions for their links, except for a few who forgot to change the default title. About ten students created subtopics, and some went so far as to move others’ links into the right category or to populate their own. Use of subcategories by students was sporadic, however, and the instructor had to devote a small amount of time to moving links into subcategories to prevent the main Neat Site page from growing overly long. Two topics were created outside of the Neat Sites topic, which probably indicated confusion as to how the system worked. More explanation of the system than was given for this assignment would be appropriate when introducing the system to new users.

Student use of the ranking and commenting features of MaSH was limited. Around one-sixth of the students (25) did not enter comments, which were required for the exercise. It is not certain whether this was because of confusion as to how the system worked, or simple failure to complete or understand the assignment. Students seldom made use of the ranking features. Although rankings were not so appropriate given the nature of the assignment, the hope was that students would experiment with them. This limited use of link ranking features continued throughout the semester, with a total of 240 link ranking activities. There were a few exceptions; sites on topics that students found amusing (the South Park television show) or strange ("Mr. T ate my balls") tended to draw a relatively high number of comments. Students did use the Neat Sites area beyond the requirements of the assignment. Some students added more than one site, and the topic was used as a jumping-off point for Web surfing throughout the semester.

4.3.2 Areas of computing

Another assignment, new to this semester, required some students (non-majors) to research information on the primary sub-areas in computer science. CS majors could refer to the information found when preparing to meet with their major advisor for the first time. Majors were encouraged but not required to use the information compiled, as part of the meeting required them to ask about their advisor's main research interests. An Areas of CS topic was created, containing several topics corresponding to the areas of CS outlined in class. The non-major part of the assignment, to build the Areas of CS topic resource, was to be completed in a week. The purpose was to see what sort of resource developed, and how well and quickly it could evolve. A secondary purpose was to determine whether students would find appropriate resources, categorize them appropriately, and provide enough information to make them useful. The third was to see if the resources would actually be used.

The results here demonstrated that a resource could be built quickly. Students added 142 links in about a week, and for the most part these had appropriate information (titles, keywords, description) and working URLs. Sites were added to reasonable categories, and students created additional categories as needed. Quality was more problematic. Some links were too vague or too specific for the nature of this assignment (e.g. "IBM's Newest Operating System—Answers to 119 questions about IBM's newest operating system, VM/ESA Version 2 Release 3.0" under the Operating Systems category). This is not unreasonable to expect from students who are not very familiar with the areas under discussion.

The CS majors, who were to use the resource built by the others, did not make extensive use of the directory. Some lines were not visited at all; the 130 that were took 316 hits, or about 2.4 hits per link. It would have been much better for the purpose of the thesis to have required majors to use the directory. This could have been used to strengthen students' resource evaluation skills and to determine whether they would bring the cream to the top of each category. However, a teacher can only inflict a certain amount of work upon students, and the assignment came at a particularly busy time during the semester. It was not appropriate to force students to use the resource, and it went mostly unused.

4.3.3 Supplementing lectures

Several lectures were supplemented with associated MaSH topics. A Citing Sources topic was given to provide guidance for research paper bibliographies. Another was created for HTML, which was covered in one lecture and used in several labs and assignments. As with most MaSH assignments, students were encouraged but not required to add links and to check periodically for useful information. The incentive was that students could benefit from a different presentation or "take" on a topic if the lecture left them confused, or they could get extra information if they were interested in the topic. The goal was to see if students would use the system to get help; a secondary goal was to see if students would contribute to the HTML topic.

This seemed to work well. Citing Sources links were hit over 100 times by the 140 students. The students added no links to this topic, which is to be expected, since—as was exhaustively proved in writing this thesis—bibliography formats are interesting in the way watching paint dry is exciting. The HTML topic was much more active. It started with four links which received over 300 hits. Several different students added links (10 links, eight contributors) that also received significant activity (about 600 hits). The HTML resources were valued outside of the course, as evidenced by an e-mail from a former student for the HTML sources to be put back on the Web the next semester (the course Web site was taken down over Christmas break).

Students created topics on their own to supplement course activities. These will be discussed below when talking about general and spontaneous use of the system.

4.3.4 Common-topic assignments

Two assignments involved using the Internet to research one of several topics. The History Paper required each student to research some element from the history of computing and produce a report. The Position Paper asked triads of students to research one of four issues and produce a presentation to support one side of the issue. In past semesters, students completed these assignments without sharing their research. Since the range of topics was fairly small, with a number of students working on each topic, these assignments seemed like an excellent place to make some serendipity happen.

For the History Paper, a History of Computing topic was seeded with a single link; for the Position Papers, students were given a subtopic for each of the four main possibilities. Students were encouraged but not required to share their research by adding links and using the links others added. The purpose was to see if they would use the system for its purpose of sharing information without having a grade hanging over them, in the context of a course assignment. Would students use the system passively or actively?

Students used the system in two different ways for these assignments. The History Paper was early in the semester, and students did not make many contributions. They did make extensive use of the first two links that were added (one by the instructor, one by a student). These were hit 164 and 94 times, respectively. One user added a number of links on the Apple II computer, but since these were not of general use, they were only hit eight times total. Contributions were much higher later in the semester on the Position Paper project. Students added 23 links across the four topics, which received a total of 78 hits. Use of the added links for research remained constant. Since students worked in groups of three for the second assignment, one would expect about one-third as many hits for its links. The actual number was 78 to 258, or just over 30 percent.

4.3.5 Ersatz newsgroups

Four links were created in MaSH under a Class Discussion topic. These "links" were set to point to themselves; their purpose was to serve as repositories for comments and facilitate class discussions. Students were encouraged to post comments and questions apropos the topics. These included a Q-and-A area for course-related questions, a course feedback area, a MaSH Q-and-A area, and a "graffiti wall" for venting and making random comments. Students could enter comments under their name or any pseudonym (including "anonymous"). The goals were to encourage use of the system and to see if it could perform simple discussion functions. There are plenty of other tools for online discussions, including old standbys like discussion lists and Usenet, and newcomers like Hypernews and Webboard. The goal was not to replace these other tools—if discussions were stuck doors, MaSH would be a screwdriver to their crowbar. But if Webmasters had already decided to use MaSH for its primary purpose, it would be serendipitous if they could use it to conduct small-scale discussions.

Results here were mixed. Students were not pushed to use the discussion areas, and the Q-and-A topics never got off the ground. The course feedback and graffiti wall topics saw more use, though the contents of the discussions left something to be desired. The graffiti wall was meant to be random, but the course feedback was about half productive discussion and half primal venting. This can be partially explained by the fact that most of the students were freshmen, but technical reasons also contributed. Moderating these discussion was difficult because there is no easy way to delete comments. Requests to stay on topic were only partially effective. The primary goal was more-or-less accomplished, though: it is possible to use the system for small-scale discussions. Adding a simple way to remove comments would go a long way toward improving this functionality.

4.3.6 Student and project homepages

Students were required to make homepages for themselves, as well as sites for group research projects. MaSH was used to collect the URLs for these pages. Each assignment received a topic, with subtopics for each of the four course sections. Students then added their links to the appropriate place. The goal was to use MaSH to help with class administration, organizing these sites so they would be easy to locate and grade. Students were assigned to grade each others' homepages; instructors graded the project pages. Another goal was to encourage students to explore each others' work.

This was a very successful use of the system. Students added their topics to the right places, and were able to locate the homepages they were supposed to grade, except in the two cases where the posting student forgot to include identifying information. The organization of the project pages made them easy for the instructors to grade as well. Finally, since MaSH tracks the date that a site is added, it was easy to determine whether the work was submitted on time. Getting students to visit each other's pages was quite successful as well, with an average of over 37 hits per homepage and 19 hits per project page. Homepages were also one of the few areas where students left comments on their own initiative.

4.3.7 General and spontaneous use

Students were encouraged to add to MaSH outside the context of any assignments. Topics and links that might be of interest to the class were fair game, and did not necessarily have to be related to the course material. The goal was to see if serendipity would really happen, if students would use the system to share interesting items and to look for them in the directory.

Students definitely used the system to visit some links above and beyond required usage. For example, students were quite interested in the Student Homepages topic, with each homepage averaging over 37 hits. This is far beyond the required three homepages that each student was to grade. Even making the unlikely assumption that students went back multiple times to the same page to do a really thorough job, most of this activity was spontaneous. The Neat Sites area was also a popular place for students looking for places to visit; a scan through the log revealed students using Neat Sites as a jumping-off point for Web expeditions throughout the semester.

Sharing found links was less spontaneous: not a great deal happened, and what did happen mostly correlated with assignments. For example, the Neat Sites area was actively populated when it was assigned, but only about 30 links were added after the assignment due date. A few users attempted to start and support topics, but these attempts were not generally successful. One eclectic user created a Poetry Corner topic that went nowhere, as well as a Computing topic that did the same (though a few students mistakenly added links here during the Areas of Computing exercise described above). Another user tried to start a Literature area with similar results. The Computing and Literature topics started with a number of links added by the topic creator and died at that point.

One topic that did get some play, Interracial Interaction, was created by “Anonymous.” It originally appeared with eight links. Most of the titles and descriptions were vague, like the link titled “Interesting: definatly [sic] worth looking at.” The links mostly pointed to sites with racist views (e.g. the KKK). One might expect complaints, or perhaps angry discussion and comments attached to the links. What happened was more interesting. These links received around 100 hits total; no comments were added and no complaints received. However, two of the links were deleted. Most of the other links were altered to point at sites such as Microsoft (the reader can decide whether this “innocuous”, or at least better than the KKK). Another site was originally described as “White Power.” This was changed to “white flour” and the link altered to point to the Betty Crocker home page.

Two students created topics on their own to supplement course materials. An assignment that dealt with how computers represent numbers and other data proved to be quite difficult for a number of students, despite a pair of lectures and a page of helpful links included with the assignment on the Web. (In retrospect, these should have been provided via MaSH.) One student took matters into his own hands, creating a Base Converters topic with several suitable links. From a participation standpoint, it was a wash—only one other student added a link, and no one added comments. But from a use standpoint, it was a wild success, with over 320 hits across five links. Another student created a Final Exam Help topic for a question on the take-home part of the final. (The student was kind enough to ask first.) Like the Base Converters topic, there was no other active participation; in fact, the student added only one link himself. However, that link received 92 hits in just a few days.

5 Discussion and analysis

Discussion of the data traditionally follows its presentation. Since the "experiment" was a case study rather than a formal experiment, there will be little number crunching here. Instead, the focus will be on discussion of how closely the results fit the theoretical framework. Whether the system approached its goals is also examined. Other interesting observations are noted, followed by a brief section on the performance of the software and of the study itself.

5.1 Did it do what it should?

The research in collaborative filtering and online communities discussed earlier suggests several items to investigate:

- Users may be lazy.
- External rewards will increase user input.
- Communities will build around topics of interest.
- Using online bookmarks will be convenient.

The case study provides evidence for the first three points, and did not directly address the fourth.

5.1.1 Users may be lazy

The collaborative filtering research suggests that without tangible rewards, users will not contribute as much as they otherwise might. Two views on the evidence give different verdicts on this claim. If total system accesses are included, then the case study clearly supports it. Table 3 shows that active uses of the system, which were defined to be adding, changing, deleting, ranking, or annotating links, comprised 3.72 percent of total accesses, or one active access per 27 total accesses. Unfortunately, the studies on collaborative filtering did not present evidence on how often users rated items versus accessing them, which would allow comparisons of the rate of contribution. Konstan et al. (1997) mention that users might only rate one to two percent of articles in a high-volume newsgroup, but do not indicate what percentage of the articles these users read. What is certain is that the vast majority of total MaSH accesses were of a passive nature.

On the other hand, over 70 percent of the system activity involved navigating the directory and administrative requests. Instead of counting total system activity, a more natural way to measure how much users benefit is to only count link visits. After all, looking through the topic directory is of little direct benefit. If we use this lens, the picture changes: user contributions rise dramatically, to about one active use per seven links visited. This view paints a much rosier picture of user contribution, where for one reason or another, users chose to contribute on a fairly regular basis.

5.1.2 External rewards will increase user input

Whether those other reasons involved the goodness of users' hearts is open to question. Some of the contributions were semi-coerced by tying them to grades. The research suggests that external rewards will increase user contributions, and the data appear to bear this out. Columns five and six of Table 5 break the semester into two parts: "Assign Days", the day before and day of an assignment that required MaSH, and "Non-Assign", which included the rest of the semester. The active-to-passive ratio is about 1 per 26.79 on non-assignment days; active use on assignment days climbs to 1 per 22.36. This is about a 16.5 percent jump in active use on assignment days, which implies that external rewards—in this case, grades—do affect user behavior.

The data in Figure 4 also support the claim that external rewards will impact user behavior. The busiest times corresponded to lab periods, when computers were available and class concerns at hand. Further support for such a connection comes from breaking down usage by day of the week. Lab days, Tuesdays and Thursdays, accounted for 42.8 percent of system accesses (23,509 of 54,916). This would be expected if MaSH were used only on work days ("Forty percent of all workplace absences come on Monday and Friday!"), but it is a large fraction for two days out of seven.

This finding is consistent with the research, but somewhat problematic for the MaSH vision of building communities. An interest-group Web site does not have much power to give rewards to its users, except perhaps for recognition of users who make the most contributions to the system. Gold stars only go so far. Educators have it easier, as an instructor can use grades to require contributions. This may not be so smart. To borrow from Lehrer (1959), MaSH "is like a sewer—what you get out of it depends on what you put into it." External rewards could skew the system so that user contributions are based on getting the reward rather than on the contribution's merits. Most people—especially, say, indigent college students—might click on things if they were given a penny a pop. The reward system may not be this blatant, but the danger that rewards might encourage spurious evaluations is real. The ideal solution would be to recognize that people *are* motivated by external rewards, and then design a reward system that would encourage reasonable contributions.

5.1.3 Communities will build around topics of interest

The case study suggests that usage is highest in topics that were of interest to students. The data from Table 4 fit nicely into three patterns. The first includes high-volume topics with a moderate hit per link count. A second pattern covers topics that have a high average number of hits per link. The third pattern contains the Maytag repairman topics, which sat around waiting for someone to call but received almost no activity. Table 6 demonstrates these patterns.

Patterns	Topics	Hits	Links	H/L
High volume	Homepages, Neat Sites, Projects, Discussions	10306	483	21.34
High hits per link	Numbers, History, Sources, Final, HTML*	1676	24	69.83
Maytags	Areas, Position, Other	407	156	2.61
*HTML had characteristics of both high volume and high hits per link.				

Table 6. Three distinct patterns of how MaSH was used

Table 6 has a row for each pattern described above. The Topics column lists topics that fit each pattern. The Hits and Links columns are the total hits and links across all topics in a pattern, and the H/L column is derived from those as the number of hits per link.

The reasonable explanation is that these three patterns correspond to why a topic matters to users. High volume topics are those where users had an intrinsic or strong interest in the subject. The Neat Sites topic was specifically designed to be of general interest, the Homepages topic allowed students to learn more about each other, and the Projects topic allowed students to see examples of each others' work. High hits per link topics suggest an operational interest: the material was useful for some task. The sources topic was required for preparing bibliographies, the numbers and final topics were helpful in answering those exercises, the HTML topic was useful toward making homepages, and the discussions topic was used for course communication. These high hit topics were used in a markedly passive way. Maytag topics were those that were forced upon students and that they did not see as useful or interesting. The Areas of Computing topic is a fine example: students added a large number of links in order to get credit for the exercise, but few used these links.

Does this suggest that communities could have formed or strengthened around the topics in MaSH? Yes, if one agrees that evidence of a topic-based community would be shown by a topic with activity both in creating links and viewing—i.e. being interested in—the links of others. This view is reasonable. A topic that is widely visited but that no one contributes to indicates a resource, not a community (though it is consistent with MaSH's goal of making serendipity happen), while a topic where everyone adds material that no one uses signals forced labor. The patterns also suggest that there is some optimal balance between passive and active use of the system. They also indicate that, somewhat oddly, a high ratio of active to passive use may indicate a dysfunctional situation where resources are added and not used. This is not necessarily the case—a high ratio of active use may indicate an extremely involved community—but it was the case for the study.

5.1.4 Using online bookmarks will be convenient

This point, claimed by the creators of online bookmark sites, was not directly addressed during the case study. The simplistic answer is, as the advertisers say, that 55,000 hits could not possibly be wrong. Still, MaSH could use several improvements toward making them more convenient. A handy, miniature link-adding interface like those of itList and MURL is in order. More online help would be nice as well. The rest of the interface could surely be improved through judicious use of principles from the field of Human-Computer Interaction. Doing this, especially in combination with a survey of users, would be a good future project.

5.2 Did it do what was hoped?

It appears that the case study results were in line with what one would expect, based on the research. The next question is whether MaSH is capable of achieving its goals:

- Communities will be strengthened.
- Resources will reflect the community.
- Resource directories will be of quality.
- Users will make use of others' discoveries.
- MaSH can support a classroom.

As with the research, the study provided support for most of these goals.

5.2.1 Communities will be strengthened

This case study does not have a prayer of proving this question, as there are far too many other variables affecting the lives of first-semester college freshmen. However, reasonable statements can be made in support of this claim. Taking the class as a community, MaSH's generating 55,000 additional hits on the course Web site probably had a net positive effect on communication. Links to course-specific discussion topics were among the most popular, indicating at least concern for the class. By using the system to publicize home pages and student work, the students were able to learn more about each other, which should have a community-building effect. Finally, one can draw a connection between the provision of the HTML help resources and the ability of students to publicize the homepage and project information above. This indirectly contributed to community-building. All of these statements are anecdotal, however. Exploring this idea is a fertile area for further work.

5.2.2 Resources will reflect the community

Community interests and values shaped the resources available in the system. Much of this argument was already made in the section above about communities forming around topics of interest. Users added links that interested them, and the most popular ones emerged supreme. The top 10 neat sites by hit count are listed in descending order below in Table 7, which gives link names and counts.

Title	Count
South Park Sound Archive	79
Mr. T ate my balls	60
Stick Figure Death Theatre	57
Microsoft Terraserver	42
sweet shots	41
South Park	37
South Park: Christmas	32
Da Ebonics Page	31
Intoximeters, Inc.	31
(site deleted, no title available)	31

Table 7. The ten most popular neat sites

These most popular sites appear to reflect students' interests. South Park is a very popular cartoon in the college student demographic; Mr. T ate my balls and Stick Figure Death Theatre appeal to the sense of the outrageous and weird; and the appeal Intoximeters, Inc. has for college students really needs no explanation. The appearance of these links is solid evidence that the students' interests helped determine norms for material available in the system. Links and topics that fell outside these norms, like the ill-fated Poetry Corner Discussion and Literature topic, failed to pass muster.

5.2.3 Resource directories will be of quality

The case study did not address the quality of the resources added to the directory. With students in an introductory computer science class, whether they can reliably judge the quality of a computer-related link is in question. This fact does not lend itself to evaluations of the resources built by this community. The concept of collaborative filtering suggests that this point is reasonable. However, actually measuring the quality of a list of resources is a fairly complex undertaking (Leighton & Srivastava, 1997), especially in light of the fact that, as discussed earlier, quality is at least partly a function of the needs of the person—or the community—using the information. This could be an excellent starting point for future work, and is discussed below along with other possible directions for building on this thesis.

5.2.4 Users will make use of others' discoveries

The high proportion and quantity of passive use indicates that serendipity did in fact happen, with users taking advantage of each others' discoveries. This is not surprising. If communities will build around topics of interest, then of course users with shared interests would like to receive additional interesting information. If no effort on their part is required, so much the better. The data bore these statements out, particularly for topics that were useful but not particularly interesting to students. These suffered from a lack of link quantity, but their hits per link numbers were sky high, almost 70, according to Table 6. This is strong evidence for the sharing of others' resources. The spontaneous addition and heavy usage of the Base Converters and Final Exam categories further support this claim.

5.2.5 MaSH can support a classroom

MaSH proved useful in supporting the class. Getting students involved with the course was a strong point. As Table 2 shows, MaSH generated almost 55,000 hits to the course Web site and generated over 12,000 visits to Web pages that students may not have otherwise seen. A number of these were student homepages or sites like “Mr. T ate my balls.” These sites could most charitably be described as having limited educational value. However, since one of the course objectives was to make students comfortable with using computers and the Internet, the visits did serve an ulterior purpose. And students did make several thousand visits to sites relevant to the course that otherwise might not have been made.

Sundry other benefits accrued as well. MaSH proved to be an adequate if limited small-scale discussion tool, allowing students to vent their concerns about the course in a public but anonymous forum. Certainly there are better discussion tools available, but this functionality is a nice bit of, well, serendipity that teachers can harness. MaSH was handy for giving students pointers to resources the instructor wanted them to see. The opposite was true as well, and was handy as a way to “collect” work online. Students posted links to their projects and homepages to the appropriate place, and suddenly there were convenient lists of ready-to-jump links with identification and descriptions.

One potential barrier to MaSH being useful in education is ease of use. This was not a problem for college freshmen. In the Neat Sites part of the study, most were able to use the system with only a limited demonstration and no help file. Adding more help and spending a few minutes showing new users how the system works should make MaSH accessible for users of most ages and computer skill levels. As for ease of use for the instructor, there is no administrative mode when using the system, so ease-of-use should be the same. Whether installation and setup for instructors are as simple as claimed in the system description is unknown, as the author was also the instructor.

5.3 Other observations

Some observations surfaced during the case study that were above and beyond confirming the research and checking the system against its goals.

- Popularity feeds on itself.
- Communities sometimes police themselves.

- Tangents are good.
- Moderation matters.
- There is a right size.

5.3.1 Popularity feeds on itself.

The data in Table 4 reveal that a small portion of the directory accounted for the lion's share of the activity. The top 50 links, or just over seven percent, received almost 33 percent of the hits. This tidbit resembles a trend found for PHOAKS by Hill and Terveen (1996). On average, the most popular URLs mentioned in a newsgroup received most of the mentions across the entire newsgroup. Konstan et al. (1997) also noted a relationship between high article rankings and high readership. These examples resemble the inverse relationship between rank and frequency of words (and many other things) discussed by Zipf (1949) in *Human Behavior and the Principle of Least Effort*.

On one hand, these findings are eminently reasonable: resources that people find to be valuable *should* be accessed more often. But there is a darker side to this observation. Early ratings will tend to have a much greater effect than later ratings. If the first few raters give thumbs down to a resource, this will tend to recommend against it for future readers. In MaSH, that Web page will sink like a stone toward the bottom and never be seen again. This is exactly what should happen for low-quality resources. But what if the first few ratings are in error: suppose Helpful Rater was having a bad day, or disliked the author or topic? Then the page will most likely forever be lost, as few will take the time to read it and correct the error. The opposite is also true, though to a lesser extent. Initial positive ratings will tend to increase a resource's viewership, though later viewers can correct an overenthusiastic initial evaluation. In either case, the recommendations get into a vicious circle: both positive and negative ratings will tend to feed back into themselves.

Can the lost pages be saved? Perhaps nothing needs to be done. Konstan et al. (1997) describe a concept called "predictive utility," which basically means that if there are lots of good items, so what if some get lost in the shuffle? Predictions become more useful as the signal-to-noise ratio tilts toward noise and the cost of missing useful information rises. For MaSH, the amount of predictive utility will depend on the topics involved, the amount of contribution by users, and their ability to provide accurate rankings. Most likely, undeserved positive and negative ratings will sometimes be a problem.

Can this problem be addressed? There are a few technical solutions which involve an element of chance: occasional random ordering, a “weighted lottery,” and variations on randomness could all give lower-ranked resources a day in the sun. Another possibility is to have a ceiling on positive or negative recommendations (MaSH has a version of this since links cannot move past the top or bottom of the list). This would help in cases where a few sites received most of the recommendations and clustered at the top (addressing Resnick and Varian’s vote early and often phenomenon). Voices that recommended less popular sites would be heard more if a ceiling were imposed, perhaps making the rating process more of a community effort. A third strategy would be to segregate sites with less than *n* ratings into a “new sites” or “need reviews” area within each category. Doing this, however, delays the benefit of recommendations—and users who see no immediate benefit may never contribute.

All of the ideas just presented for diminishing the importance of certain evaluations have a primary flaw: they involve deliberately “sabotaging” the recommendation system by withholding recommendations, at least in the short term. This could cause users to mistrust the recommendations, defeating the purpose of the system. Informing users in advance that resources will not always be presented in recommended order might help, but then, it might not. More of a problem for MaSH is that randomizing the link order works against the metaphor of a list of bookmarks that the user can manipulate. In general, however, injecting an element of randomness into recommender systems could be a good strategy for guarding against the loss of useful information to initial poor recommendations.

5.3.2 Communities sometimes police themselves

One possible objection to a system like MaSH where everyone has full and equal access is that anarchy will reign. After all, the instructor or administrator is just another user. Anyone is free to add whatever they like, including erroneous, irrelevant, or downright offensive information. Dismissing these arguments as the paranoid fears of a control freak or the need of a "sage on the stage" to be in charge of the course would not be fair. These are all valid concerns about an open system.

Fortunately, the same openness that poses the problems also offers the solutions: communities can police themselves. The "Interracial Interaction" incident was a clear case of community self-policing. Objectionable material went into the directory, and was promptly cleaned up. Hiltz and Wellman (1997) describe a similar incident as part of a discussion of the social dynamics of online communities, which they feel are much like those of an "offline" community—including having an unstated but clear social norm for behavior. As for erroneous and irrelevant information, MaSH also provides tools for the community to clean up after itself. Comments, link ranking, moving links to new topics, and deletion are all tools available to every member of the community. A low-quality link can be moved to the bottom of the list, critiqued in annotations, or just deleted.

5.3.3 Tangents are good

Categorizing a link or topic as "irrelevant" and trying to suppress such items is probably a losing strategy in the long run. Much of the activity in the case study took place through links that were not directly relevant to the course content. The continued post-assignment use of the Neat Sites directory is a case in point. Allowing the topic to remain had several advantages. It probably drew students to visit the course Web site more often, and perhaps other areas of MaSH containing course-specific material as well. It provided a catch-all category for items that did not fit well in a course topic but that students wanted to share. It most likely supported the class as a community, as well. It would be interesting to trace the paths of users during a MaSH "session," to discover patterns of using the directory; however, it would be technically difficult.

Attempting to censor "off-topic" material would likely have a chilling effect on user contributions. Doing this sets the administrator up as a de facto moderator. Users will have to consider, every time they add something, whether the item will be received well. Getting contributions is onerous enough without throwing roadblocks in the way. Concerning oneself with checking the topicality of information means that one is making more work for oneself. This is not part of the philosophy behind MaSH. For these reasons, and to garner the benefits described in the previous paragraph, it will normally be a good idea to provide a space for sharing items of general interest.

5.3.4 Moderation matters

The last point was somewhat critical of moderation, and not fairly. Moderation can be helpful in keeping a discussion or community operating well. For example, the Discussions topic suffered at the hands of off-topic (and often off-color) messages. Removal of material that is deliberately disruptive is a wise thing. Whether a central moderator is needed to do this is an open question, since as described above, communities can police themselves. There is also the danger described above that if someone takes on the moderator's role with a heavy hand, overall user contributions may drop.

5.3.5 There is a right size

On the other hand, community self-policing can turn into vigilantism. Especially in larger communities, there is the danger of just a few voices dominating the contents of the discussion or, in MaSH's case, the ratings of and comments on links in the system. This may not be a problem for some uses of MaSH, especially if most of the users are willing to use the contributions and accept the judgements of the few. The collaborative filtering research does not carry the point this far, but when a group of users accesses a rating system, and most are unwilling to contribute, the group is a community of followers (or as they are known on Usenet, "lurkers"). A lurker-dominated dynamic is a problem for settings like classes, especially if the teacher's goal is to involve all students. The problem is even worse if the contributions received from the vocal few are not very high-quality. For example, the one link added in the Final Exam Help topic for the case study was of only limited use. On the other hand, the Base Converters topic was filled with quite useful information.

The prior paragraph mostly applied to larger communities. A complementary problem exists: a system like MaSH will not work well if the number of active users is too small. The glaring failures of the Poetry Corner, Literature, and Computing topics in the case study demonstrate the fate of a topic supported by a one-man band. The term “critical mass” comes to mind. Without a certain number of contributors, and short of a certain level of activity, the system will not be well-used.

MaSH does not offer any easy answers to these situations, which are more problems of group dynamics rather than troubles specific to the software. The moral is that Computer Mediated Communication is still communication. For more detail, Hiltz and Wellman (1997) describe some of the problems they have discovered while using online communication.

5.4 Software concerns

For the most part, the software performed as intended except for a few bugs. At times, when links were deleted, their parent topics kept references to them, causing inconsistencies. Use of the back button in the browser has been implicated, but the case is still open. Also, there were occasions when moving links up or down did not seem to work at the beginning of the semester. This has been fixed. Finally, MaSH could be faster. Pages are always generated from the database, a process that took two to three seconds per page. This may not sound like much, but at three seconds per access, time required to use the directory adds up—and “speed” of downloading is the number one negative aspect of Websurfers’ experiences on the Web (Kehoe, Pitkow, & Rogers, 1998). Any of these factors could have discouraged use of the system.

Another concern is that choosing the simple Yahoo!-style interface might have had the undesirable effect of discouraging user contributions. After all, users browse in Yahoo!, not move or comment on or add links. MaSH looks and feels much like Yahoo!, and a reasonable parallel to draw is that MaSH is for browsing, not contributing—even though users were specifically asked to contribute. Emphasizing some differences between MaSH and Yahoo! would probably help. Making it easier to contribute would probably increase user activity as well. In particular, making the controls for adding links, topics, and comments more prominent would likely encourage contributions. Amazon.com, for example, puts its link that requests customer reviews in a very noticeable place at the top of the reviews. Another cheap way to encourage input would be to allow users to add an initial comment at the time of adding or changing a link's information. Finally, allowing users a simple way to add a link to the system while browsing, without their having to type in all the information and go back and forth from the MaSH interface to their surfing session, would be an excellent way to improve levels of contribution.

5.5 Methodological concerns

The study was looser than it might have been. The original plan was to phrase the thesis goals as hypotheses and design experiments to test them. However, constructing an experiment would have required a much greater commitment of time and a narrowing of scope. The homogenous nature of the class population—mostly freshmen with a fairly narrow range of backgrounds, SAT scores, and computer experience—also argued against the ability of a formal experiment to say much about the general utility of MaSH. So the case study approach was chosen instead. This exploratory sort of research allowed for a broad investigation of the usefulness of the system—an exploration which was positive and generated some interesting observations to follow up on—but made it difficult to make strong claims.

The study also probably did not go far enough. Two approaches were considered: change the course to make MaSH an integral component, or add it as a tool to the current course. The second option was chosen to demonstrate that MaSH could be useful to educators with little modification of their current curriculum. It also mitigated concerns about changing the course to rely on an unproven tool. If MaSH had been unsuccessful, a course that relied on it would have been unfair to students. However, the integral component plan would have been better for the purpose of the thesis. Making more of the course dependent on MaSH would have increased student exposure to and use of the system. This would have given the case study more weight to support its claims. Also, with more use of the system, other interesting usage patterns might have emerged.

Finally, having the author also be the instructor for the case study was not the optimal model. Parties with vested interests tend to put forth extra effort to support them, consciously or not. This could skew the results of the study. Doing this also left open the question of ease of installation for a future product. In fact, there are many things that future work could improve upon.

6 Further work and conclusion

There is plenty of research that can be done to extend this thesis. MaSH should be used in a wide variety of classrooms to determine how broadly useful it is for education. User communities should also try the system, to see if MaSH really can help bring people together and promote serendipity. Finally, the resources that are created, both in classrooms and user communities, should be compared to other resources on the Web, such as search engines, directories and clearinghouses, and automated systems like WiseWire. The software itself could stand improvement. Errors should be fixed and unimplemented features added. The system also needs to be cleaned up. As the system was modified, sometimes under the duress of making it work **now**, it became less coherent. This is a common phenomenon in software maintenance, but not an admirable one. After the problems are fixed, attention should turn to improving the system: a better interface, improving the speed, and collecting and presenting useful statistics are all possibilites. All this aside, the system survived its opening performance, confirming its research basis and fulfilling several of its goals.

6.1 Classroom use

MaSH needs to be used in a number of different educational venues to determine its exact potential for supporting education. In this thesis, MaSH proved to be useful for a freshman-level computer science course that made extensive use of the Internet. Is it useful in other situations? Would a course on music of the late 19th century benefit? Would seniors make more use of it than freshmen? How about fifth graders, or diverse populations? Could it be useful in a computer-limited situation like a middle school? Is it really as simple to use as copying the files to an appropriate place on a Web server? How could it be made more useful to educators? To students?

The world would be a slightly better place if all of these questions were answered. A release of MaSH is available; it needs to be publicized and used in as many places as possible. The more variety—different course subjects, grade levels, instructor temperaments—the better for determining how broadly MaSH could be used. Reports from instructors and students should be encouraged; these can be analyzed to determine how to improve the system.

6.2 User communities

The case study showed that in areas that interested students, use of MaSH was quite strong. However, for this case study, some use was required. If use is not required at all, would a community of users make use of MaSH? Would members of American Cat Fanciers create a fountain of feline felicity, or would they ignore MaSH in a pointed, catlike manner? The thing to do, of course, is ask: ask if they might be interested in using a system like this, ask if they would inform their members, ask if the organization would report on their experience with the system. Also, ask the individual users what they gained from the system and how much they contributed. Finally, one could ask users already in a community whether using MaSH helped them feel more a part of that community. These activities would verify whether MaSH is useful in supporting and creating user communities, and also begin to measure how much serendipity MaSH creates.

6.3 Resource creation

Another line of research involves looking at the topic directories created in MaSH and seeing how they stack up against other resource directories on the Web. The case study touched on this, asking the students to create a topic directory of computing-related sites with modest success. However, these students were not very knowledgeable about the topic. Would a more knowledgeable user base, like the cat fanciers, create a high-quality directory? The results compiled by MaSH users in courses and user communities should be compared to other Web sources. Search engines, WiseWire, manually-created clearinghouses—all of these would be fair game.

This is no simple undertaking. A metric for quality would need to be developed and applied. How do you compare a directory of ranked results against a search engine—when the engine results will depend on the search engine? How can you compare that ranked directory against an unranked one like Yahoo!, or like most clearinghouses? For a multiple-topic directory, how does one measure “appropriateness” of categorization? This is not a new topic; researchers in the field of information retrieval have been interested in this for 40 years. There are accepted procedures for measuring similarity between documents, which could help determine quality of categorization. The notion of relevance certainly carries over to directories of and searches on the Web. Measuring recall, however, is problematic in an environment with an enormous, open, and ever-changing set of documents. Leighton and Srivastava (1997) have tried to measure the effectiveness of searching on the Web, at least for the major search engines.

A system like MaSH adds additional wrinkles to the measuring process. How long and how much should a user community be able to work on a MaSH directory before comparing it to others? Even the process of creating the directory would be worthy of study. How quickly does the directory improve and grow? Does activity level off and stagnate, or does the directory continue to evolve? These questions tie back to community-building, making them fertile ground for further research.

6.4 Improving the interface

This item bridges the gap between new research directions and more practical considerations that involve improving the software. A study on how links should be presented might be interesting. Should they be ranked by users, simply alphabetical, or perhaps even random, so that links toward the bottom receive “second chances?” Perhaps users should be able to choose the presentation mode. Are links after the first few on a page ignored? Could improvements in the interface encourage user input, as discussed earlier? Would making the commands for adding a link or topic more prominent encourage user activity? How about adding a copy of the comment interface to the interface for adding or changing a link—would that encourage comments and discussion?

6.5 Fixes

A number of outstanding items should be fixed to enhance the usefulness and stability of MaSH. First and foremost, the database has no locking or exclusion implemented. Under heavy use, inconsistencies will develop. This was not a major issue for the case study, but needs to be addressed before the system is ready for more extensive use. The next most serious problem is speed: the system takes a few seconds to process each request. The cause is probably the load time of either the database implementation (DB_File, a module that comes with Perl) or the useful but massive CGI.pm module (also bundled with recent Perl versions). Either way, the bottleneck should be located and ameliorated, as faster responses should encourage use of the system. Finally, there are cases where the actual number of items for a topic do not match the number that the database has stored. This situation needs to be investigated and fixed. These three items are the primary known problems with the system.

The maintainability of MaSH also needs to be improved. The first step is to clean up the ad-hoc fixes and “improvements” that were added during the case study. These items make the code much less readable and coherent. A second improvement would be to factor out common routines in the code, especially in the display modules. This would allow MaSH providers more control over the system’s appearance. A third useful item would be to write up formal design documentation. The system was designed using a “chicken-scratch” method: a number of informal sketches and diagrams were used to outline the system’s structure before coding began. Too many decisions were left to code time, however. Documenting what happened and using that documentation to restructure the system and support further work would be wise. Finally, a number of individual tests were created in the course of writing MaSH. These should be cleaned up and automated so that they can be run as regression tests in the future, and so that individual tests can be run on demand.

6.6 Useful new features

Some useful features were planned but not implemented in the first version. A utility to export links to and from the Netscape bookmark format, which MURL has, would be quite useful—

especially if implemented so that only a portion of the directory could be selected. Another useful feature would be importing bookmark files, or using search engines to help populate topics. Speaking of search engines, the internal search engine included with MaSH is quite primitive. It could be improved, and possibly extended (as part of the feature above) to search either internally or externally, much like Yahoo!'s search facility. (Alternatively, a better search engine could be taught how to interact with the database and plugged into the system.)

More questionably useful would be implementing features discussed earlier under limitations and concerns. These features are primarily concerned with security—preventing malicious use of the system and/or tracking individual use. Security-related problems were very rare during the case study, so they are not a priority. The exception was the Web spider that came along; implementing delays and/or generating errors after repeated, quick accesses from the same IP could have blunted the spider's "use" of the system. This would be a small safeguard against malicious users as well, without unduly affecting normal use.

This leaves the idea of adding usernames and passwords. This would certainly be useful in some ways, not the least of which would be opening up possibilities for studying user behavior. More interesting to instructors would be the ability to require students to use the system, track that use, and assign grades based on use. This is not very practical as the system stands, as IP addresses do not equate to users, while usernames are both optional and captured only when adding comments and links. On the other hand, adding usernames would materially degrade users' privacy and could discourage use of the system. The software could probably be juggled so that usernames could be optionally required based on the settings file. Whether this would be worth the effort is an open question. The way to find out would be to pursue the directions for further work laid out above and discover what users want.

6.7 Conclusion

MaSH was a qualified success in its first trial. It accomplished several of its goals and was useful in teaching the introductory computer science course. Further studies are needed, but it is likely that MaSH can be a useful addition to the teacher's toolkit. Eduators and online communities can both use the software to build resources and to provide custom windows onto the rest of the Web. The software is especially suited for smaller sites where the focus is on a small range of topics, as well as places where the community of users is important. Distance learning situations, topic clearinghouses, clubs and organizations, and instructional Web sites might all benefit from MaSH.

Finally, MaSH helps individuals harness serendipity by providing a tool to collect, annotate, organize, and share the results of their explorations of the Internet. Technology is useless until and unless it boils down to making people's lives better. MaSH does its small part to make the Web a better place for people.

Glossary

Bold terms in definitions refer to other glossary entries.

ALN: Asynchronous Learning Network. A term used by Hiltz and Wellman (1997) to refer to "a teaching and learning environment located within a **CMC** system designed for anytime/anyplace use through computer networks" (p. 46).

CMC: Computer Mediated Communication. Any use of a computer's special talents of networking, data storage, or data processing to further communication between humans.

Collaborative filtering: the practice of using readers' reactions to documents in order to help other readers decide which documents to read. Coined by the creators of **Tapestry** (Goldberg et al., 1992).

GnuHoo: The original name of the **Open Directory Project**.

GroupLens: A collaborative filtering system for Usenet news developed the University of Minnesota and MIT. Notable for its relatively large-scale field test and analysis of same presented by Konstan et al. (1997).

MaSH: Making Serendipity Happen. The subject of this thesis, a system designed to facilitate the sharing of resources discovered on the World Wide Web by creating an open topic directory that users can contribute to directly. This is intended for clubs and special interest groups, as well being useful in educational settings.

NewHoo: A former name of the **Open Directory Project**.

Online bookmarks: A new business model on the Web, where users are encouraged to store their bookmarks on a free server so that they can maintain one list of bookmarks accessible anywhere.

Open Directory Project: Originally **GnuHoo**, then **NewHoo**, this system is much like Yahoo! in appearance and behavior, except that users may become editors for particular categories, and link submission is much more straightforward and rapid than Yahoo!. Also like Yahoo!, the system was founded by two men, Rich Skrenta and Bob Truel. Strangely, like Yahoo!, the project enjoyed some success. The project started in June of 1998 and by the beginning of the year contained close to 300,000 links and had been purchased by Netscape ("Open Directory Project", 1998).

PHOAKS: People Helping One Another Know Stuff. A system designed to recommend URLs on a topic by analyzing how often they are mentioned in Usenet newsgroups (Hill & Terveen, 1996).

Recommender system: Resnick and Varian (1997) prefer this term for **collaborative filtering** systems. They hold that users often are not actually collaborating, and that the emphasis can be on making recommendations, not just filtering out junk.

Tapestry: A seminal system, developed at Xerox that implemented the idea of using **collaborative filtering** to help users choose and discover what is useful from a large stream of information (Goldberg et al., 1992).

WiseWire: Originated at Carnegie Mellon University by Ken Lang, this system grew from a way of evaluating Usenet news articles into a general-purpose **collaborative filtering** system. The system attempts to categorize incoming articles into topic areas ("wires"), using artificial intelligence to interpret user votes and adjust its evaluations of articles. WiseWire powers a number of corporate Web sites, chief among them the Web Guides of Lycos ("WiseWire", 1998).

References

About Webfavorites. (no date). Webfavorites, LLC. Retrieved February 8, 1999 from: <http://www.webfavorites.com/System/aboutwf.html>.

Alexa General FAQs. (1f998) Alexa Internet. Retrieved January 8, 1999 from: <http://www.alexa.com/whatisalexa/faq.html>.

Amazon.com—Books, Music, & More. (1999). Amazon.com, Inc. Retrieved March 20, 1999 from: <http://www.amazon.com>.

Avery, C., & Zeckhauser, R. (1997). Recommender systems for evaluating computer messages. *Communications of the ACM*, 40 (3), 88-89.

Berghel, H. (1997). Cyberspace 2000: Dealing with Information Overload. *Communications of the ACM*, 40 (2), 19-24.

Brandt, D. S. (1997). Constructivism: teaching for understanding of the Internet. *Communications of the ACM*, 40 (10), 112-117.

Campbell, K. K. (1994, October). A NET.CONSPIRACY SO IMMENSE…Chatting with Martha Siegel of the Internet's Infamous Canter & Siegel. KKC.NET. Retrieved January 30, 1999 from: <http://www.kkc.net/cs/>.

Cordani, J. R., & Tucker, R. J. (1998). Tools for higher education distance teaching. In *Proceedings of the 26th SIGUCCS conference on UserServices* (pp. 71-76). New York: Association for Computing Machinery.

Definition. (1997). Penn State Department of Distance Education. Retrieved January 30, 1999 from: <http://www.outreach.psu.edu/DE/current_def.HTML>.

EZbookmark—your online bookmark system. (1998). ezbookmark.com. Retrieved January 9, 1999 from: <http://www.ezbookmark.com>.

Frankovitz, J. (1999). itList—The Online Bookmark Manager. itList. Retrieved January 9, 1999 from: <http://www.itlist.com>.

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35 (12), 61-70.

Hartley, S. et al. (1996). Enhancing teaching using the Internet: report of the working group on the World Wide Web as an interactive teaching resource. In *ITiCSE '96. Proceedings of the conference on Integrating technology into computer science education* (pp. 218-228). New York: Association for Computing Machinery.

Help/FAQ. (1999). Clickmarks.com Inc. Retrieved February 8, 1999 from: <http://www.clickmarks.com/click/faq.html>.

Herlocker, J. (1998). About MovieLens. Grouplens Research Project. Retrieved March 20, 1999 from: <http://www.movielens.umn.edu/about_frame.html>.

Hill, J. R. (1997). Distance Learning Environments Via the World Wide Web. In B. H. Khan (ed.), *Web-Based Instruction* (pp. 75-80). Englewood Cliffs, New Jersey: Educational Technology Publications, Inc.

Hill, W., & Terveen, L. (1996). Using frequency-of-mention in public conversations for social filtering. In *Proceedings of the 1996 Computer Supported Cooperative Work Conference* (pp. 106-112). New York: Association for Computing Machinery.

Hiltz, S. R., & Wellman, B. (1997). Asynchronous Learning Networks as a Virtual Classroom. *Communications of the ACM*, 40 (9), 44-49.

Kehoe, C., Pitkow, J., & Rogers, J. (1998). GVU's 9th WWW Survey Results. Georgia Tech Research Corporation. Retrieved January 30, 1999 from: <http://www.gvu.gatech.edu/user_surveys/survey-1998-04/>.

Khan, B. H. (1997). Web-Based Instruction (WBI): What Is It and Why Is It? In B. H. Khan (ed.), *Web-Based Instruction* (pp. 5-18). Englewood Cliffs, New Jersey: Educational Technology Publications, Inc.

Kinsella, W. P. (Writer), & Robinson, P. A. (Director). (1989). Field of Dreams [Film]. Gordon Company.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997). GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, 40 (3), 77-87.

Landon, B. (1999). online educational delivery applications: a Web tool for comparative analysis. Centre for Curriculum, Transfer, and Technology. Retrieved January 25, 1999 from: <http://www.ctt.bc.ca/landonline/>.

Lawrence, S., & Giles, C. L. (1998, April 3). Searching the World Wide Web. *Science* 280, 98-100.

Lehrer, T. (Vocalist). (1959). We Will All Go Together When We Go. *An Evening Wasted With Tom Lehrer*. Reprise Records.

Leighton, H. V., & Srivastava, J. (1997). Precision among World Wide Web Search Services (Search Engines): Alta Vista, Excite, Hotbot, Infoseek, Lycos. (1997). Thesis, Winona State University. Retrieved January 30, 1999 from: <http://www.winona.msus.edu/library/webind2/webind2.htm>.

Looksmart: Submitting a Site. (1997). Looksmart Ltd. Retrieved April 17, 1999 from: <http://www.looksmart.com/aboutus/partners/subsite2.html>.

MURL.COM—Free Online Bookmark Manager. (1998). murl.com. Retrieved January 9, 1999 from: <http://murl.com>.

Oakes, C. (1998, February 11). Does Yahoo Still Yahoo? *Wired News*. Retrieved January 10, 1999 from: <http://www.wired.com/news/news/technology/story/10236.html>.

Open Directory Project. (1998). Netscape. Retrieved January 10, 1999 from: <http://directory.mozilla.org/>.

PHOAKS Home Page. (1996). Retrieved January 9, 1999 from: <http://www.phoaks.com>.

Quittner, J. (1995, December 5). The AOL ‘Breast Ban’. *The Netly News*.

Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40 (3), 56-58.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 Computer Supported Cooperative Work Conference* (pp. 175-186). New York: Association for Computing Machinery.

Romiszowski, A. J. (1997). Web-Based Distance Learning and Teaching: Revolutionary Invention or Reaction to Necessity? In B. H. Khan (ed.), *Web-Based Instruction* (pp. 25-36). Englewood Cliffs, New Jersey: Educational Technology Publications, Inc.

Rucker, J., & Polanco, M. J. (1997). Siteseer: personalized navigation for the Web. *Communications of the ACM*, 40 (3), 73-76.

Skrenta, R., & Truel, B. (1998). NewHoo Announcement. Netscape. Retrieved January 10, 1999 from: <http://directory.mozilla.org/ann.html>.

Smith, T. (1998). Spinalot Documentation. ULTRALAB, Anglia Polytechnic University. Retrieved February 5, 1999 from: <http://coyote.ultralab.anglia.ac.uk/cgi-bin/Spinalot/Documentation/index.htm>.

Soloway, E. (1998). Log on Education: no one is making money in education software. *Communications of the ACM*, 41 (2), 11-16.

Sullivan, D. (1998, June 3). Three's Company: Looksmart and Snap Challenge Yahoo. *The Search Engine Report*.

WiseWire Proprietary Technology. (1998). WiseWire Corporation. Retrieved January 9, 1999 from: <http://www.wisewire-corp.com/Technology.html>.

Yahoo!. (1998). Yahoo! Inc. Retrieved January 10, 1999 from: <http://www.yahoo.com/>.

Zipf, G. K. (1949). *Human Behavior and the Principle of Least Effort*. Cambridge: Addison-Wesley Press, Inc.