2-2006

Observed Web Robot Behavior on Decaying Web Subsites

Joan A. Smith
Old Dominion University

Frank McCown Ph.D.
Harding University, fmccown@harding.edu

Michael L. Nelson
Old Dominion University

Recommended Citation
Smith, J. A., McCown, F., & Nelson, M. L. (2006). Observed Web Robot Behavior on Decaying Web Subsites. *D-Lib Magazine, 12* (2). http://dx.doi.org/10.1045/february2006-smith

D-Lib Magazine
February 2006

Volume 12 Number 2

ISSN 1082-9873

Observed Web Robot Behavior on Decaying Web Subsites

Joan A. Smith, Frank McCown, Michael L. Nelson
Department of Computer Science
Old Dominion University
{jsmit, fmccown, mln}@cs.odu.edu

Abstract

We describe the observed crawling patterns of various search engines (including Google, Yahoo and MSN) as they traverse a series of web subsites whose contents decay at predetermined rates. We plot the progress of the crawlers through the subsites, and their behaviors regarding the various file types included in the web subsites. We chose decaying subsites because we were originally interested in tracking the implication of using search engine caches for digital preservation. However, some of the crawling behaviors themselves proved to be interesting and have implications on using a search engine as an interface to a digital library.

Introduction

The web has been crawled for content since Matthew Gray launched his Wanderer program in 1993 [1], but there are few studies detailing robots' behavior while spidering a site. How quickly will they crawl new resources on an existing website? How frequently do they re-crawl dynamic content? What happens when a document that was found ('200') on a previous visit is not found ('404') on the next visit? We would like to know if the robot will try again and, if so, how often will it retry.

We conducted a study of digital preservation in website caches [2]. Our goal in the experiment was to find out how long a web page would remain in a search engine's cache (i.e., be 'preserved'). We examined the caches of Google, Yahoo and MSN, comparing the cache contents with the pages they crawled, and noting how long pages stayed in the cache. In addition, we looked at whether HTML file types were preferred over others for crawling and/or caching. Although our experiment focused on the caching patterns of search engines, we also collected some interesting data on the behaviors of search engine robots. This article discusses our findings with regard to the robots that spidered our web subsites: what was crawled, how often the crawls occured, and crawling patterns they exhibited.

Related Work

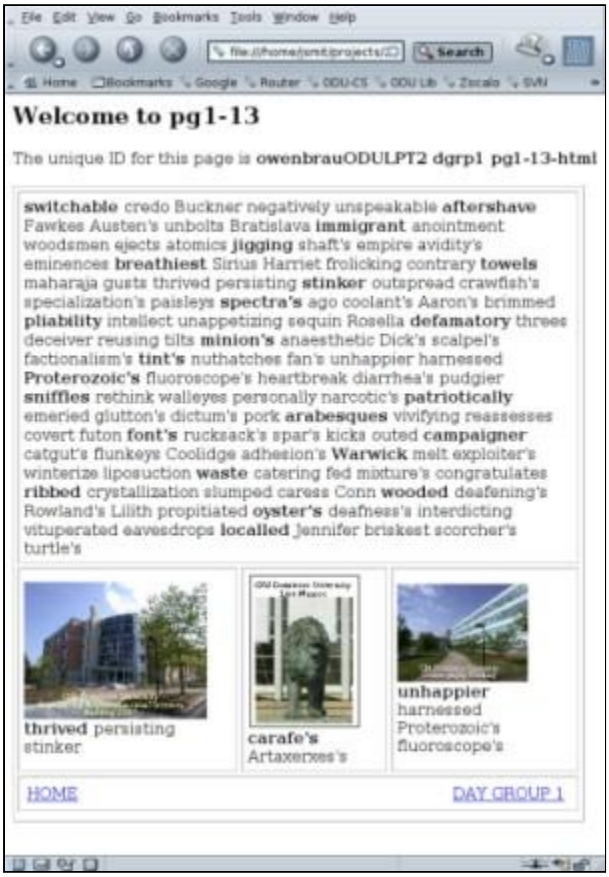
Most of the research on web crawling has focused on how to write better robots, how to improve the host web-server's performance when crawled by robots [3], crawling the deep web [4], stopping spam-crawlers [5], and attempting to influence topic-related page-ranking by search engines [6,7]. Cho and Garcia-Molina [8] looked at establishing page-refreshing rates to keep remote and local data sources synchronized. They also developed a method to identify replicated sites [9], saving crawlers valuable time

by eliminating the need to crawl the duplicated site. Others have proposed improvements to web-server caching behavior [10]. Hemenway and Calishain's "Spidering Hacks" [11] provides a collection of tools to facilitate crawls as well as to prevent them altogether. Lewandowski looked at the refresh rates [12] and search engine quality in general [13]. Oke and Bunt [14] looked at web server workloads and strategies for effective web server resource management. This article discusses the local behavior of search engine robots as they crawl through the site.

Design of the Test Web Subsites

To ensure our test pages would have unique information content and layout, we designed our own HTML and PDF files using a standard English-language dictionary, from which we randomly generated words for each page. The page style mimics one that would be found on typical medium-size websites [15]. Using these HTML and PDF documents, along with a few image files, we created test subsites. A subsite is the official World Wide Web Consortium terminology for a small subset of a larger site that is maintained by a different publisher than that of the host site [16]; we therefore use the term *subsite* to refer to the portion of the websites where our test pages were installed.

All of the test files contained common style tags (head, title, body, table, cellspacing, cellpadding, bold, tr, td, width, image, etc.), giving the subsites a 'realistic' look and feel. Figure 1 shows a sample of an HTML and a PDF page.



Full Size View

Full Size View

Figure 1: Sample HTML and PDF Pages

Each subsite had a home page that was a dynamic index file with links to each subdirectory index, as pictured in Figure 2. Links on the HTML pages led back to the home and subdirectory indexes. We were also interested in the preservation of image files, and included our own digital photos on the test pages. We used the three most common image types (JPG, PNG and GIF) throughout the subsite, but the images were small (PNG=63KB, JPG=14KB, GIF=7KB) so as to not overwhelm our own servers.

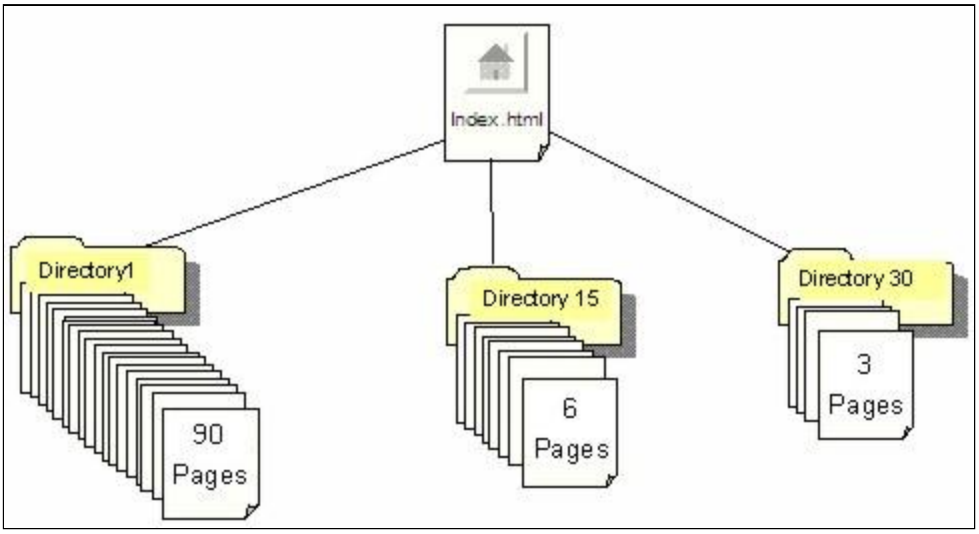


Figure 2: Examples of root and subdirectory index pages

Many sites and digital libraries implement the Robots Exclusion Protocol via a robots.txt file to reduce the load that crawlers place on the server [17]. We did not. Instead, we left the entire subsite available to both robots and humans.

Web Subsite Structure

Since we wanted to examine the preservation of expired web pages in search engine caches, we designed the subsites to gradually disappear over time. We also wished to examine the impact of average page life on cache persistence. To meet these dual goals, we decided on three 30-day cycles (i.e., 90 days total). We created 30 directories, populating each with HTML and PDF files and associated images. The number of HTML and PDF resources in each of the 30 subdirectories was based on the 90-day timeline of the experiment, and the 30 subdirectories determined how long resources would live in that subdirectory. Directory 1 had selected resources deleted every day; Directory 2 had resources deleted every two days; Directory 3 had resources deleted every three days; and so on. The total number of files in each subsite was 954, but the total subsite volume was less than 50MB, as Table 1 shows.

Table 1: Content of each test web subsite

Qty	Description	Size
31	index pages	48.5 KB
350	random-content HTML pages	735 KB
350	random-content PDF files	41,650 KB
74	PNG images	4,662 KB
75	JPG images	1,063 KB

74	GIF images	518 KB
954	**URIs per subsite**	**48,676.5 KB**

Creation of the Test Subsites

Although we used a standard English-language dictionary, we removed controversial words and phrases that might raise red flags for some of the search engines, particularly words that are often labeled "pornographic." We also randomized the word order of the dictionary, and used a random seed to set the starting point for each page. We created unique IDs for every page so that we could query for the individual pages without inadvertently advertising the new URLs to the search engines. The document examples in Figure 1 show the unique IDs clearly labeled near the top of the page. Note the construction of the ID, which combines code names for the site ("owenbrau"), the project ("ODULPT2"), the subdirectory ("dgrp1"), the page ("pg1-13") and the file type ("html" or "pdf").

Installation at the Sites

Using a custom tool we wrote for this experiment, each subsite was created off-line, tar-zipped, copied to the destination URL, and unpacked in place. A set of Perl scripts, included as part of the installation process, managed the schedule of daily changes to each test subsite. Four test subsite were deployed. Three of them were placed as subtrees (i.e., in user directories below http://www.cs.odu.edu/) in the university's computer science site, one for each of the three primary researchers (mln, fmccown, jsmit). Links were placed on the root page of FMC, MLN and JAS so the search engines could find the subsites. The fourth subsite was deployed at the root level of http://www.owenbrau.com, where the test subsite tree replaced the single placeholder page. All of the hosting sites had existed since at least January 2004 and had been previously crawled by various search engines. The primary change we introduced was the test content, and the controlled rate of decay by which we gradually deleted the test pages.

Table 2 lists the test subsites, and abbreviations used in this paper when referring to them. We also use "ODU" to refer to the root site, "http://www.cs.odu.edu/".

Table 2: Site URLs and abbreviations used in this document

Abbrev	Site Root URL
FMC	http://www.cs.odu.edu/~fmccown/lazyp/
JAS	http://www.cs.odu.edu/~jsmit/
MLN	http://www.cs.odu.edu/~mln/lazy/
OBR	http://www.owenbrau.com/

Update Cycle

At least one HTML and one PDF resource per subsite were removed every day. Links to the deleted files were also removed from the site, keeping all link references up to date. The 30 directories represented the life (in days) of directory resources. By the end of the 90-day test period, all resources, except the index pages, had been deleted from all four locations. Content availability at that point depended entirely on the search engines' cached versions of the resources. A detailed discussion of the caching behavior we observed is available [2]. Remarkably, a large number of test pages were still available in caches more than 30 days after the last page was deleted.

Crawler Profiles & Data Collection

All of the hosts at the subsites ("ODU" and "OBR") used Apache web servers with automated logging to a text file. However, the file formats differed between the ODU and the OBR servers, in both field ordering and content. The lowest-common-denominator of metadata was remote host IP address, time stamp of the request, the request itself (which contains the HTTP verb, URL, and HTTP version used), the status response returned to the requestor, and the number of bytes sent in the response. Compare the two examples taken from the JAS and OBR web logs, shown in Table 3.

Table 3: Sample Log Entries

Site	Actual Web Log Entry
JAS	207.46.98.59 - - [05/Jul/2005:00:14:06 -0400] "GET / jsmit/dgrp16/index.html HTTP/1.0" 200 815
OBR	access.log.25:66.249.66.69 - - [26/Jun/2005:17:44:42 -0400] "GET / HTTP/1.1" 200 1708 www.owenbrau.com "-" "Mozilla/5.0 (compatible; Googlebot/2.1; http://www.google.com/bot.html)" "-"

Notice how much more information is in the OBR log entry. The most useful field – user agent – where we see the Googlebot clearly identified, was only tracked at OBR. Why the difference in data logged by the servers? Web logging takes both processing time and disk space. For busy sites, reducing the fields tracked in the web logs can save hosting services time and money. In our case, the reduction in log data resulted in additional work by the researchers to track down the missing information.

To collect data, we filtered each server's logs for the period May 2005 through September 2005. Occasionally, system administrators had problems with the log rotation process or server hardware, and requests were either not recorded properly in the logs or logging was temporarily turned off. The impact of these problems is evident in the gaps seen in Table 6. Fortunately, the long-term nature of this experiment produced enough data to compensate for the gaps.

Webmasters reading this will not be surprised that we put nearly as much time into pruning and cross-checking data as we did in storing it. For the three ODU sites, we attempted to use log-resolve to find the host name or owner of the IP address. A large number of IP addresses would not resolve through DNS records. For those records, we performed manual checks of whois and similar databases to ensure that the non-resolving IPs were not newly added addresses of Google, Yahoo/Inktomi, MSN, or other major search engines. Also, a large percentage of crawls were from the ODU Bot, an in-house robot that crawls and indexes all of the university's web resources.

For example, during the month of June 2005 less than 5% of the test page requests at the MLN site came from one of the major search engines, and 63% were from the ODU Bot, as Figure 3 shows. Of the remaining 32%, nearly all were from non-resolving IPs, which were manually traced to various owners (most of them from outside the USA).

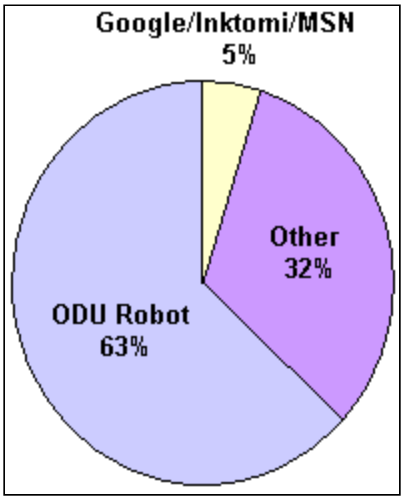


Figure 3: All MLN Site Visitors

For the ODU website, the in-house robot is again the most frequent single crawler. The ODU robot made 21% of the requests – a significant proportion of the month's activity, particularly when compared with the 9% made by Google, MSN and Inktomi combined. The "other" category, comprising 70% (over 2.8 million requests), had over 25,000 non-resolving IPs (over 1 million records) coming from a mix of human and robotic crawls. Figure 4 illustrates the relative distribution of visitors to the ODU site.

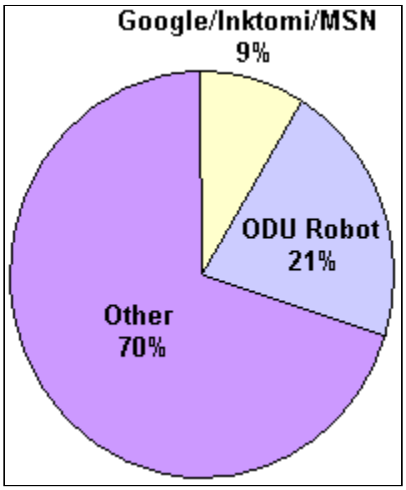


Figure 4: ODU Site Visitors

Even taking into account the fact that the ODU site has a significant proportion of non-robot requests, the burden imposed by the in-house crawler is not trivial.

Data extracted from the logs was stored in a MySQL database. We focused on the mainstream search engine agents (Google, Yahoo, and MSN) crawling our sites, posting the other crawls to a separate file. Because our focus was on the availability of our pages in search engine caches, we ignored crawls from unknown robots and spammers attempting to harvest emails since these do not have readily accessible caches. Yahoo acquired Inktomi in December 2002. Presumably Yahoo kept the Inktomi-named robots,

since no Yahoo-named robot crawled any of the test subsites, and only a very limited number (23,000 – i.e., 0.0006%) of pages at the main ODU site were requested by the yahoo.com domain.

Commercial search engines such as Google, Yahoo, MSN, and Picsearch, often employ named robots – that is, the IP resolves to a host containing robot, crawler, spider or some similarly identifiable robot-style title in the host name. See Table 4 for a list of examples from our target search engines.

Table 4: Example Robot Host Names from June 2005 ODU Web Log

Search Engine Robot-Host Names		
Crawler	Example Host Name	Unique Names
Google	crawl-66-249-71-18.googlebot.com	93
MSN	msnbot.msn.com	1
Inktomi	lj2326.inktomisearch.com	615
Picsearch	spider6.picsearch.com	13
ODU Bot	tektite ts odu edu	1

Early on, it was obvious that only Google, Yahoo (as Inktomi), and MSN were real contenders for a web preservation study. Nearly 80% of robots crawling the subsites were from one of these three search engines. Picsearch made only 315 requests, but was included since it is the image search engine for MSN. Less than 1% came from other hosts identified as robots. The remainding requests were of unknown host type. Table 5 summarizes the activity of the search engine robots we focused on in our study, as seen in the logs of our test subsites.

Table 5: Crawler activity at the test subsites May-Sep 2005

	Total Requests by Site			
Crawler	FMC	JAS	MLN	OBR
Google	2813	3384	3654	162
MSN	768	780	808	0
Inktomi	991	1735	1569	49
Picsearch	29	152	134	0

Notice the low crawl volume reflected for the OBR site. This is due in part to site web logs being unavailable for a 2-week period in July when the hosting service made some changes to its logging methods. Summary data for the site, available in a special report from the web hosting service, showed over 500 page requests occured during that period. This would bring the OBR site closer to the volume seen by the other sites. Still, lacking the details of those crawls makes it impractical to include the OBR site when analyzing crawler characteristics.

Search engines employ a large number of systems to make their crawls through a site. As observers of the crawls, we have a very limited insight into those systems. Some search engines, such as Google, associate a unique remote host name with each IP address. Others, such as MSN, may have numerous IPs but still resolve to only one remote host name. What happens at the remote host site is unknown, of course; search engines could be sharing each IP address among a large number of servers. Counting the number of distinct IP addresses is just a "fun fact" that says nothing about the computing power of the search engine nor about its in-house distribution of IP addresses. Sometimes we found that the log-resolve and whois

databases did not match precisely. For example, the IP 207.68.61.254 is attributed to Verizon in log-resolve but the whois database says that Microsoft owns the IP. This is not unusual, since resale of IPs to other business units is well documented. We therefore used the DNS entries as the final arbiter in cases where there was conflict.

Crawler Characteristics

Table 6 provides links to a series of graphs of the activities of the search engines on our subsites. The "Static Graphs" are bar charts, similar to those in Figure 6. The Animated Graphs are similar to Figure 5. The animated versions show the total activity (HTML, index files, and PDFs) for the subsite on a given day. The HTML requests plotted on the static bar graphs include both requests for index files as well as for non-index, HTML files.

Table 6: Cross Reference of Links to Data Graphs

Search Engine	Static Graphs of Crawler Activity			Animated Graphs of Crawler Activity		
Google	MLN	FMC	JAS	MLN	FMC	JAS
Yahoo (Inktomi)	MLN	FMC	JAS	MLN	FMC	JAS
MSN	MLN	FMC	JAS	MLN	FMC	JAS
ODU	MLN	FMC	JAS	MLN	FMC	JAS

The patterns for all three of the primary crawlers were similar for each of the test subsites: Request one or two index pages on the first day, then traverse the bulk of the site on subsequent day(s). Figure 5 shows the MLN subsite decay over time, and robot crawls of the resources. The X axis represents one of the subdirectories (0 is the subsite root directory). The Y axis represents pages in the subdirectories (0 is an index page). The green portion represents resources in each directory that have not yet been deleted. The red portion (deleted resources) grows as time progresses and the test subsite decays. The blue stars appear when a resource in the collection is requested. Looking at the animation of Google's crawls of the MLN site in Figure 5, the toe-dip, deep-plunge activity is obvious.

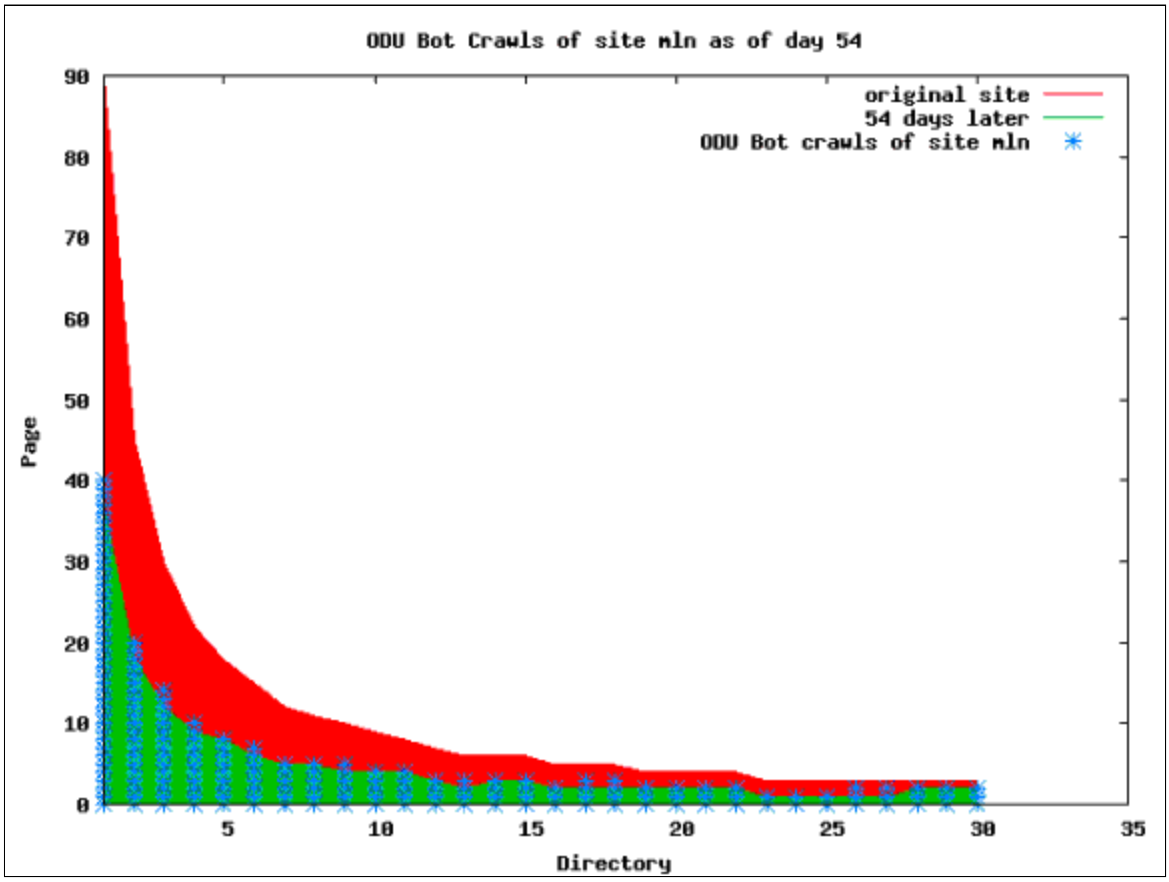
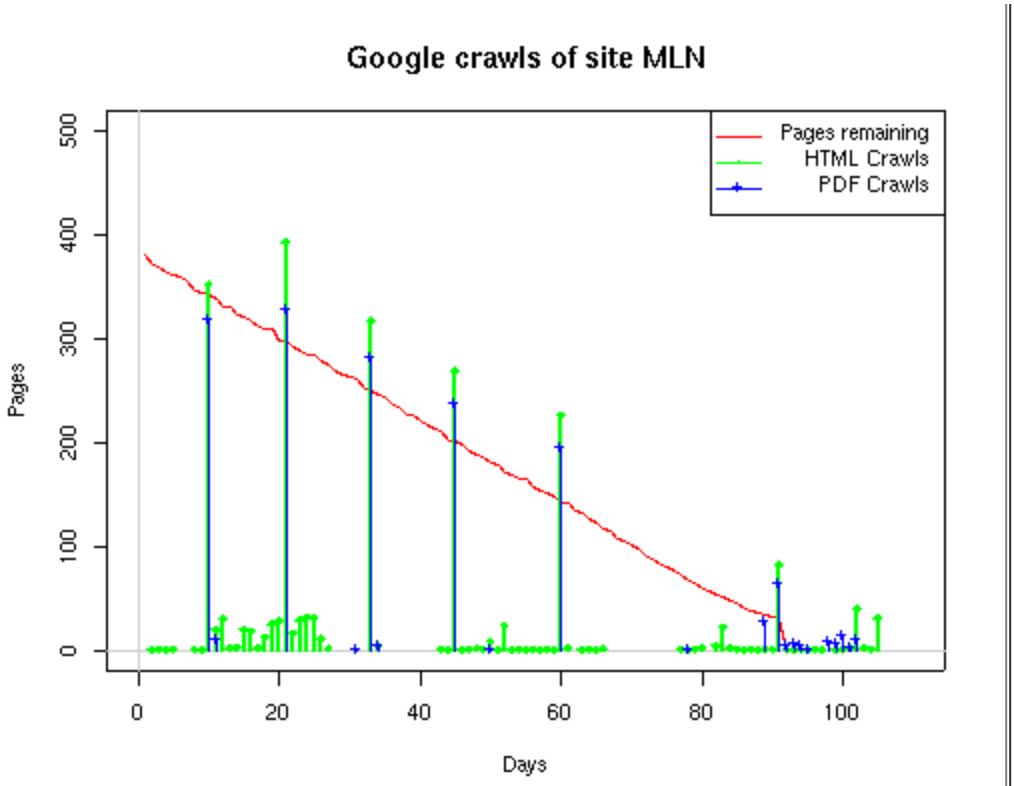
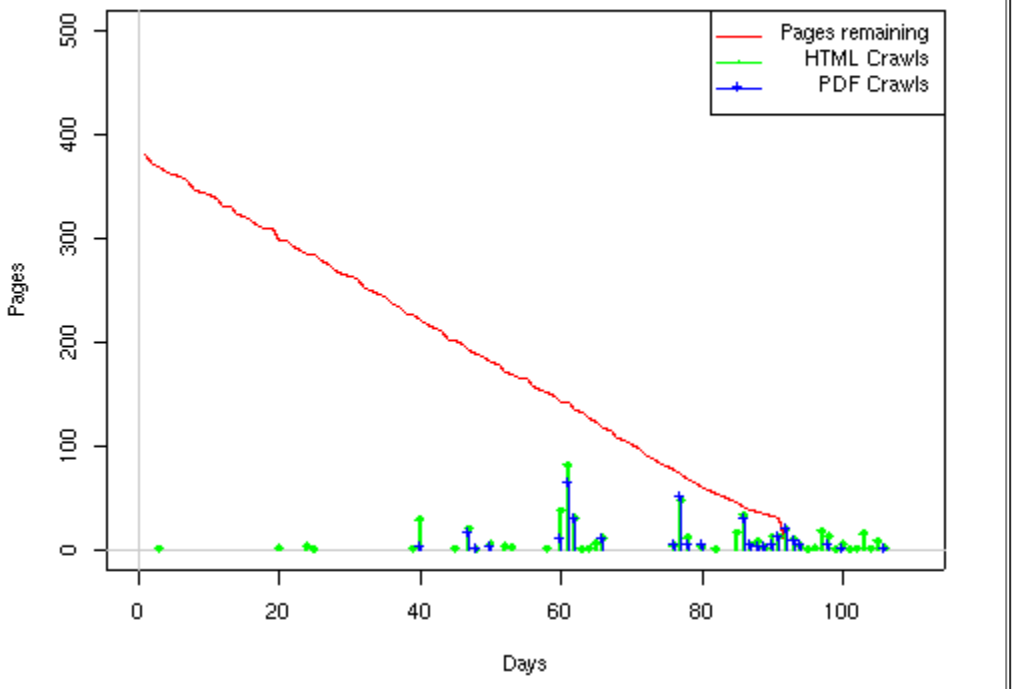


Figure 5: Day 54 of the MLN Subsite (90 Day Animation)

A static view of this behavior can be seen in Figure 6. In contrast, the ODU robot was consistently deep-crawling all of the Department's pages, including all of the test subsite pages (but none of the images). Compare the crawls by Google, Inktomi and MSN with the constant full-depth crawls of the test subsites by the ODU robot. We do not know if Google, Inktomi and MSN had analyzed the subdirectory indexes and determined that they acted as reliable keys to the current state of the test subsites; or if it was simply coincidence.


Google crawls of site MLN
Pages remaining
HTML Crawls
PDF Crawls
Pages
0
100
200
300
400
500
0
20
40
60
80
100
Days


MSN crawls of site MLN
Pages remaining
HTML Crawls
PDF Crawls
Pages
0
100
200
300
400
500
0
20
40
60
80
100
Days

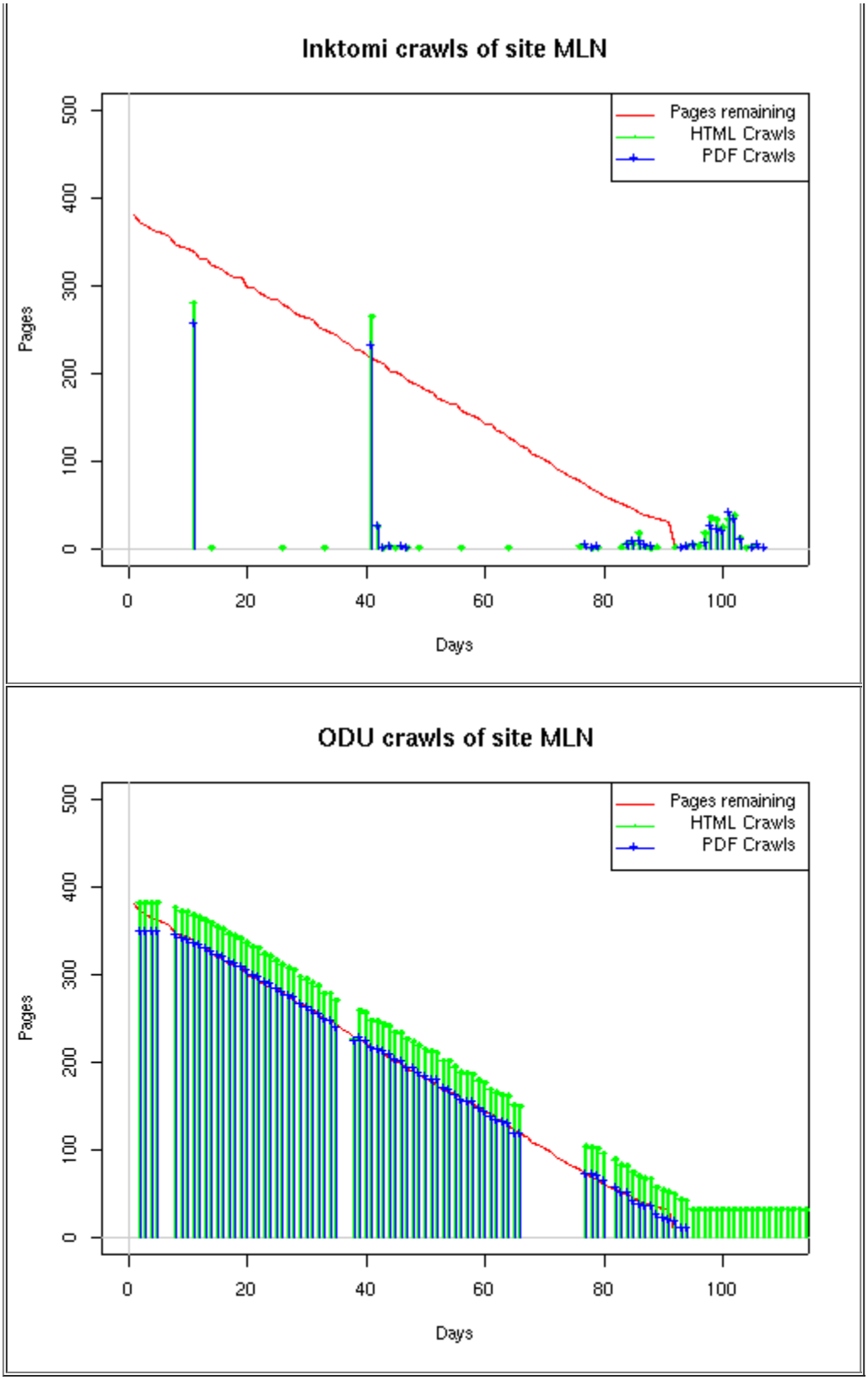


Figure 6: Crawling behavior of Google, MSN, Inktomi and ODU at site MLN

Consider the graphs of crawler behavior in Figure 6. The underlying data seems to indicate a pattern of toe-dip followed by a deep plunge. The spider requests a series of index pages on one day, then follows up a day or two later by getting all of the pages listed in the indexes. Google in particular exhibits this behavior on our test sites. We analyzed the resource types that each named robot crawled, to see if there was a pattern based on file type. For example, does Bot123 only crawl PDF but Bot124 crawl only

HTML? We did not detect a pattern. Aggregating the crawls by Robot Host Name merely proved what everyone knows, that most spiders crawl any and all types of pages.

A client can issue a conditional request for a page. That is, it can ask if it is unchanged. If it has not changed, then the server returns a '304' response which usually means the page will not be sent to the client who, presumably, has a locally cached copy. The host has just saved precious server processing time. 20% of Google's requests to the ODU site resulted in '304' responses during June 2005, whereas for the test subsites it was 48%. This is likely due to the fact that the test pages (except for the daily change to the index pages) were static, whereas the main ODU site is probably much more dynamic with many pages changing frequently.

Once a page is removed, repeat requests are rare, at least for Google, Inktomi, and MSN. Again, the MLN test subsite provides us with a good example. Only 603 requests by the Google spider returned a "not found" (404) result. Less than 6% were repeat requests. Twenty-nine pages were requested twice, two pages were requested thee times. In other words, once Google and the other spiders get a message from the server saying the file is not found, they do not ask for it again. In contrast, the ODU Robot persisted in asking for not-found resources as much as nine times each.

We also looked at the number and timing of robot requests. Hourly activity was evenly distributed across the ODU website (see Figure 7), except for the in-house ODU robot which was only active between midnight and 6 A.M.

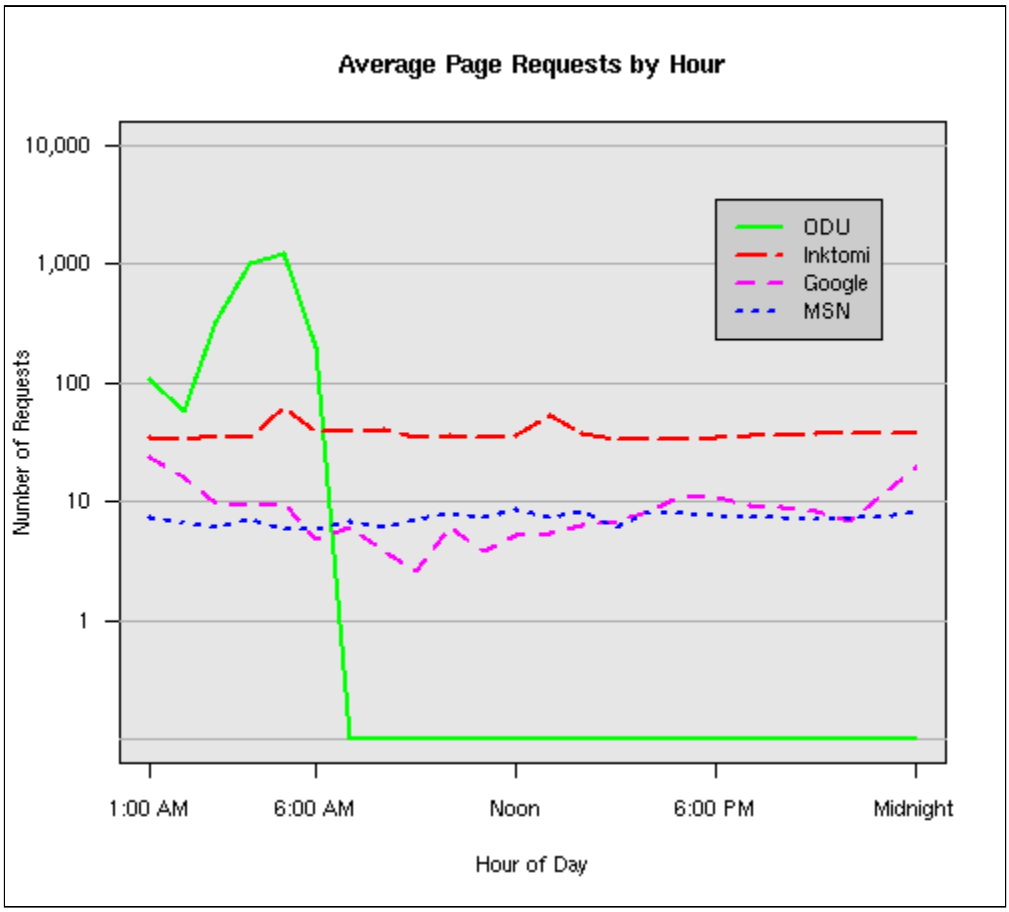


Figure 7: Average page requests per hour at ODU Website: June 2005

Restricting the in-house crawler certainly helped overall server performance at the ODU site, since the ODU indexing robot's activities was 30% greater than any other visitor, as Table 7 shows. In fact, it ranked number one, whereas the "Big 3" search engines barely make it into the list of top ten visitors to the Computer Science Department's website (cf. Table 7).

Table 7: Top 10 Visitors to the ODU Website: June 2005

Rank	Host	No. Requests	Comment
1	tektite.cs.odu (ODU robot)	885,742	Only Operates 12:01 A.M. - 05:59 A.M.
2	cs.odu.edu (non-robot)	622,779	Accesses via on-campus machines
3	cox.net	396,604	Local cable network provider
4	Inktomi	274,663	Many unique host names (bots)
5	blueice-ibm.com	128,125	4 variations on "blueice" name
6	ask.com	102,621	Also known as Ask Jeeves
7	comcast.net	88,201	Internet provider
8	verizon.net	73,500	Internet provider
9	googlebot	64,586	
10	MSN bot	52,435	

Internet Archive

One well-known host is conspicuous by its absence: the Internet Archive. We obtained the data presented here during an experiment on digital preservation, and we expected the Internet Archive (IA) to figure prominently in the results. However, fewer than a dozen test-page requests were made by IA, and all of these were initiated by the authors using the Wayback Machine advanced search interface during the first week of the test. IA does warn users that "there is a 6-12 month lag between the date a site is crawled and the date it appears in the Wayback Machine" [18]. But according to our logs, none of the subsites was crawled by IA at any point during the three months of our experiment – at least, not under the remote host names of "internet archive" or "alexa."

Implications for Search Engine Access to Digital Libraries

Anecdotally, we know that search engines are often used as the front end for contents in digital libraries. This is probably due to several factors, including the convenience of a single search interface, unhappiness with digital library's native search interface, or simply being unaware of the digital library until a search engine indexes its contents.

Although resource decay is not typical of a digital library collection, we can make some observations regarding search engine coverage of digital libraries. First, there is considerable latency between when the search engines become aware of a website (as exhibited by the toe-dip crawls that occur in the first few days) and when the crawlers begin a deep-plunge into the subsites. Google was the first to begin a deep plunge (at subsites FMC and JAS on day 9), followed by Inktomi (day 10) and then MSN (day 47). More interestingly, in only a few cases did the crawlers traverse the entire subsite even though the test subsites were not especially large in the beginning (954 files and depth of 1-3). While crawling "most" of the pages is probably good enough for regular websites, few digital libraries would consider partial coverage acceptable. Presumably the search engines would converge on complete coverage if the resources had been static, but this behavior could impact digital libraries that update their resources with comments, corrections, recommendations, etc.

Finally, the search engines seemed to demonstrate a slight preference for HTML resources although some search engine and site combinations treated PDF and HTML nearly equally. None seemed to exhibit a strong preference for image resources. Less than one-third of the images had been crawled at all. For the MLN site, only 85 images were ever requested, most of them (63, i.e., 74%) by Picsearch. Since the search engines build their image search functions based on text "near" images, this is not entirely surprising. Perhaps if the practice of "image tagging" (e.g., [19]) becomes more widespread, the search engines will respond by expanding the MIME types they crawl. All of this suggests that the practice of building "inverted repositories" (e.g., crawlable HTML pages – frequently with digital object identifiers (DOIs – that link to repository services) is especially important for digital libraries that have media types (or dynamic services) that would not otherwise be crawlable.

Further Research

Conducting longitudinal research on live systems has inherent risks. We found our experiment interrupted more than once by the vagaries and foibles of department-wide system administration activities. Lack of communication also impacted our ability to collect data, since ports were occasionally closed and resources relocated without prior coordination between departments. Occasionally, the Apache web logger would fail. The impact of these interruptions can easily be seen in Figure 6. Large gaps exist where deep crawls would otherwise be expected.

We would like to expand our web subsites to be less synthetic. To more accurately represent websites, they should feature greater variation in size and formatting, and employ more sophisticated natural language structure.

We intend to continue our investigation and comparison of persistence versus change in site content and its impact on search engine caching. Our next experiment will cover a longer period of data collection in the "live" environment, and will use an Apache module to store log information directly into a MySQL database on a research system. We hope this will circumvent the problems naturally arising from the daily activities of a very busy website.

Conclusions

Robotic crawls of websites naturally place a burden on the web server. The most popular search engines – the "Big 3" – were less demanding than the local crawling agent (the ODU robot). For Google, Inktomi and MSN, the pattern of requests was fairly evenly distributed throughout a 24-hour period. Again, the exception is our own in-house crawler, which focused its activity on the early morning hours. The ODU robot was also considerably more aggressive in its crawling, with an average of nearly 30,000 requests per day, so server performance probably benefited from the ODU crawler's restricted hours, particularly considering the daily volume of its requests and its tendency to repeatedly ask for pages that had been removed from the subsites. Google, Inktomi, and the ODU robots covered nearly 100% of the HTML and PDF resources of FMC, JAS, and MLN. Even though MSN arrived early (day 2 of the experiment), it crawled only 25% of the HTML and PDF files. Inktomi, on the other hand, did not arrive until the 10th day but still managed to cover nearly 75% of those files.

As expected, most crawlers appear willing to access all of the common file types, except for images. Power Point presentations, Excel spreadsheets, Word documents, text files, and Adobe PDF files on the ODU site were crawled by Google, Inktomi, MSN, and other robots. If there is a link to the document, it will be crawled (or at least, an attempt will be made to crawl it). This is one area where the Robots Exclusion Protocol can help, that is, if a site wants to exclude certain files, directories, or both, it should install a robots.txt file in the site's root directory.

We organized the test subsites into indexed subdirectories. The Big 3 search engines appeared to use this organization to schedule their crawls into toe-dips (look at the indexes to see what has changed), followed

by deep plunges (traverse the new list of resources). It is possible that this kind of site structure may help host sites to reduce the processing-impact of robot crawls on host servers by Google, Inktomi, and MSN.

Bibliography

[1] Allan Heydon and Marc Najork. Mercator: A scalable, extensible web crawler. In *WWW '99: Proceedings of the 8th international conference on World Wide Web*, pages 219-229, 1999. <doi:10.1023/A:1019213109274>.

[2] Frank McCown, Joan A. Smith, Michael L. Nelson, and Johan Bollen. *Reconstructing websites for the lazy webmaster*. Technical report, Old Dominion University, 2005. <http://arxiv.org/abs/cs.IR/0512069>.

[3] Onn Brandman, Junghoo Cho, Hector Garcia-Molina, and Narayanan Shivakumar. Crawler-friendly web servers. *SIGMETRICS Perform. Eval. Rev.*, 28(2):9-14, 2000. <doi:10.1145/362883.362894>.

[4] Sriram Raghavan and Hector Garcia-Molina. Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases*, 2001. Pages 129-138. <http://citeseer.ist.psu.edu/raghavan01crawling.html>.

[5] Zoltan Gyongyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Data Bases*, pages 576-587, 2004. <http://www.vldb.org/conf/2004/RS15P3.PDF>.

[6] Michelangelo Diligenti, Marco Gori, and Marco Maggini. Web page scoring systems for horizontal and vertical search. In *WWW 2002: Proceedings of the 11th international conference on World Wide Web*, May 2002. <doi:10.1145/511446.511512>.

[7] Taher H. Haveliwala. Topic-sensitive page rank. In *WWW 2002: Proceedings of the 11th international conference on World Wide Web*, May 2002. <doi10.1145/511446.511513>.

[8] Junghoo Cho and Hector Garcia-Molina. Effective page refresh policies for web crawlers. *ACM Trans. Database Syst.*, 28(4):390-426, 2003. <doi:10.1145/958942.958945>.

[9] Junghoo Cho, Narayanan Shivakumar, and Hector Garcia-Molina. Finding replicated web collections. In *Proceedings from SIGMOD '00*, pages 355-366, 2000. <doi:10.1145/342009.335429>.

[10[Virgilio Almeida, Daniel Menasce, Rudolf Riedi, Flavia Peligrinelli, Rodrigo Fonseca, and Wagner Meira Jr. Analyzing web robots and their impact on caching. In *Proceedings of the 6th Web Caching Workshop*, 2001. <http://citeseer.ist.psu.edu/almeida01analyzing.html>.

[11] Kevin Hemenway and Tara Calishain. *Spidering Hacks*. O'Reilly Media, Inc., first edition, November 2003.

[12] Dirk Lewandowski, Henry Wahlig, and Gunnar Meyer-Beautor. The freshness of Web search engines' databases. *Journal of Information Science* 31, 2005. <http://eprints.rclis.org/archive/00004619/>.

[13] Dirk Lewandowski. Web searching, search engines and Information Retrieval. *Journal of Information Services & Use* 18(3) 2005. <http://eprints.rclis.org/archive/00004620/>.

[14] Adeniyi Oke and Richard B. Bunt. Hierarchical workload characterization for a busy web server. In *Computer Performance Evaluation/TOOLS*, pages 309-328, 2002. <http://citeseer.ist.psu.edu/oke02hierarchical.html>.

[15] Edward T. O'Neill, Brian F. Lavoie, and Rick Bennett. Trends in the evolution of the public web: 1998-2002. *D-Lib Magazine*, 9(4), April 2003. <doi:10.1045/april2003-lavoie>.

[16] *Web Characterization Terminology and Definition Sheet, W3C Working Draft 24-May-1999*, May 1999. Brian Lavoie and Henrik Frystyk Nielsen, Editors. W3C Working Draft. 24-May-1999. <http://www.w3.org/1999/05/WCA-terms/>.

[17] Frank McCown, Xiaoming Liu, Michael L. Nelson, and Mohammed Zubair. Search engine coverage of the OAI-PMH corpus. Old Dominion University, 2005. To appear in *IEEE Internet Computing* March/April 2006.

[18] The Wayback Machine Frequently Asked Questions. <http://www.archive.org/about/faqs.php>. Quoted from website on 18 January 2006.

[19] Tony Hammond, Timo Hannay, Ben Lund, and Joanna Scott. Social bookmarking tools (I): A general overview. *D-Lib Magazine*, 11(4), April 2005. <doi:10.1045/april2005-hammond>.

---

doi:10.1045/february2006-smith