Lawrence, S., F. Coetzee, E. Glover, D. Pennock, G. Flake, F. Nielsen, R. Krovetz, A. Kruger, C. L. Giles.
Persistence of Web References in Scientific Research, IEEE Computer, Volume 34, Number 2, pp. 26–31, 2001.

Persistence of Web References in Scientific Research

Steve Lawrence, Frans Coetzee, Eric Glover, David Pennock, Gary Flake
Finn Nielsen, Bob Krovetz, Andries Kruger, Lee Giles

NEC Research Institute, 4 Independence Way, Princeton, NJ 08540
{lawrence,coetzee,compuman,dpennock,flake,fnielsen,krovetz,akruger,giles}@research.nj.nec.com

Abstract

The web has greatly improved the accessibility of scientific information, however the role of the web in formal scientific publishing has been debated. Some argue that the lack of persistence of web resources means that they should not be cited in scientific research. We analyze references to web resources in computer science publications, finding that the number of web references has increased dramatically in the last few years, and that many of these references are now invalid. We also find that most invalid web references can be relocated easily. We argue that, while formal references to published articles should always be used when possible, web references help to improve communication and progress in science. However, citation practices need to be improved to minimize future loss. We provide recommended practices for citing web resources, and discuss methods for relocating invalid references.

The web facilitates scientific communication in many ways. Formal references to information on the web are becoming increasingly common. However, there are many invalid links on the web, leading to user annoyance and frustration. The use of web references in research articles has been of particular concern. Some have argued that Uniform Resource Locator (URL) citations should not be contained in research papers, pointing out the lack of persistence of URLs and their contents. We examine URLs contained in computer science research articles, analyzing the volume of citations, the validity of links, and the detailed nature of invalid links.

We investigate URLs contained in research papers from the ResearchIndex (also known as CiteSeer) database [3, 6]. ResearchIndex indexes Postscript and PDF research articles on the web. A free service is available at `http://researchindex.org/` (if this URL is invalid, try searching for ResearchIndex or CiteSeer in a search engine). ResearchIndex currently contains about 270,000 research articles, including journal papers, conference papers, and technical reports. The database represents computer science papers that are available on the publicly indexable web [5].

We analyzed 270,977 articles in the ResearchIndex database. For the 100,826 articles that were cited and linked within the database, and hence the publication year was known, we extracted all URLs (67,577 URLs), and then attempted to access each URL. Redirected URLs were followed to their new destination. The experiments were performed during May 3 - May 5, 2000. URLs were extracted by searching for strings starting with (http:|https:|ftp:), and ending with a quote or whitespace. Trailing periods, commas, semicolons, parentheses, and brackets were removed from the strings.

Invalid URLs

Figure 1 shows the average number of URLs contained in the articles versus the year of publication. The number of URL citations has been increasing substantially since the inception of the web. Figure 2 shows the

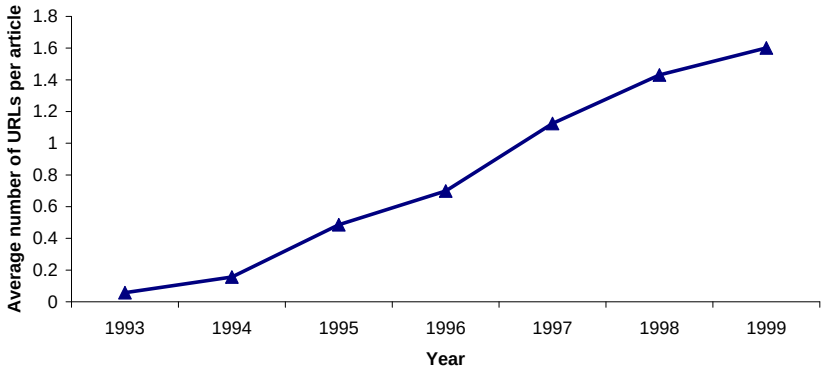


Figure 1. The average number of URLs contained in the articles versus the year of publication. The number of URLs has been increasing rapidly.

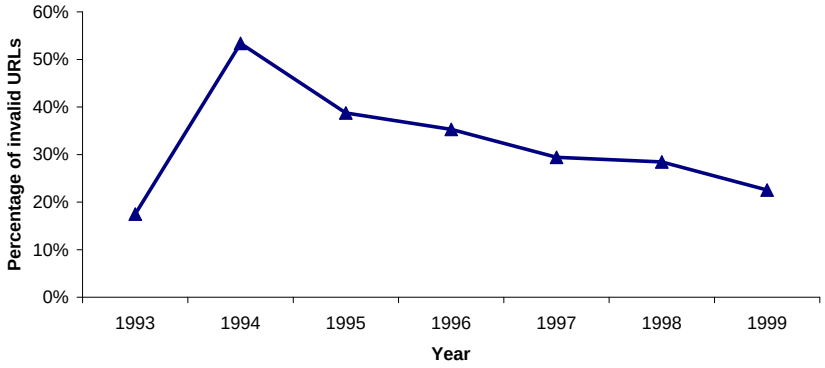
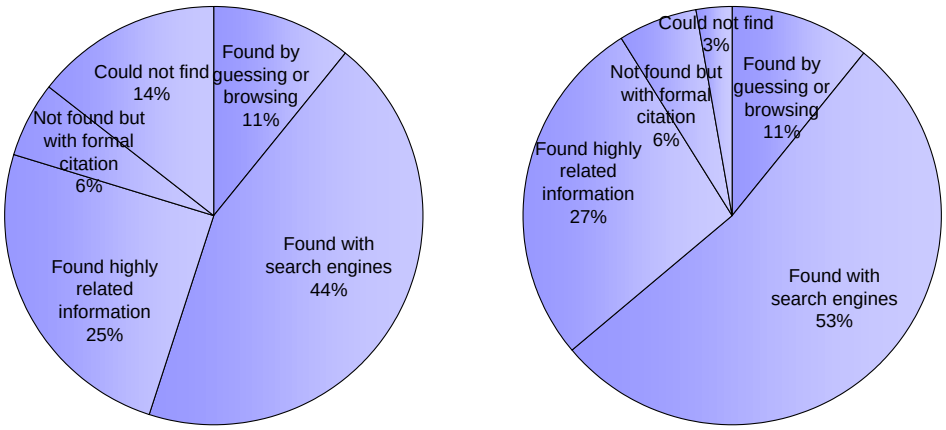


Figure 2. The percentage of invalid links contained in articles versus the year of publication of the articles. Many URLs are invalid, even for papers published in 1999 (23%). 53% of URLs from papers in 1994 are invalid.

percentage of invalid URLs in papers versus the year of publication of the source papers. The percentages are corrected so that they do not include URLs that were extracted incorrectly. The percentage of invalid URLs varies from 23% in 1999 to a peak of 53% in 1994. The lower percentage of invalid URLs in 1993 may be because many citations at this early stage of the web were to relatively well-known sites (e.g. http://www.intel.com/). However the sample size for URLs is relatively small prior to 1994, leading to lower accuracy (only 608 URLs were extracted from 1993 papers, while 21,056 URLs were extracted from 1998 papers).

For a random sample of 300 invalid URLs, we attempted to find the new location of the page cited, or highly related information. Of these URLs, 32% were either extracted incorrectly from the papers, contained a syntax error such that they could never be valid, or were example URLs that we believe were never intended to be valid. Extraction errors were typically due to the Postscript/PDF to text conversion program not converting special characters correctly or inserting spaces within the URLs (our extraction routine corrects for some easily identifiable cases, but not all). These URLs were removed from the dataset and the percentages reported are for the remaining URLs.

Figure 3 shows a breakdown of the remaining invalid URLs. We were able to find the new location of the page or highly related information 80% of the time. This 80% can be broken down into 11% of the invalid URLs that were relocated by guessing an alternate URL or browsing the web, 44% of URLs that were relocated with the help of a search engine, and 25% of URLs for which highly related information could be found (which is likely to be a good substitute for the original page, however we cannot guarantee this because we do not have access to the original page). For 6% of the URLs we could not find the new



(a) Initial classification (b) Classification after second searcher

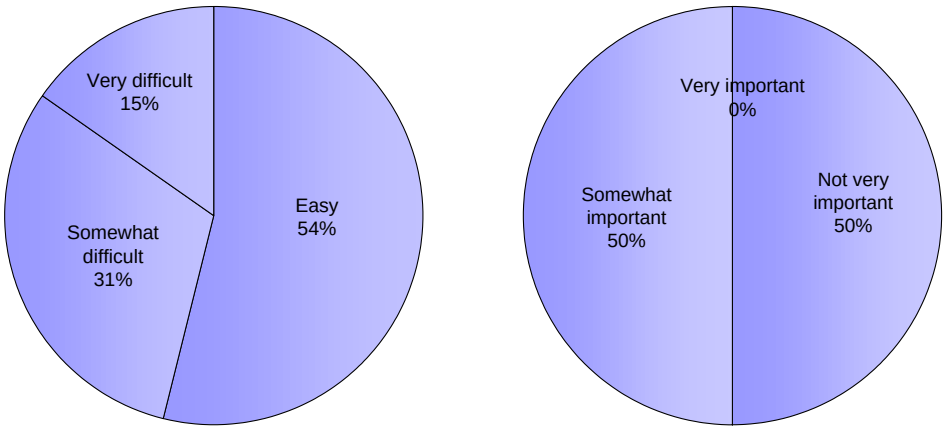
Figure 3. We examined and classified a random sample of URLs that are now invalid. The new location of the URL or highly related information could be found in many cases. A second searcher was able to find most of the URLs that could not be found by the first searcher.

location, but the URL was accompanied by a formal citation. The remaining 14% of URLs were not found. Moderate effort was put into locating moved or related information. No more than about five minutes was spent for each URL. More of the invalid URLs may be locatable given more time, more search experience, or better search tools.

URLs that were reported as lost were given to a second searcher. The second searcher was able to locate 80% of these lost URLs, bringing the overall percentage of lost URLs down to 3%. The revised percentages of URLs in each category after the second searcher can be seen in Figure 3. There was a significant difference in the success of locating URLs between the five individuals that participated in the experiment, with the most successful individual locating all URLs investigated, and the least successful individual being unable to locate 16% of the invalid URLs. These differences are due to differing search experience and abilities, different degrees of persistence, and differences in opinions regarding whether or not information was highly related in the case of related information.

For URLs where relocated or highly related information was found, the searchers estimated the difficulty locating the URLs. The following classes were used: easy, somewhat difficult, and very difficult. Figure 4 shows the percentage of lost URLs in each class. Most invalid URLs were easy to relocate.

For each invalid URL that could not be located, we examined the context of the citation in the respective paper, and estimated the importance of the URL with regard to the ability for future research to verify and/or build on the given paper. The following classes were used: not very important, somewhat important, and very important. 50% of URLs were classified to be not very important, and 41% were considered somewhat important. Only 9% of the lost URLs were considered to be very important with regard to the ability for future research to verify and/or build on the given paper. Figure 4 shows the classification of lost URLs after the URLs were sent to a second searcher. After the second searcher there were no lost URLs that were considered very important.



(a) Difficulty locating invalid URLs (b) Importance of URLs that could not be located

Figure 4. Classification of the difficulty locating invalid URLs, and the importance of lost URLs. It was easy to find the new location of most invalid URLs. None of the lost URLs were very important to the ability of future research to verify and/or build on a paper (after a second searcher).

Common causes of invalid URLs

Through our manual analysis of invalid links, we have identified a number of reasons why URLs become invalid: personal homepages tend to disappear when researchers move; URLs for academic software maintained on a personal machine may become invalid when machines are disconnected or machine names change; and sites may be restructured without maintaining old links. These problems are likely to persist without improved citation practices.

There are also a number of problems due to the initial rapid growth and evolution of the web. For example: most FTP servers have changed to HTTP; early pioneers ran their own web servers (personal machines), however the infrastructure is now typically provided by universities and corporations; it used to be more common for servers to be set up on non-standard ports; certain conventions in setting up sites have become more commonplace; URLs for homepages, for example, have tended to become standardized (e.g., http://www.x.com/~user/); and with domain names easily available, software has moved from personal repositories to dedicated sites. Increasing standardization should lead to fewer problems related to these changes.

Recommendations for generating and citing web resources

Although few critical resources cited in computer science articles appear to have been lost to date, we believe that improved citation practices are required in order to minimize future loss. Based on our experiences in labeling missing URLs, we have formulated a number of good citation practices that should improve the chances of future readers finding information that may have moved. We wish to emphasize that researchers have a vested interest in following good citation practices. A side-effect of dead links is that they may have a negative impact on the ranking of the containing material. For example, formal approaches have been proposed to bypass such pages during browsing [1], or reduce their ranking when presenting search engine results to users [8].

The following recommendations relate to all authors:

- Provide formal citations along with URL citations whenever possible. However, we believe that URL citations deemed valuable to the reader should be included even when a formal citation is not available. Although some percentage of links may become unavailable over time, this may be preferable to leaving links out (in which case 100% of them are unavailable to readers). Even when formal citations are available, the existence of an accompanying URL can significantly improve the accessibility of the information.
- Provide enough context information to enable readers to pose adequate queries to search engines in order to track down invalid links. For example, when giving the URL for a preprint, the full title of the document should be given as well, along with full details of the authors (as opposed to using "et al." for example). We found many examples where URL contents could not be inferred from the context.

Many URLs cite repositories controlled by the author. The following recommendations apply in this case:

- If possible, place materials in a reliable central repository, such as a preprint or software archive. We believe that this is particularly important for links to complete versions of papers, omitted proofs, and supporting data or results.
- Name repositories, and provide the name along with citations. This name can then be used for later searches. For software distributions, include a file with the name of the software package; this file can be indexed by some search engines. Provide a documented homepage for software, and establish a domain name if possible.
- When referencing software or software manuals, reference a URL for the entire project when possible, rather than URLs for specific versions of the software or manual. Version files frequently become unavailable when the software or manual is updated.
- Avoid URLs that depend on a personal directory, and URLs that depend on a specific machine or subnet name.

Finding the new location of information

Individual searchers in our study used different strategies when attempting to find relocated and related information. The search engines Google, ResearchIndex, and Inquirus [4] were most commonly used. Inquirus is a metasearch engine that combines the results of several search engines. Different search engines tend to index different sets of web pages, and combining the results of multiple search engines can significantly improve coverage of the web [5]. Other search engines used include AltaVista and Northern Light.

There are several common techniques that we found useful for finding the new location of information. If known, the title and/or author of a document can be searched for. It is often useful to search for the title as a phrase. The context of citations can be examined to generate alternative queries: for example, project, company, or institution names. Browsing from an alternative starting point (e.g., the top level page or a researcher homepage) within a site may be attempted in order to locate pages that may have moved to a different location on the same site. Guessing possible new locations may be attempted, for example academic software may have moved from a specific machine to its own domain, or the homepage of an individual may have changed to the standard notation (e.g., http://www.x.com/~user/). If a site has its own search engine this can be tried. If a URL has a relatively unique component then a search for this component may be attempted.

Enforcing link consistency

Alternatives to the World Wide Web such as Xanadu [7] enforce the consistency of links. However these systems are not widely used [9]. We argue that part of the reason for the success of the web may be the relative lack of requirements on the part of authors. A system that includes features such as enforced link consistency may impose too much overhead and added complexity that limits acceptance.

Rather than enforcing link consistency, which may make participation in the web more difficult by increasing resource requirements or system complexity, we recommend promotion of improved practices for citing URLs, use of services like PURL (see the sidebar), the availability of archives of the web such as Brewster Kahle's Internet Archive (http://www.archive.org/) [2], and the introduction of services that attempt to track and monitor URLs that move.

Our personal view is that it is not practical in the long term to expect individuals or small organizations to provide persistent access to on-line resources. Such material will probably ultimately move or disappear. To solve the general problem of persistence and disappearance, we believe that technical solutions and peer policies will have to be combined. Professional societies such as the IEEE and ACM, and funding agencies such as the NSF, could help by proposing and enforcing acceptable standards for citations. Ideally, all cited materials (especially those important to building on or verifying research) would be available from a stable repository, such as the Netlib Repository (http://www.netlib.org/), which is mirrored worldwide. These societies and agencies could promote preprint and software repositories, for example by sponsoring or hosting the repositories, or by requesting that authors use the appropriate repositories when possible.

References

[1] Paul De Bra and Geert-Jan Houben. A formal approach to analyzing the browsing semantics of hypertext. In *Proceedings of Computation and Neural Systems (CNS94)*, Monterey, CA, July 1994.

[2] Brewster Kahle. Preserving the Internet. *Scientific American*, March 1997.

[3] Steve Lawrence, Kurt Bollacker, and C. Lee Giles. Indexing and retrieval of scientific literature. In *Eighth International Conference on Information and Knowledge Management, CIKM 99*, pages 139–146, Kansas City, Missouri, November 1999.

[4] Steve Lawrence and C. Lee Giles. Context and page analysis for improved web search. *IEEE Internet Computing*, 2(4):38–46, 1998.

[5] Steve Lawrence and C. Lee Giles. Accessibility of information on the web. *Nature*, 400(6740):107–109, 1999.

[6] Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and Autonomous Citation Indexing. *IEEE Computer*, 32(6):67–71, 1999.

[7] Theodor Nelson. *Literary machines*. Mindful Press, Sausalito, CA, 1993. ISBN 089347052X.

[8] J. Shavlik and T. Eliassi-Rad. Intelligent agents for web-based tasks: An advice-taking approach. In *AAAI/ICML Workshop on Learning for Text Categorization*, 1998.

[9] G. Wolf. The curse of Xanadu. *Wired*, pages 137–202, June 1995. `http://www.wired.com/wired/archive/3.06/xanadu.html`.

SIDEBAR: Preserving information on the web

Scientists have long desired immediate access to all scientific knowledge. While there is much room for further improvement in access to information on the web [6], the web has already greatly improved access to scientific information. The benefits of being able to easily share a variety of materials at minimal cost,

unfortunately, are marred by the fact that most individuals or even organizations do not represent a reliable or stable publishing source. Web pages are abandoned, servers are shut down, and files are arbitrarily renamed. The lack of persistence of web references is a barrier to the goal of comprehensive shared access.

There have been many proposals for improving the situation. Some authors focus on the problem from the web designer's perspective – for example, proposing link management techniques [2]. Other authors propose augmenting existing web protocols to improve link persistence. Ingham et al. [4] suggest the development of an object-oriented network to exist in parallel with the current web that enforces referential integrity and performs garbage collection. Alternatives to the web such as Hyper-G [5] and Xanadu [8] contain built in mechanisms for enforcing link consistency.

A promising effort is the Uniform Resource Name (URN) specification [12], produced by the Internet Engineering Task Force. A URN is a persistent, location-independent identifier which can be used to uniquely identify a resource. The name stays the same even when the location of the resource moves. Implementations of URNs include the Persistent Uniform Resource Locator (PURL) [11] system, and the Handle [1] system. These are public systems that make use of resolution servers to resolve URNs into URLs. When the location of a URN moves, the resolver is updated so that the URN resolves to the new URL.

An optimal URN system would involve incorporating URN support into all Internet software such as web browsers. Unfortunately, retrofitting all Internet software is very difficult. Neither the PURL system, or the Handle system presents an optimal solution. The Handle system works by requiring the installation of software that performs the name resolution. Unfortunately, the large percentage of users that do not have the appropriate software included are unable to access handles (unless transformed, as below). The PURL system avoids the need for software support, but is not fully location-independent. The location-dependent address of a PURL resolver is part of a PURL (e.g. the PURL http://purl.org/metadata/dublin_core contains the address of the PURL resolver http://purl.org/). The use of PURLs relies on the continued existence of a particular PURL resolver, as well as the continued provision of adequate response time by the resolver. Proxy servers are available for the Handle system which make the system similar to the PURL system. For example the handle cnri.dlib/july95-arms can be transformed into a URL resolved by the proxy server hdl.handle.net: http://hdl.handle.net/cnri.dlib/july95-arms.

The PURL system is probably preferred to the Handle system currently due to the requirement for software support with the Handle system. The PURL system appears to be more popular currently, with over 500,000 PURLs registered. We recommend use of the PURL system when long-term persistence is desired, and users are prepared to maintain the validity of the redirection.

We searched for all URLs that are resolved by the main PURL resolver, purl.org, in the ResearchIndex database (we searched for all URLs containing the string "purl"). Other PURL resolvers may exist, however we believe these to be much less popular. The results show poor adoption of PURLs. Of the 67,577 URLs extracted from the papers in ResearchIndex, we were only able to locate 11 PURLs (0.016% of URLs), all of which were to the same resource: http://purl.org/metadata/dublin_core.

Note that both the PURL and Handle systems require someone to maintain the validity of resources. Despite the obvious motivation of early users, already not all PURLs are valid. A search for url:purl.oclc.org at AltaVista turned up many PURLs that return a page stating that "The requested PURL has been deactivated and can not be resolved."

Replacing HTML by improved protocols (see [7], for example) could in the future result in interesting content-based solutions. These protocols could provide support for improved content based indexing and retrieval. In principle, content summarization and indexing can be used by search engines to recognize materials that move on the web. Phelps and Wilensky [9] have shown that most documents on the web can be uniquely identified based on a small set of words that no other document shares. These words can be used to augment URLs, and may be used to locate documents that move.

These approaches can at best redirect attention to material that has moved, but not disappeared. We must

also face the issue of material becoming totally lost. It is therefore important that reasonable estimates of the problem of invalid web citations be obtained, and reasonable policies be instituted by academic societies and publishers to encourage good practices.

Even when the location of a web citation is stable, its contents can change, such that subsequent readers may not be viewing exactly the same material as cited. This issue can be addressed with version management as in Xanadu [8] and other proposals, or by periodically archiving the whole web. The Internet Archive (http://www.archive.org/) stores snapshots of information from the web. However, it is still an open question whether this approach can fully solve the problem. Alexa Internet, which creates web navigation software and studies trends in content and behavior on the web, has estimated that web pages disappear after an average time of only 75 days. Furthermore, taking a snapshot of the web is non-trivial. The time required to download a snapshot means that many pages will change while the snapshot is being generated.

Another related issue is the possibility of limited or no availability to the hardware and/or software required to read specific data formats.

Other efforts to improve permanence on the web include the Intermemory project at NEC Research Institute [3], which aims to create highly survivable and available storage systems using widely distributed processors, of which each individual processor may be unreliable and untrustworthy, and LOCKSS (Lots of Copies Keeps Stuff Safe) [10], in which multiple libraries work together to redundantly cache copies of specific documents.

References

[1] W. Arms, C. Blanchi, and E. Overly. An architecture for information in digital libraries. D-Lib Magazine, February 1997. `http://hdl.handle.net/cnri.dlib/february97-arms`, `http://www.dlib.org/dlib/february97/cnri/02arms1.html` (Search for CNRI, Magazine of Digital Library Research).

[2] M.L. Creech. Author-oriented link management. *Computer Networks and ISDN Systems*, 28(7–11):1015–25, 1996.

[3] A. Goldberg and Peter N. Yianilos. Towards an archival intermemory. In *Proceedings of IEEE Advances in Digital Libraries, ADL 98*, pages 147–156, Santa Barbara, CA, 1998. IEEE Computer Society.

[4] D. Ingham, S. Caughey, and M. Little. Fixing the ”broken-link” problem: the W3Objects approach. *Computer Networks and ISDN Systems*, 28(7–11):1255–68, 1996.

[5] F. Kappe, K. Andrews, and H. Maurer. The Hyper-G network information system. *Journal of Universal Computer Science*, 1(4):206–220, 1995.

[6] Steve Lawrence and C. Lee Giles. Accessibility of information on the web. *Nature*, 400(6740):107–109, 1999.

[7] S. Mace, U. Flohr, R. Dobson, and T. Graham. Weaving a better web. *Byte*, 23(3):58, 1998.

[8] Theodor Nelson. *Literary machines.* Mindful Press, Sausalito, CA, 1993. ISBN 089347052X.

[9] T. Phelps and R. Wilensky. Robust hyperlinks cost just five words each. Technical Report UCB//CSD-00-1091, University of California, Berkeley, January 2000.

[10] David S. H. Rosenthal and Vicky Reich. Permanent web publishing. In *Freenix*, San Diego, CA, June 2000.

[11] Keith Shafer, Stuart Weibel, Erik Jul, and Jon Fausey. Persistent Uniform Resource Locators. `http://www.purl.org/` (OCLC Online Computer Library Center).

[12] K. Sollins and L. Masinter. Functional requirements for uniform resource names. Internet Request for Comments, RFC1737, `http://www.cis.ohio-state.edu/htbin/rfc/rfc1737.html`, December 1994.