

Indexing the Internet

Michael Middleton
School of Information Systems, Queensland University of Technology

This is the text for an invited paper presented at:

Indexers - partners in publishing: First International Conference of the Australian Society of Indexers, held March 31 - April 2 1995, Marysville Victoria.

It was subsequently published as:

Middleton, M.R. (1995). Indexing the Internet. In M. McMaster (Ed.), *Indexers - partners in publishing: proceedings from the First International Conference of the Australian Society of Indexers, held March 31 - April 2 1995, Marysville Victoria* (pp. 196 - 205). Melbourne: Australian Society of Indexers. [ISBN 0 646 25050 7].

ABSTRACT

Navigation around information resources on the Internet is assisted by browsing software such as the gopher, lynx, Mosaic and Netscape. The variety and complexity of resources has meant there have been calls for meta-information so that there are resident on the network, tools for guidance to the resources using classified or index term approaches.

Searching software used in association with the browsing software has functionality that may be compared with that of vendor-operated information retrieval systems. Although this functionality is relatively limited, when used with HTML documents, relevance feedback assists retrieval via associative mechanisms.

Documents published electronically using SGML standards are able to contain their own surrogates and therefore be self-indexed and self-classified by the authors, however because the author view does not necessarily represent the user view, it is still necessary to have user-oriented retrieval aids.

Menu-linked approaches via gophers or via home pages constructed for WWW browsing software mean that users can superimpose their own classified view of resources. Therefore, rather than attempting an overall classification of what is on the Internet, multiple classification systems may be constructed, reflecting the orientations of the different disciplines of users.

A model is suggested for construction of such classified approaches with reference to the way they may use existing retrieval systems, automatic indexing, and thesauri that have been manually or automatically constructed.

INTRODUCTION

The Internet, and its growth through the proliferation of resources being made available through the various linked networks, are documented daily in the press and the specialist information technology literature. There are many documents dealing with the history and development of the Internet, and there are many that give guidance on its use. These documents appear in print form^a, on the Internet itself, or in both print and computer form^{1,2,4}.

The huge amount of information available, together with the principle of linking documents together using hypertext markup, means that the matter of finding one's way to required material can present significant challenges. The process is generally known as navigation (or perhaps *surfing* if the discoveries are serendipity).

Various approaches have been adopted to assist navigation. These may be categorised in three ways:

1. The development of **browsing software** that provides a friendly interface to resources on the network through the World Wide Web (WWW)^{2,3} and per medium of links via other communication protocols, to information via file transfer, news servers and remote login to interactive sites;
2. The development of specific WWW sites that provide **structured guidance** to users by use of classified or categorised approaches to information resources;
3. The provision of **information retrieval** support through a combination of:
 - a. creating public databases that are surrogates of WWW home pages through use of automatic web searching software known as *worms* or *robots*, and
 - b. building search functions into information retrieval software to improve the precision and recall of searches of Web sites, or of Web databases using such facilities as Wide Area Information Service (WAIS).

These approaches provide a combination of manual and automatic content analysis of Web sites, and therefore represent indexes to the Internet. They are increasingly being used in combination in order to assist users to navigate the Internet.

^a From the 19th March 1995 version of the Unofficial Internet Book List (version 1.1) at <ftp://rtfm.mit.edu/pub/usenet/news.answers/internet-services/book-list> come these figures:

Number of books in this list: 239
Additions to this version of the list: 35
Least expensive books: free (Guide for Accessing California
Legislative Info, tied with NetPages)
Most expensive book: \$100 (Internet Handbook for Law Librarians)
Thickest book: 1700 pages (Information Infrastructure Sourcebook)
Thinnest book: 10 pages (The Internet at a Glance)

BROWSING SOFTWARE

A variety of software has been developed to provide users with structured entry to Internet information resources. A typical example is the gopher developed at the University of Minnesota that enables compatible resources to structured in such a way that a menu-based approach guides entry to information of various types ranging from text files to encoded software for file transfer to remote online connection using telnet protocol.

The development of the WWW protocols that permit interlinking of hyperlinked documents has been accompanied by development of graphically-oriented browsers. Among the most heavily used of these initially was Mosaic from The National Center for Superconducting Applications at the University of Illinois. It provides a point and click interface to hypertext documents, and has been much copied.

There are now many similar browsers on the market or available for free. There is an expectation that browsers incorporate such features as:

- The ability to retain *bookmarks* or *hotlists*.

This is a facility that enables the user to record network addresses so that these addresses may be joined automatically on a point and click basis at a later date. In some cases these hotlist or bookmark facilities have been developed to enable self-categorisation of addresses by the user. The Netscape browser for example, permits the user to store, structure and label addresses, in effect providing structured classification support.

- Provide forms support.

This means that creators of home pages may set them up in such a way that they enable viewers to enter information into boxes. The entered information is then used by the server machine for such purposes as searching a database for matches to an information retrieval query.

- Provide page cache and history log facilities.

This means that the local client machine may retain for some time large amounts of data coming from remote pages, so that if the user refers back to earlier pages, the browser is able to reproduce them by reference to its history record from their local location, rather than reverting to a reload from a remote site.

A more detailed idea of the features that browsers are expected to have, is shown in Appendix 1, which is an extract from *PC Magazine* that shows part of a comparison.

STRUCTURED GUIDANCE

The HTML itself may be used so that controllers of particular sites may set up pages at the sites to provide guidance to the network as a whole.

There are many examples of such sites ranging from that of CERN (originator of much WWW software), to facilities such as Global Network Navigator and EINet.

At a local level there are Australian sites such as the Australian National University's and the National Library's, and at specific subject level many individuals make their own Home Pages to act as pointers to subject material.

In some of our units at QUT we require students who are undertaking an information management or information systems course to act as information intermediaries by building subject-specific pages linked to their own home page. Some examples of these can be found at 'Examples of Student Assignments' on one of the QUT pages^b.

The better examples of structured guidance embody a combined classification and index term approach. This type of approach can be seen utilised on CD-ROM software such as that used with Microsoft's *Encarta*. Guidance to the required material is assisted by broad classification categories that may be use in conjunction with keyword searching. Although some Internet sites are providing guidance through both categorisation and through keyword searching, they are not to my knowledge combining the two in such a way that keyword searching may be conducted within categories.

A combination of ready-build categories together with WAIS searching would make this possible.

INFORMATION RETRIEVAL

The most developed type of support for information retrieval is provided by software that provides keyword searching of contents of client machines making material available to the Internet. For example the Veronica search facility that enables searches of indexed Gopher clients permitted searches of a database of approximately 15 million items in December 1994 in the following resource categories:⁶

b	CERN	< http://info.cern.ch/ >
	GNN	< http://bond.edu.au/gnn/gnn.html >
	EINet	< http://www.einet.net/ >
	ANU	< http://polly.anu.edu.au/ >
	National Library	< http://www.nla.gov.au/ >
	QUT Student pages	< http://www.fit.qut.edu.au/frill/itb323/assign3.html >

0 -- Text File	s -- Sound
1 -- Directory	e -- Event (not in 2.06)
2 -- CSO name server	I -- Image (other than GIF)
4 -- Mac HQX file.	M -- MIME multipart/mixed message
5 -- PC binary	T -- TN3270 Session
7 -- Full Text Index(Menu)	g -- GIF image
8 -- Telnet Session	h -- HTML, HyperText Markup Language
9 -- Binary File	

Certain types of data NOT served directly by gopher servers are also included in the index if the resources are referenced on menus of indexed gopher servers. These types are: telnet sessions, CSO sessions, HTML files served by WWW servers, and type-7 searches. These items are included in the index even though they reside on non-gopher servers.

Many examples of software that perform the same type of function as Veronica, often called robots, exist for WWW as opposed to gopher sites. This software based at a particular site automatically moves around the Web searching publicly available HTML-encoded documents. Robots are set to examine particular parts of documents. From document information they produce a database that essentially is an index to pages on the WWW.

Robots may seek document information based upon:

- uniform resource locaters
- document titles based on HTML code (Titles are not mandatory and many are absent)
- document textual content
- document links to other documents

At present robots do not provide indexes to graphic or multimedia objects that are referenced by pages unless there is a corresponding textual representation.

Robots consume significant resources and there is a robot developers etiquette that recommends automatic database building to be carried out in certain ways. This is so as not to tie up information servers from which information is being obtained to construct the index databases.

The following table showing some robots and their addresses, is based upon material maintained by Kelly⁷. Some of the main searching tools are listed below:

Robot software	Uniform Resource Locator (URL)
Aliweb	http://web.nexor.co.uk/public/aliweb/aliweb.html
CUI WWW Catalog	http://cuiwww.unige.ch/cgi-bin/w3catalog
EINet's Galaxy	http://galaxy.einet.net/
Globewide Network Academy	http://uu-gna.mit.edu:8001/cgi-bin/meta/
InfoSeek	http://www.infoseek.com/
Jumpstation Robot	http://www.stir.ac.uk/jsbin/js
Lycos	http://lycos.cs.cmu.edu/
Nikos	http://www.rms.com/cgi-bin/nomad
RBSE URL	http://rbse.jsc.nasa.gov/eichmann/urlsearch.html
WebCrawler	http://webcrawler.cs.washington.edu/WebCrawler/WebQuery.html
World-Wide Web Wanderer	http://www.netgen.com/cgi/wandex
World-Wide Web Worm	http://www.cs.colorado.edu/home/mcbryan/WWW.html
Yahoo	http://www.yahoo.com

Some further information on specific robots derived from Nexor⁸ is listed in Appendix 2.

Each of these facilities is available without extra cost to users of the Internet. Use of them is extremely heavy, and we can expect to see introduction of more facilities such as Infoseek which has both a free version and an embellished commercial version, that in addition to more than 200,000 fully indexed Web pages, incorporates the following indexes to other databases such as the following:

- a month's contents of over 10,000 newsgroups;
- A variety of news services such as Associated Press, Reuters, Newsbytes, and PR Newswire;
- Cineman Movie, Book, and Music Reviews;
- Computer Select, Computerworld, InfoWorld;
- Hoover's Masterlist of U.S. Companies;

Results are returned in English-like description, rather than just HTML links, as is the case with many robots.

Robots present a number of problems in terms of their search performance:

- Robot updates are carried out periodically so there is no indication in the database created if an indexed site has been withdrawn from service;

- all sites are indexed which means that there is no differentiation between high quality edited resources and ephemeral material such as the personal interests of undergraduates;
- robots obtaining information from servers may place heavy loads on servers;
- there is no controlled vocabulary contained in the many sites that are indexed. The index terms are therefore subject to the vagaries of free-text indexing.

Hodge⁹ has reviewed automatic indexing support for large database production and notes that there is a continuum from no support... to clerical activity support... to quality control support... to intellectual support... to automatic, and notes the considerable vocabulary control that goes with it. Although there is automatic indexing using robots, there is not thesaurus application for quality control of Web pages at this stage.

However there have been calls for greater structuring of documents to permit keyword inclusion of fields. For example Desai¹⁰ has proposed a semantic header for documents that among other things includes subject keywords within documents as part of HTML. This principle is one that has been under consideration for Standard Generalized Markup Language (SGML) of which HTML is a subset. In Desai's discussion document a generalised approach to HTML structure incorporating indexing is proposed as in the following figure:

```

<semhdr>

  <title> ..... </title>
  <subtitle> ..... OPTIONAL </subtitle>
  <alttitle> ..... OPTIONAL </alttitle>
  <char-set> ..... OPTIONAL </char-set>
  <Language> ..... </Language>
  <author>
    <aname> ..... </aname>
    <aorg> ..... </aorg>
    <address> ..... </address>
    <aphone> ..... </aphone>
    <afax> ..... </afax>
    <aemail> ..... </aemail>
  </author>
  <Subject>
    <General> ..... </General>
    <Sublevel1> ..... OPTIONAL </Sublevel1>
    <Sublevel2> ..... OPTIONAL </Sublevel2>
  </Subject>
  <Keyword>
    .....
  </Keyword>
  <Dates>
    <Creatred> ..... </Creatred>
    <Expiry> ..... </Expiry>
    <Updated> ..... </Updated>
  </Dates>
  <Version> ..... </Version>
  <Hardware>
  <Software>
    .....
  </Software>
  <Coverage> ..... </Coverage>
  <Classification> ..... </Classification>
  <Annotation> ..... OPTIONAL </Annotation>
  <URL> ..... </URL>
  <URN> ..... </URN>
  <UAS> ..... </UAS>
  <Cost> ..... </Cost>
  <abstract> ..... OPTIONAL </abstract>
  <size> ..... </size>
</semhdr>

```

It may be seen that provision is made within the markup language for keywords to be supplied within HTML documents. This is a precursor of self-cataloguing and presages identifier tags along the lines of Machine-readable Cataloguing (MARC) tags that have been in use for about 20 years for print document and other library materials cataloguing. HTML self-cataloguing, if adopted by the publishing community, may be used by robots to build reference databases that refer to the full text of WWW pages.

This type of development with HTML may be contrasted with approaches suggested, not for WWW pages, but for hypertext documents being created using PC authoring software for publishing on compact disk. Liebscher¹¹, for example sees self-indexing being part of the document creation process, permitting different conceptual representations of the same document, with links that are really embedded index terms deriving their meaning from their relationship to expanded concepts in the hypertext.

Marchionini¹² proposes an algorithm for a hyperdocument starting with an index and proceeding by:

- Identification of main facets;
- Generation of an exhaustive list of terms & phrases;
- Map terms/phrases to facets;
- Determination of preferred terms (label nodes);
- Writing articles (create nodes), marking cross references;
- Reviewing articles (nodes), revising node setting according to criteria;
- Importing files into hypertext system and implementing links;
- Testing and editing hyperdocument.

The degree of functionality of retrieval software for searching databases that have been constructed by robots is generally quite limited. Although standard boolean facilities are usually available, they are not supported by the ability to put interim results into sets for recombination - a facility that is *de rigueur* in online retrieval software.

Among the more developed software in use is the WAIS¹⁴ software which in addition to provision of Boolean combinations, facilitates:

- Ability to limit searches to fields.

For data collections whose documents are structured in a semi-regular format, the regular portions of the documents can be tagged by the WAIS parser as fields. A client can then ask a WAIS server to limit its search to those documents containing a user-specified value of a particular field.

- Right hand truncation or wild card searching.
- Nesting of linked boolean terms.
- Relevance ranking.

Each document is scored based on its relevance to a user's question, where the most relevant document has the highest score, or rank -- 1000 being the highest, 1 being the lowest. A document receives a higher score if the words in the question are in the headline, or if the words appear many times, or if phrases occur as in the question. A document's score is derived using techniques such as word weighting, term weighting, proximity relationships, and word density. Questions made up of natural language, relevant documents, and boolean expressions are all weighted using these techniques.

- Weighting

A variety of weighting techniques is used. These include word weighting that takes account of where in a document the search word is found, term weighting that takes account of how frequently a term is used in a database, proximity relationships that weight conjoint terms according to their relative proximity and word density that relates the frequency of term occurrence to the size of the document that contains it.

DEVELOPING A MODEL FOR ACCESS TO WWW DOCUMENTS

Mallery¹³ in reviewing the state of indexing of the Web, has referred to the absence of indexation within the Web infrastructure, and the hermeneutics problem which he characterises as having the following types of interpretation issues:

- Explaining specialised knowledge in ordinary language for citizen access;
- Explaining across belief systems: ideological, religious, cultural;
- Interdisciplinary explanation across scientific disciplines;
- Cross paradigmatic access across incommensurate schools of thought in developing fields of knowledge;
- Transnational access across language and ethnic divides.

Familiar retrieval techniques such as use of boolean retrieval, and of statistical retrieval that permits a ranking of items retrieved, have been applied to WWW robots, and for software that will search WWW pages interactively. These retrieval facilities are also available on WAIS databases, to which access must first be obtained from the Web. More advanced retrieval facilities such as natural language processing and semantic indexing for knowledge databases have yet to be applied. Mallery anticipates the development of software that will use basic retrieval procedures to carry out a detection search or first stage filtering for retrieved items, followed by an extraction phase that may semantically index items locally.

My view is that client software will be developed to provide combined classified and indexed navigation entry to databases and page-based information resources that will enable users to

evolve personalised views of the Internet. This will require a local controlled vocabulary, particular to the discipline of the user, that may be installed and subsequently developed by the user. The vocabulary with its semantic relationships can then be used to parse Web pages for relevant material and assign pointers to these from a local database. This may be provided dynamically within the purview of a broad classification system, which for retrieval purposes may be used on a pull-down menu basis to set the scope of, or limit index term approaches.

The state of image-based retrieval is such that the controlled vocabularies may contain images synonymous with text (or with hierarchies of text). An example has been implemented with the NASA-JSC archives¹⁵. This approach requires the controlled textual description of images. However access by utilisation of icons to represent images and text is still very much in the research area and has not been implemented beyond very limited variations of images.

For those of you who have access to WWW, you will find Indexing the Web¹⁶ is a page that is a useful entry point to keep abreast of indexing principles on the Web.

REFERENCES

1. *The Internet* (1995) <<http://www.dsu.edu/internet/internet.html>>
2. *INTERNET--Introduction--History* (1995)
<<http://www.rpi.edu/Internet/Guides/decemj/icmc/internet-introduction-history.html>>
3. Hahn, H. & Stout, R. (1994) *The Internet complete reference*, Osborne McGraw-Hill, Berkeley, CA.
4. Kehoe, B.P (1995) *Zen and the art of the Internet*, 3rd ed., Prentice-Hall, Englewood Cliffs, NJ. (ISBN 0131214926) (First edition for example at
<http://sundance.cso.uiuc.edu/Publications/Other/Zen/zen-1.0_toc.html>)
5. Ayre, R. & Reichard, K. (1995) The web untangled, *PC Magazine*, Feb 7 (Magazine home page <<http://www.pcmag.ziff.com/~pcmag/pcmag.htm>>; specific item at
<<http://www.pcmag.ziff.com/~pcmag/1403/03webint.htm>>)
6. *Common Questions and Answers about veronica, a title search and retrieval system for use with the Internet Gopher*. (1995, Jan 13th)
<<gopher://futique.scs.unr.edu:70/00/veronica/veronica-faq>>
7. Kelly, B. (1995) 3. WWW Browsers, in: *Running a WWW service*
<<http://www.arnes.si/books/www-handbook/>>
8. NEXOR (1995) *List of robots* <<http://web.nexor.co.uk/mak/doc/robots/active.html>>
9. Hodge, G.M. (1992) *Automated support to indexing*, National Federation of Abstracting

and Information Services, Philadelphia, US.

10. Desai, B.C. (1995) *Cover page aka semantic header*
<<http://www.cs.concordia.ca/bcd/semantic-header.html>>
11. Liebscher, P. (1994) Hypertext and indexing, in Fidel, R. *et al* (Eds.) *Challenges in indexing electronic text and images*, Learned Information for ASIS, Medford, NJ, pp. 103-109.
12. Marchionini, G. (1994) Designing hypertexts: start with an index, in Fidel, R. *et al* (Eds.) *Challenges in indexing electronic text and images*, Learned Information for ASIS, Medford, NJ, pp.77-89.
13. Mallery, J.C. (1994) *Indexing and retrieval for the World Wide Web*, draft 8, Intelligent Information Infrastructure Project, Massachusetts Institute of Technology<<http://www.ai.mit.edu/people/jcma/pending/www-retrieve.html>>
14. *How to search a WAIS database* (199?) <http://town.hall.org/util/wais_help.html>
15. Seloff, G. (1990) 'Automated access to the NASA-JSC image archives', *Library Trends* 38, 4, pp. 682-696.
16. *Indexing the Web* (1994 -)
<<http://union.ncsa.uiuc.edu/HyperNews/get/www/indexing.html>>

APPENDIX 1

Extract from PC Magazine Comparison of Browsing Software⁵

WEB BROWSERS: SUMMARY OF FEATURES				
	AIR Mosaic 1.1	Cello 1.01a	Enhanced NCSA	InterAp 26
			Mosaic for	
			Windows 1.02	
List price	\$29.95 (also	Free	Varies by vendor	\$295.00
	included in other			
	SPRY products)			
General Features				
Allows multiple simultaneous connections	YES	YES	NO	NO
Requires third-party SLIP/PPP provider	YES	YES	YES	YES
Winsock 1.1-compatible	YES	YES	YES	YES
Configuration				
Editable .INI file	YES	YES	YES	YES
User can turn off graphics	YES	YES	YES	YES
Navigation and Storage				
Can save documents to disk	YES	YES	YES	YES
Can print text and graphics	YES	YES	YES	YES
Displays progress report while loading pages	YES	YES	YES	YES
Offers interactive hotlist	YES	YES	YES	YES
Can add current URL to hotlist	YES	YES	YES	YES
History and Customization				
History log	YES	YES	YES	YES
Unlimited page cache	NO	YES	NO	NO
Customizable page cache	YES	YES	YES (via .INI file)	YES
User can create and edit stylesheets	NO	NO	YES (1)	YES
Macro support	NO	NO	NO	NO
OLE 2.0 support	NO	NO	NO	YES
Forms support	YES	NO	YES	YES
Tools				
archie	NO	YES	NO	NO
FTP	YES	YES	YES	YES
Gopher	YES	YES	YES	YES
Jughead	NO	YES	NO	NO
Veronica	NO	NO	NO	YES
WAIS	NO	YES	NO	NO
Miscellaneous				
Usenet newsgroups	YES	YES	YES	Optional
E-mail	YES	YES	NO	Optional
MIME support	YES	NO	N/A (2)	Optional
Search capabilities	YES	YES	YES	YES
Local on-line help	YES	YES	YES	YES

(EXTRACT CUT OFF AT THIS POINT)				

APPENDIX 2

Modified extract from *List of Robots: A Public Service provided by NEXOR*⁸

The fish Search

- Run by people using the version of Mosaic modified by Paul De Bra <debra@win.tue.nl>
- It is a spider built into Mosaic. There is some documentation online.
- Identification: Modifies the HTTP User-agent field. (Awaiting details)

Harvest

- Run by hardy@bruno.cs.colorado.edu
- A Resource Discovery Robot, part of the Harvest Project; runs from bruno.cs.colorado.edu, sets User-agent and From fields.
- Pauses 1 second between requests (by default).

Note that Harvest's motivation is to index community- or topic-specific collections, rather than to locate and index all HTML objects that can be found. Also, Harvest allows users to control the enumeration several ways, including stop lists and depth and count limits. Therefore, Harvest provides a much more controlled way of indexing the Web than is typical of robots.

Indexer

- A script that sucks html URLs out of the database and feeds them to a modified freeWAIS waisindex, which retrieves the document and indexes it. Retrieval support is provided by a front page and a cgi script driving a modified freeWAIS waissearch.
- The separation of concerns is to allow spider to be a lightweight assessor of Web state, while still providing the value added to the general community of the URL search facility.
- Identification: it runs from rbse.jsc.nasa.gov (192.88.42.10), requests GET /path RBSE-Spider/0.1", with a and uses a RBSE-Spider/0.1a in the User-Agent field; Seems to retrieve documents more than once.

InfoSeek Robot 1.0

- By Steve Kirsch <stk@infoseek.com>
- Its purpose is to collect information to use in a "WWW Pages" database in InfoSeek's information retrieval service (for more information on InfoSeek, please send a blank e-mail to info@infoseek.com); The Robot follows all the guidelines listed in "Guidelines for Robot Writers" and we try to run it on off hours.
- Will be updating the WWW database daily with new pages and re-load from scratch no more frequently than once per month (probably even longer). Most sites won't get more than 20 requests a month from us since there are only about 100,000 pages in the database.

The JumpStation Robot

- Run by Jonathon Fletcher <J.Fletcher@stirling.ac.uk>.
- Version I has been in development since September 1993, and has been running on several occasions, the last run was between February the 8th and February the 21st.
- More information, including access to a searchable database with titles can be found on The Jumpstation
- Identification: It runs from pentland.stir.ac.uk, has "JumpStation" in the User-agent field, and sets the From field.
- Version II is under development.

Lycos

- Owned by Dr. Michael L. Mauldin <fuzzy@cmu.edu> at Carnegie Mellon University.
- This is a research program in providing information retrieval and discovery in the WWW, using a finite memory model of the web to

guide intelligent, directed searches for specific information needs.

- You can search the Lycos database of WWW documents, which currently has information about 390,000 documents in 87 megabytes of summaries and pointers.
- More information is available on its home page.
- Identification: User-agent "Lycos/x.x", run from fuzine.mt.cs.cmu.edu. Lycos also complies with the latest robot exclusion standard.

The NorthStar Robot

- Run by Fred Barrie <barrie@unr.edu> and Billy Barron.
- More information including a search interface is available on the NorthStar Database. Recent runs (26 April) will concentrate on textual analysis of the Web versus GopherSpace (from the Veronica data) as well as indexing.
- Run from frognot.utdallas.edu, possibly other sites in utdallas.edu, and from cnidir.org. Now uses HTTP From fields, and sets User-agent to NorthStar

The Peregrinator

- Run by Jim Richardson <jimr@maths.su.oz.au>.
- This robot, in Perl V4, commenced operation in August 1994 and is being used to generate an index called MathSearch of documents on Web sites connected with mathematics and statistics. It ignores off-site links, so does not stray from a list of servers specified initially.
- Identification: The current version sets User-Agent to Peregrinator-Mathematics/0.7. It also sets the From field. The robot follows the exclusion standard, and accesses any given server no more often than once every several minutes.

The WebCrawler

- Run by Brian Pinkerton <bp@biotech.washington.edu>
- Identification: It runs from webcrawler.cs.washington.edu, and uses WebCrawler/0.00000001 in the HTTP User-agent field.
- It does a breadth-first walk, and indexes content as well as URLs etc. For more information see description.

W4 (the World Wide Web Wanderer)

- Run by Matthew Gray <mkgray@mit.edu>
- Run initially in June 1993, its aim is to measure the growth in the web. See details and the list of servers
- User-agent: WWWWanderer v3.0 by Matthew Gray <mkgray@mit.edu>

WWW - the WORLD WIDE WEB WORM

- Maintained by Oliver McBryan <mcbryan@piper.cs.colorado.edu>
- Another indexing robot, for which more information is available. Actually has quite flexible search options.
- Awaiting identification information (run from piper.cs.colorado.edu?).