



Ernest Perez, Ph.D.

HTML Indexer

Aims to Index the Web

Search engines simply leave the user in minefields of free text keywords and

▲
NP

Librarians have been known to mutter, "All the Web needs is good indexing." But, honestly, that has not been a realistic option. Although we have endless Web search engines at our disposal, they generally use free text searching, along with a variety of ranking methods for search results.

This search engine software technology approach is useful for some purposes, and quite lacking for others. It generally ignores the strong suits of classical indexing—conceptual analysis of content, controlled vocabulary to reduce information fragmentation, and cross-referencing to guide the user in navigating the controlled vocabulary. Search engines simply leave the user alone, to traverse the minefields of free text keywords and natural language style.

Never fear, I think a partial solution is here. Brown, Inc.'s HTML Indexer

software package (www.html-indexer.com) streamlines the task of *really* indexing Web sites. It also provides the opportunity to use standard indexing approaches and controls. This package can produce high-quality, back-of-the-book-style alphabetical indexes for either hosted (at your own location) or remote Web resources.

APPETITE FOR WEB SITE INDEXES

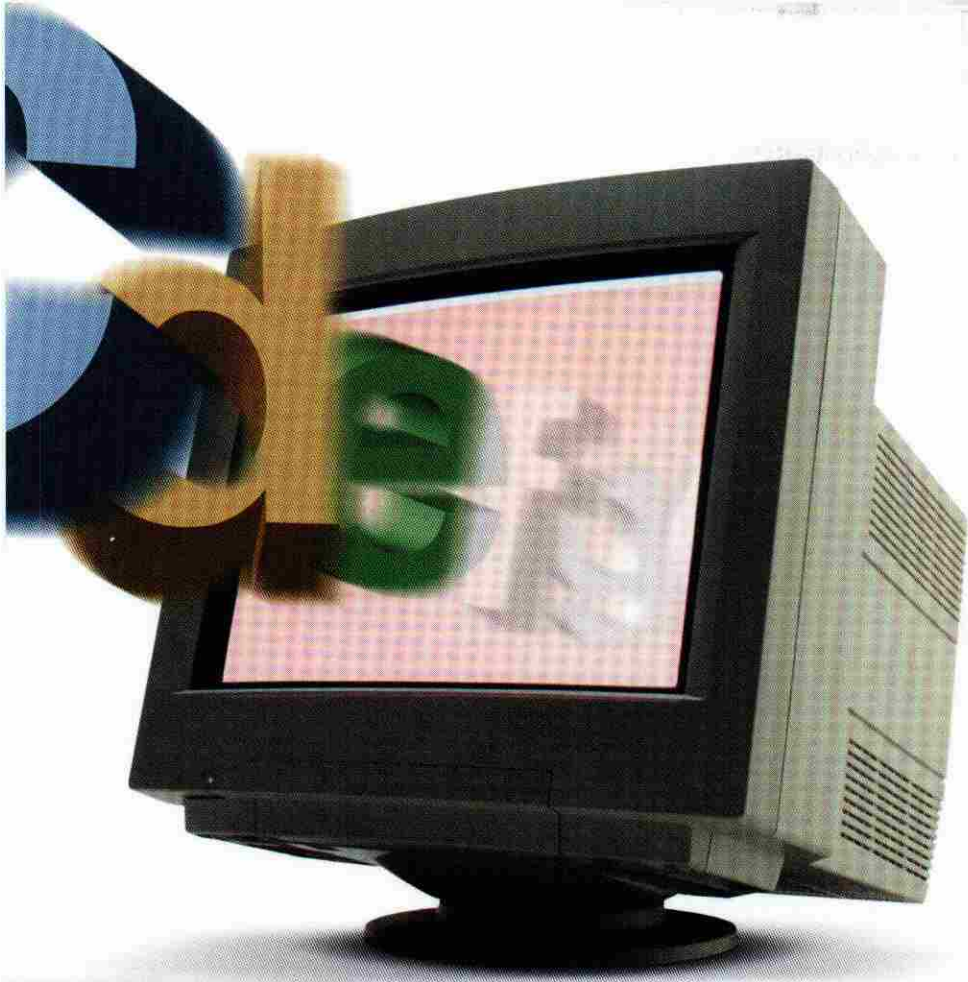
My own definition for distinguishing indexing from cataloging is that indexing points to specific item locations within specific documents or information records; whereas, cataloging results in an inventory listing of a collection of information records, with some added-value access and retrieval abilities. To wit, an index is where you're getting *specific* about finding stuff!

Right off the bat, I was favorably impressed with HTML Indexer. It's

fast, easy to learn and use, and can put out high-quality indexes. Be warned, though, the index quality is dependent upon the skill and indexing acumen of the indexer.

My baby-steps HTML Indexer learning projects involved sample indexes produced for simple but useful domains. One of these was a small index of Web resources targeting Oregon state agency employees. It covered Web resources in the special area of human resources, job benefits, job classifications, position descriptions, and union contracts.

I did want to honestly temper my uncontrollable librarian biases, to avoid seeing a librarian-type solution for which there was no problem. So I tested my (justifiably) prideful evaluation of my demonstration index. I showed it around to several non-library colleagues in Oregon state government. Happily, I received unanimous



alone, to traverse the natural language style.

positive responses and predictions that such a Web index would indeed be helpful to state employees.

HTML INDEXER KISS APPROACH

Let me stress that HTML Indexer is not truly an "indexing program" in the traditional sense of the word. It doesn't take care of vocabulary or thesaurus control, punctuation, capitalization, italicizing, or any of that fancy stuff. The software is a collection of useful index-building tools, sticking to the classic KISS (Keep It Simple, Stupid) approach. It leaves all the index management niceties for you, the operator, to handle, manage, and edit while you're planning and preparing the index. The program simply gives you some powerful manipulation tools to make preparing the index quite a bit easier.

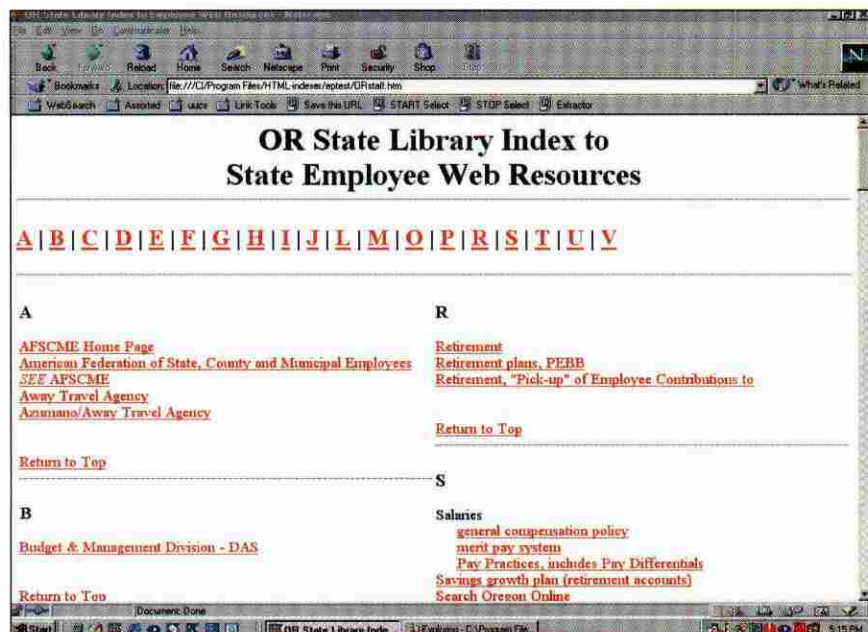
HTML Indexer provides tools for fast, easy import of basic Web site

hierarchy information. Incidentally, the program uses *relative paths* in the indexing, so you don't actually have to do your indexing work on the actual Web server host machine. Using Network Neighborhood or WebWhacker or some similar utility, simply download a full copy of the target HTML directories and source files. You can then build and fine-tune your index on your own machine, and then upload the finished index to the Web site host machine.

The program quickly scans the hierarchical directory and filename structure, reads the HTML Title information and/or Headline text for each file, and assigns one of those word strings (giving priority to Title) as the default index entry text for that file. The site structure data is automatically recorded in an HTML Indexer work file (Windows file extension .IPJ). These .IPJ files are an enduring "embedded index" map or working file of the target site. As a site changes, this work file automatically tracks changing HTML structure, file locations, HTML directory changes, and the presence of new files. Index entry links are thus continuously translated to the current location of an HTML file.

MODIFYING DEFAULT SETTINGS

The default descriptive text entries offer a quick reference to the content



A snapshot of the Oregon State Library Index to State Employee Web Resources, created using HTML Indexer.



or topic of each HTML file. You can easily modify default entry texts to fit your indexing practice, headings, and capitalization style. HTML Indexer also offers easy methods for creating added entries, nested subtopics, and “forced alphabetization,” (e.g., file under “such & such,” as if spelled “Mac,” “as if spelled out.”)

The index entry manipulation commands are simple and obvious, but you need to be aware that the program does not perform the traditional, indexing application behind-the-scenes work—tasks such as vocabulary checking, automatic editorial formatting of entries, and sub-classification cross-referencing relationships. The tool is rather an efficient WYSIWYG index editor. However, you must personally know and conscientiously apply good indexing and display practices.

As you add and work with entries within the program workspace, you are simply manipulating the individual entry item titles sitting atop a pointer to the item URL location. Thus, you really are adding or copying these pointers, and are then able to massage the content of the descriptive index term entries. Easy-function key shortcuts and Windows Copy & Paste operations greatly facilitate index entry text.

File information or URLs and entry text is displayed in a Windows Explorer-like presentation. You can quickly toggle back and forth between “title” or entry listings, hierarchical filename displays. It’s easy to sort and rearrange your working view by the different columns, such as entry status, entry text, and target URL.

As you develop and correct your index masterpiece, easy function keys and menu commands enable you to quickly produce and review final

HTML format displays, using automatic browser displays.

CROSS-REFERENCING WEAKNESS

NP The single negative judgment I had was the omission of an easy function or operation for inserting and linking of cross references. The vendor explains a manual method for doing this, in the documentation, and it’s effective, albeit clunky.

To handle this, you must first create an HTML “anchor” or named link at the location of the preferred index term. You can then create SEE or SEE ALSO entries containing links which jump to the anchor link at the referenced location in the finished index HTML file. The final Web display will show highlighted links at the cross-reference entry locations, such as Insurance-Health SEE Health Insurance. Clicking on this reference link in this example results in an instant hyperlink jump to the “Health Insurance” preferred term and its entries.

This approach works, but it involves several complicated keyboard operations, and copying and pasting of URL text—and it isn’t really all that easy. (Hey, just a minute, here, we’re back to doing indexing work in HTML, which is what the program is supposed to replace!) The program needs easy, automatic functionality to handle this important and very frequent indexing operation. HTML Indexer publisher David Brown says this improvement is high on the customer request list, and will definitely be included in the next software update, at the beginning of 2002.

CUSTOMIZING OUTPUT

There are extensive built-in controls to define the basic index display

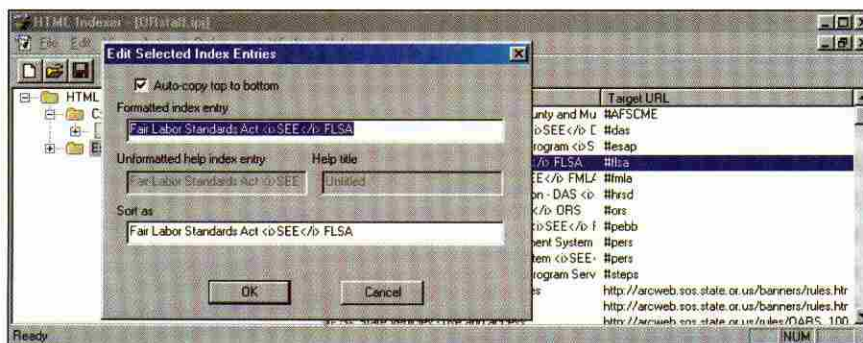
presentation format. You can also combine HTML Indexer output into customized Web page templates, or even use it along with Cascading Style Sheet coding. So you do have flexible customization and site-compatibility editing power for your site indexes.

Built-in custom definition choices include simple menu selection of such formats options as: Font and size selections; nested or run-on sub-classifications; word-wrap for longer lines; single- or multiple-column Web index listings; insertion of top and/or bottom alphabet lists for “jump to <selected letter> use; and display of graphics for fancy illuminated or “rubric” initial letter displays. Incidentally, HTML Indexer also produces HTML Help or JavaHelp format indexes as optional output formats.

REMOTE SITE INDEXING

You must work a little harder if you are indexing “remote sites” (files on Web sites other than your own). In this case, you must actually browse the remote site, and use a little extra manipulation to grab the data and transfer it into the index program. Since the HTML files are not actually on your machine, the program can’t automatically scan the files to import filenames/URLs, title and headline texts. You’ll need to run HTML Indexer simultaneously with your browser window, and use your Copy & Paste function to import URL and text information into the Indexer. Again, since the referenced files are not on your machine, the program can’t automatically check for file changes or location moves. So you will need to use a link checker to check the current status of the referenced URLs in your index.

Manual entry processing of remote files is pretty quick, although nowhere near as slick as the host-site indexing. It’s easy to create an entry and transfer the information. What kind of work-session time are we talking about? Well, my “State Employee Information” sample index took me about an hour and a half. Admittedly, this was a small, specialized index to 57 remote pages, and nowhere near the size and complexity of some of the Web indexes referenced in this article. Remember, though, this was part of my learning experience, and the time included much experimentation with



HTML Indexer offers easy methods for creating “forced alphabetization,” and modifying other default settings.



multiple-entry creation, cross-references, and different formats such as column presentation choices, type of index sub-heading display, and "return to top" links.

HTML INDEX MAINTENANCE

As you well expect, the maintenance of your Web index is where the indexing rubber is really going to meet the road! HTML Indexer does indeed speed up initial index creation, and allow for easy format fine-tuning and easy manipulation of entries.

But, with a short learning curve, you can produce a decent HTML index using practically any standard HTML editor, or even using a Notepad-style editor. But using this manual editing

approach, you can expect to pay and pay and pay for index maintenance in the long run, with endless amounts of your personal labor.

Updating ease is where HTML Indexer really excels. When you decide to update, all you need do is get back into Indexer/browser working mode, reload the database-like .IPJ source file, and get to work on your update. If you're working on a local site or site copy on your own machine, you will have the benefit of a built-in file checker to highlight and identify missing, moved, or new files.

When you're indexing remote files, like all Web information users, you'll need to do a little old-fashioned detective work to track down changes in the source site file locations. So you will want to use a standalone link checker to quickly identify any problems with the remote files. But HTML Indexer greatly simplifies the addition and modification of URL entries and indexing text. You basically just need to work on the changed files, and not regenerate the index from scratch. Best of all, the program replaces the laborious manual HTML file production steps with a simple function key press.

USER EXPERIENCE

I found several quality examples of user sites on the HTML Indexer Web site. All the customers I contacted were quite happy with the product.

Karen Lane, Co-Webmaster responsible for the index at the American Society of Indexers Web site (www.asindexing.org/backndx.htm), rated HTML Indexer as an excellent indexing tool. She commented on several minor formatting issues, but expects them to be resolved in program updates. Her greatest praise is for the considerable time-saving. "Once the project is organized, the generation and regeneration of the index is easy and fast, so that saves time. We had a hand-maintained site index before we used HTML Indexer, and it was very easy to have out-of-date items remain in the index unnoticed. That can't happen with this program. Adding new entries is very easy too. On the whole, once you learn how to use it and what it can do, it does save time."

Kathryn Varjabedian, index specialist at the Los Alamos National

Laboratory Research Library (<http://lib-www.lanl.gov/libinfo/news/newsindx.htm>), was equally positive about the program. "HTML Indexer is easy to use and saves a lot of time compared to manually adding entries in a straight HTML page. For each new monthly issue of the library newsletter, I can add it to the index in about 20 minutes. I go through the articles, edit/add one or several entries for each article, and click a button to re-generate the entire index...The ease of adding material, editing and re-generating a new index are quite valuable. Support from the company is good."

Bob Huerster, who used the program in the past for Consumers Union, also has a highly favorable personal view of HTML Indexer. (This is not an official CU endorsement.) Huerster reports he was quite happy with the program, and that it's relatively easy to learn. He says, "The great thing about it is the way the embedded indexing 'moves' as the page 'moves.'" He echoes the other users in saying that the vendor gives great support. However, CU is moving to a purely dynamic page site, which he will not be able to index going forward.

WEB INDEXING OPPORTUNITIES

Common wisdom may be right—the Web offers a new venue for indexers and librarians. This product provides a good toolkit for delivering on some of that "Web-indexing" potential. It certainly gives us an economic and realistic shot for effective indexing of networked resources, both in employed or freelance project modes.

In summary, HTML Indexer is an effective tool for Web indexing projects. It is best used for direct indexing of Web file resources resident on the machine used for indexing. However, it is also efficient for indexing Web resources on remote sites. It is not really a specialized indexing program. Instead, it gives you automated efficiency in working with Web index production. The product is an easy-to-use and powerful index production tool.

Ernest Perez (ernest.r.perez@state.or.us) is group leader for Information Services and Technical Services at the Oregon State Library

Comments? Email letters to the editor to marydee@xmission.com.

Product at a Glance

HTML-Indexer

Brown Inc.





503/292-1710

dmbrown@brown-inc.com

www.html-indexer.com

Specifications:

HTML Indexer, v3, is a 32-bit Windows application. It will thus run on MS Windows 95 and higher. The program takes up 1.75MB of hard disk space. It costs  in download version  for the CD-ROM version. There is a free demo version of HTML Indexer available at the Web site. The demonstration version creates classic index files, HTML Help index files, and JavaHelp index and mapID files, but it does not save index entries or project settings between sessions.

Documentation, ease of use:

No big learning curve here. The online Help contains an effective tutorial exercise. The program interface is clear and simple, and the Windows Help files are comprehensive. There's a good Help topical outline display, and the Help index is superb. One must really expect that with this product! There are helpful FAQ, Product Details, and Tips & Tricks pages at the product Web site. Users all reported good-quality customer support.