A book-style index for the web: the University of Texas Policies and Procedures Website

Linda Fetters

Describes the process of creating a web index for an online index of printed documents. Includes practical examples of HTML linking, and the use of named software.

In 1997 I had the opportunity to create my first Web index, the University of Texas at Austin (UT) Policies and Procedures Web Site. The Web index is an outgrowth of the online version of the policies and procedures (TxPoly), a project that started in the early 1980s.

1980s online project

The Associate Vice President and Business Manager decided that it would be good to have the policies and procedures available on the mainframe computer system where all students and faculty members could access it. Only a limited number of the policy manuals were printed and distributed to campus offices.

The online index would include the *Handbook of Operating Procedures*, the *Policy Memoranda*, the *Regents Rules and Regulations* (two parts), and the undergraduate catalog. Later the undergraduate catalog was dropped and the *Handbook of Fiscal Procedures* was added. I provided individual indexes for each manual as well as a cumulated index, all created on my first computer, a Radio Shack Model II.

The data processing staff designed the database application on the mainframe. They input each paragraph as a database record to which table of contents and index records would be linked. We were not aware at the time that we were creating a hypertext index.

To access a policy, users could browse the index by scrolling the screen, or they could jump to any part of the index by typing the first few letters of the entry at the prompt. They could access the related text file by entering the line number for the index entry. Line numbers appeared on the screen to the left of each entry. Later, a separate key word index file was added.

Maintaining the original index

After the project was complete, it was decided that I would maintain the text files as well as the index files. For the first year, I had to go to the campus to access the mainframe. I learned how the records were stored and how the text editor worked, but it was inconvenient for me to go to the University for several days in a row whenever any changes had to be made. I asked for modem access, which was unavailable at the time. Several months later, the UT administration decided to allow outside access by modem, and I could make updates from my home office.

In 1989 I left Austin and moved to Port Aransas, a small resort town located on the Gulf Coast near Corpus Christi. TxPoly remained in limbo for several months until UT administration provided a toll free method for updating the files. A few years later, when I was able to get local Internet access, I could make the updates without the University having to pay the long distance charges.

Hatching the web version

Sometime around 1993 or 1994, I heard that the Business Manager's Office was considering a Web version of TxPoly. They planned to convert the database records to text files with no index: only a search engine would be available to users. I expected to hear at any time that my services were no longer needed.

In late 1996, our local chapter of the American Society of Indexers met at the UT Health Science Center at San Antonio. Part of the program was a demonstration of building Web pages by Janna Lawrence. Her presentation convinced me that building a Web page was no more difficult than inserting typesetting codes in a document. In the meantime, I moved to Corpus Christi and signed up with a local Internet service provider and started to learn more about searching the Web. Before that, I was essentially Internet- illiterate. I had an e-mail account with AOL (America Online), but since there was no local access number, it was just too expensive to do any Web surfing.

A few months later I contacted the UT Business Manager's Office to see if they had already built their Web page. They asked me to submit a proposal for converting the database text files to HTML, to re-index the databases, and to make links between the index entries and the text files.

As part of my proposal, I created a sample Web index based on one chapter of the *Regents Rules and Regulations*, which had been mounted on a Web server (without an index) by the UT System Office. To do this, I bought Laura Lemay's *Teach yourself web publishing with HTML 3.0 in a week* and went to work. I spent one day reading through the book and then downloaded a freeware copy of **HTML Assistant** (and later purchased a copy of **HTML Assistant Pro 97**). Thus armed, I went to work with the demonstration version, a process that took about a week.

In early 1997 I took the demo index to Austin. After a second meeting a few months later, the project was approved. Only the *Handbook of Operating Procedures* and the *Policy Memoranda* would be included.

Designing and planning the pages

The TxPoly index is quite a bit different from most Web indexes. Instead of being an index of other Web sites, it is a book-style index with links to a book that has been converted to electronic format.

Even though TxPoly is a very simple Web site, there were still many design decisions to be made: what size the files should be, what the "home page" should look like, how to move from one file to another, and how to format the text and index. Before beginning to code the files, I spent a lot of time looking at the underpinnings of the ASI Web page (www.asindexing.org), especially its index, as well as the Regents Rules Web site. (I discovered while writing this article that the Regents Rule site has been completely revamped, and there is no longer access to individual sections of each chapter, completely invalidating all the links in the TxPoly files). Many examples of Web indexes can be found at ASI's Web site (www.asindexing.org) and the AusSI Web page (www.zeta.org.au/~aussi).

File size

The first decision was the optimum file size. I decided that chapter-sized files would be too big and take too long to load. Each file would contain only one section of a chapter and its subsections. That resulted in many small files and eliminated the ability to scroll through one whole chapter as you would read through a book. I compensated for that by providing links between each section to the previous section and next section. I also provided links in each file to get back to the index or the table of contents.

File names and directories

All coding and designing can take place on your own computer. After coding, you can view the files in your browser and test the links. The only links you can't test while offline are links to other Web sites.

Fortunately you can keep all the files in one folder (directory) without worrying about the name of the directory on the Web server where they will eventually reside. I named all my files *.htm only to be told when I was ready to FTP them to the Web site that they should be named *.html because they were being placed on a UNIX server. It wasn't hard to change the file names, but it meant that all the internal links in all the files had to be changed as well. Fortunately, HTML Assistant Pro is able to perform massive global changes easily, but the more information you have before you begin, the better. In this case, I didn't know where the files were going to be stored until I was almost finished with the project (www.utexas.edu/policies/hoppm/).

Formatting the text

The text files from the TxPoly database were provided by the University. I had to clean them up by removing the hard carriage returns and extra spaces that padded each line.

Formatting the text was simple. I decided what header size to

use for the chapter titles and section names and stored the HTML codes in a "template" file that I copied into each new text file as I started the formatting.

I made another template file for the links at the end of each file:

> Go to Next Section
> Go Back to Previous Section
> Return to Table of Contents
> Return to Combined Index

All I had to change were the links to the next and previous sections.

Another decision was what type of codes to use for the text files and the index entries. To simplify the process, I used the same type of codes for both: the definition/ glossary codes.

Here is a sample file:

```
<HTML>

<HEAD><TITLE>Handbook of Operating Procedures,
Chapter 4</TITLE>

</HEAD>

<BODY><P>

<H3><B><CENTER>CHAPTER 4<P>

STUDENTS</B></CENTER></H3><P><P><P><P>

<H4><A NAME="h0406"></A>Sec. 4.06 Intercollegiate
Athletics Council for Men</H4>

<DL>

<DD>The Intercollegiate Athletics Council for Men
shall advise the President, through the Vice President
for Administration, on all matters of policy, personnel,
and programs for intercollegiate athletics for men. The
President is responsible for the conduct of
intercollegiate athletics for men at The University of
Texas at Austin, and all recommendations of the
Council are advisory to the President.<P><P>

<DD>Faculty members of the Council shall review
annually the academic performance and progress of
student athletes and shall have access to academic
records and to academic counselors or other
athletics department personnel as necessary and
appropriate to carry out these faculty members'
general academic advisory responsibilities. Faculty on
the Council shall report annually to the Faculty Senate
on academic and other matters of relevance to the
General Faculty.<P><P>

<DD>As a matter of established University policy,
athletics department personnel are prohibited from
violating any rule or regulation of an organization
having regulatory authority over the intercollegiate
athletics activities of the institution with respect to
recruitment of student athletes. Violation of this policy
will result in disciplinary action up to and including
termination. As a budgeted auxiliary enterprise, no
general revenue funds are, or may be, used for
intercollegiate athletics activities purposes.<P><P>

<DD>The Intercollegiate Athletics Council for Men
shall be composed of nine voting members as follows:
a student of The University, an ex-student of The
University, two Regental appointees, and five
members of the General Faculty of The University.
Deans and other academic administrators directly
responsible for the certification of student athletes'
```

academic progress shall not be eligible to serve on the Council.<P>

<DL>

<DD>1. The student member of the Council shall be appointed by the chief administrative officer of The University to a one-year term which shall begin at the start of the Fall Term.<P><P>

<DD>2. The ex-student member of the Council shall be appointed by the Ex-Students' Association of The University to a one-year term which shall begin at the start of the Fall Term.<P><P>

<DD>3. The two Regental appointees shall be appointed by the Board of Regents to four-year staggered terms which shall begin at the start of the appropriate Fall Terms. One such appointment shall be made in each even-numbered year. A Regental appointee may not be reappointed to a succeeding term on the Council.<P><P>

<DD>4. The chief administrative officer of The University shall appoint a member of the General Faculty as a member and chairman of the Council for an indefinite term with the appointee to serve as a member and as the chairman of the Council at the pleasure of the chief administrative officer. <P><P>

<DD>5. The chief administrative officer at The University shall appoint four members of the General Faculty to four-year staggered terms on the Council, with one such faculty member appointed each year. Each four-year term shall begin at the start of the appropriate Fall Term.<P><P>

<DD>6. Each member of the Council shall continue to serve until his or her successor has been appointed, and each member may be reappointed to one or more additional terms, except for the student members as noted above. Regental members of the Council may not succeed themselves.<P>

</DL>

<DD>All recommendations of the Council shall be transmitted to the President by the prompt delivery of two copies of the minutes of all meetings. Upon review of the minutes of all meetings of the Council, the President may accept and approve, decline or modify any recommendation therein. No recommendation advanced by the Council shall have any force or effect until said recommendation has been accepted and approved by the President, and has received such other approval as may be required by the <I>Rules and Regulations of the Board of Regents</I>. Two copies of the minutes of the Council together with the action and decisions made by the President upon the recommendations of the Council shall be delivered to The University of Texas System office.<P><P>

<DD>Nothing in Sec. 4.06 shall be deemed to limit or restrict the authority of the Board of Regents.

</DL></DL>

<P><P><P><P>

<CENTER>Go to Next Section</CENTER><P>

<CENTER>Go Back to Previous

Section</CENTER><P>

<CENTER>Return to Table of Contents</CENTER><P>

<CENTER>Return to Combined Index</CENTER><P><P><P>

</BODY>

</HTML>

Creating the index

The index can be produced with your usual indexing program. I converted the existing index files from the old TxPoly database to **Cindex** files. I decided to use the original locators as visual cues to the section numbers (See example). I used the same section numbers for the links and anchors, as described in the next section.

Any of the popular indexing programs (**Cindex, Macrex,** or **Sky Index Professional**) can insert the HTML codes for the index main headings and subheadings using their code table capabilities. You still have to insert the coding for individual links and anchors for the locators. You can avoid retyping these codes, except for the actual anchor/link name, by storing the codes in the user-defined function keys, macros, keywords, abbreviations, or acronyms available in these programs.

Creating Links

By far the most difficult part of the project was inserting the links between the index entries and the paragraphs to which they pointed. Fortunately, this project had a built-in structure that was established many years ago. I was able to use the section names for anchors. No matter what names you use, the links have to be inserted in the index entries and the anchors have to be inserted into the matching text file.

The following example shows the codes used for indenting the index entries and the links for locators and cross-references. I've indented the subheadings (coded <DD>) to make the text easier to read. The <A NAME=> code indicates an anchor, and the <A HREF=> code indicates a link. When a user clicks on a link, the Hypertext Transfer Protocol (http) jumps to the matching anchor.

Locator Key

<p>

HOP = <i>Handbook of Operating Procedures</i>

PM =<i> Policy Memoranda</i>

<P>

<DT><A NAME="d"></A><B>D</B>

<P>

<DT><A NAME="daily"><i>Daily Texan</i>

<DD>authority of Board of Operating Trustees over students and employees, <A HREF="h0409. html#h040919">HOP IV (4.09) (19) </A>

<DD>editorial manager, <A HREF="h0409.html #h040910b">HOP IV (4.09) (10) (b)</A>

<DD>election of editor, <A HREF="h0409.html# h04097">HOP IV (4.09) (7)</A>

```
<DD>relationship with Journalism Department, <A
   HREF="h0409.html#h040910b">HOP IV (4.09) (10)
   (b)</A>

<DD>relationship with Journalism Department, <A
   HREF="h0409.html#h040912">HOP IV (4.09)
   (12)</A>

<DT>Dangerous materials. <I>See</I> <A HREF=
   "combh.html#hazchem">Hazardous
   chemicals, fire code for</A>; <A
   HREF="combh.html #hazmatdis">Hazardous
   materials disposal</A>

<DT>Dean's Fellows, <A href="h0310.html#h0310b2">
   HOP III (3.10) (II) (B)</a>

<DT>Deans. <I>See also</I> <A HREF="combt.html#
   vpdean">Vice president and dean of graduate
   studies</A>

<DD>established schools and colleges, consultative
   committee for selecting, <A HREF="h024.
   html#h0243">HOP II (4) (3)</A>

<DD>evaluation of, <A HREF="h025.html#h025">HOP II
   (5)</A>

<DD>new schools and colleges, consultative
   committee for selecting, <A  HREF="h024
   .html#h0244">HOP II (4) (4)</A>
```

You can look at these files in detail at the website: www.utexas.edu/policies/hoppm/.

Debugging formatting problems

I discovered that the files looked different in HTML Pro 97's internal viewer than they did in Netscape and consequently spent a lot of time reformatting the files. After that, I used the Netscape browser for viewing the files and for printing formatted versions of all the files to check for formatting problems.

Checking links

This is a time-consuming process in this type of Web page where you have hundreds (maybe thousands!) of links. I believe there are programs that will check links for you, but I checked mine all by hand. Twice. After I moved the files to the UT server, I checked every link again since I had to rename all the files from *.htm to *.html.

Transferring files

My ISP had a file transfer program (FTP) available for downloading, which was easy to use. Once the addresses and passwords required to connect with the Web server are in place in the FTP program, it's just a matter of pointing at the file(s) you want to transfer and clicking on the arrow that transfers the files from your hard disk to the server.

Helpful software

HTML Assistant Pro 97

Although there are many Web design programs available, all I wanted and needed was a program that made it easy to put codes into the file. Since I wanted to name each anchor and link individually, I didn't want to use a program that hides the code

from the user.

HTML Assistant Pro makes most the codes available on button bars or drop-down menus. You point and click where you want a code to go, or you can select a chunk of text and have the proper starting and ending codes inserted. Any time you want to see the fully formatted file, you can bring up an internal viewer, or call up your own browser. Pro 97 also allows you to open a large number of files at the same time. You can also make global replacements through multiple files, one after another. Contact Brooklyn North Software Works (www.brooknorth.com) for more information.

Cindex/ Macrex/ Sky Index Professional

As noted above, any of these programs can insert the coding for the index main headings and subheadings using their code table capabilities. You can use function keys or macros, keywords, abbreviations, or acronyms to insert anchor/link coding.

Third-party software

There are programs that can insert HTML codes in your uncoded index files. One of these is **HTML Prep** from Leverage Technologies (www.LevTechInc.com). Another program is **Webix** available from Dwight Walker, who also has training courses for Web indexing (www.wwwalker.com.au)

Linda Fetters is a medical book indexer and owner of Fetters InfoManagement Co, email: fetters@caller.infi.net.