Building Digital Libraries with Greenstone

COLLECTION MAKER

The Greenstone suite helps you build your own digital library.

BY CHI-YU HUANG

Posting documents on the web is easy, but librarians, archivists, and other experts need more sophisticated systems for organizing information within digital collections. One option is to build a network of static outlines and indexes, but that alternative is often too inflexible, requiring high overhead and constant updates. Other digital collections use homegrown scripts and other custom tools, but these tools also require high overhead and continual maintenance.

An efficient alternative for open source users who want to build fast and flexible digital collections is Greenstone. Greenstone is a suite of tools you can use to build your own digital library. The Greenstone suite not only indexes your documents but also provides an interface for defining and organizing metadata. Greenstone gives collection managers a headstart in the task of creating a smart and highly structured digital library.

In this article, I'll introduce the Greenstone digital collection suite and describe how to install and configure Greenstone. I'll also show you how to build a Greenstone library collection using the Linux Magazine archive DVD from Issue 52.

What is Greenstone

Greenstone is open-source digital library software from the University of Waikato in New Zealand [1]. The Greenstone suite provides a new way of organizing, preserving, and publishing information on the Internet or on CD/DVD. No specialist software is required for accessing a Greenstone document collection – any regular web browser will do.

A Greenstone library can handle many different document formats, including HTML, postscript, PDF, and Word. Greenstone is not limited to "text" documents; it can also handle images, audio and video.

Greenstone provides full-text indexing, enabling users to search within text documents or based on metadata such as title and author. Greenstone is also highly configurable, permitting the user to design the look and feel of the collection as well as the web interface.

Greenstone supports a variety of languages, including Arabic, Chinese, English, French, Maori, and Spanish, among others. You'll find examples of existing Greenstone-based digital libraries at [2] and [3].

Getting Ready

Greenstone requires a web server; Apache is recommended. I will assume you already have an Apache web server installed and will focus on how to configure that Apache server for Greenstone. Apache is available by default for most modern Linux distributions, or you can download it from [5].

Greenstone also requires Perl. To check if Perl is already on your system, open a terminal window, type *perl -v*, and see if a message appears specifying the version number. Again, most modern Linux distributions come with a version of Perl.

Greenstone runs on many other operating systems in addition to Linux, including Solaris, Mac OS/X, and Windows. In fact, Greenstone will work on most Unix variants. To compile the

Greenstone source code on Unix, you need the GNU C++ compiler (GCC) and the GNU database manager (GDBM).

Installation

To install Greenstone, first extract the tar file:

```
$ tar xvzf gsdl-2.62-unix.tar.gz
```

The latest Greenstone has a software installer that provides a step-by-step procedure. Run the installer:

```
$ cd gsdl-2.62-unix
$ ./setupLinux.bin
```

By default, Greenstone is installed in the directory */usr/local/gsdl* and requires root access. I set up my installation as a normal user in my home directory. There are three different installation options:

- Web Library
- Source Code
- Custom

If you select "Source Code," the installer will copy all the necessary files into the directories. To compile the source code, you need to:

```
$ ./configure
$ make & make install
```

The compile may take from ten minutes to an hour, depending on your processor. If you are running Linux on an Intel x86 PC and you are using Greenstone for the first time, I recommend you select the *Web Library* option, which will install the binaries. Installing the binaries takes just a few minutes. At the end of the installation, you will be prompted to enter a password for the administrator.

Setting Up the Web Server

Assuming you are using Apache and it is already running, you will need the appropriate privileges (probably root) to make these changes. If you do not have these privileges, you need to speak nicely to the system administrator, otherwise you could install Apache and run it as a regular user (which is what I did).

The web server needs to run the library program, which is the Greenstone web library application. Use the Apache *ScriptAlias* directive to configure a cgi-bin directory for Greenstone by adding

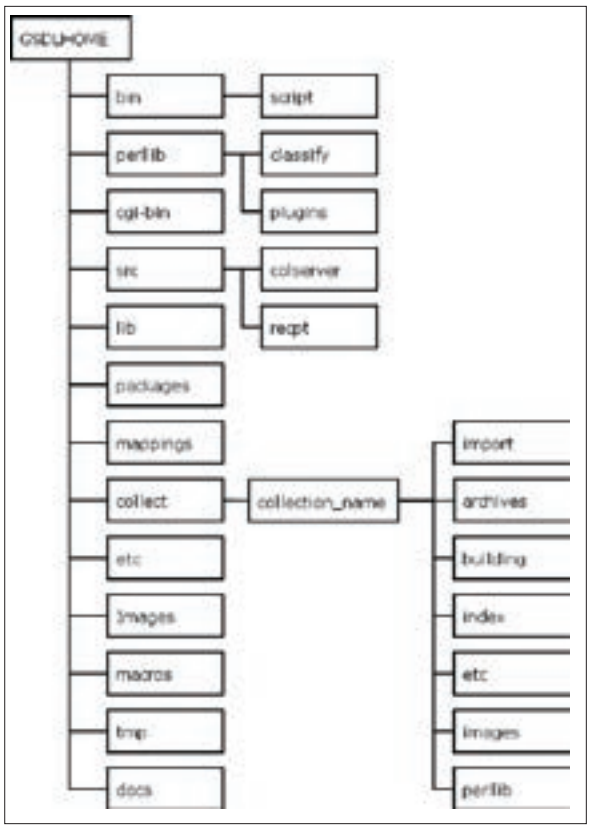

Figure 1: Structure of a Greenstone installation.

the following directives to your Apache configuration file, *httpd.conf*:

```
ScriptAlias /gsdl/cgi-bin ➥
"/home/chi/local/gsdl/cgi-bin"
<Directory ➥
"/home/chi/local/gsdl/cgi-bin">
Options None
AllowOverride None
</Directory>
```

You also need to configure the Greenstone directory to be web-accessible by adding the following *Alias* directive after the *ScriptAlias* directive to your *httpd.conf* configuration file:

```
Alias /gsdl ➥
"/home/chi/local/gsdl"
<Directory ➥
"/home/chi/local/gsdl">
Options Indexes ➥
MultiViews ➥
FollowSymLinks
AllowOverride None
Order allow, deny
Allow from all
</Directory>
```

Note the references to */home/chi/local/gsdl* in the apache directives. You will need to edit those paths to match the directory in which you installed Greenstone.

Once you restart Apache, you can access Greenstone by pointing your web browser at [6]. You may omit the port in the URL if your web server is running on the default port 80. If you are running Apache as a regular user and change the default port, you must specify the port in the URL.

System Requirements

At the time of writing this article, the latest version of Greenstone is version 2.62. Greenstone 2.62 is available at [4] in either binary executable (with statically-linked Linux binaries) or source code form.

Greenstone Structure

The Greenstone file structure is shown in Figure 1. When you build a new collection, a new *collection_name* folder is created in the Greenstone *collect* directory */home/chi/local/gsdl/collect*. Each collection has the same directory structure comprising a number of subdirectories (see Figure 1).

The *import* directory is where the original source material is placed. An *archive* directory contains the results of the import process. The *building* directory is a temporary directory used during the collection building process. Its contents are moved into the *index* directory once building is complete. The *etc* directory contains the collection's configuration information, most importantly, the *collect.cfg* file. The *images* directory holds collection-specific images. The *perllib* directory contains any Perl programs that are specific to the collection. For more details on the Greenstone system-wide directory structure, refer to the Greenstone Users Guide [8].

Building with GLI

For a first look at Greenstone in a real situation, I'll show you how to build a digital library collection using the arti-

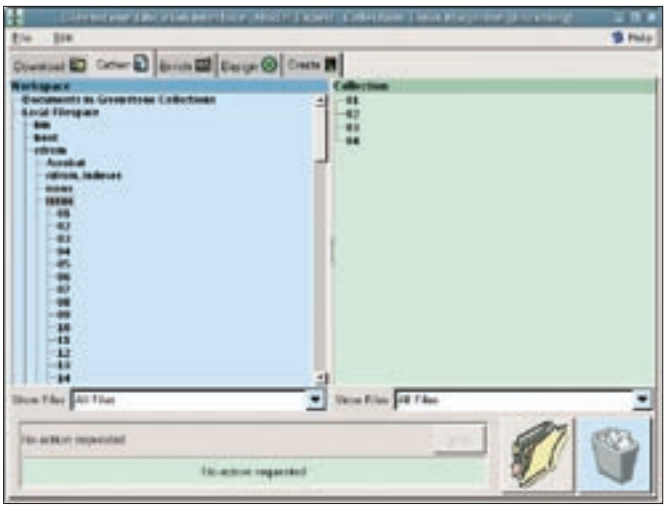

Figure 2: The Greenstone Librarian Interface (GLI).

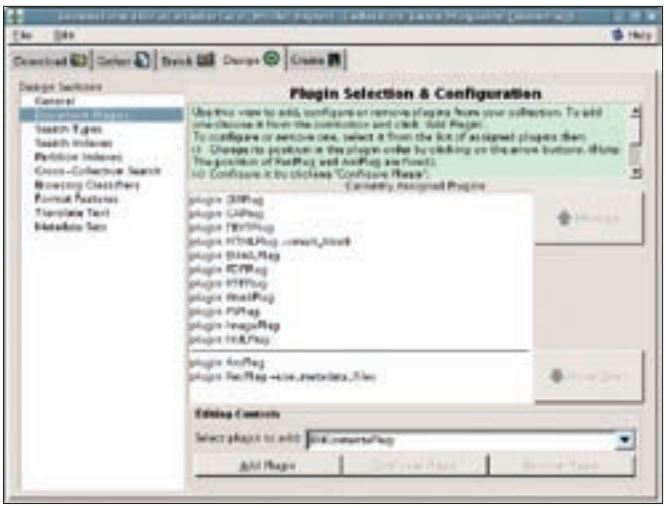

Figure 3: Design tools are available in the Design panel.

cles from the Linux Magazine Archive DVD, which last appeared in the March 2005 issue of Linux Magazine. (Keep in mind that this collection is for home use only. The license for the DVD does not allow you to post the material directly on the Internet. In general, it is important to make sure you are within the licensing requirements for any material you post in a digital library.)

Ensure that the archive DVD is loaded in the drive and mounted. You will need to know the directory where it is mounted. I am using Ubuntu, and the DVD is mounted at */media/cdrom*. Now you can build the collection using the Greenstone Librarian Interface (GLI). The GLI is a GUI application included with the Greenstone distribution that provides an easy point-and-click approach to building and customizing your library collections. GLI is a Java application that requires Sun's Java 1.4 Runtime Environment. To run GLI, type:

```
cd /home/chi/local/gli
./gli.sh
```

The first time you run the GLI, you will be prompted to fill in the URL of the Greenstone library. On my system, the setting is [6]. The port number is 9090 because I am running an Apache server as a regular user, as I described earlier.

The GLI provides you with a "walk-through" environment for building your digital collections. The basic steps in this procedure are:

- gathering documents (in the *Download* and *Gather* panels)
- assigning metadata (Enrich panel)
- designing indexing and browsing structures
- building the collection

To create a new collection, choose *File > New*. Enter the collection name (call it "Linux Magazine") and description, and hit OK. When you are prompted for a metadata selection, choose the Dublin Core metadata set. You can then select documents (or whole directories of documents) from the "Workspace" panel (on the left) and drag them over to the Collection panel (on the right). GLI behaves like a typical file manager, enabling you to copy and remove files from your "collection" (Figure 2).

Greenstone automatically extracts useful metadata from source documents during the building process. This is a very powerful feature if the documents contain metadata like title, author, subject, or keyword. Since the documents on your linux DVD do not contain this type of metadata, the GLI cannot extract anything useful automatically. We can, however, edit the metadata manually in the Enrich panel. The metadata can be managed at the folder or file level. Metadata assigned to a folder is inherited by all the files within the folder.

Once you have copied the source documents (or directories) to the Collection area, you may need to change the file permissions in order to build the collection:

```
cd /home/chi/gsdl/collect/➥
linuxmag/import
chmod -R +w *
```

Now you are ready to build the collection. For this example, I have only copied over the articles for Issues 1 to 4. This was just to save time in the building process. If you want to build the entire archive, drag over all the directories, but be prepared for a bit of a wait while the building process completes.

To build the collection, go to the *Create* panel and click *Build Collection*. Once this is complete, your collection will be ready to access. Click *Preview Collection* to view the collection in your web browser. During the build process, metadata is extracted automatically. I did some "manual" tidying up of the metadata structure. From the *Enrich* panel, I added the issue number of Linux Magazine to the *dc.Description metadata* field at the folder level. By doing this, all the articles under an issue number can be grouped together when setting up a browsing classification.

Note that there is no issue field in the Dublin Core metadata set. I therefore used *dc.Description* to store the information. Also, at the file level, I added a title entry to the *dc.Title metadata* field (as the automatically extracted title metadata does not look particularly useful). I added the Linux Magazine section (e.g., News, Cover Story, Know-How) to the *dc.Resource Identifier* metadata field. The prefix *dc* stands for Dublin Core, which is the metadata standard adopted by Greenstone.

Next, I designed the indexing and browsing structures based on the available metadata. All the designing and customizing features are available in the *Design* panel (see Figure 3).

In Greenstone, documents and metadata specifications are imported by soft-

ware modules called *plugins*. Plugins enable Greenstone to support many different document formats. You can add or remove plugins depending on what document types you have in your collection. (Note that you cannot remove *GAPlug*, *ArcPlug*, or *RecPlug*, since they are mandatory.) Because the Linux Magazine Archive documents are mainly PDF and HTML, PDFPlug and HTMLPlug are the most important plugins for this collection.

Indexing

Greenstone offers full-text searching of the documents in the collection from within a web browser window. You can search for any combination of words or phrases. By default, a Greenstone collection comes with three search indexes: text, title, and source. You can change the indexes assigned to your collection in the *Search Indexes* section of the *Design* panel (Figure 4). I removed the source index from my Linux Magazine collection, as it is just the file name of the document, and it is not a particularly useful search indicator. Also, I added the *dc.Title* metadata field as an index indicator for the titles index.

Figure 5 shows the search interface for the titles search. Greenstone also allows users to specify more complicated search terms. The advanced search interface can be set up through the *Preferences* option, located on the top right-hand corner of the collection page.

Browsing Classifications

Greenstone also allows users to browse the documents in a collection. The browsing structures are generated automatically from the metadata that is associated with each document in the collection.

You set up the browsing classifiers in the *Browsing Classifiers* section of the *Design* panel (Figure 6). All classifiers generate a hierarchical structure that is used to display a browsing index. The lowest level of this hierarchical structure is usually the documents, but it can consist of sections for some classifiers. A number of classifiers are available; refer to the Greenstone Developer's Guide [7] for details.

For my Linux Magazine collection, I used the *AZList* and *AZCompactList* classifiers to set up the browsing structures. The *AZList* classifier shows the classification terms in alphabetic order, while the *AZCompactList* classifier groups the terms that appear multiple times in the hierarchy together into a new node, shown as a bookshelf icon. The classifier settings (and associated options) for my Linux Magazine collection are:

- For browsing by title: *AZList -metadata dc.Title*
- For browsing by issue number: *AZCompactList -metadata dc.Description -buttonname issue*
- For browsing by Linux Magazine sections: *AZCompactList -metadata dc.Resouce Identifier -mingroup 1 -buttonname section*

Setting the *mingroup* option to 1 means that a bookshelf node is created at the top level even when there is just one item in the group. From the Greenstone web interface, you can select a browsing classification (for example, titles, author, and how-to) by clicking on the associated icon.

For each browsing classification, you can configure the icon. If you are not happy with the defaults, you can create your own Greenstone-style icons. For my Linux Magazine collection, I created new icons for the *sections* and *issues* browsing classification. We associate these icons with their respective classifications by adding them in the *buttonname* option (see Figure 6). I will show you how to create Greenstone-style icons later.

Formatting Features

Greenstone Library web pages are generated dynamically when requested. Format commands are used to change the appearance of these pages – particularly how documents are shown in browsing and search results lists.

To manipulate a format command, choose the *Format Features* section in the *Design* panel. You can make use of HTML tags, metadata values (enclosed in square brackets), some customized format string items (e.g., highlight, numleafdocs), and conditional expressions (like {*If*} or {*Or*}). You'll find a complete list at [7].

You can customize the look of each of the browsing classifications. For example, for the *Titles* browsing classifier, select *CL1:AZList -metadata dc.Title* from *Choose Feature* and *VList* (determines the vertical list format of the search results) from the affected component. I customized using the following format statements:

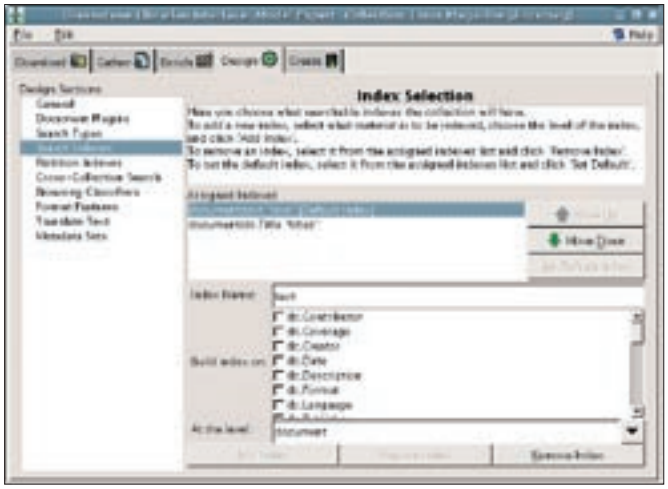

Figure 4: Define index settings in the Search Index section of the Design panel.

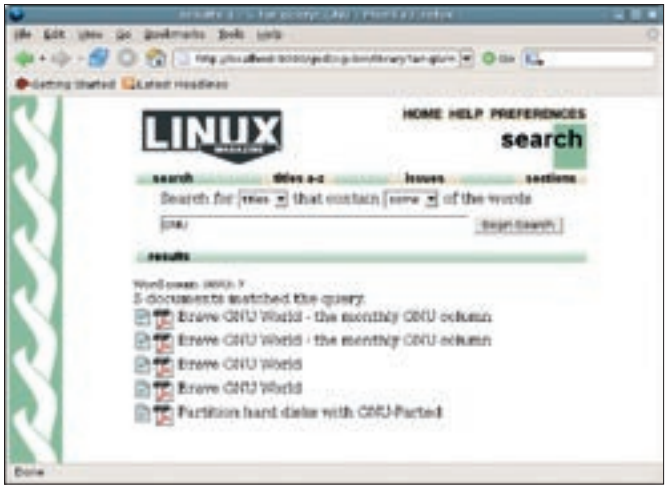

Figure 5: The interface for a titles search using our very useful sample collection.

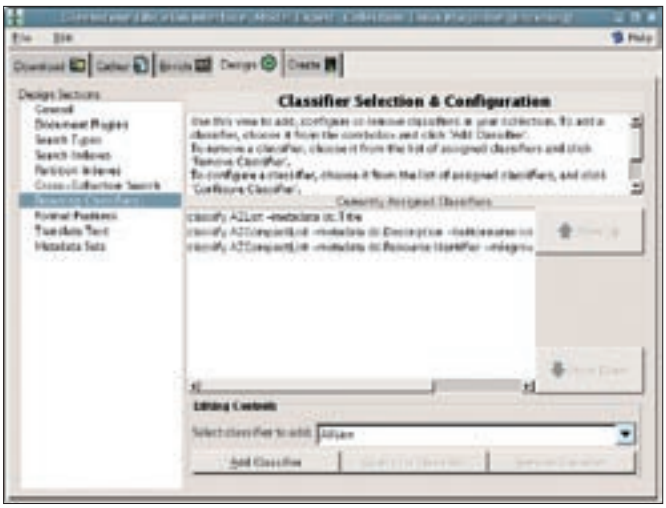

Figure 6: Setting up browsing classifiers in the Design panel.

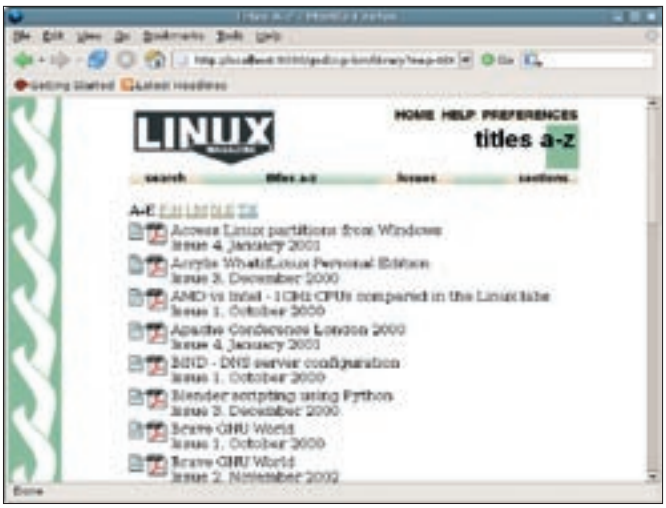

Figure 7: Browsing by titles in Greenstone.

```
<td valign=top> ➥
[link][icon][/link]</td>
<td>[srclink][srcicon]➥
[/srclink]</td>
<td>[highlight]{Or}➥
{[dc.Title],[ex.Title],➥
Untitled}[/highlight]
<br>[dc.Description]</td>
```

This format statement will show an icon that links to the Greenstone version of the document, an icon that links to the original document, the title, and issue details for each of the documents in the A-Z titles browsing list (Figure 7).

With the issues browsing classifier (using *CL2:AZCompactList -metadata dc.Description* in *Choose Feature* and *VList* in the Affected Component), I formatted it by adopting the following statements:

```
<td valign=top>➥
[link][icon][/link]</td>
<td>{If}{[numleafdocs],➥
<b>[Title]([numleafdoc])</b>,➥
[srclink][srcicon][/srclink]
<td>[highlight]{Or}➥
{[dc.Title],[ex.Title],➥
Untitled}[/highlight]➥
</td>}</td>
```

This will cause the documents to be grouped under their respective issue, and the total number of documents for that issue will be displayed.

Similarly, the setting for the section browsing classifier (select the *CL3: AZCompactList -metadata dc.Resource* Identifier for the *Choose* Feature and *VList* for the affected component) is shown below. An extra feature *< br > [dc.Description] < /br >*, which shows the issue details for the document, is added.

```
<td valign=top> ➥
[link][icon][/link]</td>
<td>{If}{[numleafdocs],➥
<b>[Title]([numleafdoc])</b>,➥
[srclink][srcicon][/srclink]
<td>[highlight]{Or}➥
{[dc.Title],[ex.Title],➥
Untitled}[/highlight]<br>➥
[dc.Description]</td>}</td>
```

All the above configuration features and settings can be easily manipulated in the GLI Design panel *Format Features* section. Format statements can be changed without rebuilding the collection.

Simple Collection Customization

Adding an icon for your collection is easy. You specify the *about page* and *home page* icons in the *General* section of the *Design* panel. Greenstone software also provides a facility for users to generate Greenstone-style collection images and classifier icons. Go to *http://www.greenstone.org/make-images.html* to create new images and icons. These images and icons should be stored in the images folder of your Greenstone installation (refer to Figure 2). The web page describes how to configure Greenstone to use the newly created images.

You can rebuild a collection at any time. Format statements can be changed without rebuilding the collection. You should be able to view any changes by refreshing the web page or by clicking the *Preview Collection* button in the *Create* panel. For more on customization and operation, refer to the Greenstone User's Guide [8].

Summary

Greenstone is an extremely useful application for storing, searching, and organizing large numbers of electronic documents. Once you have built and customized your digital library collection, you can access the collection using any regular web browser. ■

INFO

[1] The New Zealand Digital Library Project, The University of Waikato: *http://www.nzdl.org*

[2] DL Consulting Projects: *http://www.dlconsulting.co.nz/cgi-bin/index.cgi?a=p&p= projects*

[3] Examples of Greenstone in Action: *http://www.greenstone.org/cgi-bin/library?a=p&p= examples*

[4] Greenstone Software Download: *http://prdownloads.sourceforge.net/greenstone/gsdl-2.62-unix.tar.gz*

[5] Apache: *http://www.apache.org*

[6] Point your browser at the URL: *http://localhost:9090/gsdl/cgi-bin/library*

[7] Customizing your Greenstone Library: *http://www.greenstone.org/cgi-bin/library?a=p&p=faqcustomize#customizeformat*

[8] Greenstone Documentation: *http://www.greenstone.org/cgi-bin/library?a=p&p=docs*