

Customizing Greenstone User Interface

By Allison B. Zhang
Washington Research Library Consortium
October 2008

[Introduction](#)

[Greenstone User Interface Overview](#)

[Finding the Files You Need to Work On](#)

[Changing Text Header](#)

- [Collection Title](#)
- [Navigation Links](#)
- [Banner Title](#)
- [Navigation Bar](#)

[Modifying Graphic Header](#)

- [Collection Title Image](#)
- [Navigation Link Images](#)
- [Banner Title Images](#)
- [Navigation Bar Images](#)

[Modifying Document Display](#)

- [Browse Lists](#)
- [Document Display](#)
- [Search Options and Search Results](#)

[Adding a Footer](#)

Introduction

This document explains how to customize your Greenstone collection user interface. This is an updated version of the document I wrote in August 2003, which was based on Greenstone version 2.38. This updated version is based on Greenstone versions 2.74 and 2.80. You can still access the [old document](#) for comparison, especially if you want to upgrade your Greenstone from older version to version 2.74 or 2.80.

Before you read this document, you should read the Greenstone Digital Library Developer's Guide. You should install the system and build a collection. You should be familiar with the Greenstone's structure, terminology, the default user interface, and have basic knowledge and skills in web site design, HTML, and CSS stylesheet.

Greenstone's configuration system is very flexible. There are many ways to customize the user interface for your needs. There are many helpful information, guides, and tutorials on how to customize the user interface on the Greenstone web site. This document will explain the ways we used to customize about 40 collections on our web site. I will try not to repeat what the other documents have explained.

Although customizing Greenstone user interface is based on many factors such as contents of a collection, formats of the digital objects, and very important, the metadata design, this document will focus on how to change the look and feel of Greenstone user interface. For more information about how to design a user interface for digital collections, you can read a new book [Creating Digital Collections: A Practical Guide](#) by Allison B. Zhang and Don Gourley, available at [Amazon.com](#).

Greenstone User Interface Overview

The Greenstone user interface is not made in a complete HTML file or in one configuration file. We can divide the Greenstone user interface into 6 parts as illustrated in Figure 1. To make changes in different part, you need to work on different files.

Figure 1: Greenstone User Interface Overview

<div data-bbox="229 98 311 212">1</div>  <h1 data-bbox="319 112 798 212">Humanity Development Library 2.0</h1> <p data-bbox="229 212 798 246">for sustainable development and basic human needs</p>	<div data-bbox="798 98 1356 134">HOME HELP PREFERENCES SEND FEEDBACK</div> <div data-bbox="877 156 941 224">3</div> <div data-bbox="1053 134 1300 201">subjects</div> <div data-bbox="1244 98 1308 134">2</div>
<div data-bbox="229 246 311 291">4</div> <div data-bbox="311 246 391 291">search</div> <div data-bbox="391 246 630 291">subjects</div> <div data-bbox="630 246 798 291">titles a-z</div> <div data-bbox="798 246 1077 291">organisations</div> <div data-bbox="1077 246 1356 291">how to</div>	
<div data-bbox="229 291 1356 665"><div data-bbox="229 291 279 336"></div><div data-bbox="279 291 1356 336">01.00 General reference</div><div data-bbox="686 291 750 358">5</div><div data-bbox="229 336 279 380"></div><div data-bbox="279 336 1356 425">02.00 Sustainable Development, International cooperation, Projects; NGO, Organizations, Poverty and Hunger Alleviation, Basic Human Needs</div><div data-bbox="229 425 279 470"></div><div data-bbox="279 425 1356 470">03.00 Economics, Economic policies, Finance, Trade, Structural Adjustment</div><div data-bbox="229 470 279 515"></div><div data-bbox="279 470 1356 560">04.00 Business, Micro-entreprises, Management, Marketing, Finance, Microfinance, Cooperatives</div></div> <div data-bbox="590 582 654 649">6</div> <div data-bbox="670 604 766 649"> Home </div>	

- 1** **2** **3** -- Header. The header can be text or graphics, which requires different configuration in different files.
- 4** -- navigation bar that displays search and browse options. The navigation bar can also be displayed as text or as graphics which requires different configurations.
- 5** -- document display area where you can display any browse list or document metadata record.
- 6** -- footer that you can put on every page.

Finding the Files You Need to Work On

In order to change the look and feel of the Greenstone user interface, you need to work on the following files:

Collect.cfg

This is the collection configuration file. You can find this file in the /gsdl/collect/your collection name/etc/ directory. Details about how to create this file can be found in the "Developer's Guide, 1.5 Collection configuration file" and "2.3 Formatting Greenstone output". If you use Greenstone Librarian Interface (GLI), the changes in the collect.cfg file can be made by clicking the Format button and the Design button.

Macro files

A macro file has an extension **dm**. All macro files are stored in the /gsdl/macros/ directory. Details about how to create macros and macro files can be found in the "Developer's Guide 2.4 Controlling the Greenstone user interface".

Each macro file contains one or more packages. Each package contains a number of macros. A macro starts and ends with an underscore **_** and the actions and statements are included in curly brackets. It is very important to put the macros under correct package name. For example,

```
package Global

_httpiconchalk_ {_httpimg_/chalk.gif}
```

This macro name is httpiconchalk. It is in package Global. The action is to display the chalk.gif image in the directory defined by another macro named httpimg.

You can edit macro files in any text editor and save it as a text file. You can create your own macros following the format above. Macros will work within the Greenstone system.

In older versions of Greenstone, a macro can be specified for a particular collection with [c=collection name], which will overwrite the default macro. In newer versions of Greenstone, collection specified macros can be put in a macro file **extra.dm** that is placed in the collection's /macros/ directory. For example,

```
/gsdl/collect/your_collection_name/macros/extra.dm
```

You can edit the extra.dm file in GLI, under Format, by clicking the Collection Specific Macros link. The macros in the extra.dm file will overwrite the default macros.

If you want to make changes in a default macro, find the macro in a macro file, then find the package name that contains this macro. Copy the macro and the package name to the Collection Specified Macros (the extra.dm file) in GLI under Format and make changes in the copied macro. It is wise not to make changes in the original macros unless you want the changes affect all your collections. If you want to create a new macro, create it in the extra.dm file because macros in the extra.dm file can only affect one collection. You will always have the original macro file as reference if you need them.

In the /macros/ directory, you will find over 100 macro files. The most used macro files, package names, and their rolls are listed below. (Package names may be case sensitive.)

Package Name	File Name	Roles
--------------	-----------	-------

package Global	base.dm english.dm	macros that work for all display screens.
Package Style	style.dm	controls the display style for all pages.
Package document	english.dm document.dm	macros for the document display.
Package query	query.dm english.dm	macros for the Search page display.
Package preferences	pref.dm english.dm	macros for the Preferences page display.
Package help	help.dm english.dm	macros for the Help page display.
Package about	about.dm english.dm	macros for the About page display.
Package browse	browse.dm english.dm	macros for the dynamic browsing interface.
	nav_css.dm	macros for using with CSS stylesheet style.css
	nav_ns4.dm	macros providing a backwards compatibility for GSDL versions 2.62 and earlier that use graphic headers and none CSS styles. Required to download Greenstone Interface Pack (gcip-2.63.zip)

style.css

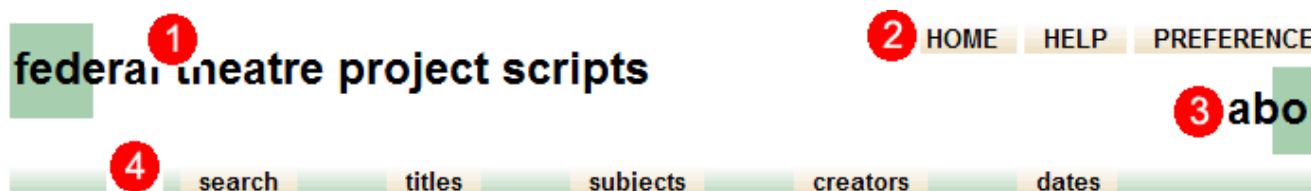
This is a stylesheet for all Greenstone collections. You can find this file in /gsdl/images/ directory. You can also create your own stylesheet.

Image files

All images files can be found in the /gsdl/images directory. You can store the collection specific images files in /gsdl/collect/your collection name/images/ directory.

Changing Text Header

Figure 2: Text Header (click a number to see how to make changes in that part)



Since version 2.63, the default Greenstone header display has been changed to text. The biggest advantage for a text based header is the convenience of changing languages as user's preference. When a user selects a display language from the Preferences page, the text on the navigation bar will change to the preferred language with the font, color and background color you specified in the configuration.

Changing collection title

In GLI, select Format - General and type your collection title in the Collection title field. You can also edit the collect.cfg file and put the title in:

```
collectionmeta collectionname "Federal Theater Project Scripts"
```

The default display of the collection title is lower case for all words. To change this, in the **style.css** file, find the following style:

```
p.collectiontitle {
font-family: sans-serif, arial, helvetica;
font-size: 24px;
font-weight: bold;
```

```

text-transform: lowercase;
white-space: nowrap;
padding: 8px 0px 12px 2px;
background-repeat: repeat-y;
background-position: top left;
margin:0;
}

```

change `text-transform: lowercase` to `text-transform: none`; The collection title will be displayed exactly as you typed in the Collection title field. Other values you can use for `text-transform` are:

capitalize	The first character of each word is converted to uppercase
uppercase	All characters are converted to uppercase
lowercase	All characters are converted to lowercase

You can also change font, font size, etc. in the `p.collectiontitle` style.

To change the color of the title, find the following style in `style.css` and change the color :

```

a.collectiontitle {
color: black;
text-decoration: none;
}

```

You can change or remove the background image under the Title.



`banner_bg.png`: this is the background image for both the Collection title and the banner title.

This image is defined in a macro `_cssheader_` in the **style.dm** file. To remove the image, copy the following lines to GLI - Format - Collection Specified Macros (or in the `extra.dm` file):

```

package Style

_cssheader_ {
_csslink_
<style type= text/css >
p.collectiontitle \{ \}
</style>
}

```

You can change the background image by replacing the existing image file in the `gsdl/images/` directory with either a new image file name or same filename `banner_bg.png` :

```

p.collectiontitle \{background-image: url( _httpimg_/your image name
); \}

```

Or use a background color instead of an image:

```

p.collectiontitle \{ background-image: none; background-color: yellow;
\}

```

Changing Navigation Links 2

To change the font, font size and text color, etc. for the navigation links, find the following style in the **style.css** file and make changes as you want.

```

p.bannerlinks {
font-family: sans-serif, arial, helvetica;
font-size: 10pt;
font-weight: bold;
margin:0;
padding:0;
}

```

To change the background image or color for the buttons, you need to make changes in the **_cssheader_** macro in the **style.dm** file. Copy the following style to GLI-Forma-Collection Specified Macros (extra.dm):

```
package Style

_cssheader_ {
_csslink_
<style type= text/css >

a.navlink \{ background-image: url( _httpimg_/bg_off.png ); \}
a.navlink_sel \{ background-image: url( _httpimg_/bg_green.png ); \}
a.navlink:hover \{ background-image: url( _httpimg_/bg_on.png ); \}
</style>
}
```

You can make your own background on and off images and replace the existing one. You can remove the background image and use background colors instead:

```
a.navlink \{ background-image: none; background-color: red; \}
a.navlink_sel \{ background-image: none; background-color: red; \}
a.navlink:hover \{ background-image: none; background-color: yellow;
\}
```

Note, these styles will affect all buttons on the navigation bar as well.

The wording on the buttons is defined in macros in the english.dm file. To change the text on the buttons, copy following macros to GLI - Format - Collection Specified Macro (extra.dm) and make change as you preferred.

```
package Global
_linktextHOME_ {HOME}
_linktextHELP_ {HELP}
_linktextPREFERENCES_ {PREFERENCES}
```

For example, if you want to change HOME to title case:

```
_linktextHOME_ {Home}
```

If you want to change HELP to Online Help:

```
_linktextHELP_ {Online Help}
```

Layout of the three buttons is defined in a macro **_globallinks_** in the **nav_css.dm** file. If you want to remove any of the buttons, copy the following macros in the nav_css.dm file to Collection Specified Macro (extra.dm) under package Global and make changes as you desire.

```
_globallinks_ {_If_( _cgiargu_ ne 1 ,_homelink_) _helplink_
_preblink_}
_homelink_
{ _navtaborig_( _httppagehome_, _linktextHOME_, _textdescrhome_ ) }
_helplink_
{ _navtaborig_( _httppagehelp_, _linktextHELP_, _textdescrhelp_ ) }
_preblink_
{ _navtaborig_( _httppagepref_, _linktextPREFERENCES_, _textdescrpref_ ) }
```

For example, to remove the Preference button:

```
package Global

_preblink_ { }
```

If you want to remove all three buttons on all pages, make the macro empty:

```
package Global

_globallinks_ { }
```

Changing Banner Title 3

The font, font size, and position, etc. of the banner title is controlled by a style in the **style.css** file.

```
p.bannertitle {
font-family: sans-serif, arial, helvetica;
font-size: 24px;
font-weight: bold;
margin: 10px 0 0 0;
text-transform: lowercase;
white-space: nowrap;
padding: 6px 2px 10px 0px;
background-repeat: repeat-y;
background-position: top right; }
```

The default display is lower case. To change the case to title case, change the following line:

```
text-transform: lowercase; to text-transform: none;
```

You can also change other values in this style.

The default background image is the same as for the Collection title: **banner_bg.png**. You can edit the **_cssheader_** macro in the **extra.dm** file by adding and changing the following line:

```
p.bannertitle \{background-image: url( _httpimg_/banner_bg.png ); \}
```

To change the wording of the banner titles, find macros starting with **_labelxxxx_** in the **english.dm** file and copy them to the Collection Specified Macro (extra.dm) and make changes. For example,

```
package Global

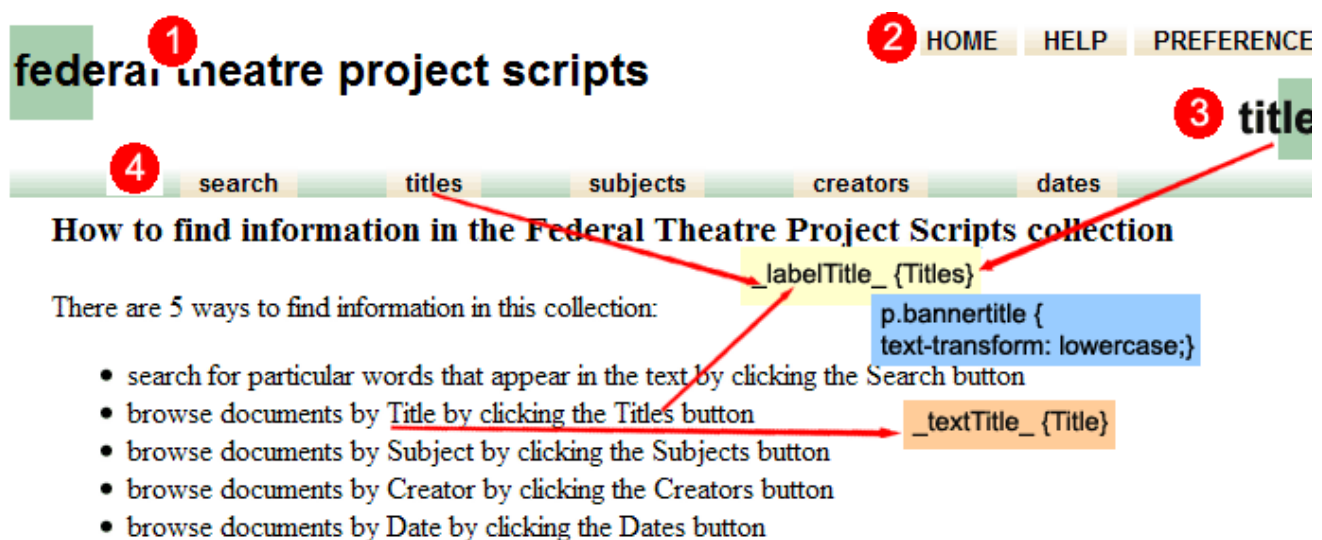
_labelTitle_ {Titles A-Z}
_labelCreator_ {Creators}
_labelSubject_ {Subject Areas}
```

These will also change the wording on the navigation bar.

You will also see macros in the english.dm file starting with **_textxxx_**. These are text showing on the "How to Find Information in Your Collection Name" section. See Figure 3 for details.

```
_textTitle_ {Title}
_textCreator_ {Creator}
_textSubject_ {Subject}
```

Figure 3: Macros/Style for Changing Wording and Case



If you cannot find the macro for your specific metadata, you can use a less used macro instead. For example, I want to create a browse option for all PDF files that I specified in the metadata element

WRLC.Admin.displayfile. I picked a less used macro/button: To. First, I define the classifier as:

```
classify AZList -metadata WRLC.Admin.displayfile -buttonname To
```

second, I add the following macro to the extra.dm file:

```
_labelTo_ {To}
```

and change it to:

```
_labelTo_ {View PDF Files}
```

The text on the navigation bar and the banner title will all changed to View PDF Files. I'll also change

```
_textTo_ {To}
```

to

```
_textTo_ {PDF}
```

The How to Find Information... section will show the following line:

- Browse documents by PDF by clicking the View PDF Files button.

Changing Navigation Bar

The font, font size, and color, etc. on the navigation bar is controlled by styles in the **style.css** file.

```
div.navbar {
min-width: 500px;
white-space: nowrap;
font-family: sans-serif, arial, helvetica;
font-size: 10pt;
font-weight: bold;
text-transform: lowercase;
border: 0;
width: 100%; /* for ns4 */
}
```

The default case is lower case as you can see. To change to title case, change the line:

```
text-transform: lowercase; to text-transform: none;
```

You can also change font family, size, and so on in this style.

To change text color, find the following style in the **style.css** file and make changes.

```
/* bg image set in style macro */
a.navlink {
vertical-align: top;
padding: 0 10px;
color: black;
text-decoration: none;
}

/* bg image set in style macro */
a.navlink_sel {
vertical-align: top;
padding: 0 10px;
color: black;
text-decoration: none;
}

/* bg image set in style macro */
a.navlink:visited {
color: black;
}
```

Changing the background color or image of the buttons on the navigation bar is the same as changing the

background color or image for the navigation links (#2) in the macro:

```
_cssheader_ {
_csslink_
<style type= text/css >
a.navlink \{ background-image: url( _httpimg_/your background image
for link ); \}
a.navlink_sel \{ background-image: url( _httpimg_/your background
image for selected link ); \}
a.navlink:hover \{ background-image: url( _httpimg_/your background
image for on link ); \}
p.collectiontitle \{ \}
p.bannertitle \{background-image: url( _httpimg_/your background image
); \}
</style>
}
```

If you don't want to use any image for the button background, you can use solid colors instead:

```
a.navlink \{ background-image: none; background-color: red; \}
a.navlink_sel \{ background-image: none; background-color: red; \}
a.navlink:hover \{ background-image: none; background-color: yellow;
\}
```

The default navigation bar image is controlled by a style under `_cssheader_`:

```
div.navbar \{ background-image: url( _httpimg_/bg_green.png ); \}
```

You can make changes in this style to use your image. If you don't want an image but use a background color instead to match the button color on the navigation bar:

```
div.navbar \{ background-image: none; background-color: red; \}
```

Changing the wording on the navigation bar is the same as changing wording for the banner title.

```
_labelTitle_ {Titles AZ}
_labelCreator_ {Creators}
_labelSubject_ {Subject Areas}
```

To add a new browse option and a button on the navigation bar, you need to define a new classifier. For example,

```
classify AZCompactList -metadata DC.Creator -buttonname Creator
```

After you build the collection, you will see a Creator on the navigation bar.

Note, there are two more styles under `_cssheader_` macro that I have not mentioned:

```
body.bgimage \{ background-image: url( _httpimg_/chalk.gif ); \}
```

is the style for the background image on the left side of each page. You can remove it or use your own image instead.

```
div.divbar \{ background-image: url( _httpimg_/bg_green.png ); \}
```

is the image for the divider. You can create your own divider line image and replace the `bg_green.png` with your file name. If you don't want to use an image, you can create any style you want. For example:

```
div.divbar \{ background-image: none; background-color: red;
height:1px; \}
```

creates a thin red line as the divider.

The following table summarizes the files you will need to edit in order to make changes in the text header.

Part	Part Name	GLI OR collect.cfg	macro	style.css
------	-----------	--------------------	-------	-----------

1	Collection title	Format -- General -- Collection title	collectionmeta collectionname		p.collectiontitle a.collectiontitle
2	Navigation links			_cssheader_ _linktextHOME_ _linktextHELP_ _linktextPREFERENCES _globallinks_ _homelink_ _helplink_ _preflink_	p.bannerlinks a.navlink a.navlink_sel a.navlink: hover a.navlink: visited
3	Banner title	Design -- Browsing Classifiers	classify	_labelTitle_ _labelxxxx_	p.bannertitle
4	Navigation bar	Design -- Browsing Classifiers	classify	_cssheader_ _labelTitle_ _labelxxxx_	div.navbar a.navlink a.navlink_sel a.navlink: hover a.navlink: visited

Modifying Graphic Header

Starting from Greenstone 2.63 release, there have been extensive modifications to support HTML2 and CSS. Previously graphic based collection title, banner titles, and navigation buttons, etc. are replaced by text that is controlled by CSS stylesheet. If you used graphic headers for your collections built on Greenstone 2.62 or earlier versions, after you upgraded to version 2.74 or later, your graphic header will look messy. You will need to make some changes if you want to keep the graphic header.

1. replace nav_css.dm with nav_ns4.dm in the /etc/main.cfg file:

```
# The list of display macro files used by this receptionist

macrofiles tip.dm style.dm base.dm query.dm help.dm pref.dm
about.dm \
document.dm browse.dm status.dm authen.dm users.dm html.dm \
extlink.dm gsdl.dm extra.dm home.dm collect.dm deposit.dm docs.dm
\
bsummary.dm gti.dm gli.dm nav_ns4.dm languages.dm \
```

2. download the new 'classic interface' pack (Greenstone Interface Pack gcip-2.63.zip), which contains most of the old text images. Unzip the file in the gsdl/images/ folder.

The following section explains how to create and make changes in the graphic header.

Figure 4: Graphic Header (click on a number to see how to change in that part)



Collection Title Image 1

You define the title image in the collect.cfg file, `iconcollection` using the following format:


```
|  |
| --- |
|  |

```

This part of navigation buttons is very flexible. You can change the position of the buttons, add new buttons, delete buttons, or do not display them at all.

To add a new button, do the following:

- 1) create a new set of images (cxxxof.gif and cxxxon.gif) and save them in the gsdl/images/xx/ directory, where /xx/ is the language code for the language you put on the button image. You can save the images in other directory that you will need to define next.
- 2) create a set of macros for the image location following the format of the other navigation links (e.g. Help, Preferences). The default macros are in the nav_ns4.dm file. For example, I want to create a new link to our digital collections page:

```

package Global

## COLLECT ## top_nav_button ## ccollect ##
_httpiconccollectof_ {_httpimg_/en/ccollectof.gif}
_httpiconccollecton_ {_httpimg_/en/ccollecton.gif}

```

- 3) create a new httppage macro in the package Global in the extra.dm file. This is the URL to open when the button is clicked. For example,

```
_httppagecollect_ {http://www.aladin.wrlc.org/dl/}
```

- 4) create a new text macro for the button. This is for the alt text in the tag. For example,

```
_textdescrcollect_ {WRLC Special Collections}
```

- 5) create a link macro

```

_collectlink_
{_gsimage_(_httppagecollect_,_httpiconccollectof_,_httpiconccollecton_,collect,_t

```

- 6) add the link macro to the _globallinks_ macro, for example,

```

_globallinks_ {_If( _cgiargu_ NE 1 ,_homelink_) _helplink_
_preblink_collectlink_}

```

Image Locations

The location macros for the images of the top navigation buttons are defined in the package Global in the **nav_ns4.dm** file. All these macros start with _httpicon followed by a c and the button name. I call the buttons as C buttons. Each button is made of a pair of images, on image and off image, which can be different colors. When you point your cursor on the button, you will see the color change effect. To change the look of the buttons, you need to make two images.

```

## HOME ## top_nav_button ## chome ##
_httpiconchomeof_ {_httpimg_/en/chomeof.gif}
_httpiconchomeon_ {_httpimg_/en/chomeon.gif}

## HELP ## top_nav_button ## chelp ##
_httpiconchelpof_ {_httpimg_/en/chelpof.gif}
_httpiconchelpon_ {_httpimg_/en/chelpon.gif}

```

To put the images in a directory other than the default directory in /gsdl/images/xx/, you need to put the path

in the curly brackets.

```
# Collections
_httpiconccollectof_ {_httpprefix_/collect/auhist/images
/ccollectof.gif}
_httpiconccollecton_ {_httpprefix_/collect/auhist/images
/ccollecton.gif}
```

Banner Title Images

3

Image macros of the banner titles are defined in the `nav_ns4.dm` file, which all start with `_titleimage` and the banner title. For example,

```
_titleimageabout_ {<img src= _httpiconhabout_ width= _widthhabout_
height= _heighthabout_ >}
_titleimageTitle_ {<img src= _httpiconhtitle_ width= _widthhtitle_
height= _heighthtitle_ >}
_titleimageCreator_ {<img src= _httpiconhcreat_ width= _widthhcreat_
height= _heighthcreat_ >}
```

If you have downloaded the Greenstone Interface Pack (gcip-2.63.zip) and replaced `nav_css.dm` with `nav_ns4.dm` in the `main.cfg` file, you have all image macros you may need, unless you want to create a new banner title. The image filenames all start with an "h_". All you need to do is to:

- create the banner images and save them as the original file names located in `/images/xx/h_xxx.gif` (jpg, png)
- change the width (`_widthhxxxx_`) and the height (`_heighthxxxx_`) in the relevant macros.

The banner titles for browse options will automatically show up after you set up a classifier. For pages other than the browse options, such as Search, Help, Preferences, etc. you will need to place a macro `_imagerthispage_` in the relevant package. For example, you want to display a banner title for the Search page, copy the following line to your `extra.dm` file:

```
package query

_imagerthispage_ {<img src= _httpiconhsearch_ width= _widthhsearch_
height= _heighthsearch_ >}
```

There is no default banner title image for the document display pages. I created a new macro and modified the `_imagerthispage_` macro in the document package to display a Description banner title in the content page.

```
document package

_imagerthispage_ {_If( _iconthispage_ NE , <img src=
_httpprefix_/collect/auhist/images/h_desc.gif >, _iconthispage_)
```

The locations of the banner title images are defined in package Global in the `nav_ns4.dm` file. All these macros start with `_httpicon` followed by an h and the banner name. I call them the "H image". You name your H images as `h_imageName`. For example, `h_title.gif`, `h_subj.gif`, etc.

```
## titles AZ ## green_title ## h_title ##

_httpiconhtitle_ {_httpimg_/en/h/_title.gif}

_widthhtitle_ {200}
_heighthtitle_ {57}
```

The width and the height are the number of pixels. You can change them if you changed the width and height of the image.

To change the location of an image, copy the macro to your `extra.dm` file and change the path to the image in the curly brackets.

```
## titles AZ ## green_title ## h_title ##

_httpiconhtitle_ {_httpprefix_/collect/your path/h/_title.gif}
_widthhtitle_ {132}
_heighthtitle_ {106}
```

To change the look of the banner images, you create an image in PhotoShop or a graphic editor and save it with the same file name.

Navigation Bar Images 4

The buttons on the navigation bar, except the Search button, are controlled by the classifiers defined in the collect.cfg file. When you define a classifier in the collect.cfg file (or in GLI, Design -- Browsing classifiers, then select classifier to add), the corresponding button will appear on the navigation bar. For example, the following classifiers will turn on the Title and the Subject buttons.

```
classify AZList -metadata Title
classify AZCompactList -mingroup 1 -metadata DC.Subject -buttonname
Subject
```

The buttons are defined in the package Global, in the nav_ns4.dm file, which all start with _image followed by the button name. For example,

```
_imageTitle_
{ _gsimage_(1, _httpiconttitleof_, _httpiconttitleon_, titles, _navlinkalt_(Title)) }
```

The locations of the images are defined in the macros in the package Global in the nav_ns4.dm file. All these macros start with _httpicon and followed by a t and the button name. I call them the "T button".

A button on the navigation bar is made of three images, the off image, the on image, and the gr image. If you want to change the look of the buttons, you need to make three images.

```
## Titles AZ ## nav_bar_button ## ttitl ##

_httpiconttitlegr_ { _httpimg_/en/ttitlgr.gif }
_httpiconttitleof_ { _httpimg_/en/ttitlof.gif }
_httpiconttitleon_ { _httpimg_/en/ttitlon.gif }
```

You also need to change the look of the space image put between the buttons. This space image is called bg_green.png and is defined in the package Global in the base.dm file

```
_httpicontspace_ { _httpimg_/bg_green.png }
_heighttspace_ { 17 }
```

You can make your own space image and define it in the _httpicontspace_ macro:

```
_httpicontspace_ { _httpimg_/tspace.gif }
_heighttspace_ { 17 }
```

The width of the button images is defined as number of pixels. If you have a long text, say "Place Names", and the button is wider than the other buttons, you put the width pixels in the curly brackets. When you build the collection, Greenstone will automatically calculate the spaces between the buttons and the tspace.gif will fill up the spaces.

There are two ways to add a new button: 1) use an existing button name and change the text on the images; OR 2) create new button images and create new macros for the new button.

1) Use an existing button name for a new button

For example, you need a button called Places and you decided to use an existing button called From for this new button. Here is what you need to do:

- define a classifier in the collect.cfg file or in GLI:

```
Classify AZCompactList -mingroup 1 -metadata DC.Coverage.spatial
-buttonname From
```

- in PhotoShop or a graphic editor, change the text From to Places on all three tfrom images, may be in different colors, and put them in the /images/xx/ directory.

```
tfromgr.gif
tfromof.gif
tfromon.gif
```

2) Created new images and new macros for a new button

For example, you want to create a new button called Museums. Here is what you need to do:

- create three images and save them in the collect/collection name/images directory.
- create a classifier in the collect.cfg file

```
classify AZCompactList -mingroup 1 -metadata DC.Corpname -buttonname
Museums
```

- create a set of new image macros in the package Global in your extra.dm file using the same format as the other image macros:

```
_imageMuseum_
{_gsimage_(1_,_httpicontmuseumof_,_httpicontmuseumon_,museums,_navlinkalt_(Museum
_imageMuseumgreen_ {<img src= _httpicontmuseumgr_ width= _widthhtmuseumx_
border= 0 >}
```

- define the locations of the images and the width of the images

```
# Museum
_httpicontmuseumgr_ {_httpprefix_/collect/auhist/images/tmuseumgr.gif}
_httpicontmuseumof_ {_httpprefix_/collect/auhist/images/tmuseumof.gif}
_httpicontmuseumon_ {_httpprefix_/collect/auhist/images/tmuseumon.gif}
_widthhtmuseumsx_ {110}
```

The following table summarizes the files and macros you will need to edit in order to create a graphic header using the old style.

Part	Part name	GLI OR Collec.cfg		Macro
1	Collection title	Format -- General -- URL to about page image	iconcollection	
2	Navigation links			_globallinks_ _pagebanner_ _httpiconcxxxof_ _httpiconcxxxon_
3	Banner title	Design -- Browsing Classifiers	classify	_imagethispage_ _httpiconhxxx_ _widthhxxx_ _heightxxx_ _labelXxxx_ _textXxxx_
4	Navigation bar	Design -- Browsing Classifiers	classify	_imageXxxx_ _imageXxxxgreen_ _httpicontxxxgr_ _httpicontxxxof_ _httpicontxxxon_ _httpicontspace_ _labelXxxx_ _textXxxx_

Modifying Document Display 5

The document display area is the place to display browse list, search options and search results, metadata records, and so on.

Browse Lists

The browse list display is controlled by the classifier and the Format string in the collect.cfg file. If you want to setup a classifier in GLI --Design --Browse Classifiers, you can select a type of classifier from a list. The most used classifiers and their roles are as following.

- AZList displays a list of documents in alphabetical order regardless of their metadata value. It is good

for displaying a Title list.

- AZCompactList groups the digital objects that have the same metadata value. For example, Subject, Creator, etc. If you use the AZCompactList, a bookshelf icon will appear indicating there are more items under this group. The default is 1 document in one group (bookshelf). If you don't define "-mingroup" number, it will display all documents with a bookshelf icon no matter if it contains a single document or multiple documents.
- DateList displays date and is good for date metadata in numeric format, e.g. 1987-09-20.
- Hierarchy displays a hierarchical list based on the separator you defined for separating the levels of hierarchy.
- GenericList displays a list of documents. It groups the documents if they have the same metadata value and displays the single document if the metadata value is unique.

The order of the classifiers you put in the collect.cfg file (or GLI --Design --Browse Classifiers) determines the classifier's number. For example,

```
classify AZList -metadata Title
classify AZCompactList -mingroup 1 -metadata DC.Subject -buttonname
Subject
classify AZCompactList -mingroup 1 -metadata DC.Coverage.spatial
-buttonname From
classify DateList -metadata Date
```




This list defines:

- CL1: is the Title list.
- CL2: is the Subject list.
- CL3: is the From list.
- CL4: is the Date list.

You can format the display list accordingly. In most cases, the following simple format will work for most AZCompact lists.

```
Format CL2VList '<td>[link][icon][link] </td> <td>[link][Title]
[/link] {If}{[numleafdcs], ([numleafdcs])}'
```

This displays the subject list as

 [Citizens' associations -- History -- United States](#) (1)
 [Commencement ceremonies](#) (76)
 [Dedications -- Washington \(D.C.\)](#) (8)

The simple Date format

```
Format CL4DateList '<td>[link][icon][link] </td> <td>[link][Title]
[/link]</td>'
```

will display Date as

[1862-1916](#) [1917-1919](#) [1920-1943](#) [1945-1950](#) [1953-1957](#) [1958-1959](#) [1960-1961](#) [1962](#) [1963-1964](#) [1965-1966](#) [1967-1976](#) [1977-1988](#) [9999](#)

1862  [Office of the 55th New York Regiment : Fort Gaines during the winter of 1861-1862](#)

1875  [Portrait of Bishop John Fletcher Hurst, Chancellor: 1891-1902](#)

1895  [Portrait of John Fletcher Hurst, Chancellor: 1891-1902](#)

In the format string, you can include any HTML tags, CSS styles and Greenstone macros, just as you design a web page. For example, the following format string displays the Title list, which is the first classifier, with a thumbnail and brief description, as illustrated in Figure 5.

```
Format CL1VList
'<td valign=top align=center width=150>[ImageHTML]</td>
<td width=10> </td>
<td align=left valign=top>
<table border=0 width=435 cellpadding=2 cellspacing=0>
<tr><td width=100 align=left valign=top><b>Title:</b></td>
<td width=335 align=left valign=top>[link]<b>[Title]</b>[/link]
```



```

</td></tr>
<tr><td width=100 align=left valign=top><b>Date:</b></td>
{If}{[DC.Date.text], <td width=335 align=left valign=top>
[DC.Date.text]</td>,<td valign=top>[DC.Date.created]</td>}</tr>
</table>
</td></tr>
<tr><td colspan=3><hr align=center size=1 noshade></td></tr>'

```

Figure 5: Title List

[A](#) [B](#) [C](#) [D](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [P](#) [R](#) [S](#) [T](#) [V](#) [W](#) [0-9](#)



Title: [Abbey Joel Butler Pavilion, Adnan M. Khashoggi Center, John M. Reeves Aquatic Complex and the Jack I. Bender Arena \(1988\)](#)

Date: 1988-11



Title: [Abbey Joel Butler Pavilion and the Adnan M. Khashoggi Center \(1988\)](#)

Date: 1988-11



Title: [Abbey Joel Butler Pavilion and the Adnan M. Khashoggi Center \(1988\)](#)

Date: 1988-11

Document Display

When you click on a title on the Title list, you see either a full-text page or a metadata record describing the digital object. The display of this page is controlled by "Format DocumentHeading" and "Format DocumentText" in the collect.cfg file. The following format strings (partial) display the record in the format illustrated in Figure 6.

```

format DocumentHeading '[Title]'

format DocumentText

'<center>

<table width="_pagewidth_" border="0">

{If}{[DC.Title.alternative], <tr> <td align="right" valign="top"
width="90" class="displaylabel">Other title:</td><td align="left"
valign="top" class="displaytext">[DC.Title.alternative]</td></tr>}

<tr> {If}{[DC.Description], <td align="right" valign="top" width="90"
class="displaylabel">Description:</td><td align="left" valign="top"
>[DC.Description]</td>,<td align="right" valign="top" width="90"
class="displaylabel"></td><td align="left" valign="top"></td>} <td
valign="top" width="150" rowspan="16" align="center"><span
class="font_thumbnail">Click to see large image</span><br> <a href="
[WRLC.Admin.imgviewer]" target="imgviewer"> </a> </td>
</tr>

```

```
{If}{[DC.Subject],<tr><td align="right" valign="top" width="90"
class="displaylabel">Subject:</td><td align="left" valign="top"
class="displaytext">[DC.Subject]</td></tr>}

{If}{[DC.Subject.persname],<tr><td align="right" valign="top"
class="displaylabel">Person:</td><td align="left" valign="top"
class="displaytext">[DC.Subject.persname]</td></tr>}

{If}{[DC.Date.text],<tr><td align="right" valign="top"
class="displaylabel">Date:</td><td align="left" valign="top"
class="displaytext">[DC.Date.text]</td></tr>}

{If}{[DC.Type],<tr><td align="right" valign="top"
class="displaylabel">Resource type:</td><td align="left" valign="top"
class="displaytext">[DC.Type]</td></tr>}

</font></table>'
```

Figure 6: Document Display

format DocumentHeading

Abbey Joel Butler Pavilion and the Adnan M. Khashoggi Center (1988)

Date: 1988-11

Place name: United States -- District of Columbia -- Washington

Resource type: Photographs

Creator: Schwab, Hilary

Original: 1 photograph : col. ; 8 x 10 in.

Digital format: image/jpeg

Language: English


Collection: [American University History Photograph and Print Collection](#)

Rights: Reproduction and copyright information regarding this document is available from the American University Library -- University Archives.

Repository: American University Library -- University Archives

Date available: 2003-04-15

Click to see large image



Format DocumentText

The font size and color, etc. for the Title is set in style.css by a style:

```
div.heading_title {
font-family: Arial, Helvetica, sans-serif;
font-size: 11pt;
font-weight: bold;
color: #000066;
text-align: center;
margin-top: 20px;
margin-bottom: 0px;
}
```

Search Options and Search Results

The default search page displays a single search box as below.

Search for that contain of the words

If you want to display advanced search with multiple search options, you need to use the indexer MGPP. You can set it up in the collect.cfg file or in GLI -- Format, in the string SearchTypes:

```
SearchTypes "form, plain"
```

the default search page will display the formed search as seen below:

Search for of

Word or phrase	... in field
<input type="text"/>	keyword in titles, subjects, names, etc. ▾
<input type="text"/>	subject ▾
<input type="text"/>	title ▾
<input type="text"/>	place ▾

In both plain and form search pages there is a drop-down list "Search for...". This is a list of search indexes you can set up in collect.cfg file:

```
collectionmeta.Keywords "keyword in titles, subjects, names, etc."
collectionmeta.DC.Subject "subject"
collectionmeta.DC.Title "title"
collectionmeta.DC.Coverage.spatial "place"
```

You can do the same in GLI -- Design -- Search Indexes to select and add the search indexes and in GLI -- Format -- Search to define your display text for the drop-down list.

Search results display is controlled by a format string "SearchVList" in collect.cfg file or GLI -- Format -- Format Features, then select "Search" from "Choose feature" and "VList" from "Affected Component". If you put the same format as listed above for Format CL1VList, the search results will display as illustrated in Figure 5.

Adding a Footer 6

Adding a footer to every page is very easy. Find the `_footer_` macro in style.dm file, copy it to your extra.dm file and placed it under package Style, and add the footer content as you want. For example,

```
package Style

_footer_ {
  <!--page footer-->
  _pagefooterextra_

  <p class="footer">
  | <a href="_httpprefix_/cgi-bin/library?c=auhist&a=p&p=help">Help</a>
  | <a href="_httpprefix_/cgi-bin/library?c=auhist&
  a=p&p=preferences">Preferences</a>

  _AUfooter_

  _endspacer_
  _htmlfooter_
}
```

This displays a footer as the following:

| [Help](#) | [Preferences](#) | [WRLC Digital & Special Collections](#) | [Contact a Librarian](#) | [Main](#) |
| [American University Archives and Special Collections](#) |

The macro `_AUfooter_` is defined in another macro file wrlc.dm as the following.

```
_AUfooter_ {  
  
| <a href="/dl/">WRLC Digital & Special Collections</a>  
| <a href="mailto:research@american.edu">Contact a Librarian</a>  
| <a href="_httppagehome_">Main</a> | <Br>  
| <a href="http://bender.library.american.edu:8083  
/php/directory.php?dept_id=arc">American University Archives and Special  
Collections</a> | </p>  
<div class="divbar">&nbsp;  </div>  
}
```

In this way, if the URL changed for this institution, the change can be made in one place - the wrlc.dm file which will affect all collections from that institution.

(Last updated October 14, 2008)



[NoeSocket](#)