GODOT
Full-text Links from CUFTS, Interlibrary Holdings Locator and Requesting
Version 3.0

Administration Guide
Last Updated: February 4, 2011
GODOT is an open source development project of the Simon Fraser University Library. For more information, see the reSearcher web site: http://researcher.sfu.ca

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/2.0/ca/ or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
# Table of Contents

INTRODUCTION...............................................................................................................6
Contact Information.......................................................................................................6
About This Document...................................................................................................6

INSTALLING GODOT...............................................................................................7
Downloading GODOT.................................................................................................7
Required Software.......................................................................................................8
Scripts.............................................................................................................................9
Values required by the install scripts........................................................................10
Installation environment variable files......................................................................12
Running the software install script...........................................................................13
Running the database install script...........................................................................15
Updating the profile cache.......................................................................................16
After installation.........................................................................................................17
Installation testing page............................................................................................20
Full text links.............................................................................................................21
Check the web for this item........................................................................................22
Export citation/reference............................................................................................22
Print Holdings.............................................................................................................24
Patron Authentication and Retrieval of Patron Data...................................................26

CONFIGURATION TOOL..........................................................................................29
Administration Main Page........................................................................................29
Adding a new site.......................................................................................................30

CONFIGURATION OPTIONS......................................................................................31
Overview of retrieving holdings from a new site.......................................................31
Configuring a catalogue target ................................................................. 31
Configuring site holdings ........................................................................... 33
ERIC and Canadian Research Index Fiche Holdings ................................ 34
Searching for local holdings ...................................................................... 36
Testing the new site's holdings ................................................................. 37
Troubleshooting and Modifying new site holdings ................................. 37
Configuring GODOT to search other libraries ....................................... 38
Error Messages ......................................................................................... 40
Requesting from other libraries (mediated, unmediated, etc.) ............... 41
Enabling Auto-Requesting (including SHOW and HIDE holdings) .......... 43
Lending .................................................................................................... 44
Full-text Options ..................................................................................... 45
Display Options ....................................................................................... 46
Citation Enhancement ............................................................................... 47
Patron Authentication and Patron Data Retrieval .................................. 47
Putting Requesting behind a Site Password ........................................ 48
Patron and Request Information Screen ............................................... 48
Checking Patron Email Addresses ....................................................... 50
Confirmation and Acknowledgment of Request ................................... 50
Borrowing Request Messages ............................................................... 52

TEMPLATES .............................................................................................. 54
Overview .................................................................................................. 54
Template for main screen ....................................................................... 57
Templates for header and footer ............................................................ 57
Template for labels in the patron information form .............................. 57
Templates for the request confirmation screen ..................................... 58
Templates for request acknowledgment screen .................................. 59
Introduction

Launched from a link embedded in your library's citation databases or other resources, GODOT provides direct links to your fulltext collections, using the CUFTS knowledge base, and also reveals holdings in your catalogue or in other locations.

GODOT also works closely with Interlibrary Loan systems (including the open source OpenILL) and Integrated Library Systems to provide direct or mediated interlibrary loan requests by your users.

GODOT is open source software developed at the Simon Fraser University Library.

Contact Information
Any questions about GODOT should be directed to researcher-support@sfu.ca.

About This Document
Code samples, filenames, URLs and class names are presented in courier typeface.

Square braces are used in code samples, filenames, URLs and class names to indicate a sample value.
Installing GODOT

If you run into any problems with this installation please contact researcher-support@sfu.ca. We welcome suggestions for improvements and reports of any problems that you encounter.

Downloading GODOT

To get the latest version of GODOT you will want to use subversion checkout, so you will need to have svn installed on your machine.

To download GODOT to /usr/local/godot:

    cd /usr/local
    svn checkout http://cufts2.lib.sfu.ca:7999/godot/trunk godot

This step and those that follow assume that you are root for the software installation and the postgres administrative user (usually ‘postgres’) for the database installation. If you are not root/postgres, you will likely need to ask your system administrator for help with some steps.
Required Software

Before installing GODOT you should have a basic system up and running with Perl, Apache/mod_perl, PostgreSQL, YAZ and the required perl modules.

Table 1: Requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perl</td>
<td>Tested with v5.6.1 and v5.8.5.</td>
</tr>
<tr>
<td>Apache/mod_perl</td>
<td>Any recent version should work.</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>Tested with postgresQL 7.0.3 and 8.0.2.</td>
</tr>
<tr>
<td></td>
<td>Postgres will need to be running in order to run the install scripts.</td>
</tr>
<tr>
<td></td>
<td>Also, please make sure that postgres is started before the GODOT parallel server and your httpd server. See the ‘chkconfig’ man page for details.</td>
</tr>
<tr>
<td>YAZ</td>
<td>GODOT uses the Perl module Net::Z3950 (version 0.51) which in turn uses the YAZ Z39.50 programmers' toolkit. YAZ may be downloaded freely at <a href="http://www.indexdata.com/yaz/">http://www.indexdata.com/yaz/</a>.</td>
</tr>
<tr>
<td></td>
<td><strong>In order to install Net::Z3950 version 0.51 you will likely need to use an older version of yaz. Yaz version 2.1.24 has been used successfully.</strong></td>
</tr>
<tr>
<td></td>
<td>If you are installing from rpms, you will need both the libyaz and yaz packages.</td>
</tr>
<tr>
<td></td>
<td>Please note that if you find you are having problems installing yaz with one of the rpms, it may be because you do not have the required openssl or xslt support. GODOT does not require yaz with either xslt or SSL support, so your best bet may be to compile from source using ‘./configure --without-xslt’. SSL support is not configured by default.</td>
</tr>
<tr>
<td>CUFTS</td>
<td>GODOT queries CUFTS in order to provide links to full-text for it to display. CUFTS may be downloaded at <a href="http://researcher.sfu.ca">http://researcher.sfu.ca</a>.</td>
</tr>
<tr>
<td>Perl modules</td>
<td>A number of perl modules are required. The install script will check to make sure you have the required ones. They should all be easily available from CPAN.</td>
</tr>
<tr>
<td></td>
<td>You can use ‘perl util/install.pl –m’ to check for the necessary modules without running the full install.</td>
</tr>
<tr>
<td></td>
<td>A good first pass at installing these modules is to install ‘Bundle::CUFTS’. If you have ‘cpan’ installed, you can use ‘cpan Bundle::CUFTS’.</td>
</tr>
</tbody>
</table>
Please note that you want version 0.51 of Net::Z3950 not the more recent ZOOM versions. Download the 0.51 version using the link at the bottom of http://perlz3950.org/download/index.html and then install the module manually (ie. do not use the cpan installer).
Scripts
GODOT consists of four CGI scripts run by your apache/mod_perl server, one standalone process and one utility script:

<table>
<thead>
<tr>
<th>Table 2: GODOT scripts</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>hold_tab.cgi</strong></td>
</tr>
<tr>
<td><strong>rw_export.cgi</strong></td>
</tr>
<tr>
<td><strong>config.cgi</strong></td>
</tr>
<tr>
<td><strong>sandbox.cgi</strong></td>
</tr>
<tr>
<td><strong>para_server.pl</strong></td>
</tr>
<tr>
<td><strong>update_cache.pl</strong></td>
</tr>
</tbody>
</table>
Values required by the install scripts

Before you run the install scripts you will need to decide on a number of values as follows.

| **User that will be running Apache/mod_perl:** | Will need write access to session, log and request number directories. For example, ‘apache’. |
| **User that will be running parallel server:** | Can be the same user as for Apache/mod_perl or different. Will need write access to the log directory. The way the parallel server is currently started requires that the user chosen must have a valid shell. This means that if your ‘apache’ user has the ‘nologin’ shell specified in ‘/etc/passwd’, you will have to use another user, such as ‘godot’ to run the parallel server. |
| **PostgreSQL user for configuration database:** | If this user does not already exist you will want to create it. If you are logged in as root, and root is not a postgres user, you will want to ‘su postgres’ before running ‘createuser’. |
| | su postgres  |
| | createuser --createdb --no-adduser godot  |
| | Shall … be allowed to create more new roles? (y/n) n  |
| | CREATE ROLE  |
| **Base directory:** | A location for the main installation to go. |
| | This directory tree will include session, log and request number files. The user that will be running Apache/mod_perl and the user that will be running the GODOT parallel server, will need to have write access to some of the subdirectories. The install script will configure the required permissions. |
| | Environment variable for the installation script is GODOT_ROOT_DIR.  |
| **Email address for admin messages:** | If you install the demo data (highly recommended), this address will also be used for the ILL system email address options, so you can see what ILL messages are being sent. |
| | Environment variable for the installation script is GODOT_ADMIN_EMAIL.  |
| **Directory for GODOT package CGI files:** | This directory needs to be web accessible and CGI enabled. This directory will be created as a symbolic link to the main installation directory. **You need to provide a directory name that doesn’t currently exist.** For example if you have directory: |
| | /usr/local/apache/htdocs  |
| | you might specify  |
/usr/local/apache/htdocs/GODOT

But GODOT must not already exist.

Environment variable for the installation script is GODOT_CGI_DIR.

Directory for GODOTConfig CGI files: Same as above, but specify a different directory. For example:

/usr/local/apache/htdocs/GODOTConfig

Environment variable for the installation script is GODOT_CONFIG_CGI_DIR.

URL that will be used for the hold_tab.cgi file: For example, if your root document directory is

/usr/local/apache/htdocs

and you have specified ‘/usr/local/apache/htdocs/GODOT’ as your GODOT_CGI_DIR, then you would likely be using something along the lines of

http://myserver.sfu.ca/GODOT/hold_tab.cgi

but this will depend on your apache and machine configuration.

Environment variable for the installation script is GODOT_URL.

URL that will be used for the config.cgi file: Same as above but for the GODOTConfig package. For example,

http://myserver.sfu.ca/GODOTConfig/config.cgi

Environment variable for the installation script is GODOT_CONFIG_URL.

URL for CUFTS server: OPTIONAL. Default values is the SFU CUFTS server. The SFU CUFTS server URL is provided for demo purposes only. It will work for the demo site profiles that you have the opportunity to load as part of the install. You will however, at some point, want to install your own version of CUFTS or inquire about having a hosted copy installed at SFU.

Environment variable is GODOT_CUFTS_SERVER_URL.

URL for citation manager server: OPTIONAL. Default value is SFU Citation Manager server. The SFU Citation Manager URL is provided for demo purposes only.

Environment variable is GODOT_CITATION_MANAGER_URL.
Installation environment variable files

Some of the values required by the installation scripts can be saved as environment variables which will minimize the amount of typing required when the install script is run and is especially useful if you need to run the script more than once.

If you are using the C shell (‘csh’):

```
cd /usr/local/godot
vi util/install_enviro_csh
source util/install_enviro_csh
```

If you are using the Bourne (or ‘Bourne-Again’) shell (‘sh’ or ‘bash’):

```
cd /usr/local/godot
vi util/install_enviro_sh
source util/install_enviro_sh
```
Running the software install script

Now you are ready to run the install script. **It is easiest if you are running this as root.** If not, there may be some parts of the installation process for which you will need the help of your system administrator.

```
cd /usr/local/godot
perl util/install.pl
```

Remember to explicitly run install.pl with perl. Otherwise you will get the following message if perl is located at a different location on your machine than what is specified in the install script.

```
util/install.pl: Command not found.
```

The install script displays information that you will need after the script is complete in a box made of ‘#’ characters. You will want to note this information as you will need it later.

Some of the questions that you will be asked by the install script follow.

Note that in the following ‘/usr/local/godot’ will be replaced by whatever you have specified for GODOT_ROOT_DIR.

**Do you want to copy the GODOT tree to /usr/local/godot?:** Answer yes if this is your first time installing GODOT.

A copy of /usr/local/godot/GODOT/lib/GODOT/BasicConfig.pm.backup already exists. **Proceed?:** Answer yes if this is your first time installing GODOT.

**Do you want to use the current or the installation values as the default? [C/i]:** If you are installing GODOT for the first time, then you will likely want to answer ‘i’ for ‘use installation values’.

The next set of prompts allows you to change the default settings for the values in /usr/local/godot/GODOT/lib/GODOT/BasicConfig.pm. If you are installing GODOT for the first time, you will probably want the default values.

**Do you want to update the configuration file with the new settings?:** Unless you have made mistakes, you will likely want to answer yes.

**Would you like to set the directories to world writable?:** This will be asked if you are not running as root (and therefore can’t change directory permissions so they are owned by the user running the web server). If you are unsure, ask your system administrator.
**Do you want to configure the web tree?:** You will likely want to answer yes if this is your first time installing GODOT.

You are now finished installing the GODOT package. Next you will be prompted to install the GODOTConfig package. The questions will be similar to those described to above.

Before continuing, please make a note of the information in the ‘#’ box. You can edit the Apache configuration file after this install script is finished.

```
# The file '/home/kristina/godot/util/godot_httpd.conf' has been created. Copy the following line into your Apache config file:
# Include /home/kristina/godot/util/godot_httpd.conf
```

**Would you like demo data for the site specific templates, CSS files and modules to be installed:** Again it is recommended that you choose this option if you are installing GODOT for the first time.

At the end of the install script, a URL for the Installation Test page will be displayed. Make a note of this. For example:

```
# The Installation Test Page should be found at  
# http://proxy2.lib.sfu.ca:8777/GODOT/index.html
```
Running the database install script

This will be easiest if you can run as the postgresql administrative user, usually ‘postgres’. If not, there may be some parts of the installation process that for which you will need the help of your system administrator.

In order to load the demo data (recommended), you will need write access to the directory ‘[base directory]/util/distrib/sql’ and the file in it, namely ‘demo_data.sql’. Please change permissions as required before proceeding.

```
    cd /usr/local/godot
    su postgres
    perl util/install_db.pl
```

Directory installation complete. Would you like to install the database?: After the GODOTConfig package has been installed, you will be asked if you want to install the database. Answer yes if this is the first time installing GODOT.

Initialize the database with demo profiles?: You will be asked whether you want to install the demo profiles, templates, css files and modules. It is highly recommended that you do so, if you have never used GODOT before. The demo configuration is referred to a number of times in this document and will allow you to get your installation of GODOT working faster.

A copy of util/distrib/sql/demo_data.sql.backup already exists. Proceed?: Answer yes if this is your first time installing GODOT.
Updating the profile cache
The public interface to GODOT uses profile cache files instead of making direct calls to
the GODOT configuration database.

The ‘update_cache.pl’ script should be used to update the profile cache after installing the
demo profiles.

It can also be used to update the cache anytime modifications are made to the profile
database outside of the GODOT configuration tool.

**If possible you will want to run this step as root.** Otherwise, you may need the help of
your system administrator.

    perl util/update_cache.pl
After installation

1. After the install script has been run, you will need to edit your Apache `httpd.conf` file and add the ‘Include’ line as displayed in one of the ‘#’ character boxes when you ran the install script.

   Include [base directory]/util/godot_httpd.conf

   Alternatively, you can add the ‘godot_httpd.conf’ file to the httpd configuration directory.

   cd [base directory]
   cp util/godot_httpd.conf /usr/local/apache/conf.d/

2. If you have installed the GODOTConfig database as a different postgres user than you will be running it, you will need to modify access:

   vi util/grant_rights.sql

   and then run it:

   su postgres
   psql -d GODOTConfig
   \i util/grant_rights.sql

3. You will also need to create a start up file for the parallel server. You will need to be root to do this.

   cd /etc/init.d
   cp /usr/local/godot/util/godot_para_server .
   vi godot_para_server

   Edit as necessary, including the constants at the top and the user for the ‘su’ statement.

   After that, update the run level information for system services, ie. ‘# chkconfig: 345 75 80’.

   You will want to make sure that postgresql starts before ‘godot_para_server’.

   Finally, you will want to make sure that the user running ‘para_server.pl’ has write permission for the parallel server log file (ie. ‘LOG_FILE’ in the ‘godot_para_server’ start up script). You may want to implement this by making a unix group containing both the user that will run apache and the user that will run the parallel server.

   vi /etc/group
   cd /usr/local/godot/GODOT/logs
chgrp godot .
chmod 775
chmod g+s
ls -lad

Then you can run ‘chkconfig’:

chkconfig --add godot_para_server
chkconfig godot_para_server reset

You can check that your ‘chkconfig’ configuration is correct with the following:

chkconfig --list | grep godot_para_server

4. You will also need to start up the parallel server

   cd /etc/init.d
   ./godot_para_server start

   To check that the parallel server is running:

   ps -fu[user]

   If it is not running, then you will want to look at the parallel server logs:

   more /usr/local/godot/GODOT/logs/godot_para_server.log

5. In order to run GODOT you will need to restart apache. If /etc/init.d/apache_perl
is your start up script:

   cd /etc/init.d
   ./httpd stop
   ./httpd start

6. If everything has gone OK, GODOT should now be up and running.

   At the end of the install script, a box was displayed with the URL of an
installation test page. For example:

   http://myserver.sfu.ca/GODOT/index.html

   Open this page up, and check to see that your installation is working. GODOT’s
main features are explained in the next section.

7. If you are having problems, it may be helpful to look at the apache server logs and
the GODOT parallel server logs.
cd /usr/local/godot/GODOT/logs
tail -f godot_para_server.log
tail -f /usr/local/apache/logs/error_log

8. When GODOT is running a number of temporary files will accumulate over time. You will want a job that runs once a day to delete any of these files that haven't been modified in the last couple of hours:

/tmp/*.cookie
/tmp/*.patron
/tmp/*.refworks
/tmp/*.output
/usr/local/godot/GODOT/sessions
/usr/local/godot/GODOT/sessions/lock
/usr/local/godot/GODOTConfig/sessions
/usr/local/godot/GODOTConfig/sessions/lock

9. You will also want to rotate the following logs:

/usr/local/godot/GODOT/logs/godot_para_server.log
/usr/local/apache/logs/access_log
/usr/local/apache/logs/error_log
Installation testing page

The following section uses the demo data included in the distribution as examples. To follow along, go to the test installation page which was displayed in a box at the end of the installation script. For example:

http://myserver.sfu.ca/godot/index.html

GODOT Installation Testing

Test the public interface:
- Example Citations
- Blank Citation Form

Test the configuration tool.

Figure 1: Installation Testing page

Try the ‘Example Citations‘ link. If you choose YOUR GODOT and have installed the demo, you will see the main GODOT page.
Figure 2: Main screen

Full text links

The first links on the pages are for full text. If you installed with the default values, the links will have been retrieved from the SFU CUFTS server. The links are prefixed by our ezproxy URL so you will not be able to actually follow them to the full text. The content of the links is controlled by CUFTS. If you choose the ‘show options’ link then you will see a list of links to the same resource but at different levels (eg. Article, TOC, journal).

A Z39.50 search of the SFU catalogue also took place and retrieved anything found in an 856 link for this title. We have configured our 856 links to only show if no links from CUFTS have been retrieved so we do not see these links. This was configured by editing site specific templates.
We will now skip down past the next two items until we are at ‘Check the web for this item’.

Check the web for this item

We have configured a link to Google using the article title if available and if not, using the title of the item. This link to Google was created using the template system which is used for all the GODOT screens. Links to other services could easily be created in the same manner (eg. Amazon, Teoma).

Export citation/reference

Links are provided to Refworks and Citation Manager. Citation Manager is another component of the reSearcher suite and is available at researcher.sfu.ca.

Figure 3: Citation Manager Login
Figure 4: RefWorks Login
The link to Refworks is actually to the rw_export.cgi script which was installed as part of the GODOT package. This script saves the citation in the RIS format to a file in ‘/tmp’. It then passes RefWorks a URL it can use to retrieve the file.

OK, now let’s go back to looking at the results of the search for print holdings.

Print Holdings

A Z39.50 search of the SFU catalogue took place looking for print holdings. If you choose ‘SFU Burnaby’, you will see our print holdings. SFU is configured to search only our local catalogue on this first screen, but it is also possible to configure GODOT to search multiple catalogues on the first screen.

The next link ‘Search other libraries / Request this item’, is configured to do a broadcast search of a number of other catalogues in the ELN and COPPUL consortia. In the case of the demo configuration, only a few libraries are searched.
You can configure this second screen so that a user selects the library from which they want to borrow or so that GODOT chooses the library. This second choice is called ‘Auto Requesting’ and is how SFU is configured. In the case of this citation, full text is available so it is unlikely that a user would want to place an ILL request. However, for the sake of this overview, we will assume that we do.

It is also possible to configure GODOT to block requesting if full-text is found or local print holdings are found. There are several different blocking behaviours that can be configured.

Select ‘Request this item Now’.
Patron Authentication and Retrieval of Patron Data

The configuration of SFU’s production system requires that patrons authenticate themselves against our ILS’s patron database before they place an ILL request. A patron database from a source other than an ILS could also be used.

![Patron authentication before placing an Interlibrary Loan request](image)

**Figure 7: Patron authentication before placing an Interlibrary Loan request**

The data in the patron record can also be used to populate the ‘Borrower information’ section of the request form.

The installed demo configuration does not include any authentication, so we go straight to the request form.
After you choose ‘Send Request’, you will either be prompted to confirm your order or just get an acknowledgment that the order is placed. SFU is configured to not require a confirmation, so you just get the acknowledgement screen.

Now go and check whatever email address you specified to the install script just before it installed the demo data. In the install script the prompt was ‘What email address do you want to use for the lending and borrowing email options for the demo profiles?’ For the purposes of this demo, SFU is configured as an OpenILL system site and is configured to send the patron an acknowledgement email. You should see two email
messages. Normally, the first email message would be sent to an email address that is automatically checked by your ILL system.
Configuration Tool

Log in to configuration tool from GODOT Installation Testing page. The first time you login, use ‘admin’ as your ‘login’ and the password that was typed in during the installation procedure.

Administration Main Page

From the GODOT Administration Page, you will see a menu bar providing access to all of the necessary configuration options.

![Menu](image)

**Figure 10: Menu**

**Change Site** allows you to switch to a particular site. Until you do this you will only be able to modify global items.

**Global Templates** allows you to edit the default templates.

**Global Style Sheets** allows you to edit the default style sheets.

**System Administration** allows you to add sites and accounts. Sites refer to a library or library branch. Accounts are for people doing the configuration at each library. Each site can have one or more accounts associated with it.
**Account Administration** allows you to edit the account details of the account with which you are logged in.

If you select **Change Site** then the sidebar will expand to include some site-specific choices.

![GODOT Menu](image)

**Sandbox** allows you to see what your template changes will look like without running GODOT. This is particularly useful for testing the main screen since it may take a few seconds due to the broadcast Z39.50 search.

**Configuration** allows you to configure a number of options for the current site.

**Templates** allows to edit the templates for the current site.

**Style Sheets** allows you to edit the style sheets for the current site.

**Site Administration** allows you to edit the name and the template chaining for the current site.
Adding a new site

1. Sign on with the login 'admin' and whatever password you entered during the install procedure.
2. Select 'System Administration'.
3. Select 'Sites'.
4. Choose 'New Site'.

![Table of Sites]

<table>
<thead>
<tr>
<th>key</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>view</td>
<td>BVAS</td>
</tr>
<tr>
<td>edit</td>
<td>BVASB</td>
</tr>
<tr>
<td>delete</td>
<td>BVASS</td>
</tr>
<tr>
<td>view</td>
<td>AEU</td>
</tr>
<tr>
<td>edit</td>
<td>BVAU</td>
</tr>
<tr>
<td>delete</td>
<td>BVAU-EL</td>
</tr>
<tr>
<td>view</td>
<td>BVAU-KRNR</td>
</tr>
<tr>
<td>edit</td>
<td>BVIV</td>
</tr>
<tr>
<td>delete</td>
<td>MVUC</td>
</tr>
</tbody>
</table>

5. The key should not contain any spaces. We have generally used the site's National Union Code (eg. BVAS for SFU) but it doesn't have to be, and in some case can’t be, because a branch library that you want to set up as a separate GODOT site may not have its own National Union Code (or your country’s equivalent).

After you have created a new site, you will need to change to that new site before you can configure it. Use the 'Change Site' option in the sidebar.
Configuration Options

Overview of retrieving holdings from a new site

First you have to determine the catalogue from which you are going to retrieve the holdings. This may be a union list or a single or multiple site catalogue. If the catalogue target is not already configured, then this needs to be done first.

If the catalogue target is associated with the current site then you will need to configure the 'site catalogue' groups. If the catalogue target is associated with another site, then this site will need to be added if this has not already been done (see Adding a new site, p. 32). After that the 'site catalogue' configuration groups of this other site will need to be configured (see Configuring a catalogue target, p.33).

Some examples of site holdings and site catalogue relationships may be useful.

Example #1: The simplest case is where there is a one to one relationship between the site holdings and the site catalogue. For example, at the University of Winnipeg, only one GODOT site is configured and holdings are retrieved from their own catalogue.

Example #2: SFU is a little less straightforward. We have three GODOT sites, one for each of our branches. They are BVAS (Burnaby campus), BVASB (Downtown campus) and BVASS (Surrey Campus). Only BVAS's 'site catalogue' options are configured. Both BVASB and BVASS point at the BVAS catalogue for holdings. The logic that separates out the holdings records is contained in site specific modules for BVAS. See 'Custom modules: Using one catalogue for multiple GODOT sites' (p. 65).

Example #3: Another example is some of the libraries that belong to the ELN consortium, a group of British Columbia post secondary libraries. For journal holdings a number of these libraries are configured to use holdings from the ELN Serials Catalogue for the brief display (ie. on the main screen). GODOT at the ELN is managed centrally, so it made sense to use a union catalogue since then there was only one target to set up and troubleshoot. Some of the smaller ELN libraries have limited systems support and are not able to easily set up, nor troubleshoot, Z39.50 access on their catalogues. Again, the logic to map the locations codes found in the ELN Serials Catalogue to their GODOT site equivalents is done using custom modules.

Configuring a catalogue target
Before you can configure holdings for a site, you first need to configure a catalogue target.

From the configuration tool:

1. Select 'Change Site' on sidebar, if necessary.
2. Select 'Configuration' on sidebar, if necessary.
3. Select 'Site Catalogue', to display the two associated configuration groups.
4. Select 'general catalogue'.
5. The 'Target name' will be something like 'SFU Catalogue'.
6. If you do not see your system type listed then you can either use 'Other' if your catalogue requires little or no special handling. Otherwise, you will want to add a new catalogue type. See 'Custom modules: Adding a new catalogue system type' (p. 65).
7. Choose 'Submit'.

![Figure 13: Z39.50 Configuration](image)

8. Choose the 'z3950' configuration group. This assumes that the primary means of searching for holdings in your catalogue is Z39.50. If not, a couple of catalogue specific modules will have to be written to override the default Z39.50 ones. Contact researcher-support@sfu.ca for further information.
9. Select 'Use Z39.50 to search' and then configure 'Host', 'Port' and 'Database name'.
10. Most catalogues do not have access restricted, but if yours does, configure 'ID' and 'Password'.
11. If your catalogue has a journal index which is Z39.50 accessible then configure 'Journal title use attribute'. The value for this is often '33' but you will want to check with your catalogue administrator. A journal index improves holdings retrieval.
12. For a first pass, this is probably all the Z39.50 options that you need to fill in.
13. Later you may want to adjust the searches by configuring the other options.
14. If you have access to a Z39.50 client in which different attribute combinations can easily be configured, this can be quite useful while you try to optimize your
Z39.50 configuration. As part of the GODOT installation, the YAZ programmers' toolkit was installed. If you have access to the command line of your GODOT server you can use 'yaz-client' for testing. See 'man yaz-client'.

15. Submit

Next, you will want to configure holdings retrieval and display using the 'site holdings' options.

Configuring site holdings

1. Select 'Change Site' on sidebar, if necessary.
2. Select 'Configuration' on sidebar, if necessary.
3. Select 'Site Holdings'.
4. Set 'Are there holdings associated with this site?'
5. Fill in 'Site name to appear next to holdings'. Probably a short version of your site name (eg. SFU-Burnaby).
6. For now, don't select either 'No holdings statement ...' or 'No item records ...'. You can adjust these as necessary later.
7. If you are using a different catalogue for retrieving holdings for the brief display (ie. the main screen) as opposed to the 'detailed holdings' display, then configure the 'Catalogue for journal for brief display' and 'Catalogue for non-journals for brief display' accordingly. For instance, in the case of the ELN libraries in Example #3 from above, they would specify 'ELN-AG' as that is the GODOT site profile name for the 'ELN Serials Catalogue' in the SFU copy of GODOT.
8. If the default catalogue is different, then configure 'Default catalogue'. For instance, in the case of SFU (Example #2) above, the two previous options are left blank, but 'Default catalogue' is configured as 'BV AS' for our 'BVAS' and 'BVASS' sites.
9. We did not configure all the site holdings options on this first pass. After you have seen how the holdings statements and item/circulation records display, you
can adjust the options accordingly. See 'Troubleshooting and modifying new site holdings' (p. 38) below.

10. Submit.

Next you will want to test your configuration by adding this site’s holdings to the list of holdings that your site wants to search.

**ERIC and Canadian Research Index Fiche Holdings**

If the citation includes an ERIC number or CRI (aka Microlog) number, then in addition to holdings retrieved from catalogue searches you can also configure a site to include ERIC and CRI fiche statements. Type your holdings statement for these fiche collections in the appropriate box and then turn the retrieval of these statements on by choosing ‘yes’ for **You own the ERIC microfiche collection** or **You own the Canadian Research Index microfiche collection** options.

![Figure 15: Collections Configuration](image)

**Figure 15: Collections Configuration**
Searching for local holdings

In order to display your own holdings or those of another site on your main GODOT screen, you will need to configure the 'holdings screen' configuration group options.

1. Select 'Change Site' on sidebar, if necessary.
2. Select 'Configuration' on sidebar, if necessary.
3. Select 'Sites that Use GODOT'.
4. Select the 'holdings screen' configuration group.
5. Choose 'configure' next to 'Ranking for journals'.
6. Under the 'Site' column, choose the site associated with the 'site holdings' configuration from above. In the case of local holdings this will be your site.
7. For 'Display Group' and 'Search Group', choose '1'.
8. For now, select 'show' option for the 'Auto Requesting' column. See 'Enabling Auto-Requesting (including SHOW and HIDE holdings)' (p. 44) for details of this feature.
9. Select where in the list you want to add the new holdings. For now, add your site somewhere near the top since we want to see the holdings on the first iteration of the main screen.

![HOLDINGS SCREEN CONFIGURATION](image)

**Figure 16: Holdings Screen Configuration**
10. Select 'add before' or 'add after' as appropriate.
11. Now do the same for 'Ranking for non-journals'.
12. For now, do not configure any of the other options. Later, see ‘Configuring GODOT to search other libraries’ (p. 39) for details on some of these options.
13. Submit

You should now be ready to test that your local holdings are being properly retrieved and display.

Testing the new site's holdings

1. Choose a journal and non-journal that you know your site has and make a note of the title, ISSN and/or ISBN.
2. Go to your 'GODOT Installation Testing' page. The URL for this page was printed out at the end of the installation process.
3. Choose 'Blank Citation Form'.
4. The field names are the same as used in OpenURL 0.1. The field 'atitle' is article title and 'aufirst' and 'aulast' are for the first author.
5. Choose your genre and then enter your title, ISSN or ISBN.
6. Submit.

Troubleshooting and Modifying new site holdings

**Problem #1:** 'All I am seeing is a 'holdings are available' message. Where are the holdings and item records?' If in your catalogue is ENDEAVOR, III or SIRSI then the logic for extracting the holdings and item/circulation records has been worked out and the related modules written. The extraction logic for the other systems has not yet been written. In the case of these systems, only the bibliographic record is retrieved (using Z39.50 USMARC record syntax).

If your catalogue returns holdings and circulation/item information via Z39.50 then it should be relatively straightforward to adapt one of the existing modules for ENDEAVOR, III or SIRSI to your catalogue system. See ‘Custom modules: Writing catalogue system modules’ (p. 65) for more information. If Z39.50 is not available, it is also possible to write modules to retrieve holdings and item/circulation data using another method. Email researcher-support@sfu.ca for more information.

If your catalogue is SIRSI and you are not getting holdings or circulation/item information, it is likely because your Sirsi Z39.50 server is not configured correctly. Please see: 'http://web.mala.bc.ca/library/marlin/docs/circ_holdings_z39.50.htm' or contact Sirsi support.

**Problem #2:** The circulation/item records are too long for journals. I only want the holdings statement.
If you only want the holdings statement then you will likely want to choose the 'site holdings' configuration group and then select 'No item records for brief journal display'. However, please note that while an attempt is made to retrieve both a holdings statement and circulation/item records, not all catalogues (or their Z39.50 servers) provide both. In some cases it may not be clear from the display which is the holdings statement and which are the item/circulation records. If selecting 'No item records for brief journal display' makes all your holdings disappear, then your catalogue may only supply circulation/item records for your journals.

**Problem #3:** I have an ENDEAVOR catalogue. Not all my holdings statements are being retrieved. This is a problem with the Endeavor Z39.50 server (Endeavor problem ID: 63639). Specifically, the Z39.50 server should send all 866 tags (Enumeration/Chronology) to Z39.50 clients that use the OPAC-1 record syntax / OPAC Holdings Schema, however only the last of the tags is sent. One way to handle this problem, until it is fixed, is to just display the location of the holdings on the main screen and then have the 'Check Detailed Holdings' link go to the Endeavor web interface. If you only want the location in the holdings statement displayed you will have to create a site specific module to override the default behaviour. See 'Custom Modules: Handling Endeavor Holdings Statement Bug' (p. 65).

**Configuring GODOT to search other libraries**

This section discusses the settings that control HOW searching and borrowing takes place within GODOT.

One of the most important features of GODOT is its ability to allow for searching and requesting from other libraries. GODOT provides you with the ability to choose which remote libraries your users can search and request from, and how those requests will be sent.

To select your list of libraries for searching and requesting, go to the ‘holdings screen’ configuration group. See ‘Searching for local holdings’ (p. 37) for the basic procedure to add a site to the ‘Ranking for journals’ and ‘Ranking for non-journals’ options.

You will probably want to select your local library as the first choice, and any branches (if configured separately in GODOT) as second or third choices.

GODOT allows you to control whether a library appears on the initial GODOT screen or on the second GODOT screen. Most libraries will want to put only their local library on the first screen. To do this, set the ‘Display Group’ and ‘Search Group’ columns to 1 and 1.
If you want your branches, or any other library to appear on the first GODOT screen, set them to 1 and 1 as well.

**Figure 17: Holdings Screen Configuration**

All other libraries will by default appear on the second screen, and no further modification is required. The order of this list will reflect the order in which they will appear on the second GODOT screen (if they have holdings for the item in question).

Note: the ‘Auto Requesting’ column is discussed in ‘Enabling Auto-Requesting (including SHOW and HIDE holdings)’ (p. 44).

When holdings are found for any of the libraries you have just configured, they will appear on the GODOT screen, along with a link for patrons to make a request if this has been configured. To have GODOT, instead of a library patron, decide which library to request from see the ‘Enabling Auto-Requesting (including SHOW and HIDE holdings)’ section (p. 44).

This list was for searching for journal. Further down the ‘holdings screen’ configuration group page is another list for non-journal items. You can either set it up to match the journal list, or you can customize it for book searches.
The ‘Display Group’ and ‘Search Group’ options in combination with the screen templates should provide you with a number of options with regards to how you want libraries searched and their holdings displayed.

The default screen templates are set up for two screens with one grouping of sites on each page. However, there are many other options. For instance the first screen could have one grouping of holdings at your library’s different branches. A second grouping on that first screen could be for libraries in your city. A link on the first page could then result in the screen being run again, but this time for libraries in your state or province. From this screen a link could result in libraries outside of your state or province being searched.

Two other important fields on the holdings screen page are ‘Search sites not specified in journal ranking’ and ‘Search sites not specified in non-journal ranking’. They allow you to determine whether libraries beyond what you set up in the previous configuration list will be searched. These two options both use the ‘Groups of which your site is a member’ option (in the ‘general’ configuration group) to determine which sites are considered ‘other’ sites. The list should have been set up during installation to reflect the appropriate list of groups.

If you want to force users directly to an ILL form if no holdings are found in any of your chosen libraries, then select the ‘holdings screen’ configuration group and select the ‘If there are no holdings, then just skip to the next screen’ option.

Error Messages

If no parser is configured for a source in which you have a GODOT link, then this message (or the default) will be displayed. This is less of a problem now that many sources have OpenURL linking, however depending on the content of the OpenURL link it may still be necessary to add that source to the appropriate configuration file. See ‘Custom Modules: Adding GODOT links to new sources’ for further information (p. 65).

![Errors Configuration](image)

Figure 18: Errors Configuration
Requesting from other libraries (mediated, unmediated, etc.)

Now that you have selected your list of libraries for GODOT to search, and the order in which they will be searched and appear on the GODOT screens, you will need to determine how requests will be sent to those libraries.

The first options to decide upon are in the ‘blocking’ configuration group. The ‘Use instead of site code for blocking’ can generally be ignored. It would be useful in the instance where a site has none of its own holdings but is associated with a site that does. The ‘Full-text are holdings for blocking’ controls whether or not full-text items in your collection are considered ‘holdings’. Most libraries have this enabled.

![Blocking Configuration](image)

**Figure 19: Blocking Configuration**

The next step in setting up requesting in GODOT is to determine whether requests should go directly to the library with holdings (unmediated a.k.a. D for Direct), should go through your ILL department first (M for mediated), result in a configurable text message (I for Information), or be blocked (N for Not allowed).

By default, all libraries are set to N. If you want to change the default, go to the ‘Default journal request type’ section and select D, I, or N.

If you’d like to set individual libraries up with different choices, go to the ‘Type of journal requesting allowed’ and select ‘configure’.

From here you can select individual libraries and give them the designation you prefer. These choices will override any defaults you had set up previously for those libraries.

Remember to repeat this procedure for monographs further down the page with the ‘Type of non-journal requesting allowed’ and ‘Default non-journal request type’ options.
If no holdings have been found by the search, or holdings have been found but your item is not available (eg. missing issues), you can allow patrons to send a request for the item to your local ILL department. This type of request is considered by GODOT to be different than requests for which holdings have been found. The configuration options for a request without associated holdings are in the ‘request link’ configuration group.

The first few options control whether the patron is allowed to make a request without associated holdings at all. The ‘ILL request link text’ text appears under ‘Place an Interlibrary Loan request’ link on the main screen.

The next option in the ‘request link’ group is ‘Limit messages for each type of patron’. This option allows you to restrict this type of interlibrary loan to only some of your patrons. For instance at SFU, we do not allow undergraduates to place interlibrary loan requests unless GODOT has already found holdings. Such requests require more staff time and so are limited to SFU faculty and graduate students. The ability to limit requesting based on patron type requires, of course, that your site is configured to retrieve patron records and authenticate on them before a patron is able to place a request. For more information on setting this up, please see ‘Patron Authentication and Patron Data Retrieval’ (p. 48).
Enabling Auto-Requesting (including SHOW and HIDE holdings)

Another feature in GODOT is the ability to enable Auto-Requesting from other libraries. By default, GODOT will search the libraries you have configured and produce a list of those libraries with holdings for the item in question. GODOT places a request link next to each of these libraries with holdings. Patrons then select one of these links to place the request.

Auto-Requesting takes the responsibility out of the hands of the patron and allows GODOT to pick the library to send the request, based on the prioritized list of libraries configured earlier.

This can be a very effective feature, especially when combined with Direct requesting (as described previously). For example, with CISTI set to D, and if CISTI is your preferred library, and if CISTI has holdings for the item in question, the request will go with the click of the user’s mouse directly to the CISTI office with no involvement of your library whatsoever.

One important consideration with Auto-Requesting is that GODOT is unable to read detailed holding statements. If a requested article is in a missing issue, GODOT will send the order anyway (although it is questionable how many users actually read the detailed holding information, either).

![Auto Requesting Configuration](image)

**Figure 21: Auto Requesting Configuration**

To enable Auto-Requesting you will want to select the ‘auto requesting’ configuration group. To turn on Auto-Requesting for journals select the ‘Turn on automatic requesting feature for journals’ option.

The next option, ‘If auto requesting is turned on, should journal holdings be displayed?’ allows you to specify a default for whether the holdings are displayed on the main screen or not. It is also possible to override this default setting for individual libraries by using the ‘Ranking for journals’ option (Auto Requesting column) found in the ‘holdings screen’ configuration group.

Turning on Auto-Requesting for monographs is the same procedure except use the non-journal options instead.
The advantage of hiding all the remote holdings is that it provides a much simpler interface for the patron to work with. The GODOT screen will essentially consist of any full-text links, local holdings, and a single request button.

The disadvantage can be for patrons who would rather identify nearby holdings and pick them up in person rather than placing a request (e.g., SFU students driving across town to UBC).

**Lending**

If you have configured holdings for a site, then you can also configure the site profile to allow lending. First you will need to configure the message format that the site expects for lending requests from remote sites. ‘National Library Generic Script’ is a common choice for ELN/COPPUL library systems. These are all email message formats so you can see what the output looks like by configuring ‘Email address for lending requests’ with your own email address. See ‘Custom Modules: Writing new ILL system formats’ (p. 65) for details on adding new formats.

The **If lending & borrowing sites have the same NUC** option may be useful if you are setting up GODOT for a library with multiple branches, creating a GODOT site for each branch and these branches all use a central ILL system. For example, at SFU we have three branches but we have a central ILL system (and office) at our main branch. In our case we have configured the **If lending & borrowing sites have the same NUC** to only send to the borrowing system, so that borrowing between branches will only create one request in the system. The NUC field option can be found in the **borrowing message** configuration group.
Full-text Options

In order to enable the search and display of full-text links in GODOT you must select ‘Search for links in MARC 856 field’ and/or ‘Query CUFTS for full text links’.

If you want to query CUFTS, then you need to have configured a URL for these queries when you installed GODOT. Alternatively, you can edit [base directory]/GODOT/lib/GODOT/BasicConfig.pm and set ‘CUFTS_SERVER_URL’ appropriately.

![Full Text Configuration](image1)

**Figure 23: Full Text Configuration**

The next two options, ‘Be this site for MARC 856 searches’ and ‘Be this site for CUFTS searches’ allow you to do these searches as another site. This would generally be used in the case of a library that has multiple branches and each branch is configured as a separate GODOT site for ILL reasons, but all branches have access to the same electronic resources.

For instance, at SFU we only have CUFTS set up for one site, BVAS, which is our main library. The GODOT configuration profiles for Surrey (BVASS) and Downtown (BVASB) both are configured with ‘Be this site for CUFTS searches’ set to ‘BVAS’.
Display Options

Display parallel server error messages only: By default GODOT will display search status messages. If you do not wish to see these messages select this option. Error messages will still be displayed.

Use javascript popup window for messages: By default GODOT displays search status messages in a javascript popup window so you can see how your search is progressing. If you do not wish to see this, then select this option.

Display citation manager links: Selecting this option will provide a link to Citation Manager and/or RefWorks. Remove the check mark if you do not want this information displayed.

Display ‘Back to Database’ link: This will create a 'Back to Database' link on your GODOT screen whenever the GODOT screen is NOT generated from a popup window (i.e. within the same browser window as the original database).

Display Internet search engine links: Checking this option will turn on the Internet search engine links (i.e., Google, Teoma, etc.).

Expand fulltext by default: Selecting this option will result in the fulltext “other services” (i.e., journal, issue, database level links) to appear in full, rather than collapsed.

Expand holdings by default: Selecting this option will result in the detailed holdings information for library catalogues to appear in full, rather than collapsed.

URL for image for OpenURL link: Image for OpenURL link if it was enabled using the 'cookie pusher' method. Some vendors use this method, but most do not.

Figure 24: Display Configuration
Citation Enhancement

If a citation has a DOI and/or a PMID then GODOT can be configured to query CrossRef and/or PubMed to retrieve citation data.

Configuring citation enhancement is important if the links from your source to GODOT typically do not contain at least a title, ISBN or ISSN as without one of these fields GODOT will return an error message. Google Scholar and PubMed are examples of such sources.

The following options will allow you to configure citation enhancement.

---

Patron Authentication and Patron Data Retrieval

You can configure GODOT so that a patron needs to authenticate themselves against your patron database before they place an ILL request. You can also use any patron data returned by your patron database API to populate the Patron and Request Information screen. If you want to do this, you will need an API to your patron database which GODOT can query. If you have an III system then you can purchase their HTTP patron api. Alternatively you can write patron database API’s that conform to that API (III HTTP) or the the Default or Default HTTP APIs. For more information see ‘Patron Database API Specification’ (p. 65).

---

Figure 25: Patron API Configuration
If a PIN or some other second piece of information is required in order to authenticate, choose ‘PIN required’. This will result in the patron being prompted for a second piece of information. To modify the wording of either the primary identifier (i.e. Library card number) or this secondary piece of information, edit the config_request_form_fields template. See ‘Templates’ (p. 55) for further information about template editing.

You only will want to fill in ‘Allowed fines’ if your patron API passes back this information.

Putting Requesting behind a Site Password

If you want you, instead of requiring patron level authentication, you can put requesting behind a site wide password. The main reason to do this is to force patrons to see a staff member before placing a request. A site might only do this to limit one type of request (use Password required to request? option).

![Request Password Configuration](image)

**Figure 26: Request Password Configuration**

For example, at some of the smaller sites that use GODOT here in BC, they allow students to request journal articles themselves but want to limit requests for books and so require a staff member to approve the request.

In other instances, the book requests are handled by ILL staff after students have placed the request by another method. In these cases you may want to select the No check for 'required' fields if using a password option so that ILL staff do not have to input all the patron details normally required for by that site.

Patron and Request Information Screen

To configure the screen that prompts the user for patron and request related information (e.g. ‘need by date’) select the ‘request form’ group.

The first set of options allows you to specify whether the field should appear on the screen, and if it should, whether the information is required. The actual field labels are
configured by modifying the ‘config_request_form_fields’ template. See ‘Templates’ (p. 55).

The ‘Default text for 'library ID’’ option allows you to specify the characters that should be added to the start of the characters your patrons enter. This is useful, for instance, if all your library cards start with the same string of numbers. You will probably want to keep this consistent with the way your other interfaces ask for your library card numbers.

The next two options, ‘Allow user to edit the patron type’ and ‘Display patron type’ only apply if you have patron authentication configured and are retrieving patron data.

The next few options (i.e., ‘Patron type choices’, ‘Choices for department’) allow you to configure the drop down lists for the related fields. If you are retrieving patron records

---

### Figure 27: Request Form Configuration

<table>
<thead>
<tr>
<th>Field</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last name</td>
<td>Use and required</td>
</tr>
<tr>
<td>First name</td>
<td>Use and required</td>
</tr>
<tr>
<td>Library ID</td>
<td>Use and required</td>
</tr>
<tr>
<td>Patron type</td>
<td>Use and required</td>
</tr>
<tr>
<td>Need by</td>
<td>Use and required</td>
</tr>
<tr>
<td>Province</td>
<td>Do not use</td>
</tr>
<tr>
<td>Department</td>
<td>Use and required</td>
</tr>
<tr>
<td>Patron email address</td>
<td>Use and required</td>
</tr>
<tr>
<td>Pickup location</td>
<td>Use and required</td>
</tr>
<tr>
<td>Phone</td>
<td>Do not use</td>
</tr>
<tr>
<td>Work phone</td>
<td>Do not use</td>
</tr>
<tr>
<td>Building</td>
<td>Do not use</td>
</tr>
<tr>
<td>How to notify patron</td>
<td>Do not use</td>
</tr>
<tr>
<td>Street</td>
<td>Do not use</td>
</tr>
<tr>
<td>City</td>
<td>Do not use</td>
</tr>
<tr>
<td>Postal Code</td>
<td>Do not use</td>
</tr>
<tr>
<td>Rush requested</td>
<td>Do not use</td>
</tr>
<tr>
<td>Paid by</td>
<td>Do not use</td>
</tr>
<tr>
<td>Account number</td>
<td>Do not use</td>
</tr>
<tr>
<td>Note</td>
<td>Use but not required</td>
</tr>
</tbody>
</table>

**Default text for 'library ID':**

**Allow user to edit the patron type:**

**Display patron type:**

**Patron type choices:**

**Choices for department:**

**Pickup location choices:**

**Paid by choices:**

**Enable server-side caching of patron info:**

[submit] [cancel]
then you will probably not want to configure ‘Patron type choices’ as this information will likely be available in your patron record.

The ‘Enable server-side caching of patron info’ only works if GODOT is being passed some sort of session number from the database from which it was invoked. This is true only for some of the older non-OpenURL links to GODOT.

Confirmation and acknowledgement screens can also be set up on this page.

Once all of the changes have been made on the Borrowing page, select Update Configuration from the top of the screen. Please wait for the results of your changes to appear before doing anything else, to avoid locking the system.

Checking Patron Email Addresses

Some ILL departments may wish to restrict communication with a patron to the student’s local email address. You can force a patron to use a local email address by configuring the Perl regex for matching patron email address option to an appropriate regular expression. For instance at SFU we would specify ‘\.sfu\..ca$’ in order to limit email addresses to those such as ‘jdoe@sfu.ca’ or ‘jdoe@godot.lib.sfu.ca’.

![Figure 28: Patron Email Configuration](image)

Confirmation and Acknowledgment of Request

The next set of options, in ‘request confirmation’, allows you to enable a request confirmation screen as well as an acknowledgment screen and acknowledgment email to be sent to the patron. The text for the confirmation and acknowledgment screens can be configured in the screen templates. See Templates (p. 55) for more information.
Figure 29: Request Confirmation Configuration
Borrowing Request Messages

The ‘borrowing message’ configuration group contains options related to the request message that gets sent to your local ILL system and/or the request message that gets sent to the remote library.

### Figure 30: Borrowing Message Configuration

The ‘National Union Code’ is whatever unique ILL site code is used in your country.

The ‘Message format’ option controls the type and format of the message sent to your local ILL system. All but the RSS message formats are sent via email. The RSS messages are sent via an HTTP GET request. If you require another message type you can add one using a custom module. See ‘Custom Modules: Writing new ILL system formats’ (p. 65).

‘Email address’ and ‘Local ILL system host name’ are used as the destination for the email and HTTP GET message types respectively.

You will likely want ‘copy requests to other sites to your local system’ to be selected. This will allow you to keep track of what requests have been sent directly from GODOT to remote libraries by your patrons. This option only affects some of the older ILL systems. Newer systems such as RSS and OpenILL route both direct and mediated requests though the local ILL system, with the direct requests automatically being sent on to the remote library without any staff intervention.

You will also probably want to select the ‘Only “can borrow from” sites in holdings list’. This means that only sites that you have configured in GODOT as potential borrowers will appear in the lending list.

The ‘National Union Code’ is whatever unique ILL site code is used in your country.

The ‘Message format’ option controls the type and format of the message sent to your local ILL system. All but the RSS message formats are sent via email. The RSS messages are sent via an HTTP GET request. If you require another message type you can add one using a custom module. See ‘Custom Modules: Writing new ILL system formats’ (p. 65).

‘Email address’ and ‘Local ILL system host name’ are used as the destination for the email and HTTP GET message types respectively. You will likely want ‘copy requests to other sites to your local system’ to be selected. This will allow you to keep track of what requests have been sent directly from GODOT to remote libraries by your patrons. This option only affects some of the older ILL systems. Newer systems such as RSS and OpenILL route both direct and mediated requests though the local ILL system, with the direct requests automatically being sent on to the remote library without any staff intervention.

You will also probably want to select the ‘Only “can borrow from” sites in holdings list’. This means that only sites that you have configured in GODOT as potential borrowers will appear in the lending list.
The ‘Maximum cost to request’ and the ‘Configure account numbers’ options are currently only used in the CISTI message format. The ‘ILL department FAX number’ appears in the message note field. It is only necessary if you will be borrowing from libraries that do not already know your FAX number.
Templates

Overview

This section will allow you to change the wording, headers, and other functions of your GODOT screens. You will encounter some programming language in these templates, but you only need to concentrate on the text or html that you wish to change. No programming knowledge is required!

If you are interested in making significant changes to the default templates you should be able to do this as the Template Toolkit system is powerful and well documented. For more information about Template Toolkit, please see http://www.template-toolkit.org.

Editing Templates

To edit the site specific templates, choose the ‘Templates’ option on the sidebar. There is also a ‘Global Templates’ option which system administrators can use.

There are a number of template groups listed. If you cannot determine which template group you should look at to find the template you need please let us know.

<table>
<thead>
<tr>
<th>TEMPLATE GROUP</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>main screen layout</td>
<td>Headers, footers, etc.</td>
</tr>
<tr>
<td>main screen content</td>
<td>Main GODOT screen with citation display, fulltext and print holdings, ILL request</td>
</tr>
<tr>
<td>main screen miscellaneous</td>
<td>Helper templates, variables setup, etc.</td>
</tr>
<tr>
<td>patron information</td>
<td>Collect patron information screens: authentication and patron details</td>
</tr>
<tr>
<td>request information</td>
<td>Request information screens: acknowledgements, etc</td>
</tr>
<tr>
<td>warning screen</td>
<td>&quot;Your library has holdings&quot; warning screen for ILL requests where local holdings were found</td>
</tr>
<tr>
<td>incomplete citation form</td>
<td>Article form requesting extra article information if the citation is incomplete</td>
</tr>
<tr>
<td>SFU DocDirect passwords</td>
<td>Screens for entering password information when placing DocDirect requests</td>
</tr>
</tbody>
</table>

Figure 31: Template Groups

In general the templates that you will be most likely to want to edit are those that start with ‘config_’. They are generally used for adding just text to your screens and contain
little or no template toolkit code. The templates that end with ‘_screen’ control the content of a whole screen. There are also templates that end with ‘_title’.

For example, to change the ‘Check the web for this item’ heading or the resources listed under it, in the SFU templates, you would do the following:

<table>
<thead>
<tr>
<th>BASIC TEMPLATES</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TEMPLATE</strong></td>
<td><strong>ACTIVE</strong></td>
</tr>
<tr>
<td>config_auto_req</td>
<td>☑</td>
</tr>
<tr>
<td>config_help_url</td>
<td>☑</td>
</tr>
<tr>
<td>config_holdings_text</td>
<td>☑</td>
</tr>
<tr>
<td>config_no_fulltext_available_text</td>
<td>☑</td>
</tr>
<tr>
<td>config_no_holdings_text</td>
<td>☑</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INTERMEDIATE TEMPLATES</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TEMPLATE</strong></td>
<td><strong>ACTIVE</strong></td>
</tr>
<tr>
<td>config_catalogue_title</td>
<td>☑</td>
</tr>
<tr>
<td>config_main_holdings_title</td>
<td>☑</td>
</tr>
<tr>
<td>main_holdings_contact_us</td>
<td>☑</td>
</tr>
<tr>
<td>main_holdings_related_info</td>
<td>☑</td>
</tr>
<tr>
<td>main_holdings_screen</td>
<td>☑</td>
</tr>
</tbody>
</table>

**Figure 32: Templates**

1. First, make sure your current site is Simon Fraser University (BVAS). This will be the ‘Active site’ displayed in the top right hand side of the screen. If this is not your active site, you will want to use ‘Change Site’ in the sidebar to switch.
2. Next, choose the ‘Templates’ item in the sidebar.
3. Select the ‘main screen content’ template group. Find the template named ‘main_holdings_related_info’.
4. In the SANDBOX column, click on the middle Edit icon.

![EDIT TEMPLATE](image)

Figure 33: Editing Templates

5. At the top of the edit box is the ‘Check the web for this item:` heading. Further down are the resources such as Google that you might want to search. Links to other resources can easily be added.

6. Once you have made these changes, select ‘Submit’ from the lower right corner. This will load your changes to the Sandbox.

7. If you have screens configured in the Sandbox, you could now click on Sandbox from the left menu bar, select the appropriate main holdings screen, and the view your changes.

8. When you are satisfied with the changes, return to the Templates screen. From the “main_holdings_contact_us” row, click on the TRANSFER arrow. This loads your changes to your “live” GODOT screens.
Template for main screen

Another template you may want to make wording changes in is the ‘main_holdings_screen’. This contains much of the wording that appears on the main GODOT screen. This, in most cases, is the first GODOT screen, which if using the default setup, contains full text links and local holdings.

<table>
<thead>
<tr>
<th>INTERMEDIATE TEMPLATES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TEMPLATE</strong></td>
</tr>
<tr>
<td>config_catalogue_title</td>
</tr>
<tr>
<td>config_main_holdings_title</td>
</tr>
<tr>
<td>main_holdings_contact_us</td>
</tr>
<tr>
<td>main_holdings_related_info</td>
</tr>
<tr>
<td>main_holdings_screen</td>
</tr>
</tbody>
</table>

Figure 34: Templates

Templates for header and footer

Adding your own header and footer can be done in the ‘config_page_header’ and ‘config_page_footer’ templates. The easiest way to do this is to copy and paste from another document. You can also type directly into the edit box. These templates are found in the ‘main screen layout’ template group.

<table>
<thead>
<tr>
<th>BASIC TEMPLATES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TEMPLATE</strong></td>
</tr>
<tr>
<td>config_page_footer</td>
</tr>
<tr>
<td>config_page_header</td>
</tr>
</tbody>
</table>

Figure 35: Configuring Headers and Footers

Template for labels in the patron information form
To modify the labels (eg. First name, Last name, Library ID) that appear on the request information and patron information form, edit the ‘config_request_form_fields’ template. You will find this in the ‘patron information’ template group.

Templates for the request confirmation screen

request_confirmation_screen
config_confirmation_local_text (for mediated requests)
config_confirmation_text
config_request_confirmation_title
request_confirmation_text

These can all be found in the ‘request information’ template group.
Templates for request acknowledgment screen

request_acknowledgment_screen
config_acknowledgment_local_text (for mediated requests)
config_acknowledgment_text
config_request_acknowledgment_title
request_acknowledgment_text

These can all be found in the ‘request information’ template group.

Template Chaining

Most GODOT libraries have only one location and therefore only one set of GODOT templates. Some sites, however, do have multiple locations or branches (e.g., SFU Library has the Burnaby, Surrey, and Vancouver branches), each with its own set of templates.

While it is possible to configure each branch’s templates separately, allowing for complete individualization, most libraries want a common design for all of their GODOT screens. Template chaining allows you to configure one set of master templates, and "chain" the other branch templates to those. The means that you can configure and maintain one group of templates, and essentially forget about the rest.

To create a template chain, you will first need to configure one set of templates, usually for your "main" branch (e.g., SFU Burnaby). Once these are complete:

1. Use Change Site to configure one of your branches
2. Select Site Administration
3. Under General Settings, confirm all of the information is correct.
4. Under Template Chaining, enter 1 and then select the master branch. It is possible to chain through two sets of templates. This is done by adding a 2 and the name of a secondary branch.
5. Press Submit.
Sandbox

Overview

The Sandbox allows you to experiment with your GODOT templates and style sheets before making them ‘live’ for your patrons..

The way it works is you first capture the basic screen content with no formatting (ie. templates and style sheets have not been applied) and label it for future use. This saved data is called a sandbox object. You can then change the sandbox templates and style sheets and quickly check your changes. Since the basic content (eg. holdings search results) has already been retrieved and saved in this sandbox object, displaying a sandbox view of a screen is fast.

Please note that the sandbox tool is for checking template and style sheet changes only. Changes made to most of the configuration options will not appear in the templates unless you capture a new sandbox object. The exception to this is some ‘text only’ options which may be seen in their modified form without saving a new sandbox object.

Saving Sandbox Screens

1. Select ‘Sandbox’ in the sidebar.
2. Set ‘Save sandbox objects?’ to ‘save’.
3. Enter a descriptive name in the ‘Name for sandbox objects?’ box (eg. no_fulltext or local_holdings’)

![Saving Sandbox Objects](image)

Figure 37: Saving Sandbox Objects

4. Select ‘Submit’.
5. Open up a new browser window.
6. Run a GODOT search using any GODOT link you want. If none are set up yet, just use the ‘Example Citations’ or ‘Blank Citation Form’ links from the ‘GODOT Installation Testing’ page.
7. Go back to the configuration tool. RELOAD the page and you will see the new sandbox objects. If you want to capture a sequence of screens you can do this without using a new sandbox object name. This is because the screen name is added to the front of the name you entered.

8. Click on any of the objects to see how your sandbox templates and styles sheets look. When you are satisfied make your templates and style sheets live. See ‘Templates’ (p. 55) and ‘Style Sheets’ (p. 64) for instructions on doing this.

Troubleshooting the Sandbox

**Problem #1:** You have enabled ‘Save sandbox objects?’ and entered a name for the object, but no object appears to be being saved. First, make sure you have selected the ‘submit’ button and go back and run GODOT again. Then reload the sandbox screen.

If you still don’t see your new sandbox objects, then check to see the hostname that appears in the URLs for the configuration tool and for GODOT. If they do not match (eg. one is an alias) then the problem is that GODOT cannot read the cookie created by the configuration tool. To fix this, use URLs with the same host name for both the configuration tool and GODOT.
Style Sheets

This section allows you to change the colours, font sizes, font faces, etc. in your GODOT screens. It works the same way as any other Cascading Style Sheet in HTML.

For example, to change the background colour of your GODOT screens:

1. Select Style Sheets from the left menu bar.
2. Click on the middle Edit icon in the SANDBOX column.

3. Within the edit screen you will see something like:

   ```
   body {
     background: #D0DEED;
     font-family: Arial, sans-serif;
     margin: 0px;
   }
   ```

4. Change the hexadecimal colour code (#D0DEED) to anything you wish (e.g., #ffffff is white).
5. Select the “Submit” button from the bottom right corner.
6. Go to the Sandbox and view your changes.
7. If you are happy with your choices, return to the Style Sheets section and select the TRANSFER arrow to make your changes “live” for all to see.
Custom Modules

Under development.