Islandora as an Open Source Institutional Repository Solution

Islandora can provide a robust and full-featured institutional repository solution. The following documentation inventories the range of modules and configurations that enable core institutional repository functionality in Islandora.

Content Models for ETDs and Archived Publications

Islandora Scholar Solution Pack

The Islandora Scholar Solution Pack is a suite of modules that provides the foundation for building your institutional repository in Islandora. The modules that comprise this solution pack:

- Define the Citation Content Model and Thesis Content Model (and associated default forms)
- Generate meta tags to meet requirements for indexing in Google Scholar
- Permit content embargoes
- Provide links on Citation and Thesis objects to search for them in Google Scholar

Batch Import

Islandora DOI Populator, Islandora XML Form Builder, Islandora Webform Module

DOI can be added to a metadata entry/ editing form as an identifier field. See Islandora XML Form builder. There is a DOI batch ingest module that allows a user to upload a .txt file of comma separated DOIs or compile a list of DOIs in a GUI for ingest. Some community members have built custom workflows that mint DOIs when a research object is published or embargoed.

Author Profiles and Gallery

Islandora Scholar Profiles Module

The Islandora Scholar Profiles module uses Person and Organization entities to support the creation of individual author profiles. Profiles can display biographical information drawn from MADS metadata and a list of all items in the repository that have an association with the author. The module also automatically generates a Scholar's Directory page that displays a searchable, sortable, filterable list of all authors in the repository. The Scholar Profiles Module:

- Adds Person and Organization (department) collections/content models
- Provides forms for creating new scholar and department entities
- Provides basic template for scholar profile page
- Provides taxonomy for classifying research interests
- Provides form for scholars to request updates to their profile

Visits and Downloads Statistics and Automated Readership Reports

Islandora Matomo Modules

The ISLE Matomo Docker image and Islandora Matomo module and optional plug-in capture usage statistics and automatically generate author dashboards and real-time usage maps. Matomo can be configured to collect information for individual authors, departments, publications or other segments of your repository. The module also generates automatic, monthly readership reports that can be sent to authors via email.

ETD Workflows Including Self-Deposit

Islandora Scholar and Islandora Webform Module

The Islandora Scholar suite of modules provides support for thesis/dissertations and citations. ETDs can be batch ingested via DOI, PMID, EndNote, RDF, etc. The Islandora Webform Module supports self-deposit workflows.

Data Portability

Islandora Metadata Export

The Islandora Metadata Export module automatically creates an embeddable block with metadata downloads in a range of formats, including BibTex, EndNote, MARC, MARCXML, DublinCore, RIS, and JSON.

SEO and Google Scholar Indexing

Search engines like Google, GoogleScholar, and Bing likely drive significant traffic to your institutional repository. In order to return relevant search results, search engines crawl and index site content. Repository administrators can implement a few key practices to ensure that search engines properly index their sites. Proper indexing in search engines will improve your repository's visibility on the web, driving more traffic to your site, and encouraging future deposits. The following practices promote efficient site indexing, making your content more discoverable to crawlers while reducing strain on your site.

Islandora Scholar Solution Pack

Google Scholar requires a browse interface that allows its search robots to discover the URLs of articles in your repository. The Islandora Scholar module automatically generates the following views of your content, as recommended by Google Scholar.

- **For small collections (less than one thousand papers)**, e.g., papers written by a single author or a small group, all articles are listed on a single HTML page, such as www.example.edu/~professor/publications.html, including links to the full text PDF.
- **For medium sized collections (thousands of papers)**, articles are grouped by date of publication or the date of record entry. Other forms of browse interfaces, such as browse by author or by keyword, often generate more URLs than your website can deliver to the search robots in a reasonable amount of time. This view appears at yourdomain.edu/gs_years.
- **For very large collections (over one hundred thousand papers)**, an additional browse interface lists only the articles added in the last two weeks. This smaller set of webpages can be recrawled more frequently than your entire browse interface, which will facilitate timely coverage of your recent papers by the search robots. This view appears at yourdomain.edu/gs_updated.

Google Scholar recommends that the URL of every article is reachable from the homepage by following at most ten simple HTML links. Some institutions report success with placing hidden links to the above landing pages on prominent pages in their repository.

The Scholar solution pack also adds meta tags to citation and thesis object pages to be crawled by GoogleBots. Tags are also recognized by Zotero.

Customizing your repository's look and feel can make it harder for search engines (and other tools and services such as Zotero, Connotea and SIMILE Piggy Bank) to correctly identify item metadata fields. To address this, the Islandora Google Scholar Module automatically places item metadata in the element of each item's HTML display page.

Dublin Core Metadata

```
<meta name="DC.type" content="Article" />
<meta name="DCTERMS.contributor" content="Tansley, Robert" />
```

If you have heavily customized your metadata fields away from Dublin Core, you can modify the crosswalk that generates these elements by modifying [islandora]/config/crosswalks/xhtml-head-item.properties.

Google Scholar Metadata

Google Scholar requires publications to have title, at least one author, and a publication date for inclusion. The Islandora Google Scholar Module includes these required fields in each item's HTML display page.

```
<meta content="Tansley, Robert" name="citation_author" />
<meta content="Donohue, Tim" name="citation_author" />
<meta content="Ensuring your DSpace is indexed" name="citation_title" />
<meta content="2018" name="citation_publication_date" />
```

Tip:

- If you have heavily customized your metadata fields, or wish to change the default "mappings" to these Highwire Press tags, they are configurable in [islandora]/config/crosswalks/google-metadata.properties.

Islandora JSON-LD Module

The Islandora JSON-LD Module creates a standards-compliant JSON-LD record for an object based on a set of predefined MODS XPaths. Metadata structured as JSON-LD is increasingly important for dataset discovery through services like Google’s Dataset Search.

Islandora XML Sitemap module

The XML Sitemap module allows Islandora to expose its content in a way that search engines can easily crawl. Sitemaps allow crawlers to index your site without having to visit every page in your repository, meaning they can dow their work more quickly and efficiently). Without sitemaps, search engine indexing activity may significantly tax your repository. The Islandora XML Sitemap module works in conjunction with the Drupal XML Sitemap Custom module to automatically include Islandora objects in the Drupal sitemap. Detailed documentation on installing and configuring this module is available at https://wiki.lyrasis.org/display/ISLANDORA/Islandora+XML+Sitemap.

Make your sitemap discoverable to search engines

Even if you've enabled the Islandora XML Sitemap module, search engines may not locate your sitemaps unless you provide a direct link. There are two main options for directing search engines to your sitemaps.

1. Provide a hidden link to the sitemaps on your repository's homepage. If you've customized your site's theme, ensure that there is a link to /htmlmap on your front or home page.

```
<a href="/htmlmap"></a>
```

2. Announce your sitemap in your robots.txt. For example:

```
# The FULL URL for your sitemaps (HTML and XML)
# XML sitemap is listed first as it is preferred by most search engines
# Make sure to replace "[islandora.url]" with the value of your 'islandora.url' setting in your islandora.cfg
file.
Sitemap: [islandora.url]/sitemap.xml
Sitemap: [islandora.url]/htmlmap
```

3. Confirm that the sitemap was successfully submitted to Google via the Search Console (Crawl > Sitemap). Note that it may take a few days for Google to start crawling, and perhaps many days for it to finish, based on the priority Google assigns your content for indexing. If necessary, you can also manually submit a sitemap through the Search Console. Visit the Sitemap Report and enter the relative URL for the sitemap. Google should process the sitemap immediately, but site indexing may take some time. Follow the same steps to resubmit a sitemap if there are significant changes to your site structure.

Tips:

- `Sitemap:` lines can be placed anywhere in your robots.txt file. You can specify multiple "Sitemap:" lines, so that search engines can locate XML and HTML formats. For more information, see: http://www.sitemaps.org/protocol.html#informing
- Always include the FULL URL in the `Sitemap:` line. Relative paths are not supported.

Other Good Practices for SEO in Islandora

Keep Islandora up to date

New Islandora releases may include improvements to indexing tools.

- Best practices include keeping up to date with Islandora and Drupal module updates, especially these modules: -- https://wiki.lyrasis.org/display/ISLANDORA/Islandora+Core+Module -- https://wiki.lyrasis.org/display/ISLANDORA/Islandora+XML+Sitemap -- https://www.drupal.org/project/xmlsitemap -- https://wiki.lyrasis.org/display/ISLANDORA/Islandora+Scholar (if used)

Confirm that your site is visible to search engines

- Check that your site is visible by performing a "site:" search on each relevant search engine (e.g., search Google for "site:wikipedia.org")
- If your site does not show up, each search engine has its own process for manually submitting a URL for inclusion. The processes for common search engines include:
    - **Google:** Verify ownership in Google Search Console and submit URL for indexing via the Inspect URL tool.
    - **Yahoo and Bing:** Submit your URL for inclusion through Bing Webmaster Tools.
    - **Google Scholar:** Apply for inclusion through the Scholar Inclusions portal (discussed in more detail below).

Create a strong robots.txt

Your site should include a robots.txt file, which indicates to search engine crawlers which pages or files they can request from your site. A robust robots.txt file must strike a balance between overloading your server with crawler requests and ensuring access to the content needed to comprehensively index your site. In the case of a repository, crawlers should be able to index item, collection and community pages, and all bitstreams within items. Crawlers access your site as an anonymous user; they will not be able to access restricted content.

Ensure that your robots.txt file is at the top level of your site: i.e. at http://repo.foo.edu/robots.txt, and NOT e.g. http://repo.foo.edu/islandora/robots.txt. If your Islandora instance is served from e.g. http://repo.foo.edu/islandora/, you'll need to add /islandora to all the paths in the examples below (e.g. /islandora/browse-subject).

Ensure your robots.txt allows access to critical indexing paths

You may wish to block crawlers from accessing some URLs in your repository that do not contain relevant information, such as log-in or registration pages or contact and feedback forms. However, blocking certain URLs can impede crawlers from properly indexing your site. **Never put the following paths on `Disallow:` lines, or your repository might not be indexed properly:**

- /bitstream
- /browse (UNLESS USING SITEMAPS)
- /*/browse (UNLESS USING SITEMAPS)
- /browse-date (UNLESS USING SITEMAPS)
- /*/browse-date (UNLESS USING SITEMAPS)
- /community-list (UNLESS USING SITEMAPS)
- /handle
- /html
- /htmlmap

Tips:

- `Disallow:` lines are case sensitive.

Ensure your robots.txt allows access to item "splash" pages and full text.

Full-text access is critically important for effective indexing, enabling keyword searching and citation analysis among other functions.

Example good robots.txt

The following example robots.txt includes highly recommended settings (uncommented) and additional optional settings (in comments). Your local configuration will determine whether you choose to enable optional settings. To do so, uncomment the corresponding `Disallow:` line.

```
#
# robots.txt
#
# This file is to prevent the crawling and indexing of certain parts
# of your site by web crawlers and spiders run by sites like Yahoo!
# and Google. By telling these "robots" where not to go on your site,
# you save bandwidth and server resources.
#
# This file will be ignored unless it is at the root of your host:
# Used:    http://example.com/robots.txt
# Ignored: http://example.com/site/robots.txt
#
# For more information about the robots.txt standard, see:
# http://www.robotstxt.org/robotstxt.html
User-agent: *
Crawl-delay: 10
# The FULL URL to the Islandora sitemaps
# XML sitemap is listed first as it is preferred by most search engines
# Make sure to replace "[islandora.url]" with the value of your 'islandora.url' setting in your islandora.cfg
file.
Sitemap: [islandora.url]/sitemap.xml
Sitemap: [islandora.url]/htmlmap
# CSS, JS, Images
Allow: /misc/*.css$
Allow: /misc/*.css?
Allow: /misc/*.js$
Allow: /misc/*.js?
Allow: /misc/*.gif
Allow: /misc/*.jpg
Allow: /misc/*.jpeg
Allow: /misc/*.png
Allow: /modules/*.css$
Allow: /modules/*.css?
Allow: /modules/*.js$
Allow: /modules/*.js?
Allow: /modules/*.gif
Allow: /modules/*.jpg
Allow: /modules/*.jpeg
Allow: /modules/*.png
Allow: /profiles/*.css$
Allow: /profiles/*.css?
Allow: /profiles/*.js$
Allow: /profiles/*.js?
Allow: /profiles/*.gif
Allow: /profiles/*.jpg
Allow: /profiles/*.jpeg
Allow: /profiles/*.png
Allow: /themes/*.css$
Allow: /themes/*.css?
Allow: /themes/*.js$
Allow: /themes/*.js?
Allow: /themes/*.gif
Allow: /themes/*.jpg
Allow: /themes/*.jpeg
Allow: /themes/*.png
# Directories
Disallow: /includes/
Disallow: /misc/
Disallow: /modules/
Disallow: /profiles/
Disallow: /scripts/
Disallow: /themes/
# Files
Disallow: /CHANGELOG.txt
Disallow: /cron.php
Disallow: /INSTALL.mysql.txt
Disallow: /INSTALL.pgsql.txt
Disallow: /INSTALL.sqlite.txt
```

```
Disallow: /install.php
Disallow: /INSTALL.txt
Disallow: /LICENSE.txt
Disallow: /MAINTAINERS.txt
Disallow: /update.php
Disallow: /UPGRADE.txt
Disallow: /xmlrpc.php
##########################
# Default Access Group
# (NOTE: blank lines are not allowable in a group record)
# trailing slash sections courtesy of advice from https://www.volacci.com/blog/fix-problems-drupal-default-robotstxt-file
##########################
# Paths (clean URLs)
Disallow: /admin/
Disallow: /comment/reply/
Disallow: /contact/
Disallow: /logout/
Disallow: /node/add/
Disallow: /search/
Disallow: /user/register/
Disallow: /user/password/
Disallow: /user/login/
# Paths (no clean URLs)
Disallow: /?q=admin/
Disallow: /?q=comment/reply/
Disallow: /?q=contact/
Disallow: /?q=logout/
Disallow: /?q=node/add/
Disallow: /?q=search/
Disallow: /?q=user/password/
Disallow: /?q=user/register/
Disallow: /?q=user/login/
# Paths (clean URLs) – no trailing forward slash
Disallow: /admin
Disallow: /comment/reply
Disallow: /contact
Disallow: /logout
Disallow: /node/add
Disallow: /search
Disallow: /user/register
Disallow: /user/password
Disallow: /user/login
# Paths (no clean URLs) – no trailing forward slash
Disallow: /?q=admin
Disallow: /?q=comment/reply
Disallow: /?q=contact
Disallow: /?q=logout
Disallow: /?q=node/add
Disallow: /?q=search
Disallow: /?q=user/password
Disallow: /?q=user/register
Disallow: /?q=user/login
```

WARNING: for your additional disallow statements to be recognized under the `User-agent: *` group, they cannot be separated by blank lines from the declared `user-agent: *` block. A blank line indicates the start of a new user agent block. Without a leading user-agent declaration on the first line, blocks are ignored. Comment lines are allowed and will not break the user-agent block.

This is OK:

```
User-agent: *
# Disable access to Discovery search and filters
Disallow: /discover
Disallow: /search-filter
Disallow: /statistics
Disallow: /contact
```

This is *not OK*, as the two lines at the bottom will be completely ignored.

```
User-agent: *
# Disable access to Discovery search and filters
Disallow: /discover
Disallow: /search-filter

Disallow: /statistics
Disallow: /contact
```

Tips:

- To determine whether Google has access to a particular URL on your site, you can use the robots.txt tester in the Google Webmaster Tools suite.
- For more information on the robots.txt format, please see the Google Robots.txt documentation.

Avoid redirecting file downloads to Item landing pages

Some repositories use custom URL redirects in order to facilitate capturing web analytics (e.g., Google Analytics). While these URL redirects may seem harmless, they may be flagged as cloaking or spam by Google, Google Scholar and other major search engines. This may hurt your site's search engine ranking or even cause your entire site to be flagged for removal from the search engine. **Make sure that you never redirect "direct file downloads" (i.e. users who directly jump to downloading a file, often from a search engine) to the associated Item's splash/landing page.**

Acknowledgements

Development of the Islandora JSON-LD module, Islandora Matomo Docker image, Islandora Matomo module and real-time map plug-in, Islandora Scholar Profiles module, and Islandora Metadata Export module, as well as enhancements to the Islandora Webform module and the Islandora Scholar solution pack was supported by a grant from the Andrew W. Mellon Foundation. To learn more about this work, visit the LASIR project page.

Parts of this documentation were adapted from the Samvera Community's Digital Commons Feature Matrix.