Islandora:
Creating and Sustaining an Open Source Community

Melissa Anez
Islandora Foundation
Nick Ruest
York University
Where We’ve Been
History

Developed at UPEI in 2006

AlF Grant 2010 - 2013

Islandora Foundation - July 2013
The old way...

Writing Islandora code begins in one of three ways:

1. Idea for improvement or client request
   - Two items must be created by the developer:
     1. A JIRA ticket
     2. A message on the developers list "I am developing/I would like to develop" referencing the JIRA ticket number.
   - Discussion in mailing list and committers meeting. If consensus cannot be achieved, the issue is escalated to the Roadmap management meeting.

2. Bug
   - Bug is assigned to or assumed by most appropriate programmer. In the advent that there is not consensus, the issue is escalated to the Roadmap management meeting.

3. Contribution from community external to UPEI or DGI
   - UPEI or DGI developer takes stewardship of code to integrate into system.
   - A current module is forked into a developers private repository OR a new repository is created. While developing, a developer pushes regularly to GitHub, allowing for others to track progress.
   - Testing & Review by Group coordinated by release manager.

Pull request

Code Committed
The new way...

Documentation for developers
- Working With Fedora Objects Programmatically Via Tuque
- Working Programmatically With XML Forms
- Programming Solution Packs
- Coding Standards
- Git guidelines and best practices
- How Islandora Code Gets Made

Welcome!

If you are reading this document then you are interested in contributing to Islandora, and that's awesome! All contributions are welcome: use-cases, documentation, code, patches, bug reports, feature requests, etc. You do not need to be a programmer to speak up!

We also have an IRC channel on freenode.net. Feel free to hang out there, ask questions, and help others out if you can.

Workflows

Commit Messages

Commit messages should follow the guidelines.

In summary:
- First line: Issue ID in all caps (e.g. \[450\])
- Second line: blank
- Following lines: more detailed description of the changes (one paragraph per line)

Using the git command line tool you can add multi-line commit messages by typing a ' at the end of the line and continuing to type. E.g.,

```
$ git commit -m "Islandora 7.x.x cannot delete content model when there is only one content model that references it and it is not the last one remaining."
```

Islandora Committers (7.x-1.x)

Islandora is open source and released under GPL3. The software and associated documentation is developed collectively by a community of contributors and committers. All interested community members are encouraged to contribute to the project. Contributors who demonstrate sustained engagement with the project through quality participation in meetings, mailing lists, documentation and code updates can be nominated by existing committers to also become a committer. It should be emphasized that committers need not be limited to software developers. Community members with skills in documentation and testing, for example, can also be committers.

Contributing to Islandora

Islandora is committed to being a great open source project and to help improve that we are introducing new policies to improve our contributions. A successful project and we want to make sure that our community of contributors effectively works together.

All new features should be discussed with the community in advance to make sure that the new features will be broken down into smaller pieces and tested as a whole.

Getting started

To start contributing to Islandora, you need a few things:
- A GitHub account
- Knowledge of Git
- Familiarity with the Islandora codebase

You can contribute to Islandora by:
- Submitting bug reports
- Submitting pull requests
- Providing documentation updates
- Developing new features

For more information, please visit the Islandora website.
Where We Are
Islandora Community

- 150+ public sites,
- 1240 listserv members
- 104 GitHub members
- 20 Committers
- 23 Contributed Modules
- 26 Foundation members
- 10 Interest Groups
- 5 Community-led software releases
- 16 Camps
- 2 major versions
- 2 full-time employees
Collaborators:

- Andrews University
- University of Connecticut
- Common Media
- American Philosophical Society
- Florida State University
- Florida Virtual Campus
- Marmot Library Network
- Delft University

Members:

- California Institute of Technology
- PALS (Minnesota State Colleges and Universities)
- Technical University of Denmark
- Chinese University of Hong Kong
- BCELN
- Agile Humanities
- Berklee College of Music
- METRO
Membership Funding

$200,000
$180,000
$160,000
$140,000
$120,000
$100,000
$80,000
What does membership get you?

• **A voice!**
  ○ Sustainability
  ○ Governance
  ○ Collaboration

• **Project & Community Manager**
• **Technical Lead**
Islandora is...

...not static. It is ever evolving, dynamic, and requires organizational commitment.
Contributing to open source digital asset management/preservation systems and their communities...
...is a core component of digital preservation; sustainability, and risk mitigation at the very least.
Board of Directors

The Board is primarily concerned with legal and financial aspects of the Islandora community.

It is chiefly supportive of the Roadmap Committee and Islandora Committers, but directional in the sense that when the Roadmap Committee looks to the Board for direction and lays out the issues in a way that the Board can act on, the Board will provide direction. Roadmap Committee provides the core directions of the community.

http://islandora.ca/if/board
Roadmap Committee

- Consults and liaises with the Board of Directors as needed, ex. fiscal or legal issues;
- Consults and liaises with the Committers as needed, ex. technical direction, policy and process changes;
- Review, consult with the Community, and vote on timelines for adoption of major versions of the main Islandora components, ex. next major version of Drupal;
- Review and vote on submissions made under the Licensed Software Acceptance Procedure;
- Facilitate the creation of Interest Groups, as well as review and dissolve Interest Groups, and appoint convenors when necessary;
- Solicit and receive reports, as well as review and approve requests from Interest Groups;
- Review and approve proposals for Islandora Camps, conferences, or other Islandora events; Propose, review, and approve policies and guidelines for community conduct as needed, ex. community practice, committers practice, community conduct.

https://github.com/Islandora/islandora/wiki/Roadmap-Committee-Terms-of-Reference
Islandora Committers

Rights:
- Write access to the codebase
- Nomination privileges of new committers
- Release management privileges
- Binding votes on procedural, code modification, and release issues
- Access to the private Committers mailing list

Responsibilities:
- Monitor and respond to project mailing lists
- Attend project and technical meetings
- Monitor and vet bug-tracker issues
- Review and commit code contributions
- Ensure code contributions are properly licensed
- Guide and mentor new committers

https://github.com/Islandora/islandora/wiki/Islandora-Committers
Where We’re Going
Future Development

http://islandora.ca/CLAW
Islandora CLAW

I make use of existing tools integrating myself into each one. I’m not a layer, I play roles in multiple ones.
Islandora CLAW is developed by the Islandora community, for the Islandora community.
Community Centred

“All contributions are welcome: use-cases, documentation, code, patches, bug reports, feature requests, etc. You do not need to be a programmer to speak up!”
When is CLAW going to be done?
Minimum Viable Product: MVP

Clearly defined goals for community members to work towards, and present the point at which the CLAW team feels the software is ready to be tested out in small installations.

Needless to say, since this is a minimum product, a lot of parts of the previous prototype are either going to get replaced by an existing project to ease the maintenance load, or be removed altogether until time and resources allow other development.

https://islandora-claw.github.io/CLAW/mvp/mvp_doc/
CLAW Design Goals

- Big picture: Utilize Fedora 4/ Drupal 8 to their fullest
- Easier to install
- Easier to develop and contribute
- Expose Linked Data
- Easier to use
  - Enhanced user experience
  - Faster searching, display, and ingest
  - Expose more control of your repository through UI
- Easier to scale
High Level Features

- Content modeled in Drupal as Entities using PCDM 1.0
- REST API exposed for Drupal Entities
- Support for collections, images, books, and pages
- The ability to control metadata mappings between Drupal and RDF
- Provide RDF based default descriptive metadata profile in Drupal
- The ability to export/import JSON-LD
- Automatic backup of Drupal content in Fedora 4

- Ability to restore/bootstrap a Drupal site from a properly structured Fedora 4 repository
- The ability to index and search resources with Apache Solr
- The ability to restrict access to collections and/or individual resources across all representations (Drupal, Fedora, Solr, etc…)
- Asynchronous derivative generation
- Vagrant environment for development purposes, which will serve as a starting point for more complicated, distributed installs
What about Islandora 7.x?
Islandora 7.x-1.x

- Still actively developed.
- Supported until CLAW is production ready (and, let’s be honest, a little beyond).
- New releases twice yearly, managed by volunteers.
Questions?