Fedora 4 and Islandora Integration, Atonement, Upgradation, and more!
Project history and background
Fedora 4 Interest Group
Islandora Fedora 4 Interest Group

Convenors

- Nick Ruest (York University)
- Daniel Lamb (discoverygarden)

Terms of Reference

- The name of the iIG is the Islandora Fedora 4 Interest Group.
- The purpose of the Islandora Fedora 4 Interest Group is to implement Fedora 4 in the context of Islandora.
  - Islandora/Fedora 4 Prospectus
  - Islandora/Fedora 4 Project Plan
- Specific goals, activities, outcomes may include:
  - Create a generic "upgration" document for Islandora and:
    - Identify pilot partners
      - York University
  - Reviewing Fedora 3 to 4 Upgration Checklist and:
    - Identifying what features are necessary in Fedora 4
    - Mapping Fedora 3 features to Fedora 4 features
  - Mapping Fedora 4 features to Islandora
    - Determining how Islandora will leverage new Fedora 4 features
  - Outreach and communication
    - Engaging other interested Islandora community members
    - Recruiting developers to work on integration tasks
- The interest group will meet once a month virtually (e.g. via Skype). A call for agenda items will be posted to the Islandora Google Group 1 week prior to the meeting. The convenor will appoint a note taker for the meeting and meeting notes will be made available at some url.
- The convenors will produce a report to be submitted to the Islandora Roadmap Committee following the IF4iG’s meeting.

Links
Atonement
Documentation
About Islandora

Islandora is an open-source software framework designed to help institutions and organizations and their audiences collaboratively manage, and discover digital assets using a best-practices framework. Islandora was originally developed by the University of Prince Edward Island’s Robertson Library, but is now implemented and contributed to by an ever-growing international community.

Islandora consists of:

- **Sync** - Event driven middleware based on Apache Camel that synchronizes a Fedora 4 JCR with a Drupal CMS.
- **Islandora - Fedora 4 Repository module**
- **Install** - The is a development environment virtual machine for the Islandora and Fedora 4 project. It should work on any operating system that supports VirtualBox and Vagrant.

About this guide

The **Technical Design documentation** will help you:

- Understand the Islandora 7.x-2.x design rationale
- Importance of using an integration framework
- Using camel
- Inversion of control and camel
- Camel and scripting languages
- Islandora Sync
- Solr and Triple store indexing
- Islandora (Drupal)

The **How to build documentation** provides and overview on how the documentation is created, built, and deployed.

Installation

The **installation section** provides and overview on how to create a virtual development environment.

Contributing

If you would like to contribute, please get involved with the Islandora Fedora 4 Interest Group, and check out the contributing section. We love to hear from you!

If you would like to contribute code to the project, you will need to be covered by an Islandora Foundation **Contributor License Agreement** or **Corporate Contributor License Agreement**. Please see the **Contributors** page on Islandora.ca for more information.

Sponsors

- Discoverygarden
Contributing

http://islandora-labs.github.io/islandora/contributing/contributing
DevOps
Introduction

This is a development environment virtual machine for the Islandora and Fedora 4 project. It should work on any operating system that supports VirtualBox and Vagrant.
vagrant up
Upgration
Upgration

Pilots
Fedora Repository 3... / Fedora 3 to 4 Upgration Pilots

Upgration Pilot - York University

Created by David Wilcox, last modified on Feb 11, 2015

Project Overview

The York University Libraries upgration project identifies collections that cover the range of object models that the repository uses. The conservative goal is to perform an upgration on the collections listed below. The stretch goal is an upgration of all objects in the repository.

By upgration, we mean upgrating and migrating objects and datastreams, along with security restrictions (XACML), in Fedora 3.8.0 to Fedora 4.x. Moreover, we will develop a strategy for upgrading and migrating our content models, including inline XML datastreams, managed datastreams, and external datastreams.

York University Digital Library (YUDL) is an Islandora repository that run on the HEAD version of all Islandora Foundation modules. The repository is run as close a stock/generic Islandora instance where possible. Therefore, this upgration pilot can serve as a basis for a generic Islandora Fedora 3.x to Fedora 4.x upgration.

- Collection Description(s)
- Object Models
- Fedora 3 Details
  - Storage: Legacy storage (or Akuba)
  - XML metadata : datastreams
  - XML metadata: inline
  - Content models
  - Datastream types (inline, managed, redirect, and external)
  - Identifiers
  - Indexing strategies (GSearch, RI-Search vs. F4 approaches)
  - Replication/Journaling
  - Security policies: XACML
  - OAIPMH
  - Versions
  - Disseminators
  - Audit history
- Fedora 4 Details

Collection Description(s)

York University Digital Library contains approximately 200,000 unique digital assets.

Jean Augustine fonds
Upgration

Property mappings
fcrepo3->fcrepo4

Object properties
fcrepo3->fcrepo4

Datastream properties
fcrepo3->fcrepo4

RELS-EXT/RELS-INT
Islandora Ontology

http://islandora.ca/ontology/relsext/#
http://islandora.ca/ontology/relents/#
Islandora Ontology

https://github.com/Islandora-Labs/islandora_ontology
migration-utils

https://github.com/fcrepo4-labs/migration-utils
7.x-2.x
Technical design
Technical Design

Introduction
Technical Design

Integration Framework
Technical Design

Apache Camel
Technical Design

Apache Camel
Technical Design

Drupal
Middleware Services

Introduction
Middleware Services

Collection
Middleware Services

Object
Middleware Services

File
Middleware Services

Derivative
Middleware Services

Ingest
Middleware Services

Zip Ingest
XML Forms
The Future of Forms in Islandora

Submitted by mane on Mon, 05/11/2015 - 10:10

If you have been following events in the Islandora community lately, you probably know that we are in the midst of an ambitious project to build an Islandora that works on top of Fedora 4. The project is going great, but it has raised some questions about how to proceed with one of the most crucial yet under-appreciated tools in the Islandora stack: XML Form Builder.

Largely the work of a single developer (discoverygarden's awesome Nigel Banks), XML Form Builder is a powerful (and sometimes difficult-to-master) tool that allows you to leverage the ease of Drupal forms to edit metadata for your Fedora objects. All Islandora Solution Packs come with standard MODS forms, but XML Form Builder lets you go far beyond those basic building blocks to meet almost any use case. A simple form for students to add objects with just a few fields; a brand new form to manipulate esoteric metadata standards; a tiny tweak to an existing Solution Pack form that brings it in line with your institution's specific needs.

The Fedora 4 project team recognizes that this functionality needs to exist in the future version of Islandora. The question now is: How? Do we port over XML Forms and untangle the legacy of compromises that allows it to work in the current environment? Do we sidestep the issue and build a new tool that more closely leverages the underlying structure of Fedora 4? Do we build a new tool and make it look like XML Form Builder so it's comfortable to use, but still works completely differently? XML, RDF, Xpath, etc. It's a big task and it comes with a lot of big questions.

So, we turn to you, the Islandora Community, to get some answers. What tools do you need to work with metadata for your collections? How are you using XML Form Builder now? What parts of it don't you use? What parts are critical? To that end, Nick Ruest has put together a template to collect use cases. Please add yours to the list and help us shape Islandora's future.
Questions/Discussion