State of The CLAW Address;
In 20 minutes or less

Nick Ruest @ruebot
Diego Pino @dpinonavarro
Background
Fedora 3 is End of Life
Phase 1
December 2014-August 2015
Atonement

ethos
“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!”
Thank you to our sponsors:
Phase 2

November 2015-Present
Community

We have to do this together
Islandora CLAW is developed by the Islandora community, for the Islandora community.
Islandora CLAW

Introduction

Islandora CLAW (also known as Islandora 7.x-2.x) is the next generation of Islandora upgrade will be compatible with Fedora 4 (link is external).

For more details, please check out the following resources:

- Weekly Tech Calls
  - 2015
  - 2016
- CLAW Lessons - webinars on how to develop CLAW
  - Week 01: Intro to Fedora 4.x
  - Week 02: Hands-on Creating Fedora 4.x Resources

Repository Structure

- Alpaca: Event driven middleware based on Apache Camel that synchronizes a Fedora 4 with Drupal
- Chullo: A PHP client for Fedora 4 built using Guzzle and EasyRdf.
- Crayfish: Top level container for the various Islandora CLAW microservices, lovingly known as Cray.
- docs: Documentation!
- install: Bleeding edge development environment
- islandora: CLAW Drupal modules
- PDX: Top level container for the various PCDM specific Islandora CLAW microservices.

Sponsors

- UPEI
- discoverygarden inc.
- LYRASIS
- McMaster University
- University of Limerick
- York University
- University of Manitoba
- Simon Fraser University
- PALS
- American Philosophical Society
- common media inc.

Maintainers

- Nick Ruest
Developer documentation

Update vagrant to use Drupal 8, and remove all Drupal 7 CRUFT.

June sprint; Kick-off call notes

Crayfish code coverage

chullo code coverage

Build docs from CLAW Lessons

Setup codecov.io on PDX and Crayfish

Provide scholarly content types
CLAW Lessons
CLAW Calls

Wednesdays at 1PM EST
Time/Place

This meeting is a hybrid teleconference and IRC chat. Anyone is welcome to join. Here is the info:

- Time: 1:00pm Eastern Daylight Time US (UTC-4)
- Dial-in Number: (641) 715-3570
  - Participant Code: 304589#
  - International numbers: Conference Call Information
- IRC:
  - Join the #islandora chat room via Freenode Web IRC (enter a unique nick)
  - Or point your IRC client to #islandora on irc.freenode.net

Attendees

- Nick Ruest
- Jared Whiklo
- Ed Fugikawa
- Melissa Anez 🌟
- Peter Murray
- Diego Pino
- Ben Rosner
The future is **us**.

Make the lobster *happy*, not *sad*.
Project Plan & Priorities
islandora.ca/CLAW
ruestn@yorku.ca

Don’t be afraid to contact your Project Director
Technical Introduction to Islandora CLAW
What we know about current Islandora (7.x-1.7)

https://github.com/islandora/islandora
Design Goals for Islandora CLAW

Utilize Fedora 4 & Drupal to its fullest

- Easier to use
  - Better user experience
  - Faster searching, display, and ingest
  - Expose more control of your repository through UI
- Easier to install/configure
- Easier to develop and contribute
- Easier to scale
- Community Based design.

Islandora is no longer Middleware
Islandora CLAW (zoom level 0)

Where is Islandora here?
Islandora-CLAW is a bit everywhere (and among us)
I make use of existing tools integrating myself into each one. I’m not a layer, I play roles in multiple ones.

Multi Language Distributed Role Separation Not a layer
Diagrams are good
Quick Facts about Islandora CLAW (or how we achieve the Goals)

- Makes use of Drupal CMS completely (tight and close)
- Islandora Solution Packs? Real Drupal Modules
- Resources Live in Drupal and in Fedora 4 (same meaning, different representations)
- There is no “Source Master”.
  - Bi-directional Sync between Drupal and Fedora via CAMEL
- Exposes endpoints everywhere (RESTful API)
- It’s written in PHP(variety of frameworks), JAVA, Spring, Blueprint XML
- It deals with data in a very Async fashion (event driven)
- We have Microservices (Silex, PHP)
- It’s a stack of moving pieces that depend on each other
Islandora generates real Drupal Content. (In the future Entities)

Islandora CRUDs Real Fedora 4 Resources
Apache Camel

(Java?)
Don’t be scared of Apache Camel

Camel is a framework for building middleware

- Routes can be written in multiple languages (not only Java)
- It’s all about Messages: Enterprise Integration patterns and delivery management
- Works out of the box with everything you can think of
The immediate Future
¿microservices in CLAW?
A simple idea

No more Big chunks of all purpose code

Instead lightweight PHP Services that do a particular task, reusing code, classes or full packages

I’m a Specialist in all styles

Chullo Here. I pass messages to FEDORA 4

Crayfish Here. I handle RDF, do some stuff, pass it to Chullo
A simple idea

**Micro Services:** Run at an *HTTP* URL/PORT listening and processing what you send them. Not php code only. Real RESTful endpoints and routes

**ResourceService:**
I will take that HTTP POST “BULLET” for you

**Chullo:**
Invoke me with that message when you are done processing (whatever you do, I don’t care)
Role separation: Chullo, where it all started (not a service)

https://github.com/Islandora-CLAW/chullo

Uses EasyRDF and Guzzle(PSR-7) to talk to Fedora 4.

Provides Interfaces and Classes

**Islandora\Chullo**
- FedoraAPI (full api)
- Chullo (simplified access to api)
- TriplestoreClient (talks to triple store)

**Islandora\Chullo\Uuid**
- UuidGenerator (UUID V4 and V5 generator)
Simple ideas require new knowledge

**there is More:** WE built these Microservices using a PHP Framework named **SILEX**!

Let’s avoid reinventing the wheel and do some cool apps!

I’m not SILEX

ResourceService
I was built in SILEX. I can use YOU, simple PHP PACKAGE!!
Silex in a few words

http://silex.sensiolabs.org/doc/usage.html

- It’s a Micro-framework (you can build a one file app!)
- Imagine something “like” Camel but in PHP with less stuff.
- Ready for PHP 7
- Makes handling HTTP requests so simple
- It’s based on a larger framework named Symfony
  (DRUPAL 8 is Symfony!)

No WAY! You are better at Async. But I’m PHP. My people loves PHP

You want to replace ME!
Benefits of the CLAW approach

- Fedora 4 is better, faster, safer and better designed than Fedora 3
- You get the **Real Drupal Experience**
- 3rd Party Modules (Drupal)
  - Use all that is out there as normal Drupal users do
- Async is **good**
- Common and flexible Data modelling (Standard Ontologies)
- Simpler contributing to code
- Less code / Better Test Coverage
- Community based
Questions or Discussion

..or, time ran out?
ruestn@yorku.ca