Final Lesson

(let's have an open discussion)

May 17, 2016

(My name is Diego Pino Navarro
I work at metro.org)

Data Modeling in Claw, Migration to FCRepo 4, Derivatives. All tied together

Warning: little to none doodles

Data modeling or better: Semantic Data modeling

Facts that hurt:

Until now you/we people had little chances to do real semantic data modeling. (We did structural modeling)

Content Model?

- Mix of blueprinting (ds composite model) and application logic with practically no semantics
- Most of the meaning itself was given by XML metadata, attached to our Fedora3 objects.
- Rels-ext used liberally

And when we did it, we did it sometimes wrong!

Islandora RELS-EXT

`isSequenceNumberOfmynamespace%3A125`

https://github.com/Islandora/islandora_ontology/blob/master/relsext.rdf#L167-L171

Yes, we are creating "on the fly" a property name in Islandora 7.x
Which is 200% semantically incorrect

Further Illustrating our lack of Semantic Data modeling

Example for 7.x-1.x/Fedora 3:

How do i define “this is a digital representation of a Postcard”

1) I choose a CMODEL able to display a digital acquisition of a Postcard
2) The winner is: islandora:sp_large_image_cmodel
3) What Datastreams does islandora:sp_large_image_cmodel provide?
    a) OBJ(TIFF or JP2)
    b) MODS
    c) TN
    d) JPEG
    e) TECHMD
4) Ok, i make my master a TIFF
5) I fill my MODS, upload the TIFF
6) Hu. That works for verse side.
7) Now Same for the inverse (repeat steps 1-5)
8) How do i join this? A compound! islandora:compoundCModel
9) People can now see both sides! Hey i got a postcard?

Well. No.

Only you know it **is** a Postcard. (**is** == digital representation of a real object)

But other humans are smart enough to understand it's a postcard because:

- You added some hints in the title
- Metadata(MODS) points to the idea
  - Mods Genre: Postcards, postcards, picture postcard, etc?
- The pictures we display resemble the notion we have of what a postcard is.

By the way? When was the last time you sent a postcard?

- Maybe in 50 years no one will. The notion can get lost!
- Which part is the front? Does our MODS metadata describes also our multiple visual formats? MODS Links to the next object in this "compound"?
- What is a COMPOUND? ahhhhhhhhhhhhhhhhhhhhh...h

So: The difference between a Postcard and two selfies?

For Islandora 7.x not much:

- Our CMODEL decisions were made based on visualization (display)

As hard as this sounds: almost on Mime Types and available Drupal viewers!

Sorry: I forgot about MODS (present in all CMODEL defining Solution packs).

- Yes. We also got MODS to meta describe our Object.
- Mmm. Are we describing the object or the real thing?

But: nothing is lost. Structural Modeling was the first step (And we still need it on Fedora 4)

Data modeling in Islandora Claw

Needs:

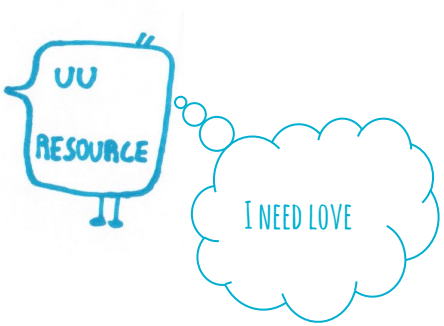
Structural Modeling composed of two layer:

- LDP (needed by the Fedora 4 Platform)
- PCDM Ontology: Defines aggregation of resources (similar to the idea of a FOXML)
  - Allows us to “bind” multiple related resources (RDF and Non RDF)
  - Defines some base rdf classes (pcdm:Object,pcdm:Collection, pcdm:File)
  - https://github.com/duraspace/pcdm/wiki

Semantic Data modeling (many layers):

- Based on Formal Ontologies (RDFS or OWL)
- Allows us to give our Resources and their connected ones(Graphs) a shareable/computeable/expectable meaning
  - rdf:type -> class
  - Literal Properties that extend the base description of a resource inside a knowledge domain (many of those can replace XML schemas completely)
  - Properties that give relationships between Resources a meaning inside a knowledge domain (predicates)

So where is the problem?



We need to make decisions on how to implement this in CLAW!

Spoiler alert. We already did some decision, but not enough ones. Not even sure if good ones

Ontologies define rules

Each time you apply an rdf:type to a resource you must be sure it does not conflict with the previous rdf:types it already has.

Questions that need answers:

- Which ontologies do we want to use to rdf:type our Islandora Resources?
  - Use Existing ones? Schema? Bibframe? EDM?
    - Criteria for deciding?
  - Create/Derivate our own Semantic/Ontology Stuff
  - Derive from which ones? From PCDM? Or higher semantic ones?
- Is PCDM enough?
  - PCDM defines structure. It organizes Resources
  - A pcdm:Object can be a Book, a Postcard, and Idea, an Agent, etc

Some ideas

———

And if we allow people to decide semantics on their own needs? How do we validate this?

We need an extra logic layer:

Semantic Reasoning system/Reasoner

On Ingest:

- Validate the passed RDF.
  - On properties and datatypes
  - On inter resource relationships (quality and quantity)
  - On incompatible types

On Retrieval:

- Given a “understandable/predictable base structure” (LDP+PCDM)
- Given other rdf:types
  - To be able to traverse the related resources based on the properties/restrictions these define for this resource
  - To know where to stop traversing
  - To know what resources are missing

Etc, etc...

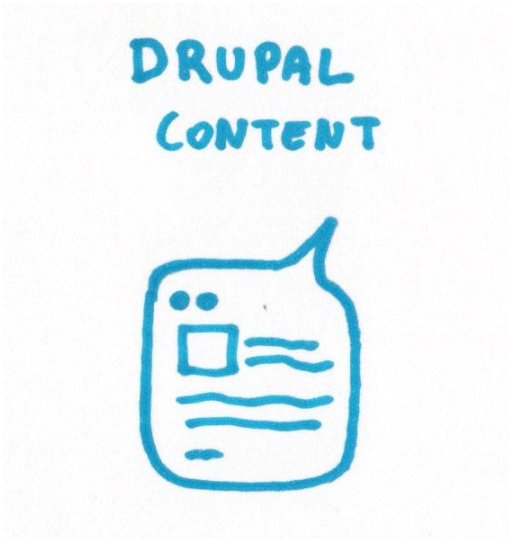
Ok. Nice Theory. But let’s Migrate to Fedora4

Sorry People: We need to migrate. We can migrate. But we need to solve our data modeling concerns first

Migrating without thinking about data modeling is like putting a square box inside a ball. You are losing potential!

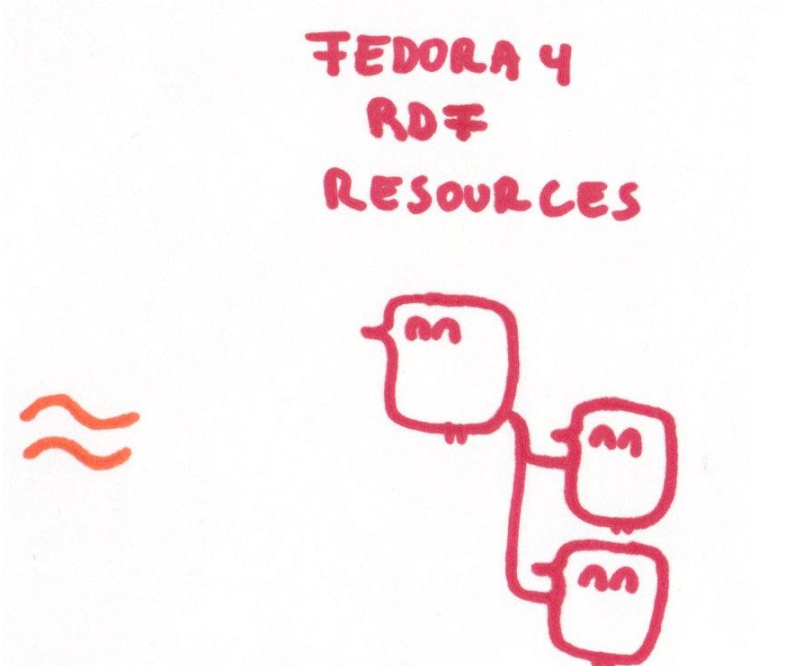
(Or not understanding the problem at all)

And don't forget Drupal (i think i used this slides too many times)



Islandora generates real Drupal Content.

≈



Islandora CRUDs Real Fedora 4 Resources

Drupal Content the Islandora-CLAW way

THIS NEEDS RETHINKING!!

- DRUPAL provides Content types and Entity types
  - We extend the Drupal **NODE** type to make a **BUNDLE**
  - Our **BUNDLES** include RDF fields properties)
  - Other fields too (configurable)
- We integrate **UUID** (it's back!)
- This whole content (**Entity**) lives in DRUPAL (MYSQL)
- Versionable as a whole
- We get an **URL**
  - http://somedomain.com/node1 (or UUID)

Lesson learned?

Islandora CLAW needs your interaction on this

- Data modeling affects your digital Assets
- Data modeling affects your workflows
- Data modeling affects your display
- Migration makes only sense if we are moving from non-semantic to semantic aware preservation
  - Not all of our current RELS-EXT properties make sense in Fedora 4
  - MODS does not solve all metadata needs (nor describes enough in an RDF world)

We need to start talking the same language:
Grab this book and put it under your pillow!
http://www.amazon.com/Semantic-Web-Working-Ontologist-Second/dp/0123859654

Migration
(Why, planning, Software tools)

Why do we need to migrate?

- Fedora 3 is No longer supported
- Performance: Fedora 4 can handle bigger data/more data/better data
- Web Semantics and Linked data
- Additional functionality

First steps

- We need a motivation (fedora 3 is dead?)
- We need a plan
- We need software
- We need storage space
- We need time

We have motivation, let's plan: 1. Data

- Object properties
  - Which ones can be migrated?
- Metadata
  - Moving as files?
  - Transforming XML to rdf?
  - Move literal values(agents?) to additional linked resources?
- Binaries
  - Every asset?
  - Only preservation masters/recreate derivatives?
- RELS-EXT/INT?
  - New Semantic Modelling?
  - CMODELS to which rdf:type?
- Identifiers: PID to URI?
  - Namespacing: an rdf:property?

let's plan: 2. Preservation

- Datastream versioning
  - Fedora 4 handles versions differently
  - In fedora 4 versions are optional(triggered by you)
  - In fedora 3, Object properties are not versionable
    - But there is some history on the Audit Log
- BagIt workflows?
- Premis/Audit?
- External Resources? Archivematica, S3, whatever

let's plan: 3. Functionality

- Access Control
  - XACML to WEBAC?
- How will Islandora CLAW /Drupal understand my migrated assets? Viewers, etc
- Do i have custom services that depend on Fedora 3/API?
- Integrations/indexing?
  - Solr/elasticsearch index? (forget about gsearch)

let’s plan: ask yourself

- Is any loss/ambiguities in metadata tolerable?
- Is any transformation of metadata representation/serialization tolerable?
- What did you promise your users?
- Do you have the resources (human included)

Sofware: migration-utils

---

https://github.com/fcrepo4-exts/migration-utils

- Developed as a framework or starting point for migration efforts
- Extensible for common institution-specific needs

Sofware: Other (own crazy ideas)

- As-you-go Migration (good for testing/refining)
- Based on Camel routes/activeMQ Messaging
- Run Fedora 4 and Fedora3 side by side
- When you ingest/update a Fedora 3 Object
  - Route does something similar to gsearch
  - Transforms your Fedora 3 Object and datastreams to Fedora 4
  - Uses blueprint like system, matching cmodels to pre made RDF graphs (mapping)
  - Can also be manually triggered to handle groups of PID

Derivatives

Current (7.x-1.x) approach

Each solution pack provides php code to derive their datastreams

Based on mime/type matching

More or less configurable

Sync processing: Ingest->wait-while-derive->derived

Islandora CLAW approach (WIP)

Derivatives will be provided by a service (Inside Alpaca?)

- Not necessarily based on rdf:type matching (previous cmodel idea) but on file types+semantics mix.
- Idea is to mimic/use/adapt File format registry https://www.archivematica.org/en/docs/fpr/
- Means shareable transform policies
- More dependable format detection that just mime/type
- Multiple different rdf:type (image, book, postcard, letter, whatever!) can share same transformations
- Really not a need anymore for an “OBJ”, any pcdm:file can be the preservation master
- Async, all based on Camel Routes.
- We need to expose this to drupal/rule/workflow builder?

Thank you all for being part of this!