“Contentless”
Digital Collections
“Contentless” Digital Collections

William G. Cowan, Library Technologies
Randall D. Floyd, Library Technologies
Daniel L. Pierce, Library Technologies

Digital Library Brown Bag, October 14, 2015
#dlbb
“Contentless”? 

Curated lists
Bibliographies
Representative images
Televised Opera and Musical Comedy Database
Hydra Community

Started as a joint development project between Stanford University, the University of Virginia and the University of Hull in 2008 (with Fedora Commons)
Hydra Community

- University of Notre Dame
- Northwestern University
- Columbia University
- Penn State University
- Indiana University
- London School of Economics and Political Science
- Rock and Roll Hall of Fame
- The Royal Library of Denmark
- WGBH
- Boston Public Library
- Duke University
- Yale University
- Virginia Tech
- University of Cincinnati
- Princeton University Library
- Cornell University
- University of Oregon (as Oregon Digital)
- Oregon State University (as Oregon Digital)
- Case Western Reserve University
- Tufts University
- Duoc UC
- University of Alberta
- Digital Public Library of America (DPLA)
- University of Michigan
Hydra Projects

- Hydra-in-a-Box
- Hydra and the Portland Common Data Model (PCDM)
- Hydra Solution Bundles
  - Avalon
  - Sufia
- Spotlight for Exhibits
Hydra, Sufia . . .

- Hydra Community Developed
  - Penn State
    - ScholarSphere
  - Notre Dame
    - CurateND
  - University of Cincinnati
    - Scholar@UC
Data Curation Experts
  ○ Worthwhile
  ○ Based on Sufia
• Why is this important?
  ○ Hydra and Fedora 4
BiblioCat

- Simple cataloging interface to manage “contentless” curated resources
- Based on Worthwhile
- What we’ve done with Worthwhile to make it even more useful
Two concepts emerged in CurateND and again in Worthwhile

- **Curation Concerns**
  - Adds simple IR-like self submission abilities to a Hydra-based application
  - Adds the ability to manage related objects as "Work Types"

- **Work Types**
  - Custom content type definitions that items can derive their attributes and behaviors from
  - Are added as needed by running a “generator”
By default in Worthwhile:

• Registers the Work Type for use in the system
• Generates code for that Work Type that can be used to customize its metadata, behaviors, etc.
  ○ A developer would then customize that generated code to make that Work Type different from the others
But we want:

- To register the Work Type for use in the system
- To make its metadata and behaviors configurable by a non-developer
- To not be editing code to accommodate the special needs of each Work Type
Generating a Work Type...

Dynamic Configuration Files

Creates configuration for work type

Registers a work type for use by catalogers
Configuration File

```json
work_types:
  Televised Opera:
    fields:
      title:
        vocab: "RDF::DC"
        label: "Work Title"
        multiple: "false"
        views: ",index,show,edit"
      duration:
        vocab: "RDF::Vocab::EBUCore"
        label: "Duration"
        multiple: "false"
        views: ",index,show,edit"
  Orchestration:
    vocab: "RDF::Vocab::MO"

Changes to these should be effective in the system immediately.
```

Configuration determines metadata and behaviors.

**Bibliocat**

![Bibliocat interface for "The abduction of Figaro" (Televised Opera)](image)
Records are immediately available to add to exhibits in Biblio when created/updated in Bibliocat.

And to take it one step further:
Bibliocat is our cataloging tool for curated bibliographies. It uses Curation Concerns in a Worthwhile application.

Biblio is the public access interface for bibliographic records. It is an exhibit oriented Spotlight application.

Flexible dynamic Work Types without touching code!

Generating a Work Type...

Dynamic Configuration Files

Creating configuration for work type

Records are immediately available to add to exhibits in Biblio when created/updated in Bibliocat.
Biblio

What about user access?

Spotlight
Spotlight
Building Exhibits

- Access through Solr to Fedora objects
- Widget based site development
- Faceted Search
- Blacklight based
Why not Omeka?

- An open source web publishing platform
- A tool for creating digital content
- An exhibit builder for digital objects
Spotlight and Omeka

**Spotlight**
- Access to Fedora Objects
- Widgets based (limited)
- Theme based (limited)
- Close to Production

**Omeka**
- No direct access to Fedora
- Plugin based (hundreds)
- Theme based (many)
- Since 2008
Illuminating repositories using Spotlight
Spotlight Exhibits

Adds formatted pages on top of Blacklight faceted search results
Blacklight vs Spotlight
Information Flow

Apache Solr index populated by Bibliocat
Customized to read Work Type configurations from Bibliocat
Labels will match what is seen in the cataloging interface
Futures
Things to do

Update to Fedora 4
Align to the New Curation Concerns
Next Collection
Questions?