bibliotek-o :
a BIBFRAME Implementation

Folsom, Steven
Cornell University
sf433@cornell.edu

Kovari, Jason
Cornell University
jak473@cornell.edu

Younes, Rebecca
Cornell University
rebecca.younes@cornell.edu
Background & Motivation
LD4L Labs & LD4P (2016-2018)

- Andrew W. Mellon Foundation funded projects
- Aimed to expand linked data within library domain
- Joint effort: Ontology Group

https://ld4l.org/ | http://ld4p.org/
Motivation

• Engage in BIBFRAME evaluation
• Provide extension to BIBFRAME
• Demonstrate select alternative patterns
• Accommodate legacy data in real-world orientation

NOT 'competition'
Development Process
bibliotek-o development process

• Articulated principles & best practices
• Used BIBFRAME as a starting point / default modeling
• Developed alternative models for discrete patterns
• Submitted recommendations to LoC for consideration
Design Principles
Modeling principles (selected)

- Reuse and align with existing external vocabularies
- Use OWL axioms (in moderation)
- Prefer object properties and structured data to literals
- Adopt single way of expressing a relationship or attribute
- Prefer atomic data representation
- Simplify and generalize
Vocabulary reuse & alignment

Align & reuse well-established ontologies

- FOAF, schema, prov, Web Annotation
- rdau relationships: bibliographic resources

Concerns: stability & semantic alignment
RDF and OWL constructs (in moderation)

rdfs:domain, rdfs:range

owl:ObjectProperty, owl:DatatypeProperty, owl:SymmetricProperty, owl:TransitiveProperty, owl:inverseOf

Avoid over-constraint / over-specification
Uniform representations

Single method of expressing relationship or attribute

Minimize query paths
Structured data versus literals

Prefer object properties & resources, where logical
Simplify and generalize

• Prefer generalized properties when classes indicate semantics

• Demonstrated via Activity pattern
Select Modeling Patterns
Activities
Bibliotek-o activities

Design principle: Prefer the simplest and most general model capable of faithfully representing the data. Use common patterns to capture shared semantics across entity types.
Content/Carrier/Media
Two patterns:

• `bf:content/bf:Content | bf:carrier/bf:Carrier | bf:media/bf:Media`

• Subclassing of `bf:Work | bf:Instance`
Design principle: Adopt a single method of expressing a relationship or attribute in order to eliminate redundancy and minimize query paths.

Subclasses of bf:Work | bf:Instance
Design principle: Prefer decomposed (atomic) to precomposed (composite) values.

- **BIBFRAME:**
  - Content resource “two dimensional moving image”
  - Content resource “three dimensional moving image”
- **bibliotek-o:**
  - bf:MovingImage
  - bf:Object + bf:MovingImage
Legacy Literals
The challenge of legacy literals

Migrate and preserve unstructured legacy data while defining a forward-looking model for original cataloging in RDF.

508 #$a Photographer, Richard Beymer ; film editor, Charles Pavlich.

• *In general*, BIBFRAME tends to accommodate legacy data by defining datatype properties.
  • `bf:credits` (datatype property)
bibliotek-o strategy

• Focus on a model suited to original RDF data creation and real world data.
• But conversion tools not yet able to fully parse this complex data.
• Attach a custom datatype to the string to flag the data for future parsing and structuring.
  • [http://bibliotek-o.org/datatypes/legacySourceData](http://bibliotek-o.org/datatypes/legacySourceData)
• Create the target object structure.
• Store the typed literal as the rdf:value of the object.
• Preserves legacy data without distorting the model with unwanted datatype properties.
Tooling
Editors

• **VitroLib**
  • Cataloging editor built on Vitro
    • Ontology-neutral semantic web application for ontology and instance editing and browsing
    • Foundation of VIVO application
  • Loaded with bibliotek-o framework ontologies
  • Customizations for data entry and display based on these ontologies and application profile (in progress)
  • Lookups to external data sources
  • Developing SHACL application profiles to drive custom form configurations

• **CEDAR**
  • Create and edit RDF-based, ontology-driven forms
Status and Community Engagement
bibliotek-o's influence on BIBFRAME

- owl:inverses declared
- datatype properties --> object properties (select)
- selective soft-alignment
bibliotek-o status

Further development frozen; however...
Community engagement

We welcome GitHub issues around modeling:

https://github.com/ld4l-labs/bibliotek-o
Advocacy

bibliotek-o represents:

LD4_ Ontology Group's BIBFRAME 2.0 assessment

We encourage active engagement with BIBFRAME
Documentation

Home page: [http://bibliotek-o.org/ontology](http://bibliotek-o.org/ontology)

OWL file: [http://bibliotek-o.org/ontology.owl](http://bibliotek-o.org/ontology.owl)

Human-readable documentation: [http://bibliotek-o.org/ontology.html](http://bibliotek-o.org/ontology.html)

Visualization: [https://bibliotek-o.org/overview/overview.html](https://bibliotek-o.org/overview/overview.html)

GitHub repository: [https://github.com/ld4l-labs/bibliotek-o/tree/v1.0.1](https://github.com/ld4l-labs/bibliotek-o/tree/v1.0.1)

FAQ, pattern recommendations & RDA discussion: [https://wiki.duraspace.org/x/H5TBB](https://wiki.duraspace.org/x/H5TBB)
Thank you!

Folsom, Steven
Cornell University
sf433@cornell.edu

Kovari, Jason
Cornell University
jak473@cornell.edu

Younes, Rebecca
Cornell University
rebecca.younes@cornell.edu