Linked Data Best Practices and BibFrame

CNI 2015 Fall Forum

Washington DC

December 15th, 2015
Rob Sanderson (@azaroth42)

(google doc)
Overview

• Quick Linked Data Refresher

• Best Practices Assessment
  • Define Your Domain
  • Use URIs for Identity
  • Provide Useful Information
  • Reuse Existing Work
  • 700s vs 500s

• Practical Concerns
Linked Data Refresher: RDF

RDF encodes a Graph Data Structure

- W3C standard(s)
- Nodes are identified by URIs
- Data is included as literal values
- Anyone can assert anything about anything
- Blank nodes don't have URIs
- Order is hard
Linked Data Refresher: RDF

RDF encodes a Graph Data Structure

- W3C standard(s)
- Nodes are identified by URIs
- Data is included as literal values
- Anyone can assert anything about anything

- Blank nodes don't have URIs
- Order is hard
- **No honestly, it is really hard!**
Linked Data Refresher: LOD

Linked Open Data provides best practices for RDF

1. Use URIs as the names of things
2. Use HTTP URIs so people can look up those names
3. When someone does, provide useful information, using standards
4. Include links to other URIs, so they can discover more things
Linked Open Data provides a more consistent framework

- Constraints on RDF derived from successful usage
- Adapted and extended
- Demonstrable improvement for adoption and usability
Linked Data Best Practices

Linked Open Data provides a more consistent framework

• Constraints on RDF derived from successful usage
• Adapted and extended
• Demonstrable improvement for adoption and usability

• Does BibFrame follow them?
• Do recent "2.0" updates help?
Define Your Domain

A Domain Model keeps you honest by constraining scope

- Define appropriate terms from your domain model
- Define **only** terms from your domain model
- Define **only** one pattern for each feature
- Consider dynamic resources carefully
Define Your Domain

• Define appropriate terms from your domain model
• Define **only** terms from your domain model
• Define **only** one pattern for each feature
• Consider dynamic resources carefully

• ✔ Work, Instance, Item, Title, Identifier ...
• ✗ Person, Place, Annotation, Relator, Resource, ...
• ✗ title vs titleStatement, parts, notes vs relationships, ...
• - Unclear. Circulation?

Score: 1 / 4
Define Your Domain: 2.0

- Define appropriate terms from your domain model
- Define **only** terms from your domain model
- Define **only** one pattern for each feature
- Consider dynamic resources carefully

- ✔ Work, Instance, Item, Title, Identifier ...
- ½ Person, Place, Annotation, Relator, Resource, ...
- ½ title vs titleStatement, parts, notes vs relationships, ...
- - Unclear. Circulation?

Score: 2 / 4
Use URIs for Identity

URIs are globally unique identifiers, fundamental to Linked Open Data

• Use URIs, not strings
• URIs must identity one thing
• Use HTTP URIs
• Use Natural Keys in URIs
• Clients treat URIs as opaque
• Avoid Dates, Hash URIs
Use URIs for Identity

- Use URIs, not strings
- URIs must identify one thing
- Use HTTP URIs
- Use Natural Keys in URIs
- Clients treat URIs as opaque
- Avoid Dates, Hash URIs

- MANY uses of strings for identity, esp authorities
- Resource vs Metadata, Part vs Whole
- HTTP used in general
  But blank nodes overused?
- Ontology & examples are good; compare RDA's P10001
- No URI construction or inferencing required
- No dates, versions in URIs, no recommendation for hashes

Score: 4 / 6 (charitable)
Use URIs for Identity: 2.0

- Use URIs, not strings
- URIs must identity one thing
- Use HTTP URIs
- Use Natural Keys in URIs
- Clients treat URIs as opaque
- Avoid Dates, Hash URIs

- ½ Fewer uses of strings for identity, esp NOT authorities
- ✔ Resource vs Metadata; Part vs Whole
- ✔ HTTP used in general
  But blank nodes overused?
- ✔ Ontology & examples are good; compare RDA's P10001
- ✔ No URI construction or inferencing required
- ✔ No dates, versions in URIs, no recommendation for hashes

Score: 5½ / 6 (somewhat charitable)
Provide Useful Information

Someone else will do the most interesting thing with our data

• Provide useful information when URI is requested
• Describe your own resources ... individually
• Include links to other resources
• Avoid reification, lists, and blank nodes
Provide Useful Information

- Provide useful information when URI is requested
- Describe your own resources individually
- Include links to other resources
- Avoid reification, lists, and blank nodes

- ✔ Promoted for main classes, Identifier needs attention
- ✓ Not discussed but Annotations need attention
- ❌ Only internal references, not external (e.g. language)
- ❌ Reification (e.g. related)
  - ✔ lists
  - ❌ Blank nodes (everywhere)

Score: 2 / 6
Provide Useful Information: 2.0

• Provide useful information when URI is requested
• Describe your own resources individually
• Include links to other resources
• Avoid reification, lists, and blank nodes

• ✔ Promoted for main classes, Identifier got needed attention
• ✔ Annotations got attention
• ✗ Only internal references, not external (e.g. language)
• ✗ Reification (e.g. contribution) ✔ lists
  ✗ Blank nodes (still everywhere)

Score: 3 / 6
Reuse Existing Work

Don't ignore giant shoulders!

- Reuse existing vocabularies
- Define terms in your own namespace
- Relate new terms to appropriate existing ones
- Name terms consistently
- Only define what matters
- Inverse relationships matter
Reuse Existing Work

- Reuse existing vocabularies
- Define terms in your own namespace
- Relate new terms to appropriate existing ones
- Name terms consistently
- Only define what matters
- Inverse relationships matter

- ✔ Fundamental and ignored
- ✗ Somewhat faint praise?
- ✗ Minimally related outside of BibFrame
- ✗ Terms don't follow best practices, or internal convention
- ✗ Over engineered, doesn't reuse own properties
- ✗ Very inconsistently done

Score: 1 / 6 (charitable)
Reuse Existing Work: 2.0

- Reuse existing vocabularies
- Define terms in your own namespace
- Relate new terms to appropriate existing ones
- Name terms consistently
- Only define what matters
- Inverse relationships matter

- X Fundamental and still ignored
- ✔ Still faint praise
- X Minimally related outside of BibFrame
- ½ Terms don't follow best practices, or internal convention
- ½ Over engineered, doesn't reuse own properties
- X Very inconsistently done

Score: 2 / 6 (still charitable)
Conclusion: Improvement in 2.0!

Charitable Scores:

- BibFrame total score: $8 / 22 = 36\%$ ... fail
- BibFrame 2.0 total score: $12.5 / 22 = 57\%$

- 57\% is still a C grade, but now passing
- But ... Perfection is the enemy of the good enough

Work Still Needed:

- Reuse existing ontologies and vocabularies
- More consistency in design
- More linking = more part of the semantic web
- Drop remaining strings that provide identity
700s vs 500s
700s vs 500s

Okay, Who needs a hint?
700s vs 500s

Hints:

• Not: Added Entries vs Notes
700s vs 500s

Hints:

• Not: Added Entries vs Notes

• http://id.loc.gov/authorities/subjects/sh85008324
  vs
  http://id.loc.gov/authorities/subjects/sh85118553
700s vs 500s

Hints:

• Not: Added Entries vs Notes

• http://id.loc.gov/authorities/subjects/sh85008324
  vs
  http://id.loc.gov/authorities/subjects/sh85118553

• dewey:700 vs dewey:500

Ahem ... natural keys???
700s vs 500s

Hints:

• Not: Added Entries vs Notes

• http://id.loc.gov/authorities/subjects/sh85008324
  vs
  http://id.loc.gov/authorities/subjects/sh85118553

• dewey:700 vs dewey:500

Ahem ... natural keys???
Art vs Science

They're more like... guidelines

- Any assessment is subjective
- There are no unbreakable rules
- Context and expected use are important to consider
Practical Concerns

- Documentation is currently insufficient for third parties to develop implementations
  - Needs to be updated and maintained

- MARC to BibFrame converter from LC
  - Wrapped by Stanford for local data improvements
  - Wrapped by Cornell for ontology improvements
  - Wrapped by ...

- BibFrame "Lite" converter from Zepheira
  - ... Doesn't really implement BibFrame
Conversion Now

- **Written in XQuery**
  - Suited for XML, not Graphs
  - Very limited community
  - Limited functionality

- **Lack of Tests / Documentation**
  - Difficult to determine if current behavior is correct
  - Difficult to know if code changes break existing behavior

- **Inflexible**
  - Difficult to customize for local requirements
  - Difficult to keep up to date with changing ontology
  - Difficult to use external enrichment or transformation tools
Insufficient for Production

We will need to:

• Re-run repeatedly as data, ontology and code change
• Handle enhancements to the MARC data
• Handle enrichments to the resulting graphs
• Customize it for local practices
• Know when it doesn't work
  • And what needs to be fixed
• Share the development, configuration and understanding
Desirable Conversion Features

- Supported and developed by the community!
- Documented
- Testable and auditable
- Efficient
- Configurable
- Robust
- Integrated with local systems
How to Get There?

• Thoroughly document the ontology
  • If it's too hard to document, it's too hard to implement!
• Use proposals process to solicit feedback on updates
• Document transformation processing algorithms used
  • Per MARC field, and/or per BibFrame feature
  • Updating with new proposals
• Engage with community to determine requirements
  • Make it possible for stakeholders to implement their own patterns
• Seek partners for development
  • LD4L keen to participate!
Thank You!

ROB SANDESON

@azaroth42
azaroth42@gmail.com / azaroth@stanford.edu

(google doc: final draft of analysis report)

(these slides on slideshare)

http://ld4l.org/