

URI Management Working Group

The URI Management Working Group formed in April 2016 to make recommendations for creating and maintaining URIs that are necessary for Samvera Community projects to express appropriate RDF properties and property values. The working group collected its meeting notes and other documents on the following wiki page:

[\[https://wiki.duraspace.org/display/samvera/URI+Management+Working+Group\]](https://wiki.duraspace.org/display/samvera/URI+Management+Working+Group).

The working group produced two documents, the Predicate Decision Tree, and the Samvera Vocabulary Manager Functional Requirements; as well as a list of next steps. The Working Group completed this work in February 2017.

Goal

This group is interested in making recommendations for creating and maintaining URIs that are necessary for Samvera community projects to express appropriate RDF properties and property values. This group is a working group of the [Samvera Metadata Interest Group](#) and is considering use cases from requests and questions documented on the [Metadata IG Requests and Priorities list](#). Examples of properties that might require minting new URIs include the following:

- Community terms (Sufia, CurationConcerns, possibly PCDM)
 - `ss:arkivoChecksum` = `premis:hasMessageDigest` or a checksum property from LC (not sure which)
 - `ss:relativePath` = `premis:hasContentLocationValue` or a property from ebucore (`ebucore:locator?`)
 - `ss:importURL` = `premis:historyInfo` but this might not work
 - `ss:proxyDespoitor` = `marcrelator:dpt` (depositor)
 - `ss:onBehalfOf` = `marcrelator:cre` (creator)
- Accepted vocab (standard) but not available through RDF - I.e. GeoRSS, PBCore
- Local vocab that everyone wants - you just don't know it yet!
- RDF that doesn't work for me, i.e. Premis

To keep the scope reasonable, the group limited itself to only RDF Properties (e.g. resources of type `rdf:Property`).

Samvera Metadata Survey

The group prepared and ran a survey on metadata needs between June 3 and July 11, 2016, asking the following seven questions:

- What repository software are you using? Hydra/Islandora/Other
- Have you needed predicates (properties) that you couldn't find? Yes/No

- Have you needed objects (values or controlled vocabularies) that you couldn't find?
Yes/No
- Are you currently storing literals or URIs for RDF objects (values)? Literals/URIs/Both
- If you are storing literals, what types of literals are you currently storing? [Strings][Strings with language code][Boolean values][Numeric values][Date/Time values][Other]
- Have you ever needed to access a previous version of an ontology (either for predicates or objects)? Yes/No
- Is there anything else you would like to share regarding using RDF with Hydra or Fedora?

The survey gathered 18 responses, from Hydra or other repository software users, and no responses from Islandora users. Only one response mentioned the need to support a previous version of an ontology. Every response said they were storing metadata using literals, a few were also storing a URI. No one is storing only URIs. Responses also indicated usage of other literal types, such as date/time types, numeric types, and strings with a language code.

[Predicate Decision Tree](#)

The predicate decision tree was created as a guide to finding predicates to be used for Samvera applications. An initial draft was created by the working group and then released for community comment. The main principle behind the tree is to try to reuse existing predicates when possible. It provides many links to existing ontologies as well as resources to find existing predicate terms.

[Functional Requirements](#)

The Samvera Community Vocabulary Manager is intended to be software that can provide both human and machine readable responses to RDF predicates created by the Samvera Community. The list of functional requirements lists abilities this software should have and considers the gathered use cases and survey responses. In building out these requirements, the minimum (M1 and M2) and extended (E1) requirements from the [W3C's Best Practice Recipes for Publishing RDF Vocabularies](#) were consulted and implemented. M1 requires that a serialization is provided when dereferencing a URI (N-Triples, JSON-LD, and TTL); M2 requires that an HTTP URI be consistent in its interpretability (HTML using text/html, TTL using application/x-turtle, JSON using application/json, and N-Triples using text/plain); E1 requires that a dereferenced URI provide human-readable documentation (URIs resolve in a web browser to human-readable HTML pages). E2 requires that versions of a vocabulary be differentiated but at this time the Samvera Community Vocabulary Manager is not required to manage versions. The working group accepts that both the requirements and the implementation will change over time.

The working group assembled some examples of what a possible implementation of the vocabulary manager could look like. The examples use two properties (arkivoChecksum and

`importURL`) that may or may not be actual Samvera predicates. Each example screenshot is linked at the end of this document.

The route [/index.html](#) would list all properties in HTML with links to a complete RDF serialization of the same list in [N-Triples](#), [JSON-LD](#), and [TTL](#) (turtle).

Each property has its own HTML page showing metadata about the property term itself and includes links to the same data in the following RDF serializations: N-Triples, JSON-LD, and TTL.

One example property is the HTML page for [arkivoChecksum](#) and its serializations: [N-Triples](#), [JSON-LD](#), [TTL](#)

Another example property is the HTML page for [importURL](#) and its serializations: [N-Triples](#), [JSON-LD](#), [TTL](#)

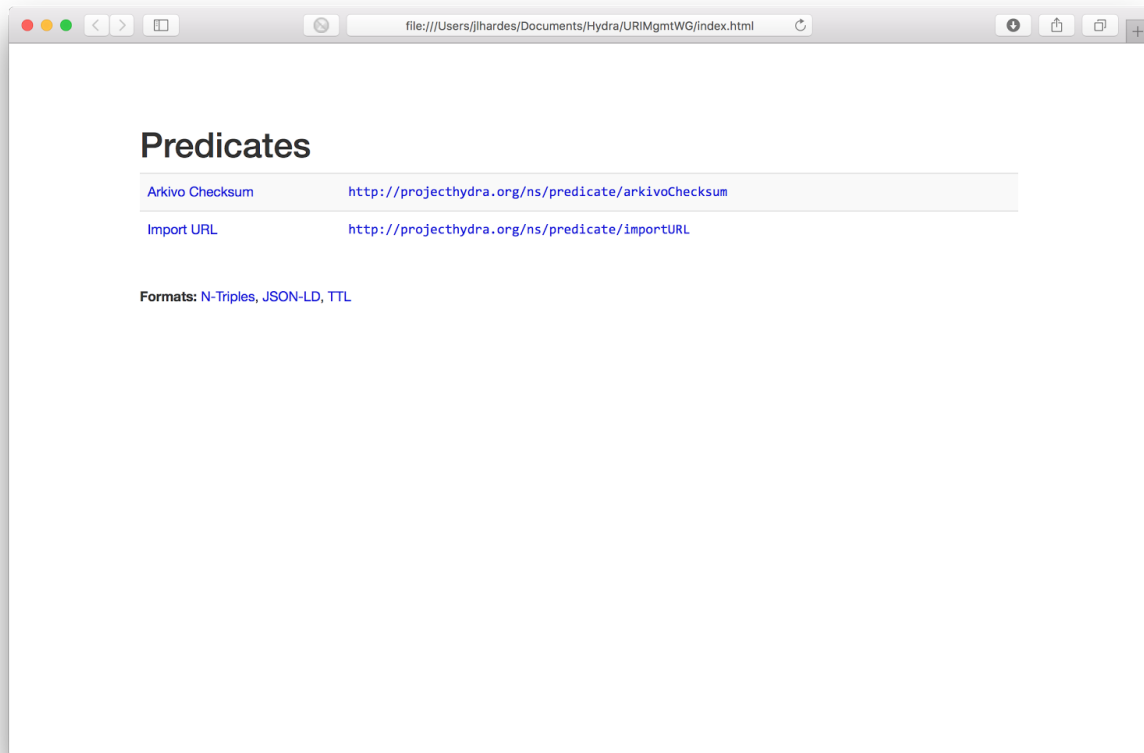
The HTML pages would show all the metadata for a given predicate. The *Usage Example* text comes from the property `skos:example`.

Open Questions/Future Work

- The Development and Deployment of the Vocabulary Manager software is still open.
- The URI Management Working Group recommends a standing working group be created to handle the work of reviewing requests for new predicates and creating them. See [draft charter for URI Selection Working Group](#).
- The issue of handling RDF vocabularies, especially large ones, was not addressed. However, it is expected that the vocabulary manager should be able to handle it in later phases of development.
- Institutions seem to be split on the importance of honoring the ranges for predicates proscribed in ontologies. (Which is reasonable, since RDFS range statements are simply assertions and do not have a notion of conformance.)

Mock-ups

/index.html



arkivoChecksum

Arkivo Checksum
<http://projecthydra.org/ns/predicate/arkivoChecksum>

rdfs:subPropertyOf

rdfs:range <http://www.w3.org/2001/XMLSchema#string>

rdfs:domain

rdfs:label Arkivo Checksum English [en]

rdfs:comment An MD5 checksum for a bitstream uploaded by Zotero's subscription service, Arkivo. English [en]

rdfs:seeAlso <https://github.com/projecthydra/sufia/issues/1327>

rdfs:isDefinedBy <https://www.npmjs.com/package/arkivo>

dcterms:creator spr7b@virginia.edu

dcterms:issued 2016-11-11

dcterms:modified

dcterms:description

Other Formats: N-Triples, JSON-LD, TTL

skos:example:
A repository object has an MD5 checksum generated by Arkivo that should be stored with the bitstream.
<<http://example.org/object>> <<http://hydra.vocab.manager/predicate/arkivoChecksum>> "C7F2C101A6A74332C72483C48D5E202D" .

importURL

file:///Users/jlhardes/Documents/Hydra/URIMgmtWG/importURL.html

Import URL

<http://projecthydra.org/ns/predicate/importURL>

rdfs:subPropertyOf

rdfs:range

rdfs:domain

rdfs:label Import URL English [en]

rdfs:comment The URL from which content hosted in a cloud storage service is imported. English [en]

rdfs:seeAlso https://github.com/projecthydra/curation_concerns/issues/561

rdfs:isDefinedBy

dcterms:creator spr7b@virginia.edu

dcterms:issued 2016-11-11

dcterms:modified

dcterms:description

Other Formats: N-Triples, JSON-LD, TTL

skos:example:
A repository object was imported from Google Drive cloud storage.
<<http://example.org/object>> <<http://hydra.vocab.manager/predicate/importURL>> <https://drive.google.com/file/d/0B_3cn7zuYfw-UVVhTmdTc2VYd1k/view?usp=sharing> .