Descriptive Metadata and Controlled Vocabularies

Context

Now that we have built some consensus around the structure of Hydra Works and Collections, what descriptive metadata do we need to agree on to have common tools that work on them?

The Hydra community has an opportunity to converge on DPLA MAP v4 as a common baseline for descriptive metadata, part of which would be a move to RDF and usage of classes/things instead of strings

Discussion

Is there a need for folks to agree on a certain subset of fields, like title? Possibly, but perhaps not relevant (could be mitigated via different Solr indexing strategies)

Sufia’s implementation of descriptive metadata: there is a list of terms, some for display-only and others for user-editing, that can be asked, e.g., if they are repeatable or not

LinkedVocabs is a newer, better version of what Oregon (and Sufia adopters) are using for Questioning Authority-like needs (for instance, it includes validations). Every CV resource class has a QA-compliant mixin (which hits remote services or uses sparql to hit a triplestore). The controlled vocab resource class can be configured to use multiple vocabs per term. Need to link fields to multiple vocabs and merge results (querying happens at resource class, not in controlled vocab). Does the community want this? (YES!) Requires a triplestore with an RDF.rb adapter (does not exist for Fedora 4, which doesn’t seem like a good fit, so this means we’d need another system dependency in our Hydra infrastructure to support LinkedVocabs). We discussed whether Solr could be a good back-end, and concluded it wouldn’t comply with the RDF.rb repository interface so it's not a great fit either. MongoDB is working out well for Oregon Digital.

Other products, such as Marmotta (used by DPLA), can automatically cache linked data vocabs, which prevents having to hit remote services all the time. Its LDcache usage means you don’t need to have the same URI all over the place (scattered across object-specific graphs), but may be one of the only LDcache implementations in existence.

Most of us have implemented this by overriding our objects’ to_sorl method to do deep indexing into the graph (e.g., to extract the proper labels from nested resources). This also may require expensive periodic full reindexing.

How much of this stuff do we want or need for Hydra Works? Apparent consensus that descriptive metadata seems orthogonal to these needs, though still of great interest. These steps to improve usage of controlled vocabularies within our descriptive metadata were identified (in order of complexity) by Trey Pendragon and much nodding was seen:

1. Handle only the CVs we know about as Ruby entities
2. Use QA against those CVs, and return JSON
3. Do #2 fast and for arbitrary remote data sets
4. Index complex objects (labels)
5. Display indexed complex objects and maintain them over time

Actions

* A loose “working group” will be formed by some combination of Trey Pendragon, Unknown User (escowles@ucsd.edu), Tom Johnson, Karen Estlund, and Lynette Rayle to explore in more detail the topic of better (and more performant) support for linked data in our tooling. They will use the hydra-tech list and Hydra Tech calls to solicit interest, organize effort, and report progress.
* UCSD may work on extending DPLA MAP for Works with PREMIS for rights
* The Hydra Works application profile should include usage guidelines for baseline metadata (look at DC and DPLA MAP)