Infrastructure - Decision #8601

Decide on a URI space for DataONE resources

2018-06-05 00:39 - Bryce Mecum

Status:	New	**Start date:**	2018-06-04
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Milestone:	None		

Description

Summary

In the past, we've needed to represent DataONE resources (e.g., Objects, "datasets", etc.) in Linked Open Data contexts. Currently, these resources don't have have canonical URIs.

For example, take a dataset on search.dataone.org with the following URL:

```
https://search.dataone.org/#view/doi:10.18739/A28K74W2F
```

Candiate URIs for this resource include:

1. https://search.dataone.org/#view/doi:10.18739/A28K74W2F: Depends on implementation details in MetacatUI's router, uses fragment URLs in the URL which we're deprecating soon anyhow
2. https://cn.dataone.org/cn/v2/resolve/doi:10.18739/A28K74W2F: Depends on implementation details of the DataONE API and is tied to a specific version of the DataONE API. e.g., When API version 3 is release, will ../v2/resolve/doi:10.18739/A28K74W2F and ../v3/resolve/doi:10.18739/A28K74W2F refer to the same resource?

Proposal

Create a URI space that all services can integrate against. This space follows the convention of:

https://dataone.org/{resource_type}/{resource_identifier}

where {resource_type} is a singular name of a top level DataONE resource such as "dataset", "person", or "object" and {resource_identifier} is a type-appropriate identifier (e.g., PID of a science metadata Object for the dataset type, DN for the person type, etc.). The collection of top level resources (e.g., datasets), follows the form https://dataone.org/{resource_type_plural} (e.g., "datasets").

Examples:

- Dataset: https://dataone.org/dataset/doi%3A10.18739%2FA28K74W2F
- Person: https://dataone.org/person/https%3A%2F%2Forcid.org%2F0000-0002-0381-3766
- Object: https://dataone.org/object/urn%3Auuid%3A3c80e9d6-277c-4a32-bc7a-d85c499f370f
- All datasets on DataONE: https://dataone.org/datasets

Expected outcomes

- A normative document describing the URI space will be added to the DataONE documentation (https://releases.dataone.org/online/api-documentation-v2.0/)
- Other project will make use of these URIs

Affected projects

- Metacat (sitemap functionality will make use of these URIs)
- MetacatUI (Google Structured Data integration via JSON-LD will use these URIs for resources)

- GeoLink (DataONE LOD graph will use these URIs for resources)

Future work

With the URI space decided, we can start working on a unified, content-negotating DataONE resolve service (different from the CN/MNStorage.resolve API method). See https://hpad.dataone.org/GYJgjAJgpgHMwFoBGBWALItBjGMEE4kBDAZgKzSWBSS2ADYoSg==#detail-object-content-negotiation-for-objects. How this works is not being decided in this Redmine ticket.

Previous work / discussions (chronological order):

- https://docs.google.com/document/d/1yU-d-aFdtiSB91Wk0sFj1xthW8skBPof9qKwx00bjxE/edit?usp=sharing
- https://hpad.dataone.org/GYJgjAJgpgHMwFoBGBWALItBjGMEE4kBDAZgKzSWBSS2ADYoSg==

History

#1 - 2018-06-05 05:32 - Matthew Jones

I am generally in favor of these canonical linked data URIs for our principal objects, and I like this proposal with one minor change. In general with REST endpoints, I think the pluralized collection should be the container for the individual objects, rather than having a separate pluralized collection endpoint and a singular item endpoint. It makes it easier to distinguish the collection from the individual items **in** the collection. So, in your examples, they would become:

Examples:

- Dataset: https://dataone.org/datasets/doi%3A10.18739%2FA28K74W2F
- Person: https://dataone.org/people/https%3A%2F%2Forcid.org%2F0000-0002-0381-3766
- Object: https://dataone.org/objects/urn%3Auuid%3A3c80e9d6-277c-4a32-bc7a-d85c499f370f
- All datasets on DataONE: https://dataone.org/datasets
- All people on DataONE: https://dataone.org/people
- All objects on DataONE: https://dataone.org/objects

My two cents. There are certainly people out there that argue to have a plural collection and singular item, but I find it confusing to do so. I think the following design principles are nicely reasoned and thought out, and represent common practice:

- https://blog.philipphauer.de/restful-api-design-best-practices/

#2 - 2018-06-05 19:02 - Bryce Mecum

Thanks for the feedback. I'm glad you caught that as I imagine the singular/plural thing is the thing about this proposal people are more likely to differ on. I'm okay with consistently using plural forms as you suggested, especially if others agree with it.

#3 - 2018-06-05 19:23 - Bryce Mecum

Talked to Dave V on Slack. He asked:

> is there a vocabulary of resource_type that we can leverage?
> That's really the sticking point I think. Maybe we could leverage schema.org types as the "resource_type" ?

I asked for some pros/cons and Dave said:

plus is that we're not introducing terms that are possibly used in other places in a different way. Negative is that we have to be sure that the terms we choose to reuse align with their intent and use in other contexts

#4 - 2018-06-05 19:25 - Bryce Mecum

Maybe https://www.w3.org/TR/vocab-dcat/ is a starting point?

#5 - 2018-06-05 22:01 - Dave Vieglais

What is the list of resource_types that need to be supported? Quick draft (while listening to yet another discussion on the "taxonomic concept"):

- Object: Any component of a Dataset
- Dataset: The set of Objects aggregated in a resource map (including aggregations of resource maps).
- Agent: An entity that can authenticate and may create, modify, or access an Object, a person is a type of Agent.
- Catalog: Equivalent to a Member Node
- Type: equivalent to a formatId

#6 - 2018-06-05 22:53 - Bryce Mecum

Right now: Just Dataset, so I can merge some changes into MetacatUI and Metacat.

In the future: Probably also Person, Organization, maybe Project.

In the far future: Other things we haven't yet thought of?

#7 - 2018-06-11 18:49 - Bryce Mecum

Modeling people and orgs as Agents is interesting. It's programmatically difficult in some contexts to distinguish a person from an organization and Agent nicely skirts around making that distinction. I also like your idea of including Catalog and Type. I could see using Catalog for sure, maybe Type too.

Would this be good to discuss a bit more here, on Slack, or on a call sometime soon? For now, all I want to decide on is the URI space for datasets while designing in space for us to model the other types of resources listed here.

#8 - 2018-06-11 20:36 - Matthew Jones

Good discussion. BCO-DMO uses the following linked data categories (https://www.bco-dmo.org/data):

- Programs
- Projects
- Deployments

- Platforms
- Datasets
- Parameters
- Instruments
- People
- Affiliations
- Funding
- Awards

I think a number of these should be in scope for us as well, particularly Projects, Programs, Awards, and Affiliations (which is for organizational affiliation). The Program breakdown is particularly useful in showing our value to funders (as would a Funder top-level breakdown). Some of the others would be of interest too but we have less chance of standardizing that across DataONE. Can we get agreement on the datasets and objects endpoints, then continue the discussion on the rest?

#9 - 2018-06-11 23:47 - Bryce Mecum

^ That makes sense.

Before I forget (again): We also need to determine which type of Object we reference by PID in our /datasets/X URIs. The metadata PID (as DataONE Search does) or the Resource Map PID? Referencing by metadata PID isn't a reliable way to refer to a "dataset" in DataONE so I'd prefer to use the resource map PID. I think this is possible to do from MetacatUI without too much pain.

#10 - 2018-06-12 02:54 - Dave Vieglais

Agree that the Dataset URIs should point to the Resource Map PID since that contains pointers to the components of the Dataset.

What is the expected behavior? i.e. are there any plans for supporting content negotiation of any sort? When a user requests /Datasets/{PID} in their browser, do they get an ORE document or do they get a view of the Dataset, perhaps as a HTML page with json-ld pointers to the components of the Dataset and augmented with properties from the metadata?

Also, what is the response when visiting /datasets/? Presumably a list of dataset PIDs, but what if the total is say a million or more? Are there paging semantics? Is this the same thing as DataONE /v2/object?formatId=http%3A%2F%2Fwww.openarchives.org%2Fore%2Fterms ?

#11 - 2018-06-12 17:15 - Bryce Mecum

Hey Dave, good questions.

I purposely made this proposal minimal in order to reduce the number of things to decide on. Content negotiation and response implementation details are certainly w/in scope to discuss though. If those issues don't affect the URI space decision in this ticket, I'd still like to decide this before we move on.

Proposals for the details you mention are linked in the Previous work section in the original post. The short answer is that yes, fronting those URIs with resolvable endpoints at some point would be the goal, and that service could do content negotiation.

For 'dataset', browsers could go to MetacatUI when requesting */* or HTML and other clients could get appropriate responses when requesting rdf+xml andd ld+json. For 'datasets', I'd have to think on that more. A thing to do there could be to return a VoID Dataset description for appropriate requests (RDF/XML, JSONLD), or redirect to DataONE search for web browsers.

#12 - 2018-06-12 22:17 - Bryce Mecum

A note that just came up: If we go the route of using the Resource Map object's PID in the canonical URI for a Dataset, we also need to define the behavior when a dataset doesn't have a Resource Map which is pretty common.

#13 - 2018-06-12 22:33 - Matthew Jones

Yeah, our search system is oriented towards using the metadata PID as the main controlling PID that is displayed, and both the data files and RM redirect to the metadata PID view service. So, although the RM has some logical advantages, its not what we decided to do several years ago with the search system. Many sites assign their DOIs to their metadata docs specifically because of this behavior, so I'd not want to see them switching to a UUID for the RM as the main entrypoint for a package with a DOI assigned to metadata. So, I would argue that we should stick with the metadata doc being our main entry point for data sets to be consistent with how search and the view service works. If we want to switch to using the ORE PID, many sites would probably need to change their identifier assignment workflow/approach, which is a major change.

#14 - 2018-06-15 21:43 - Bryce Mecum

> So, I would argue that we should stick with the metadata doc being our main entry point for data sets to be consistent with how search and the view service works. If we want to switch to using the ORE PID, many sites would probably need to change their identifier assignment workflow/approach, which is a major change.

I agree with this.

Mark Schildhauer and I chatted a bit today about another option for the URI space: Making use of our existing purl.dataone.org space. i.e., https://purl.dataone.org/datasets/{PID}. This has some advantages, including:

- Signals our intent to maintain that URI space. i.e., this is not just a URI but a PURL
- It's easier to deploy the resolve service separately from other dataone.org components

Any thoughts on this?

#15 - 2018-06-17 11:21 - Matthew Jones

I think that's an interesting proposal.

It would be nice if the URI space for our dataset landing pages were integrated in with our search application. For some reason I think this would be easier to do if we weren't using the purl because the https://dataone.org/datasets/{PID} can fit in with MetacatUI if MetacatUI is mounted at the root of dataone.org (probably complicated though). Alternatively, we could root our persistent URIs at https://search.dataone.org for ease of mounting the app. Either way, I wouldn't want all MetacatUI URIs to be PURLS. If the persistent url space is outside of our search app URL space, then our persistent URIs for datasets will always have to be a redirect into the MetacatUI view space, which seems confusing to me. I'd rather have just one URI.

#16 - 2018-06-17 18:43 - Dave Vieglais

A couple comments:

I disagree that only the metadata PID should be the dataset PID. The metadata document does not contain relations to other components of the dataset, so it is not obvious how to find those other components. It is also contrary to the operational practice of other repositories such as EDI for example.

Instead, the behavior of /dataset/PID should be that PID may be any PID of any component (data, metadata, or the resource map) for a dataset. That behavior is compatible with the practices in place. Potential problem may be if a component is used in more than one dataset.

I'm a bit hesitant of using purl.dataone.org as an API endpoint - it is not setup for that and all the APIs are at cn.dataone.org. Seems like it would be fragmenting the API endpoints without compelling justification. Access control would add a layer of complexity as well. My preference would be to use dataone.org/dataset or perhaps dataone.org/api/dataset before purl.dataone.org.

#17 - 2018-07-17 19:02 - Bryce Mecum

Thanks Dave. I think your idea to make {PID} in {whatever}/datasets/{PID} be any Object is perfectly fine.

Matt has been lobbying for us to homogenize our URLs a bit here: Basically, why do we need MetacatUI and our LOD URI space to have different URI structures anyway? What if both had the same URIs? e.g., MetacatUI could switch from serving on https://search.dataone.org to https://dataone.org.

Dave: What do you think the feasibility of this is? I imagine there's a historical reason the Search UI is mounted at the subdomain search.dataone.org.

If it's hard to mount MetacatUI at dataone.org and we make it a goal to have the LOD URI space and MetacatUI have the same URI space, I think that would lead us to use a LOD URI space of search.dataone.org/datasets/{PID} instead of dataone.org/datasets/{PID}. I'm totally okay with these URI spaces separate but understand we don't all feel this way.

#18 - 2018-07-18 19:53 - Matthew Jones

I think one major advantage of rooting the URI space at https://dataone.org/datasets and homogenizing so that our search app and the LOD URIs are the same is that it helps when people want to cite data. Basically, they currently copy the #view URI from the browser location bar and cite that all over the place, and I think it would be best if the browser showed our LOD URI for the location of the landing page.

#19 - 2018-07-18 21:19 - Dave Vieglais

Using dataone.org/dataset/{PID} makes sense to me, and similarly for other access URIs. We should be able to promote that as the citable or at least reusable link to the data when being accessed through DataONE.

There's no particular functional reason for using separate domain names for search and cn other that the simplicity of such an approach for technical implementation.

Enabling that scheme is a bit complicated right now because of the public website sitting at https://dataone.org/. It will be necessary to move the website to another location, a process that involves a few headaches, but the end result will be worth the effort.

#20 - 2018-07-18 21:42 - Bryce Mecum

Glad to hear it your thoughts on the technical details, Dave, thanks! Sounds like it's a go to use https://dataone.org/datasets/{PID} in our JSON-LD for now and we can hash out the details of building out the content-negotiation and resolution service and changing where MetacatUI is deployed at a later date.

I'll leave this issue open for a week but I'm moving forward with shipping this URI space in MetacatUI's JSON-LD feature.

Before this is closed, I will write this up on the official API docs.

Thanks all!