



A GA4GH Data Repository Service for iRODS

Mike Conway
Data Systems Architect/Engineer
National Institute of Environmental Health Sciences



NIEHS Office of Data Science

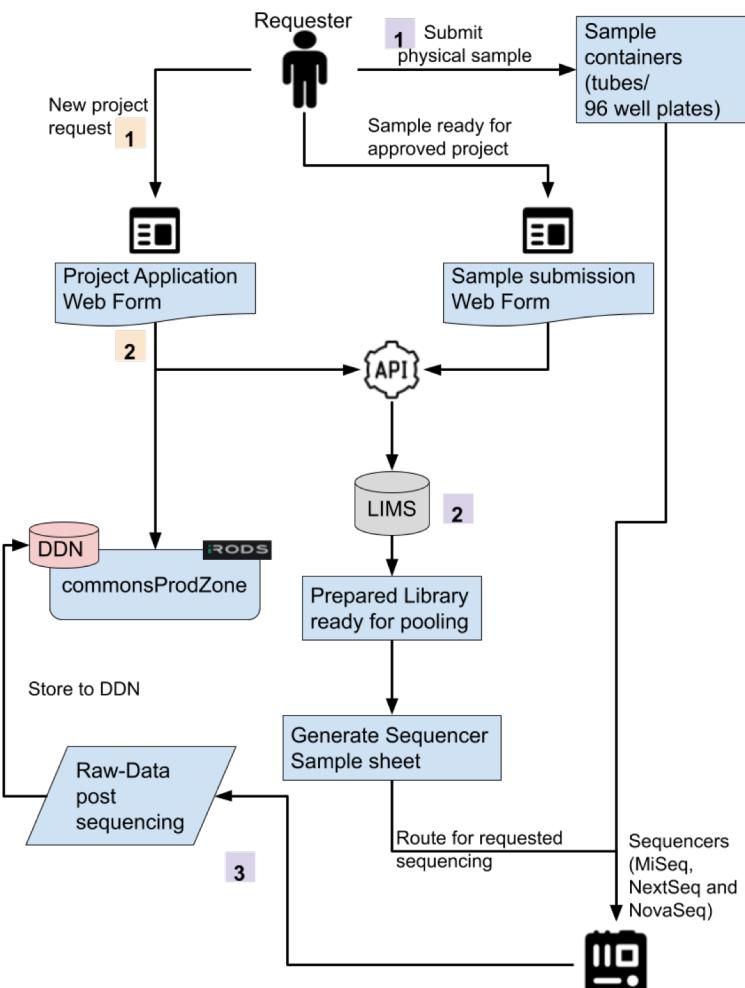
Developing an NIEHS Data Commons

Developing a Commons to manage research data, using iRODS as a platform for unifying and managing local and cloud resources.

The screenshot shows a web-based interface for managing research data. At the top, there's a navigation bar with links for 'For Lab Staff', 'For Office Staff', 'For Managers', 'Working Here', 'Computer Support', 'Library', 'News', 'Policies & Ethics', 'Purchasing', 'Safety & Security', 'Training', and 'Divisions'. Below the navigation bar, a sidebar on the left lists various management functions: Resources, Rules, Users, Groups, Profiles, Collections, Search, Templates, Shared Links, Favorites, Public, and Trash. The main content area displays a list of files under the heading 'Action' with columns for 'File Type', 'Modified', 'Size', and 'Action'. The files listed are:

File Type	Modified	Size	Action
hetal	Mar 12 2018 15...	-	View Info Edit
i have spaces	Jun 25 2018 15...	-	View Info Edit
testupload	Sep 19 2018 15...	-	View Info Edit
DataCommonsDesigns_01312018.pdf	Feb 01 2018 18...	1.2 MB	View Info Edit

Data Commons integrated with processing pipelines and workflow systems.



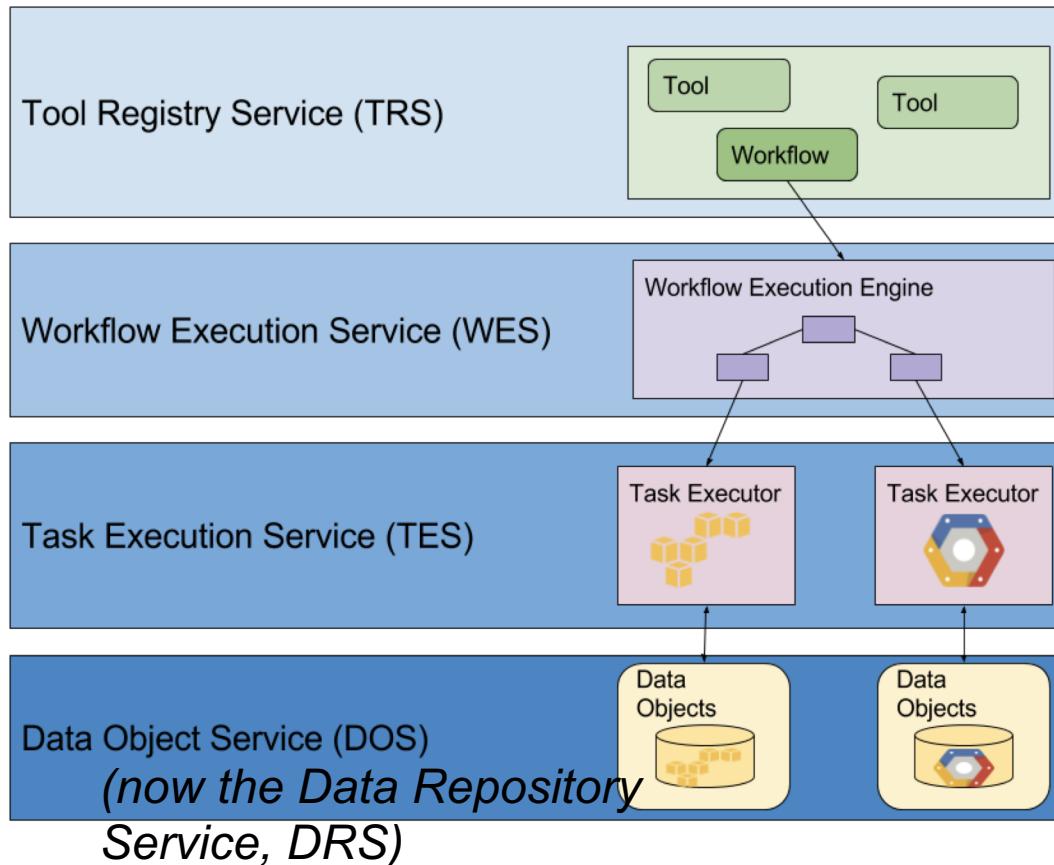
Use Case:

- Data Commons as the hub for managing research projects in an ISA model
- Sample submission integrated with Clarity LIMS triggers NextFlow pipelines
- Data Commons as delivery mechanism gathering metadata and pipeline results

Setting future strategy anticipating move to cloud over time, with a hybrid of local research data, published artifacts and tiered storage in the cloud.

How can we develop strategies that work for cloud and local use cases?

GA4GH Cloud Work Stream APIs



**Sharing Tools
and Workflows**

**Executing
Workflows**

**Executing
Individual Tasks**

Accessing Data

O'Connor, Frian, and David Glazer. n.d. "20190319 - GA4GH Cloud Work Stream Overview - Google Slides."

Accessed June 25, 2019.

https://docs.google.com/presentation/d/1_MFTCw1uDrfNtbki2Nvyh2I2IYOIQKTHmrZgMTspdM4/edit#slide=id.g54dc8a46d6_0_0.



GA4GH Data Repository Service

Described by GA4GH:

“The Data Repository Service (DRS) API provides a generic interface to data repositories so data consumers, including workflow systems, can access data in a single, standard way regardless of where it’s stored and how it’s managed. The primary functionality of DRS is to map a logical ID to a means for physically retrieving the data represented by the ID.”



Search or jump to... Pull requests Issues Marketplace Explore

Unwatch 1 Star 0 Fork 1

michael-conway / irods-ga4gh-dos

Code Issues 2 Pull requests 0 Projects 0 Wiki Security Insights Settings

GA4GH Data Object Service for iRODS

Manage topics

65 commits 2 branches 0 releases 1 contributor View license

Branch: master New pull request Create new file Upload files Find File Clone or download

Mike Conway #1 add mime type detection Latest commit 300d8ed 1 hour ago

etc/irods-ext #1 preparing docker 20 hours ago

ga4gh-bundle-management-service #1 completed tests for demo 1 hour ago

ga4gh-common #1 add mime type detection 1 hour ago

ga4gh-dos-service #1 completed tests for demo 1 hour ago

swagger #1 restructure for drs last month

.gitignore #1 access methods work 25 days ago

LICENSE.txt init swagger last year

README.md Update README.md 5 months ago

eclipse-formatting.xml init swagger last year

pom.xml #1 access url at service layer 4 days ago

sample-settings.xml #1 restructure for drs last month

README.md

GA4GH Data Object Service for iRODS

This is an implementation of the GA4GH data object service for the iRODS data grid, based on the iRODS Jargon API.

- GA4GH DRS implementation for ‘native’ iRODS collections.
- Service to designate an iRODS Collection ‘in place’ as a Data Bundle.
- URL creation, including ticket based access via https are supported.
- Low barrier to entry, no special setup, stateless Docker image.

<https://github.com/michael-conway/irods-ga4gh-dos>

Demo – Designate an iRODS Collection as a Data Bundle

Code snippet designates a collection root as a bundle

```
79 String bundleRoot = irodsCollectionRootAbsolutePath + "/" + bundleDir;
80 DosConfiguration dosConfiguration = new DosConfiguration();
81 DosServiceFactory factory = new ExplodedDosServiceFactoryImpl(irodsFileSystem.getIRODSAccessObjectFactory());
82
83 DosBundleManagementService explodedDosService = new ExplodedDosBundleManagementServiceImpl(
84     irodsFileSystem.getIRODSAccessObjectFactory(), irodsAccount, factory, dosConfiguration);
85 String guid = explodedDosService.createDataBundle(bundleRoot);
86 Assert.assertNotNull("no guid returned", guid);
87
```

Marks bundle with AVUs for GUID and checksum of checksums

```
test@server1:~$ imeta ls -C /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle
AVUs defined for collection /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle:
attribute: ga4gh:bundleChecksum
value: d68d4685be3b13168b63bb31df1c0a4ee9c267a0cbe300869b08768c43d9b6dd
units: irods:ga4gh:dos
-----
attribute: ga4gh:bundleId
value: aa713031-e421-4783-b522-068b63020def
units: irods:ga4gh:dos
test@server1:~$
```

Demo – Designate an iRODS Collection as a Data Bundle

Child objects (nested) flattened and marked as a Data Object. GUID is added as AVU and checksum is computed.

```
test@server1:~$ ils
/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0:
testFile0.txt
testFile1.txt
testFile2.txt
testFile3.txt
testFile4.txt
testFile5.txt
testFile6.txt
testFile7.txt
testFile8.txt
C- /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/testPutCollectionWithTw
oFileslvl2nbr0
 C- /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/testPutCollectionWithTw
oFileslvl2nbr1
 C- /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/testPutCollectionWithTw
oFileslvl2nbr2
 C- /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/testPutCollectionWithTw
oFileslvl2nbr3
test@server1:~$ imeta ls -d /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/
testFile0.txt
AVUs defined for dataObj /zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle/testPutCollectionWithTwoFileslvl3nbr0/tes
tFile0.txt:
attribute: ga4gh:dataObjectId
value: 4de8f3ee-73c0-4e47-9f95-617e05998d91
units: irods:ga4gh:dos
test@server1:~$
```

Running DRS via Docker – Swagger API

The screenshot shows the Swagger UI interface for the Data Repository Service. At the top, there's a green header bar with the 'swagger' logo, a dropdown menu set to 'default (/api-docs)', and a 'Explore' button. Below the header, the title 'Data Repository Service' is displayed, along with its GitHub repository link (<https://github.com/ga4gh/data-repository-service-schemas>) and contact information ('Contact the developer' and 'Apache 2.0'). The main content area is titled 'DataRepositoryService' and contains four API operations listed in a table:

DataRepositoryService		Show/Hide List Operations Expand Operations
GET	/bundles/{bundle_id}	Get info about a Data Bundle.
GET	/objects/{object_id}	Get info about a Data Ga4ghObject.
GET	/objects/{object_id}/access/{access_id}	Get a URL for fetching bytes.
GET	/service-info	Get information about this implementation.

At the bottom left, there's a note: '[BASE URL: / , API VERSION: 0.6.0]'

Service Info - Configurable

404 Not Found

Try it out!

[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' 'http://localhost:8080/service-info'
```

Request URL

```
http://localhost:8080/service-info
```

Request Headers

```
{
  "Accept": "application/json"
}
```

Response Body

```
{
  "version": "0.0.1-SNAPSHOT - 2019-06-25T14:40:59Z",
  "title": "iRODS GA4GH Data Repository Service Demo",
  "description": "",
  "contact": "NIEHS Office of Data Science",
  "license": "BSD 3-Clause"
}
```

Response Code

```
200
```

Response Headers

Retrieve a Data Bundle via GUID

```
curl -X GET --header 'Accept: application/json' 'http://localhost:8080/bundles/aa713031-e421-4783-b522-068b63020def'
```

Request URL

```
http://localhost:8080/bundles/aa713031-e421-4783-b522-068b63020def
```

Request Headers

```
{  
  "Accept": "application/json"  
}
```

Response Body

```
{  
  "id": "aa713031-e421-4783-b522-068b63020def",  
  "name": "/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle",  
  "size": "0",  
  "created": "2019-06-25T11:15:11Z",  
  "updated": "2019-06-25T11:15:11Z",  
  "version": "0",  
  "checksums": [  
    {  
      "checksum": "d68d4685be3b13168b63bb31df1c0a4ee9c267a0cbe300869b08768c43d9b6dd",  
      "type": "sha256"  
    }  
,  
    "description": "iRODS exploded bundle collection",  
    "aliases": [  
      "/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle"  
    ],  
    "contents": [  
      {  
        "name": "testFile0.txt",  
        "size": "0",  
        "checksum": "d68d4685be3b13168b63bb31df1c0a4ee9c267a0cbe300869b08768c43d9b6dd",  
        "type": "sha256"  
      }  
    ]  
  ]  
}
```

Checksum of Checksums

Response Code

```
200
```

Response Headers

Data Bundle links to child Data Objects

```
"/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementServiceImplTest/testCreateDataBundle"
,
contents": [
  {
    "name": "testFile0.txt",
    "id": "4de8f3ee-73c0-4e47-9f95-617e05998d91",
    "drs_uri": [
      "http://localhost:8080//objects/4de8f3ee-73c0-4e47-9f95-617e05998d91"
    ],
    "type": "object"
  },
  {
    "name": "testFile1.txt",
    "id": "9ade673d-83df-4d2e-ad6b-a468ff9c0243",
    "drs_uri": [
      "http://localhost:8080//objects/9ade673d-83df-4d2e-ad6b-a468ff9c0243"
    ],
    "type": "object"
  },
  {
    "name": "testFile2.txt",
    "id": "4de8f3ee-73c0-4e47-9f95-617e05998d92",
    "drs_uri": [
      "http://localhost:8080//objects/4de8f3ee-73c0-4e47-9f95-617e05998d92"
    ],
    "type": "object"
  }
]
```

Child Data Objects listed in bundle with GUID and access URI

Accessing a Data Object by GUID

Curl

```
curl -X GET --header 'Accept: application/json' 'http://localhost:8080/objects/27119224-78ff-42f6-8e99-a64d39e092ec'
```

Request URL

```
http://localhost:8080/objects/27119224-78ff-42f6-8e99-a64d39e092ec
```

Request Headers

```
{
  "Accept": "application/json"
}
```

Response Body

```
{
  "id": "27119224-78ff-42f6-8e99-a64d39e092ec",
  "name": "testFile3.txt",
  "size": 144,
  "created": "2019-06-25T11:15:30Z",
  "updated": "2019-06-25T11:15:30Z",
  "version": "",
  "mime_type": "text/plain",
  "checksums": [
    {
      "checksum": "zwSgAvMofVr0R4j1zx/LJ8uGjzPSUnt0KY0fxR41fyk=",
      "type": "SHA256"
    }
  ],
  "access_methods": [
    {
      "type": "file",
      "access_url": {
        "url": "irods://test1@server1.local:1247/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagementService"
      }
    }
  ]
}
```

Individual iRODS file as a Data Object, has GUID, checksum, computed MIME Type

Generating an Access URL on demand

An access method without a URL requires a call to obtain the URL.
In this case generating an iRODS ticket on demand for read access.

Try it out!

[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' 'http://localhost:8080/objects/27119224-78ff-42f6-8e99-a64d39e092ec/acc
```

Request URL

http://localhost:8080/objects/27119224-78ff-42f6-8e99-a64d39e092ec/access/irods-rest

Request is for GUID and
Access id

Request Headers

```
{  
  "Accept": "application/json"  
}
```

Response Body

```
{  
  "url": "http://example.com/irods-rest/fileStream?path=/zone1/home/test1/jargon-scratch/ExplodedDosBundleManagement",  
  "headers": [  
    "Authorization : Bearer IxXb42lajFdNsAr"  
  ]  
}
```

iRODS Ticket is generated on demand, can be shared



Next Steps

- Complete packaging and unit tests
- Validation with GA4GH
- Incorporate the ability to attach descriptions to bundles and data objects
- Beta release
- Implement https download access as first service in new irods-rest REST API revision
- Possible command line tool or rule set:
 - CRUD on bundles
 - Rules enforcing optional immutability?
- Possible ‘quick download’ util that can download irods:// URIs via high speed transfer

What iRODS needs!

- Focus on i/o performance of streaming.
- Standard way of computing MIME type (via extension inspection or optional file content scanning) and storing computed MIME type for subsequent query.
- Possible iCommand support for irods:// URI download
- Work with GA4GH to put iRODS semantics into the mix in DRS, add to CI.
- Standard notion of a file ‘Description’, is it the ‘comment’? Is it a standard AVU?
- Mark as ‘immutable’ at collection level?



Thank You!

Mike Conway
NIH/NIEHS
Office of Data Science

<https://www.niehs.nih.gov/research/atniehs/dntp/osim/index.cfm>

mike.conway@nih.gov

GitHub:

<https://github.com/michael-conway/irods-ga4gh-dos>