iRODS and NIEHS Environmental Health Science

Mike Conway – mike.conway@nih.gov

Deep Patel – deep.patel@nih.gov

National Institute of Environmental Health Sciences

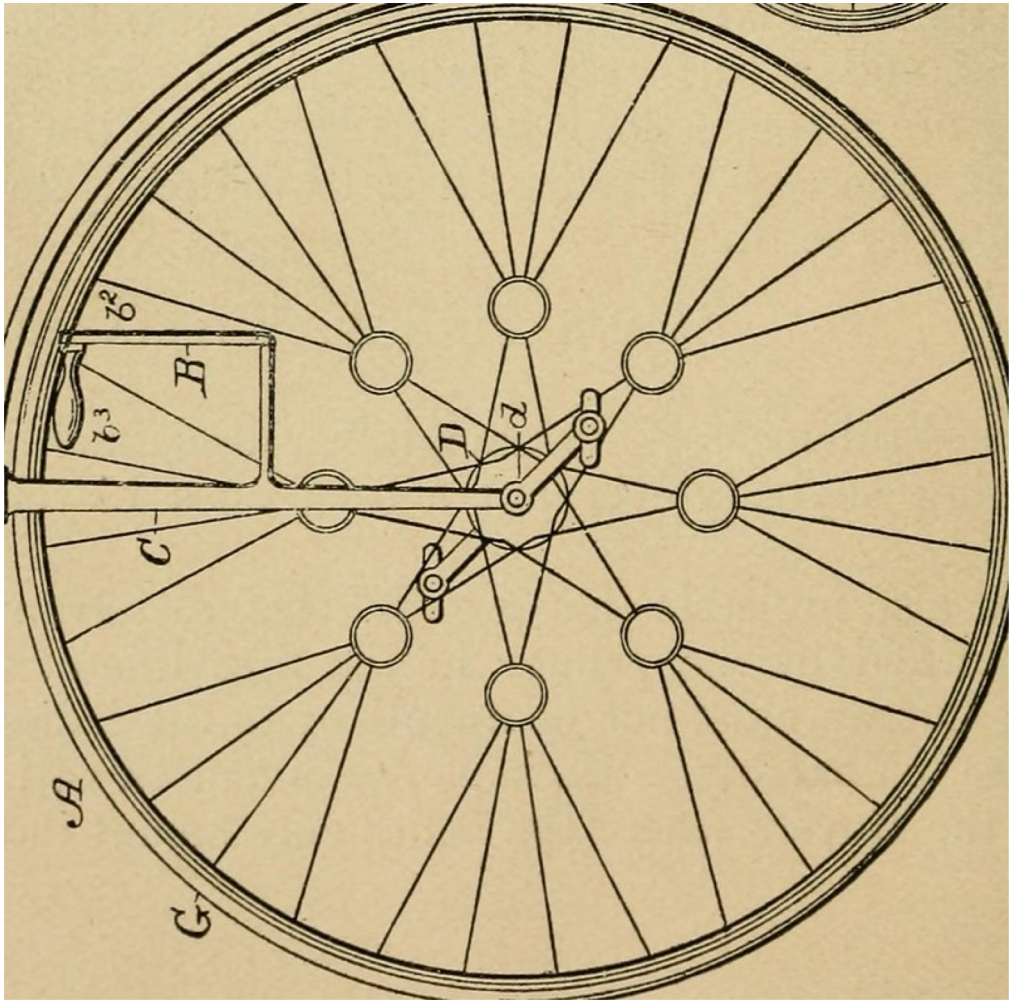
What we do now

- Focus on environmental health, managing intramural research data
- iRODS is heavily involved in managing sequencing data
- We are tracking the NIH Strategic Plan for Data Science
- We are getting more and more involved with the Health Sciences community via GA4GH

Building Flexibility is the theme

- Real world problems
- Cheap and simple solutions that don't pollute community code
- In the spirit of open source we want to develop broadly usable tools/patterns that can provide foldback of effort from the community
- How to add microservices around core iRODS capabilities that simplify common tasks
- Look at how some of these raw materials lead towards some 'bigger fish' that the iRODS community can tackle together

iRODS Role



iRODS strength is as a hub that can serve as the canonical data and metadata store

- Monitor and manage the data
- Index and leverage existing search technology, offloading the search problem to existing solutions
- Serve as a data curation hub, a curation ‘bus’ in a cloudy, multi-repository world

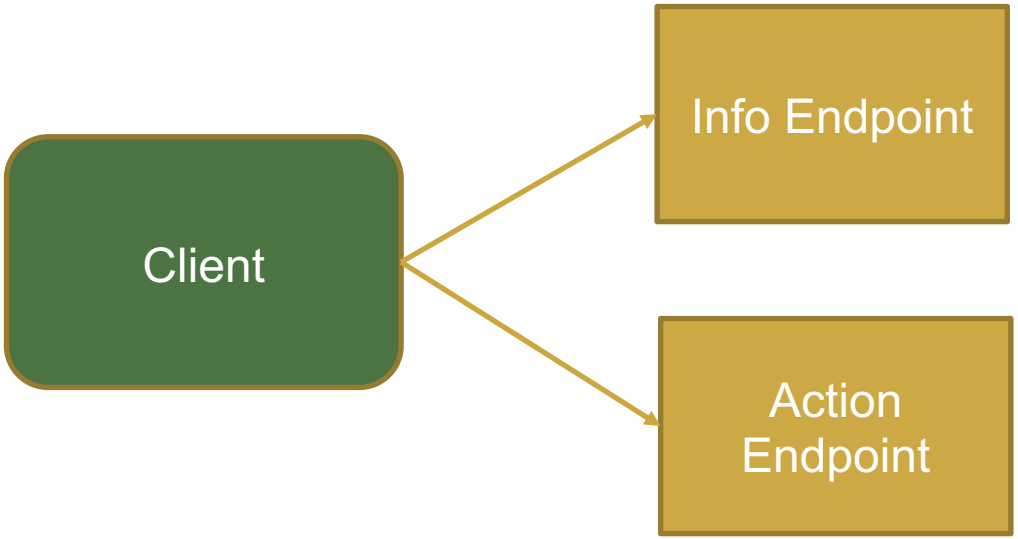
https://flic.kr/p/S8qcUo

Search and Indexing

https://flic.kr/p/2gZPyqN

File Carts and Publishing

Using REST-ful API with a general 'plugin' model for MetaInx or any other microservice architecture



Plugins advertise themselves, describe their capabilities, including user guidance

Carry out a focused action based on iRODS user in JWT claims

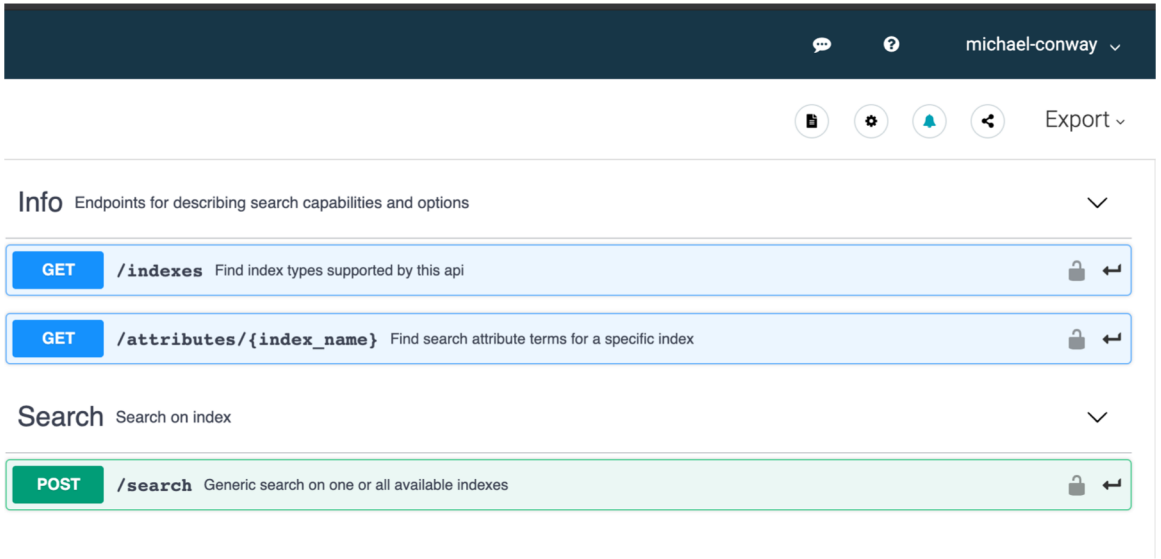
Client:

- Obtains a JWT by iRODS auth
- Has a configured list of endpoint addresses, interrogates info endpoints to establish the available plugins

Search!

- Simple way to plug in light-weight search capabilities (focused on a particular task/persona)
- Agnostic about underlying search technology, simpler than the QueryArrow approach?
- User friendly
  - Default Google-like search using baked in assumptions for unstructured strings
  - Attribute:Search Term advanced functions with wildcards/query operators for building more sophisticated searches
- UI and API-friendly output
  - Google like search results with sublinks
  - ils –LA listings

Search API



← Endpoint for registration
← Describe search schema

← Execute a search

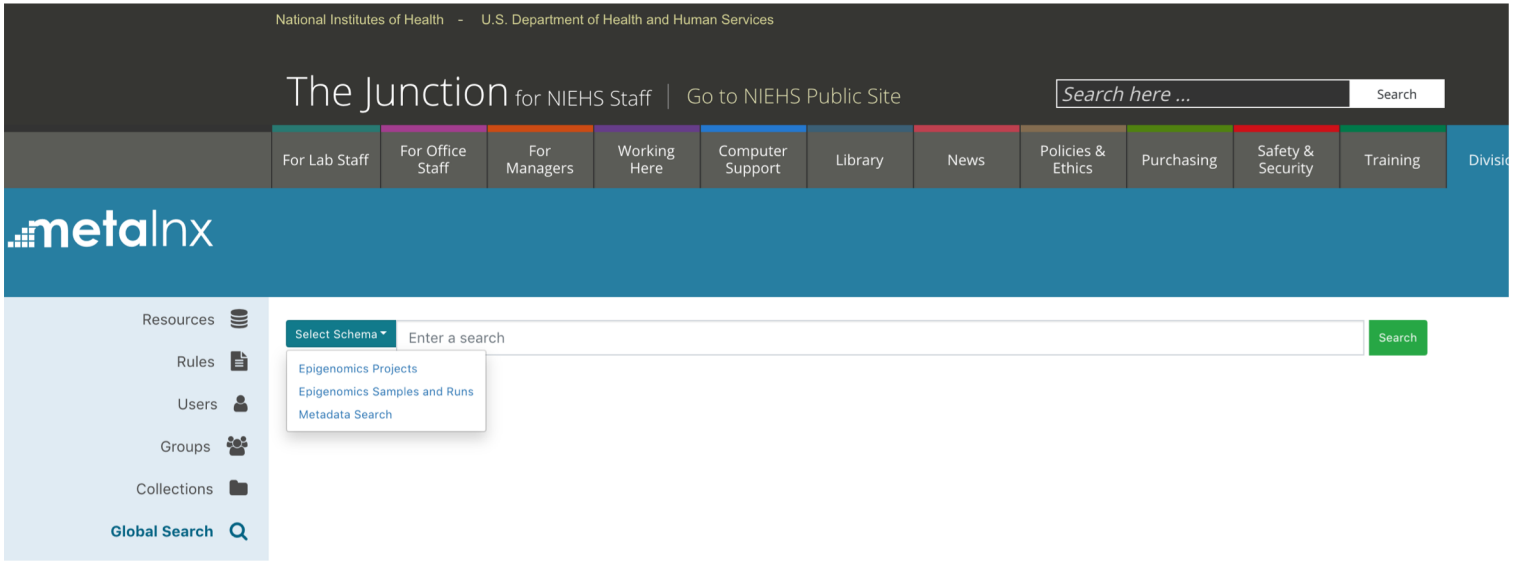
A search plugin has a schema that describes its purpose and targeted persona. The available search attributes are described by the /attributes endpoint.

Many small plugins versus one complicated, multifaceted plugin.

MetaInx configuration

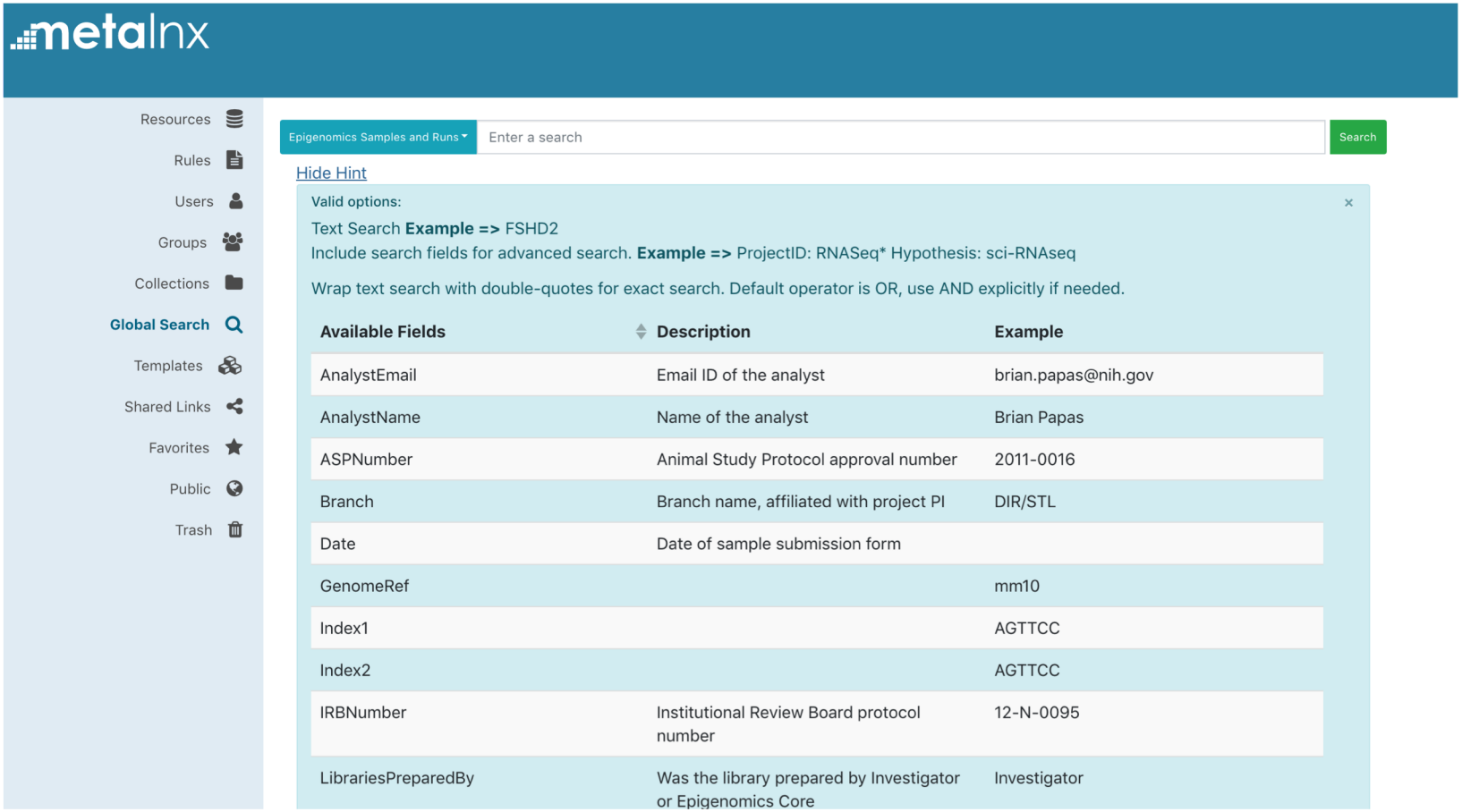
```
#################################
# Pluggable search configuration. Turn on and off pluggable search globally, and configure search endpoints.
# N.B. pluggable search also requires provisioning of the jwt.* information above
#################################
# configured endpoints, comma delimited in form https://host.com/v1
pluggablesearch.endpointRegistryList=http://proj_sample_search:8082/v1,http://metadata_search:8082/v1
# enable pluggable search globally and show the search GUI components
pluggablesearch.enabled=true
classicsearch.enabled=false
# JWT subject claim used to access search endpoint for data gathering. User searches will utilize the name of the individual
pluggablesearch.endpointAccessSubject=pluggablesearch
# timeout for info/attribute gathering, set to 0 for no timeout
pluggablesearch.info.timeout=0
# timeout for actual search, set to 0 for no timeout
pluggablesearch.search.timeout=0
```

Post-discovery in a web UI



Available schema have been discovered by polling the /info endpoints

Describing the schema attributes



Available Fields	Description	Example
AnalystEmail	Email ID of the analyst	brian.papas@nih.gov
AnalystName	Name of the analyst	Brian Papas
ASPNumber	Animal Study Protocol approval number	2011-0016
Branch	Branch name, affiliated with project PI	DIR/STL
Date	Date of sample submission form	
GenomeRef		mm10
Index1		AGTTCC
Index2		AGTTCC
IRBNumber	Institutional Review Board protocol number	12-N-0095
LibrariesPreparedBy	Was the library prepared by Investigator or Epigenomics Core	Investigator

Search Options

- Enter a plain string…
  - The plugin, appropriate to the persona, will translate into a default query
  - e.g. sample name, sample name unique, run id like NOVA01*
- Enter Attribute:Query
  - The plugin will fashion a query appropriate to the targeted attributes
  - e.g. GenomeRef:mm10 PreperationKit:Nextera
- Attribute:Query style can also support ‘builder’ type query interfaces in later iterations

Search result options

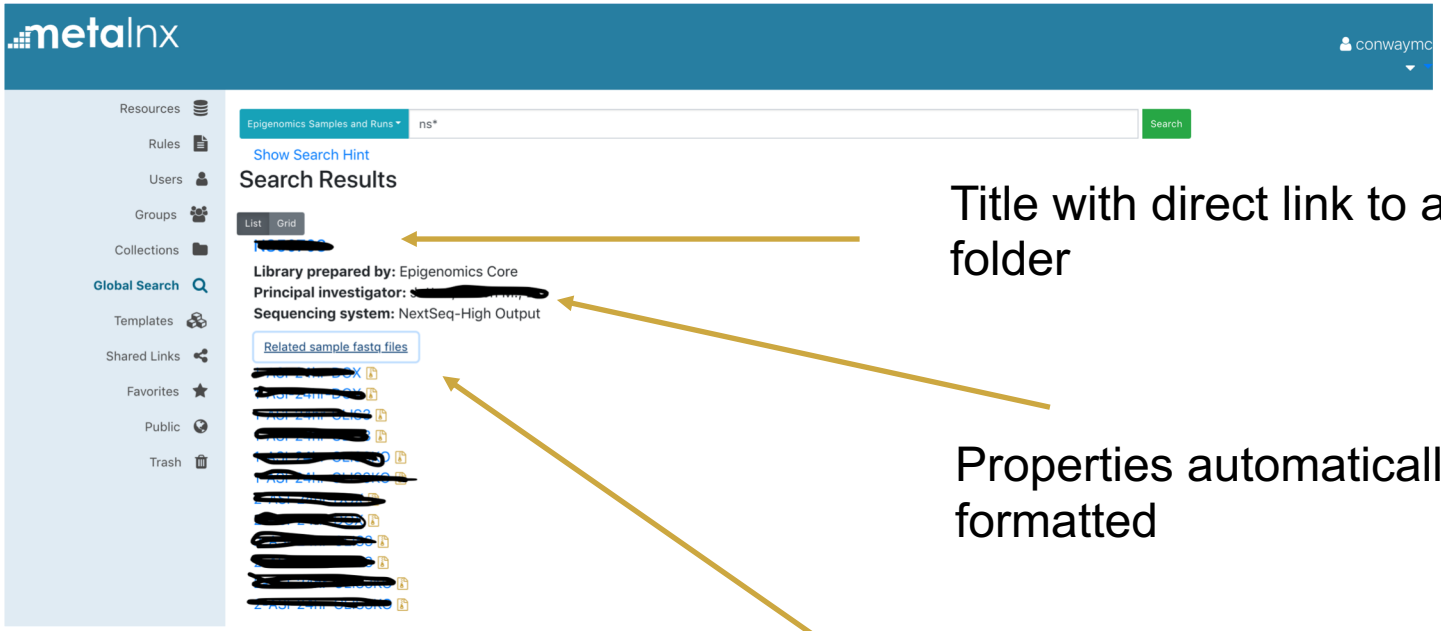
Search can support familiar web search results with customizable properties and sublinks

```
search_data ⌄ {                                                              ↵
    index-schema-
    description     index-schema-description › {...}↵
                    example: OrderedMap { "name": "name", "id": "id", "version": "version", "info": "info" }
    search_result   ⌄ [ ⌄ {
                      title                 string
                                            Descriptive title for search result
                      url_link              string
                                            Resolvable https link to result data location
                      subtitle              string
                                            Optional subtitle that can be presented as a highlight, publication info, etc
                      content_text          string
                                            Bag of attribute-value paired metadata attached to search hit
                      properties            result_properties › {...}↵
                      links                 search_data_linkset › {...}↵

                    }]

}
```

Results in familiar web search format with built-in support for headers and sub-links



Title with direct link to a run folder

Properties automatically formatted

Sub-links or ‘see also’ links

Grid style results now available

List Grid

Name	Modified	Size (kB)
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-
[redacted]	Wed Dec 31 1969 19:00:00 GMT-0500 (Eastern Standard Time)	-

Virtual Collections and FAIR data

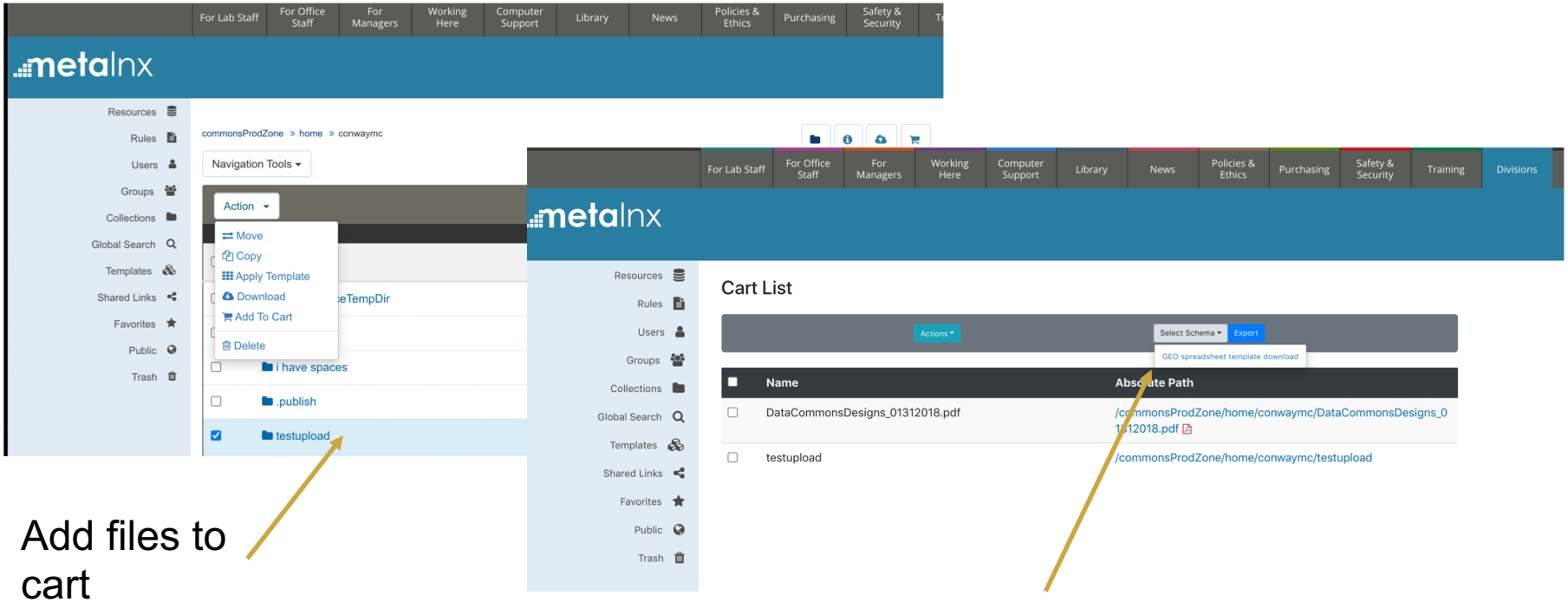
- Originally implemented in DFC but is fairly powerful when combined with this plugin pattern
- Any query that returns an ils –LA listing can serve as a virtual collection. Saved queries can become a new iRODS collection
- Virtual collections can span federations, these query endpoints could be added to ils, icd
- MetaInx already supports proxy pages (off by default) for query results that show available metadata and allows requests for access for ‘metadata is public, data is private’ scenarios

The iRODS indexing capability and the search plugin architecture are being aligned

- Current NIEHS file props and AVU search plugin – https://github.com/angrygoat/pluggable-search-props-and-metadata
- Current file props and AVU indexer implementation
  - Based on a utility visitor framework
    - https://github.com/DICE-UNC/jargon/tree/master/jargon-data-utils/src/main/java/org/irods/jargon/datautils/visitor
    - Filtering
    - Restart
    - Automatic metadata stack management
    - https://github.com/trel/metadata-and-file-props-indexer

Publication and the file shopping cart!

A file shopping cart is an iRODS ‘hidden’ file in the user home area that contains a manifest of selected paths in iRODS



Add files to cart

Apply publishing plugins to cart contents

Publishing plugin API

- Like the search plugin, publishing plugins advertise themselves through an info endpoint.
- Publishers are simple but can accommodate complex operations through integrating with an external endpoint.
- Options:
  - Publish and **return a notification**
  - Publish and **download** the results to the client
  - Publish and **follow a link to an iRODS path** as a result
  - Publish and **follow a link to an external site** to complete the action

GEO publishing

- Our current production scenario is to publish sequencing data to GEO (Gene Expression Omnibus). Currently this gathers data from our iRODS based Commons and generates a spreadsheet
- The publishing plugin is set to a processing type of ‘download’ so a spreadsheet is generated by the plugin
- Next step is go generate a zip with the selected FastQ files assembled together with the spreadsheet
- Final step would be to add an intermediate curation service that would capture user data entry and tie back to original data

iRODS for managing research data via FAIR principles in a multi-repository world

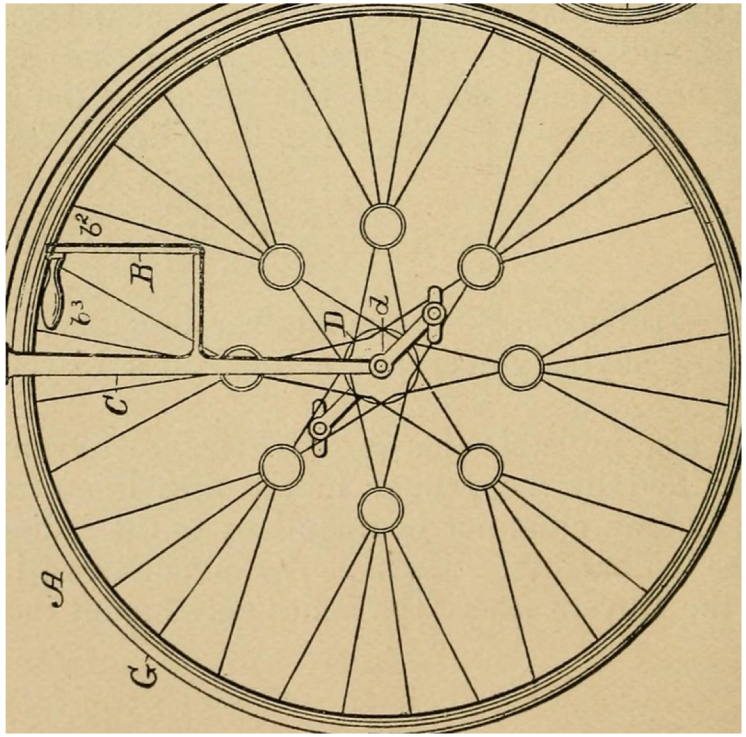
WHERE DO I PUT MY RESEARCH DATA?

https://flic.kr/p/8JLUEQ

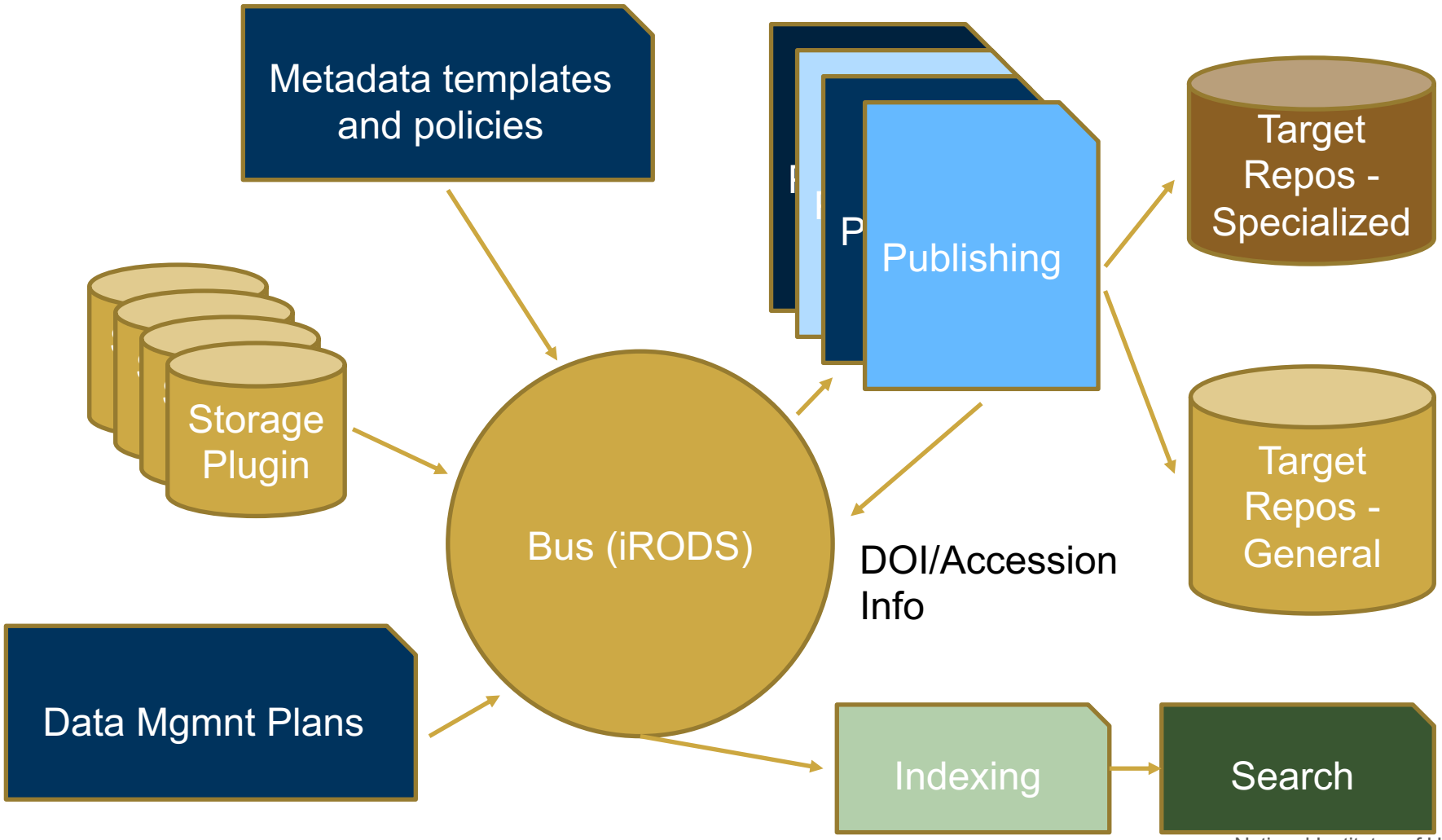
iRODS for preservation/archival storage with a twist

We need for a data submission ‘bus’ that can support:

- The assembly and curation of data sets and metadata models that are then pushed in segments to appropriate specialized repositories (e.g. dbGaP, GEO, SRA)
- The recording of DOIs and accession numbers via publishing
- Data model crosswalk to target repositories as part of publishing



Data flow through a data submission bus



Some useful references on this ‘bus’ concept

- COPO as a working example of a ‘bus’ (CyVerse UK) - https://f1000research.com/articles/9-495
- Geoscience Digital Data Resource and Repository Service (GeoDaRRS) Workshop Report - https://opensky.ucar.edu/islandora/object/technotes:570