OpenSearch-based Federated Space-Time Query

Chris Lynnes and the ESIP Federated Search Cluster
Why So Difficult???

- Example Use Case: fetch all the aerosol-related data for a volcanic ash plume

- Today:
  - Search Google, GCMD, ECHO, and individual data centers, each with its own search tool
  - Query colleagues, etc. for unpublished datasets

- Tomorrow:
  - User makes a single (federated) search for relevant datasets
  - Then a single (federated) space-time query for granules from desired datasets
Version 0 Solution

• Back in the day...
  ○ Federated dataset query of DAACs
  ○ Federated space-time query of DAACs for granules

• BUT...
  ○ Slow
  ○ Non-standard protocol
  ○ Expensive to implement individually at each site

• However, these are no longer impediments
  ○ Speed: fast networks have mitigated this
  ○ Standards: HTTP, OpenSearch
  ○ Cost: HTTP/OpenSearch are easy to add as thin layers on top of existing database query engines
Proposed Solution: OpenSearch

- OpenSearch Description Documents
  - Describe in machine-readable form how to form a URL to execute a query
    - e.g., http://mirador.gsfc.nasa.gov/cgi-bin/mirador/granlist.pl?
      dataSet=AIRIBRAD.005&page=1&maxgranules={count}&pointLocation={geo:box}&startTime={time:start}&endTime={time:end}&format=atom
  - Allows access to many heritage query servers simply by creating the description document

- Recursive OpenSearch Concept
  - Dataset-level search returns links to OpenSearch Description Documents for granule-level search

- Responses in Atom
  - With additional ESIP conventions (under development)
The Two-Step Query

- Rationale for splitting query into two steps
  - Most dataset-level queries have
    - low "precision": \( \text{precision} = \frac{\text{desiderata}}{\text{desiderata} + \text{dreck}} \)
    - small results set (dozens)
  - Space-time granule queries for a given dataset have:
    - large results set (tens of thousands), but
    - high precision
  - Combining the two in one step produces:
    - mammoth results set (dozens * tens of thousands)
    - with low precision
- Therefore, concept is:
  - Step 1: dataset search
  - Step 1.5: user / client selection of datasets
  - Step 2: granule search for selected datasets
Recursive OpenSearch

Discovery

Space-Time Query

Client

Dataset Query Engine

OpenSearch Description Document Repository

Granule Query Engine

keyword query

dataset results*

OpenSearch Description Document request

OpenSearch Description Document**

space-time query

granule results

*Dataset results include links to OpenSearch Description Documents

**OpenSearch Description Document includes template for forming space-time granule queries
OpenSearch Block Diagram

Client Application

OpenSearch Environment

(1) Discover

(2) Query

Data Set Description (XML)

(3) AtomRSS results with file URLs to client

Space-Time Query Service

(4) Access Online Data

Granule Information

Return search results

Catalog Search

Dataset Directory

Granule Inventory

Data Repository

Data Files (online)
FROST

- Federated - data providers and third parties provide search services
- Recursive - the two-step search
- Open Search - the standard
- Toolset - drop-in server
  + database
  + engine for small data providers and P.I.s
History

- July 1994: Version 0 goes operational
- Oct 2008: "Whatever happened to federated search?"
- Jan 2009: FROST concept proposed at ESIP
- July 2009: FROST demoed at ESIP
- July 2009: Federated Search cluster started in ESIP
- Sept 2009: Federated Search abstract submitted for AGU
What Now?

- Prototype servers in progress
  - GHRC
  - NSIDC
  - ECHO
  - SciFlo nodes
    - AQUA ECHO Client
  - MODIS Web Services
  - ACCESS-NEWS
  - Mirador (GES DISC)
  - GCMD (dataset-level)
- Resolve ambiguities in Atom responses
- Prototype clients
  - XSLT: need volunteer(s) to make this more robust
  - Mirador
  - Talkoot?
  - Others?
ESIP Conventions for Atom Response

- Spatial Info: georss
  - [http://www.opensearch.org/Specifications/OpenSearch/Extensions/Geo/1.0/Draft_1](http://www.opensearch.org/Specifications/OpenSearch/Extensions/Geo/1.0/Draft_1)

- Time Info

- Data hyperlinks

- Documentation hyperlinks

- Service hyperlinks

- Connecting dataset results to granule query

- Dataset-level hyperlinks vs. granule-level hyperlinks
Representing Time Information

• Time in the Query
  ○ Following Draft extension for Time: [http://www.opensearch.org/Specifications/OpenSearch/Extensions/Time/1.0/Draft_1](http://www.opensearch.org/Specifications/OpenSearch/Extensions/Time/1.0/Draft_1)
  ○ [http://example.com/?q={searchTerms}&pw={startPage?}&dtstart={time:start}&dtend={time:end}&format=rss](http://example.com/?q={searchTerms}&pw={startPage?}&dtstart={time:start}&dtend={time:end}&format=rss)

• Time in the Response (Not covered by the extension)
  ○ Namespace: xmlns:time="[http://a9.com/-/opensearch/extensions/time/1.0/](http://a9.com/-/opensearch/extensions/time/1.0/)"
  ○ include elements within each item as:
    ▪ <time:start>YYYY-MM-DDTHH:SS:MMZ</time:start>
    ▪ <time:stop>YYYY-MM-DDTHH:SS:MMZ</time:stop>
Representing Data Hyperlinks

- HTML in `<atom:content>`?
  - but not very parseable or consistent

- XHTML in `<atom:content>`?
  - more parseable, *but* requires RDFa parsing code
    - hence steep adoption curve

- XHTML in `<atom:content>` with `@type` or `@title`?
  - e.g., `<link rel="enclosure" title="Browse" type="image/jpeg"...`
  - type has limited expressibility; title should be used for other more readable purposes

- `<atom:link>` with non-standard "rel" values?
  - e.g., `<link rel="http://www.esip.org/fedsearch/browse"...`

- `<atom:link>` elements with additional contents, like machine tags, e.g.:
  - `<link rel="enclosure" title="Browse" href="http://disc.gsfc.nasa.gov/daac-bin/airs/displayPreviewImage.py?filename=AIRS.2006.06.10.204.hdf"length="10000" type="image/jpeg">esip:LinkType=Browse</link>`
  - But very uncommon; behaviour is undefined in Atom processors

Agreement to use `<atom:link>` with non-standard "rel" values. Namespace and ontology for those rel values needs to be decided.
Representing Document hyperlinks

• Similar issues to Data hyperlinks

• Insert in granule-level results or just dataset-level results?

• Splitting vs. lumping of different kinds of documents
  ○ Type as just "datasetDocument" or...
  ○ User's guides
  ○ Dataset disclaimers
  ○ Dataset home page
  ○ Dataset news feed
  ○ Dataset OpenSearch Description Document

• Split out "client-actionable" document types?
  ○ Dataset disclaimer
  ○ Dataset OpenSearch Description Document

Suggestion that handling of documents not be included in framework. Maybe it should be a separate query? Counter-proposal to defer until needed by use case, and keep as simple as possible.

Consensus seemed to favor dataset-level, if and when we address documents.
Representing Service Hyperlinks

- Same typing issue as data and documents
- Potential service hyperlinks
  - OPeNDAP
  - OGC
  - Web Services
- Tie to scast conventions?
Connecting Dataset-Level Results to Granule-Level Query

- FROST concept
  - Dataset results includes links to Open Search Description Documents for granule query
    - Need to be tagged as such for machine recognition
    - OSDD template has dataset identifier (whatever it is) "hard-coded"
      - e.g., http://mirador.gsfc.nasa.gov/cgi-bin/mirador/granlist.pl?
        dataSet=AIRIBRAD.005&amp;page=1&amp;maxgranules={count}
        &amp;pointLocation={geo:box}&amp;startTime={time:start}
        &amp;endTime={time:end}&amp;format=atom
  - Alternatives using {searchTerms} placeholder?
What's Next

• Testing out the conventions with robust, operational, public clients
  ◦ A robust, common reference client would be helpful...
  ◦ Eventually: Convention / standard validator

• Link to services and servicecasting (scasting)

• Semantic tagging