ESIP LAB PROV CHALLENGE

Tom Narock and Doug Fils

Project Scope

- This project looked at provenance challenges from both the user and the creators perspective
- First, we created a prototype system for capturing data life cycle provenance as it happens.
- Second, we created a generic visualization service that can graphically display complete provenance traces irrespective of dataset.
- The combination of provenance capture and visualization services demonstrates a complete end-to-end system over the entire spectrum of provenance
- ***The goal of this project is not to create a production ready end-to-end system.***
- ***Rather, by working with a prototype implementation of existing standards, we aim to identify impedance points.***
- ***An impedance point is any point along the data lifecycle in which provenance generation/capture is ambiguous due to lack of standards and/or loosely defined specifications.***

PROV-AQ

- The W3C Provenance Access and Query (PROV-AQ) Working Group is developing specifications to define how to locate, retrieve, and query provenance records.

- The component that we are interested in here is the "ping-back" mechanism

- The “ping-back” mechanism can be used to register related provenance information that the data creator does not otherwise know about; e.g. provenance describing how it is used after it has been created.

- A “ping-back” feature extends the capabilities of PROV by addressing questions such as:

1. *What new resources are based on this resource?*
2. *What has this resource been used for?*
3. *Who has used it?*
4. *What other resources are derived from the same sources as this resource?*

Recommendation 1: PROV-AQ provides a standard for capturing downstream provenance

Data Providers Perspective

1. Advertising a PROV-AQ service
2. Considerations for generating provenance records
3. Receiving provenance records via pingback

Advertising a PROV-AQ service

- PROV-AQ is based on Linked Data and Semantic Web principles.
  - *Earth Science data systems are not there yet*
  - *We don't have URIs for all our datasets*
  - *The "average" user shouldn't be bothered to learn them anyway*

Recommendation 2: Adopt an advertising convention based on recognizable identifiers such as DOIs

In our summary document we have details for using

1. Dataset landing pages to advertise PROV-AQ services
2. JSON-LD to include PROV-AQ service details in a machine understandable format
3. Using REST with a DOI to send downstream provenance info back to a data provider

Considerations for Generating Provenance Records

- Provenance can exist in many encodings. The W3C PROV Data Model (PROV-DM) provides the generic structure of provenance and offers multiple encodings for this structure such as XML, XML-RDF, and Turtle.

Recommendation 3: We recommend the Earth science community adhere to semantic documents even if underlying systems are not fully Linked Data compliant. We also recommend data providers supporting PROV-AQ adhere to:

1. Providing a PROV-O description of how each of their datasets were generated.
2. Using rdfs:label throughout their PROV documents. The W3C Provenance Data Model (PROV-DM) stipulates that each provenance item have a label. PROV-O suggests that rdfs:label be used to implement such a label; however, the use of rdfs:label is not required. **We suggest that it should be required.**
3. Every provenance document returned from a PROV-AQ system should utilize the notion of prov:Bundle. In PROV-O terminology, a Bundle is a collection of related provenance statements, the grouping of which may have provenance itself. **PROV-AQ systems should return all provenance as a prov:Bundle providing return date, time, and associated service information as provenance of the bundle.**

Considerations for Generating Provenance Records

- We are aware of at least four “dialects” of provenance within the Earth sciences: PROV-M, PROV-ONE, OGC, and PROV-ES.

1. PROV-ES is a generic extension of PROV-O with Earth science specific Entities and Actions, such as Dataset, Instrument, and ProcessStep.
2. The OGC model (more formally defined in the Closa et al. paper) adds geospatial concepts such as Feature, Point, and Polygon.
3. PROV-ONE, originating from the DataONE community, is primarily aimed at scientific workflows and focuses on how outputs of one activity become the inputs of another activity.
4. PROV-M is focused on system reporting and specializes terms such as Document and Report.

Recommendation 4.

- All provenance documents within the Earth science ping-back environment should adhere to one of these dialects. In other words,
  - *All attempts should be made to conform to existing dialects of PROV-O. Any new implementations should first be vetted via open forum - potentially via ESIP.*
  - *Don’t just use PROV-O base concepts of Entity and Activity. Rather, data providers should provide additional subclasses to indicate which dialect they are using*

For example, use

```
:dataset
  a prov:Entity, eos:product;
  rdfs:label "Some dataset from USGS"^^xsd:string;
```

Instead of

```
:dataset
  a prov:Entity;
  rdfs:label "Some dataset from USGS"^^xsd:string;
```

Where eos:product refers to the Product concept in PROV-ES, which is a subclass of prov:Entity

We believe the W3C Shapes Constraint Language (SHACL) can be used to detect which dialect is being used (and validate its conformity), although we have not tested this yet.

Receiving Provenance via Ping-Back

- The PROV-AQ spec stipulates that provenance should be sent (ping-backed) as URIs.
- In other words, one does not send the entire provenance record. Instead, one simply sends a URI, which at a future point can be dereferenced to retrieve the full provenance description.
- Submitting only a URI requires that the full provenance document be stored online indefinitely awaiting deference.
  - *Infeasible for individual researchers who create derivative datasets,*

Recommendation 5.

The Earth science community commits to building and maintaining a provenance hosting service. This service would serve as an intermediary between data users and providers. A user would generate provenance on how a dataset was used. This provenance would then be uploaded to the provenance hosting service, which would return a dereferencable URI to that provenance. The user would then send the received URI as a pingback to the PROV-AQ service.

- PROV-AQ stipulates that a service is not required to do anything with the URIs it receives. Further, the spec does not indicate in which form a PROV-AQ system should return provenance.

Recommendation 6.

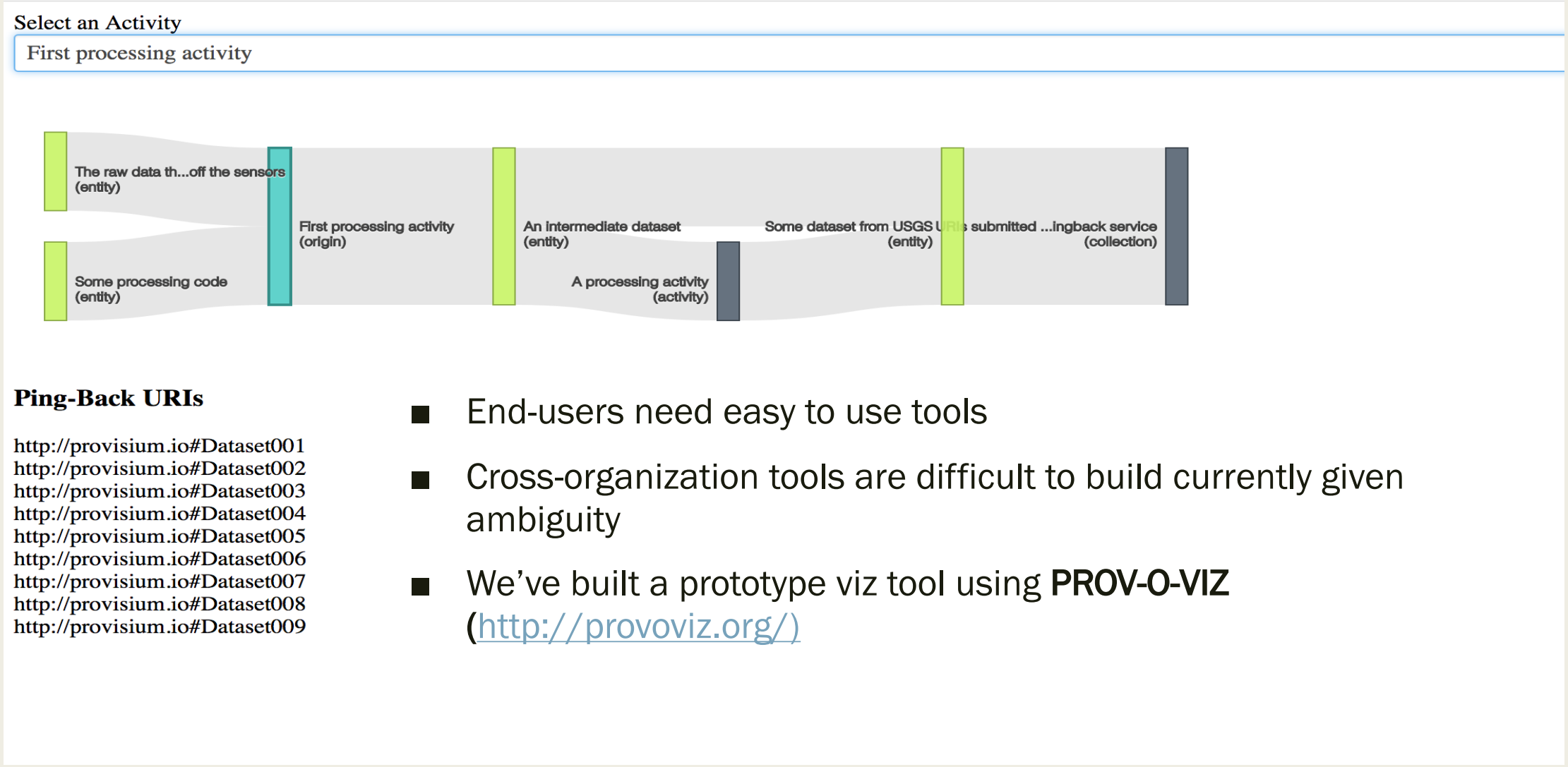
We suggest a standardization of this for the Earth sciences. When queried for a dataset a PROV-AQ service should

- **Return the provenance of how the dataset was created along with each URI it received via pingback. The URIs should be enclosed within a prov:Collection. (examples in the final report)**

User’s Perspective

1. Easy to use tools for exploring provenance
2. Incentivizing provenance generation and scope

Easy to use tools



- End-users need easy to use tools
- Cross-organization tools are difficult to build currently given ambiguity
- We've built a prototype viz tool using **PROV-O-VIZ** (http://provoviz.org/)

Incentivizing provenance and scope

- It's obvious why data providers would adopt PROV-AQ
- It's not clear what would incentivize people to send ping-backs
- *In addition, what does it mean to be a "new resource based on this resource"?*
  - *Posters?, data analysis?, etc.?*

- Maybe a good place to start is limiting ping-backs to

1. Notifying a data provider regarding the creation of a new derivative dataset that is publicly available to the community
2. Notifying a data provider regarding errors and usage issues in a dataset

Final Thoughts

- We needed to create additional PROV subclasses for our work
- We'd like to integrate these into PROV-ES
- We'd recommend that ESIP take over the development of PROV-ES. We believe PROV-ES should reside within ESIP via the ontology portal and have active community development much like SWEET.

- GitHub repo: https://github.com/esipfed/provisium
- Full report, visualization and prov-aq code, example provenance