Data Provenance Service Prototype for Collaborative Data Infrastructures

Vasily Bunakov
Science and Technology Facilities Council
Harwell Campus, United Kingdom
e-mail: vasily.bunakov@stfc.ac.uk

Javier Quinteros
GFZ German Research Centre for Geoscience
Potsdam, Germany
e-mail: javier@gfz-potsdam.de

Linda Reijnhoudt
Data Archiving and Networked Services
The Hague, Netherlands
e-mail: linda.reijnhoudt@dans.knaw.nl

***Abstract*—We report on the ongoing work of augmenting the services of EUDAT Collaborative Data Infrastructure with data provenance components. These will support the progression of existing software platforms for research data management to mature solutions for accountable data curation to improve reproducibility of results and authenticity of data. The approach and technology considered may be of interest to other collaborative data infrastructures.**

Keywords – data infrastructure; data curation; provenance.

I. INTRODUCTION

EUDAT Collaborative Data Infrastructure (CDI) [1] is a European e-infrastructure that has emerged as a result of two consecutive European projects bearing the same name [2]. An e-infrastructure is comprised of a few software platforms [3] that support different cases of data management for research: individual data publishing, data sharing, large-scale (institutional) data management, data staging for computation, data annotation, and building a public service (catalogue) for research data discoverability.

The collaborative nature of the EUDAT CDI implies multiple instances of the same service being run by different organizations, and data movements across multiple human, organizational and software agents. This diversity and complexity raise questions about data origin, data traceability, and accountability for all actions performed over data through its entire lifecycle. The lifecycle spans from ingestion through movements and transformations to the eventual distribution and consumption. This group of questions can be referred to by an umbrella term *data provenance* which is an aspect of wider considerations for the selection, collection, preservation, and maintenance of data that are known as *data curation,* which is a prominent topic in EUDAT [4].

The problems of data provenance become more acute in the operation of software platforms that handle big amounts of data with high level of automation for all the actions performed over data. EUDAT B2SAFE [5] – a robust, safe and highly available service for storing large-scale data in community and institutional repositories – is a perfect example where automated and scalable data provenance is in high demand.

We consider design and implementation of data provenance components for EUDAT B2SAFE that can be considered a prototype for a distributed data provenance service spanning different locations and a variety of research communities.

The rest of the paper is structured as follows. In Section II, we describe the first implementation that was made for GEOFON Data Centre in Potsdam, Germany, that runs an instance of EUDAT B2SAFE. We outline the use case, explain design and implementation of data provenance components, and indicate future developments. In Section III, we describe Provgen software component for generation of provenance records. In Section IV, we discuss coupling of Provgen with other software components and possible routes for publishing provenance records. We conclude our work in Section V.

II. GEOFON USE CASE

GEOFON [6] is a seismological data infrastructure that consists of a number of data centres across the globe with GFZ, the German Research Centre for Geosciences, being one of the prominent data nodes. GEOFON is the earthquake information provider worldwide; it is also one of the largest nodes of the European Integrated Data Archive (EIDA) for seismological data under the ORFEUS umbrella [7].

For partner networks, GEOFON acts as a data centre that saves a replica of the original data and as a data distribution centre at the same time. Most of the data is Open Access, but there is a small amount of data under an embargo, usually for a limited period of 3 to 4 years.

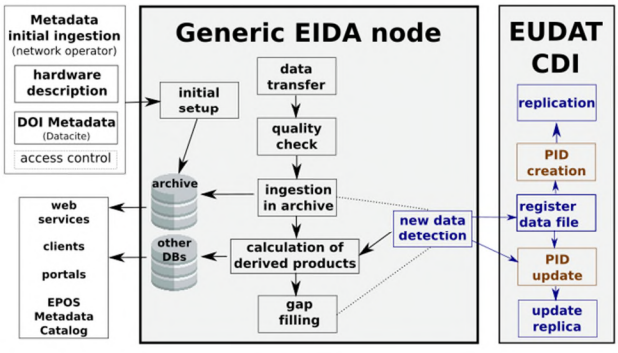

Figure 1. Data workflow in GEOFON.

 ISBN: 978-1-61208-631-6

Data is supplied in GEOFON along with its metadata, and data quality checks are performed upon ingestion. Also, data coming from different sources is checked for possible overlaps. The data workflow in GEOFON is illustrated in Figure 1.

Many services have been implemented by GEOFON itself, and the GFZ participation in EUDAT projects allowed to integrate two EUDAT services: B2SAFE and B2HANDLE. B2SAFE is used for data management at scale, and B2HANDLE allows minting and managing persistent identifiers (PIDs) for data [8]. Each PID is stored along with a set of key-value pairs called "PID record". This allows tracking down replicas of the file in different data centres and offers a reliable identification of not only a data asset itself but of all derived information, such as calculated checksums for data integrity checks. So, having clear provenance information for PID records, which can be considered valuable data themselves, is important and this is why it has been the focus of an initial testing for our implementation.

III. PROVGEN: A COMPONENT FOR PROVENANCE DATA GENERATION

One of the clearer requirements for a data provenance service is to have a high configurability to adjust to different user needs. Another requirement was that it should be decoupled from other EUDAT services, and should not have many software dependencies.

To fulfil these requirements, we designed a templating system where templates can be loaded by the operator of the system. Templates are in the Resource Description Framework (RDF) Notation3 format and each template is the result of the design of a certain Provenance record type that depends on a particular workflow.

```
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix datacite: <http://purl.org/spar/datacite/> .
@prefix provgen: <http://provgen.eudat.eu/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

provgen:EUDATCreatePID_at_{EUDAT_LITERAL:timestamp}
    a prov:Activity;
    prov:endedAtTime
"{EUDAT_LITERAL:timestamp}"^^xsd:dateTime ;
    prov:wasAssociatedWith provgen:EUDATCreatePID;
    prov:generated provgen:{EUDAT_ESCAPE:PID}; .

provgen:EUDATCreatePID
    a prov:Agent;
    a prov:Type prov:SoftwareAgent;
    prov:atLocation <https://github.com/EUDAT-B2SAFE>; .

provgen:{EUDAT_LITERAL:node}:{EUDAT_ESCAPE:irods_path}
    a prov:Entity;
    rdfs:label "{EUDAT_LITERAL:irods_path}";
    prov:atLocation "{EUDAT_LITERAL:node}";
    datacite:hasIdentifier provgen:{EUDAT_ESCAPE:PID};
    prov:atLocation "{EUDAT_LITERAL:irods_path}"; .

provgen:{EUDAT_ESCAPE:PID}
    a prov:Entity;
    dct:identifier "{EUDAT_LITERAL:PID}";
    datacite:usesIdentifierScheme
          <http://purl.org/spar/datacite/handle>; .
```

Figure 2. Provenance template *createPID*

As an example, the template of a record for the creation of a PID for a data file is expressed as the RDF snippet in Figure 2. It uses the PROV ontology [9] to express relations between Agents and Entities, through Activities. The EUDATCreatePID functionality (the Agent) generated a PID (the Entity) during the Activity. Other ontologies were used to record more details, such as the actual identifier generated and the identifier scheme used. When the calling service creates a PID, it calls the Provgen with the variables required for the *createPID* template.

Figure 3 shows the completed provenance document, in which the placeholders in the template have been replaced with the actual. To facilitate the correct markup of the resulting completed document, we defined at least two different ways to do the replacement of the variables: literal or escaped.

```
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix datacite: <http://purl.org/spar/datacite/> .
@prefix provgen: <http://provgen.eudat.eu/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

provgen:EUDATCreatePID_at_2017-12-02T17:29:23
    a prov:Activity;
    prov:endedAtTime "2017-12-02T17:29:23"^^xsd:dateTime;
    prov:wasAssociatedWith provgen:EUDATCreatePID;
    prov:generated provgen:PREFIX1\/PID1; .

provgen:EUDATCreatePID
    a prov:Agent;
    a prov:Type prov:SoftwareAgent;
    prov:atLocation <https://github.com/EUDAT-B2SAFE>; .

provgen:server1:\/path\/filename
    a prov:Entity;
    rdfs:label "/path/filename";
    prov:atLocation "server1";
    datacite:hasIdentifier provgen:PREFIX1\/PID1;
    prov:atLocation "/path/filename";

provgen:PREFIX1\/PID1
    a prov:Entity;
    dct:identifier "PREFIX1/PID1";
    datacite:usesIdentifierScheme
          <http://purl.org/spar/datacite/handle>; .
```

Figure 3. Completed provenance document for the creation of a PID.

It can be seen in Figure 3 that elements such as a file path which includes characters like "/", will be invalid if they are literally replaced where a subject or an object is expected in the triple. However, they should still appear without modifications in the case of being used as literals.

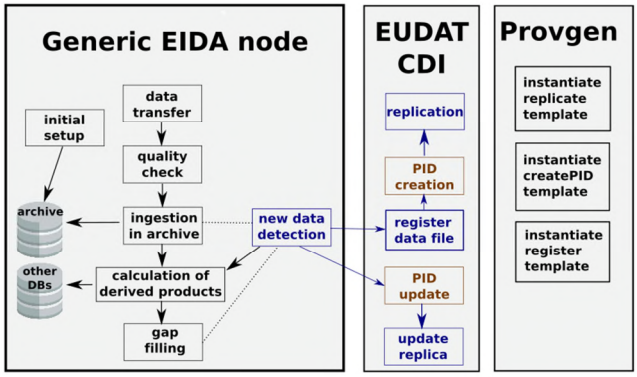

Figure 4. Calls to the Provgen service in the GEOFON work-flow

To decouple Provgen from other components of the EUDAT CDI and avoid the already mentioned dependencies, we specified an API [10] which allows the following:

- list available templates in Provgen including the specification of each template (expected variables and their description),
- instantiate a template,
- show the Provgen documentation,
- show the Provgen configuration.

Before using Provgen, a developer needs to identify points in the data workflow where provenance information should be generated.

Every call identifies the template to be used and the replacements for the placeholders in the template. If no suitable template exists, the developer needs to create a template and put it into the "templates" folder. Then, the program can make a call to the Provgen API in the respective point of a data workflow.

The template should be documented with comments on the first lines of the document, where a list of the variables to be replaced within the document is presented and explained in detail. For instance, the template in Figure 2 can be documented with the following comments to explain the parameters used:

```
# EUDAT_PARAM:timestamp - the time at which
the PID was generated
# EUDAT_PARAM:PID - the handle PID that was
generated
# EUDAT_PARAM:irods_path - the absolute path
of the file in iRODS
# EUDAT_PARAM:node - the domain name of the
server where the node resides
```

The template-based design makes Provgen quite universal: it can be used not only in the B2SAFE service where it was tested but with any other EUDAT service, or by other data infrastructures.

IV. COUPLING PROVGEN WITH OTHER COMPONENTS, AND SERVICE DESIGN CONSIDERATIONS

Provgen per se is just a flexible, configurable component for provenance records generation. It can be interacted with using its API. In the simplest and default installation, records will be stored in the free online Provenance backend storage called ProvStore [11][12]. This could cover the needs of most users, as the only technical requirement is to open a free account. The free ProvStore service includes the capability to store, share, export and visualize the documents generated by Provgen.

In the case that a considerable number of provenance records are expected (e.g. millions), we also provide the option to make the records available in files and upload them to a more powerful external backend storage (e.g. triple store). This gives also the possibility of exposing them for querying using an endpoint for RDF Query language (SPARQL).

For testing this bridge to a triple store, we used Jena TDB, a native triple store, as the backend with Fuseki server as a frontend [13]. The ingest of Provgen-generated records was straightforward, and Fuseki allowed to expose them via a SPARQL endpoint – so Jena framework allowed a persistence layer for provenance records and a decent commonly understandable API in the form of a SPARQL endpoint. As Provgen-generated records are primarily based on the popular PROV specification, it makes the service interface quite universal and self-specified, with good prospects of adoption across different data infrastructures and user communities.

We are considering further experiments on Provgen-generated records ingestion in a neo4j graph database [14] as EUDAT services, B2SAFE in particular, favour this database engine for metadata management. Provenance records then may become additional metadata in a common metadata storage, which may allow insightful inquiries into data and metadata quality and into data maintenance procedures; as an example, using provenance records in order to judge on the quality of data policy implementation. The inclusion of provenance records in a common metadata store will have its conceptual advantages, also operational advantages as one will not need additional software components (RDF triple store and a frontend for it), but can just rely on the graph database component already in use.

This approach will have its disadvantages, too, as one of the strengths of the RDF representation of provenance records is an out-of-box ability to conduct standardized machine reasoning over provenance. Provenance exposure via SPARQL endpoint will have an architectural advantage, too, as SPARQL endpoints allow building up natural service federations with simultaneous requests to multiple endpoints from the same software components and thus can support the development of a universal infrastructure-wide provenance service (which in the case of EUDAT, according to this e-infrastructure naming conventions, could be named a B2PROV service). Also, SPARQL as a query language, as well as RDF APIs for high-level programming languages are more common than specific query languages and APIs for graph databases, which is important for a favourable adoption of the data provenance service by a wider community of software developers.

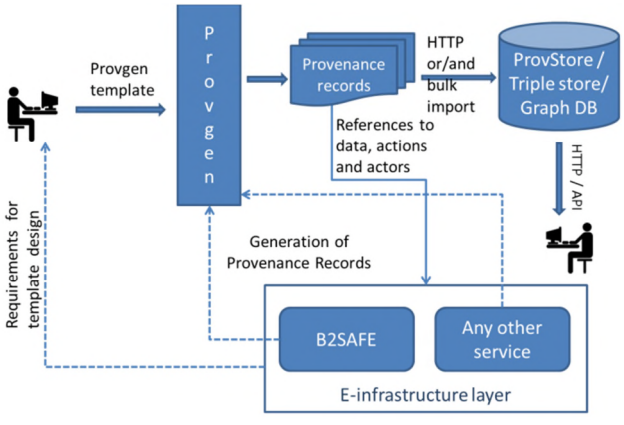

Figure 5. Provgen integration with a triple store or/and a graph database.

Which route, RDF-based or with graph databases, will prevail in EUDAT Collaborative Data Infrastructure, will depend on its operational and sustainability considerations. For other data infrastructures, especially for newly emerging ones without a burden of years-long particular implementations, we expect the RDF-based approach could be preferred for publishing provenance records and performing machine reasoning over them. Figure 3 indicates both possible routes.

V. CONCLUSION

This work reports on the progress made with design and implementation of a provenance service applied to the use case of the thematic GEOFON seismological data infrastructure [6] that interacts with the common domain-agnostic EUDAT data infrastructure [1]. The service prototype implemented [10] prepares and publishes well-structured provenance records for the GEOFON data managed by the EUDAT B2SAFE service [5]. The future work will involve further testing and promotion of the provenance component to its adoption in EUDAT B2SAFE Production. The opportunities will be explored for a wider use of the provenance component, or a generic service based on it, in other EUDAT services [3] and for other use cases beyond seismological data.

ACKNOWLEDGMENTS

This work is supported by EUDAT 2020 project that receives funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 654065. The views expressed are those of the authors and not necessarily those of the funding project or institutions.

REFERENCES

[1] EUDAT Collaborative Data Infrastructure. [Online]. Available from: https://www.eudat.eu/eudat-cdi [retrieved: March, 2018]

[2] EUDAT project. [Online]. Available from: https://www.eudat.eu/ [retrieved: March, 2018]

[3] EUDAT services. [Online]. Available from: https://www.eudat.eu/services-support [retrieved: March, 2018]

[4] V. Bunakov et al. "Data curation policies for EUDAT collaborative data infrastructure", Selected Papers of the XIX International Conference on Data Analytics and Management in Data Intensive Domains (DAMDID/RCDL 2017). CEUR Workshop Proceedings Vol-2022, 2017, urn:nbn:de:0074-2022-6, pp. 72-78

[5] EUDAT B2SAFE service. [Online]. Available from: https://www.eudat.eu/b2safe/ [retrieved: March, 2018]

[6] W. Hanka and R. Kind, The GEOFON Program. Annals of Geophysics v. 37, n. 5, Nov. 1994. ISSN 2037-416X. doi:10.4401/ag-4196 (1994)

[7] L. Trani et al. "The European seismological waveform framework EIDA". In *Geophysical Research Abstracts*, vol. 19, EGU2017-13770, Vienna, Austria, April 2017.

[8] J. Quinteros et al. "Moving towards persistent identification in the seismological community". In *Geophysical Research Abstracts*, vol. 18, EGU2016-15619-1, Vienna, Austria, Apr 2017.

[9] PROV ontology. [Online]. Available from: https://www.w3.org/TR/prov-o/ [retrieved: March, 2018]

[10] Provgen API specification. [Online]. Available from: https://raw.githubusercontent.com/javiquinte/provgen/master/swagger.yaml [retrieved: March, 2018]

[11] T. D. Huynh and L. Moreau (2014) ProvStore: a public provenance repository.At *5th International Provenance and Annotation Workshop (IPAW'14) , Cologne, Germany,09 - 13 Jun 2014.* 3pp

[12] ProvStore service. [Online]. Available from: https://provenance.ecs.soton.ac.uk/store/ [retrieved: March, 2018]

[13] Apache Jena framework. [Online]. Available from: https://jena.apache.org/ [retrieved: March, 2018]

[14] NEO4J graph platform. [Online]. Available from: https://neo4j.com/ [retrieved: March, 2018]

 ISBN: 978-1-61208-631-6