eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

DataCite

EOSC PIDGraph Service

euroCRIS tutorial 16 May, 2024

Mike Bennett, DataCite
Matt Buys, DataCite

Funded by
the European Union

PID Graph Background

Project FREYA

Developed by the FREYA project, funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 777523.

PID Graph – Concept

The connected searchable Graph of publications, datasets, other research outputs, people and organizations.

PID Graph – User Stories

At the beginning of the FREYA project we collected more than 40 user stories (see https://pidforum.org/c/pid-best-practices/8) that were difficult to address with currently available services.

PID Graph – Architecture

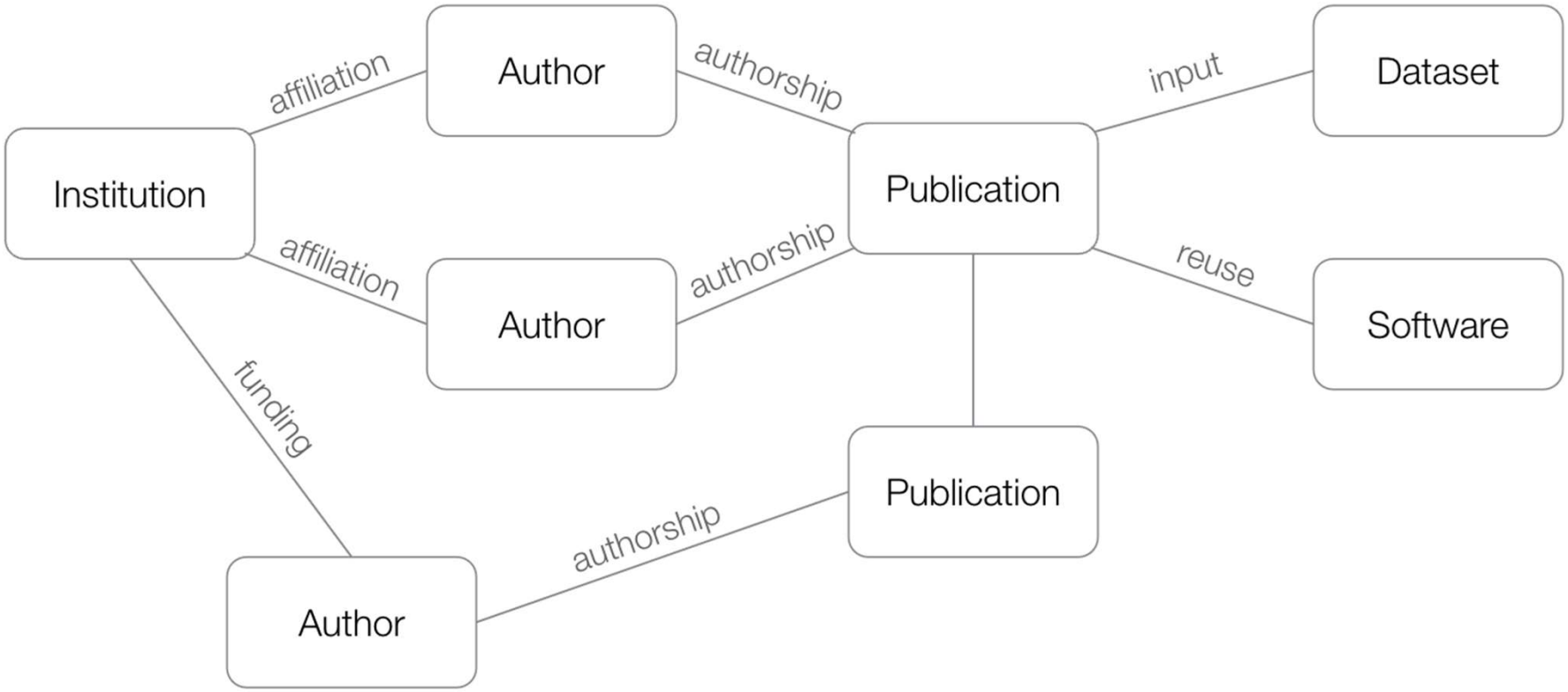
We discussed and prototyped PID Graph architecture to address user stories, taking into account existing PID infrastructure, in particular from unfunded partners, and sustainability beyond FREYA project duration.

PID Graph

In a nutshell

The graph formed by the collection of scholarly resources such as publications, datasets, people and research organizations, and their connections. The PID Graph uses persistent identifiers and GraphQL, with PIDs and metadata provided by DataCite, Crossref, ORCID, and others.

https://www.project-freya.eu/en/blogs/blogs/the-pid-graph

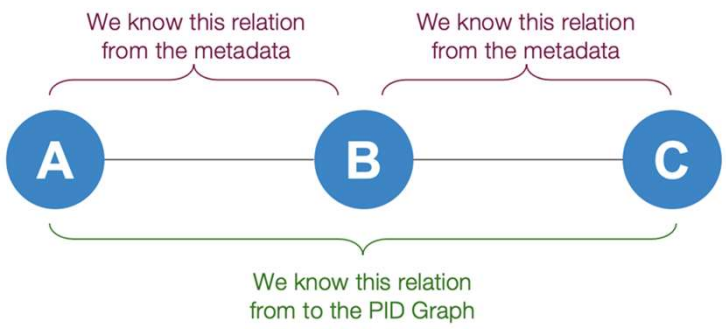

Author
Dataset
Institution
Publication
Author
Software
Publication
Author
affiliation
authorship
input
affiliation
authorship
reuse
funding
authorship

Connecting Research

Having unique persistent identifiers for researchers and their outputs is crucial to connecting pieces of the research landscape together.

PIDs already have the potential to enable the connected research graph, but we're not yet taking full advantage of their connecting powers.

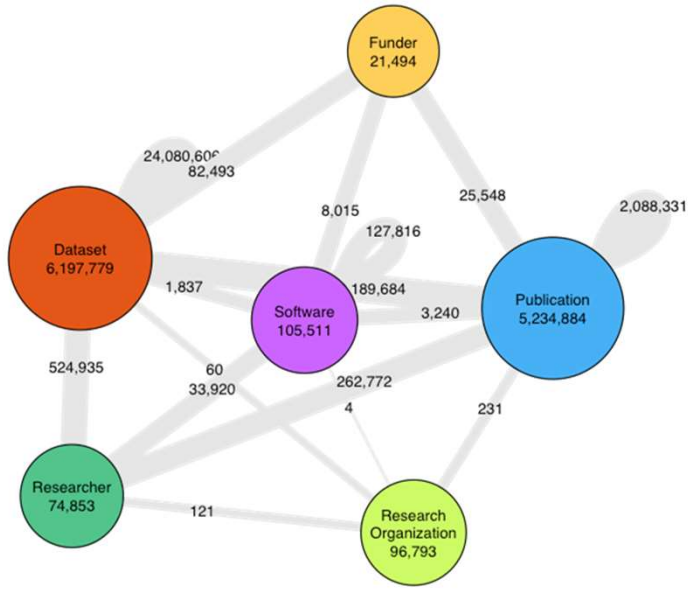
We can now clearly link PIDs together via relations in their metadata to enable the discovery of connections at least two "hops" away



PID Graph Evolution

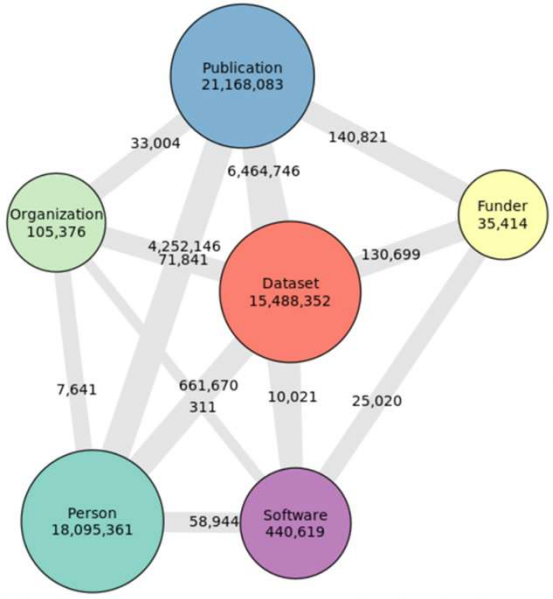
PID Graph

Nodes and connections (as of September 2019)



PID Graph

Nodes and connections (as of October 2023)



PID Graph Use Cases

In 2018 the FREYA project partners started collecting user stories that address important needs of their respective communities. In an August 2018 workshop we discussed these user stories, grouped them together, and prioritized them. The main categories were the aggregation of scholarly outputs, e.g. by research institution, funder, or researcher; the versioning and granularity of data and software, and the grouping of all research outputs and other resources (e.g. data, software, people, funding) for a given publication. All these user stories depend on a PID Graph, with typically two connections needed in the graph, e.g. “show me all citations for datasets funded by a particular grant”.

https://pidnotebooks.org/

https://pidforum.org/c/pid-best-practices/8

PID Graph Iterative Development

DataCite continue to support the sustainability and iterative development of the PID Graph.

Currently our primary focus is on extending the metadata connections and harvesting functionality (of PID relationships).

In addition, we are continuing to innovate on DataCite Commons (frontend to the PID Graph)

https://commons.datacite.org/

eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

DataCite

The PIDGraph Data

DOI metadata and vertices to other PIDs

Main PIDGraph nodes are DOIs, containing data modelled with the DataCite Metadata Schema

Additional sparse nodes for selected PIDs – ORCiD IDs for people, and ROR IDs for Research Organisations

Rich relationship data exposing the links between the involved PIDs

eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

DataCite

Methods of Access

APIs

DataCite REST API – https://api.datacite.org/dois

- Contains DOI metadata for DataCite DOIs in JSON format
- Use Cases: Ingesting individual records or small subsets (e.g from an individual repository)

DataCite Event Data API – https://api.datacite.org/events

- Contains relationships between PIDs, modelled as triples (object, subject, relationship) with additional metadata
- Use Cases: Discovering links between PIDs, tracking citations

eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

DataCite

Methods of Access

APIs

DataCite GraphQL API – https://api.datacite.org/graphql

- Graph-based view of DOI metadata, containing many node types (e.g datasets, articles)
- Use Cases: Exploring the graph of PIDs and their relationships, retrieving “two-hop” connections

DataCite OAI-PMH API – https://oai.datacite.org

- Standard protocol based harvesting of DOI metadata in Dublin Core or DataCite XML formats
- Use Cases: Ingesting metadata into systems that already support OAI-PMH

Methods of Access

Data Files

DataCite Public Data File – https://datafiles.datacite.org

- A full data dump of all DataCite DOI metadata records in JSON format
- Use Cases: Bootstrapping other systems, performing large analysis of DOI metadata

DataCite PID Links Data File

- Upcoming release containing all the vertices in the PIDGraph in JSON format
- Use Cases: Bootstrapping other systems, retrieving connections between PIDs

Methods of Access

Frontend

DataCite Commons – https://commons.datacite.org

- Interactive web-based front end for exploring the PID Graph
- Detailed pages for DOIs
- Summary pages for ORCiD IDs and ROR IDs
- Analysis of PID data, graphs and charts
- Faceted searching and filtering

eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

DataCite

Demonstrations

REST API – Single DOI retrieval, basic search

GraphQL API – More complex searching

Data File – CLI processing using JQ

Commons – Exploring the graph of a DMP