GraphQL API

Intro & motivations

The FREYA project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 777523.

DataCite GraphQL API

DataCite has developed a GraphQL API for consuming the FREYA PID Graph.

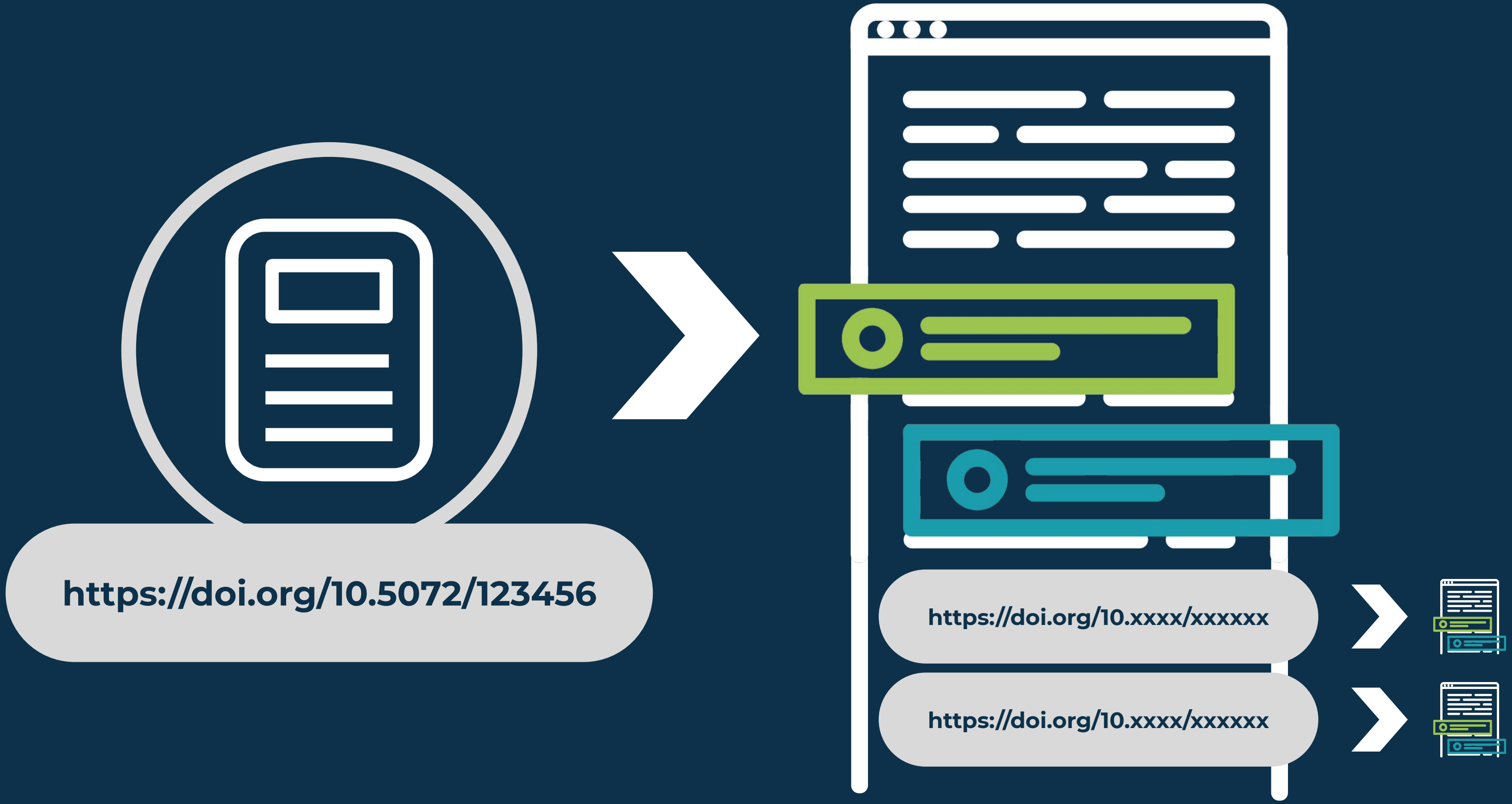
GraphQL is a convenient way to access the information in the PID Graph with DataCite metadata as a starting point.

What is the PID Graph?

The FREYA PID Graph is a conceptual model for linking together research objects using their PIDs and their metadata.

We mean “graph” in the network sense. Who knows who? Who created what? How are all those things related?

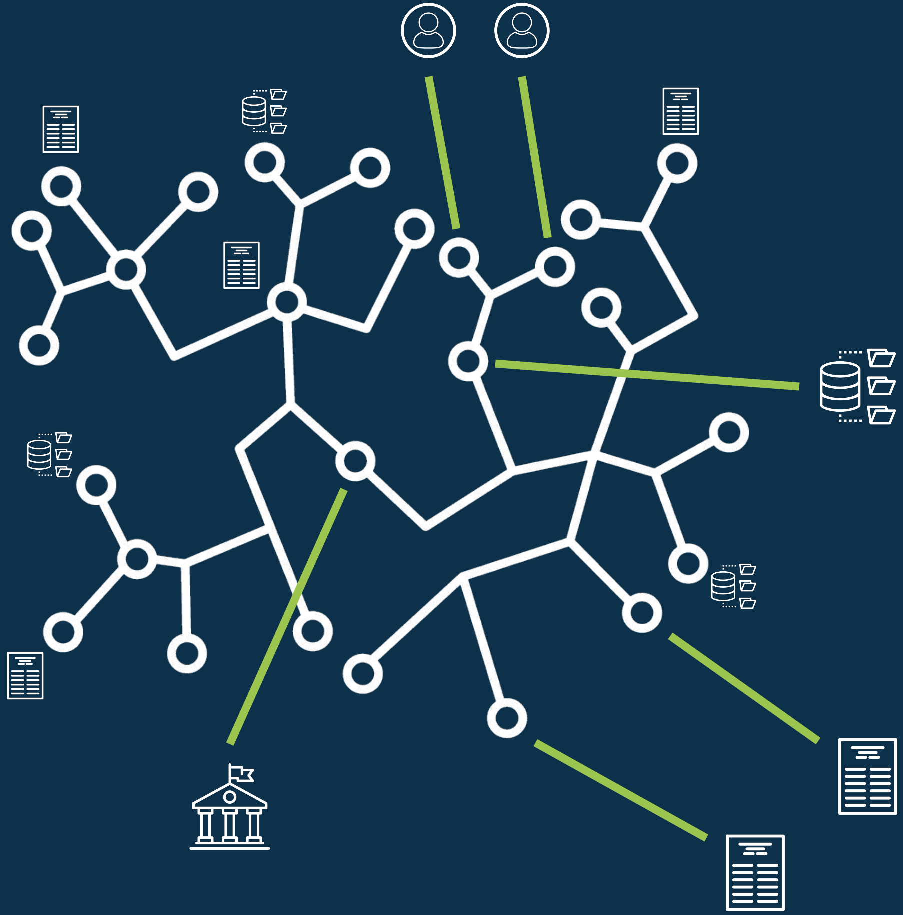
https://doi.org/10.5072/123456



https://doi.org/10.5072/123456

https://doi.org/10.xxxx/xxxxxx

https://doi.org/10.xxxx/xxxxxx

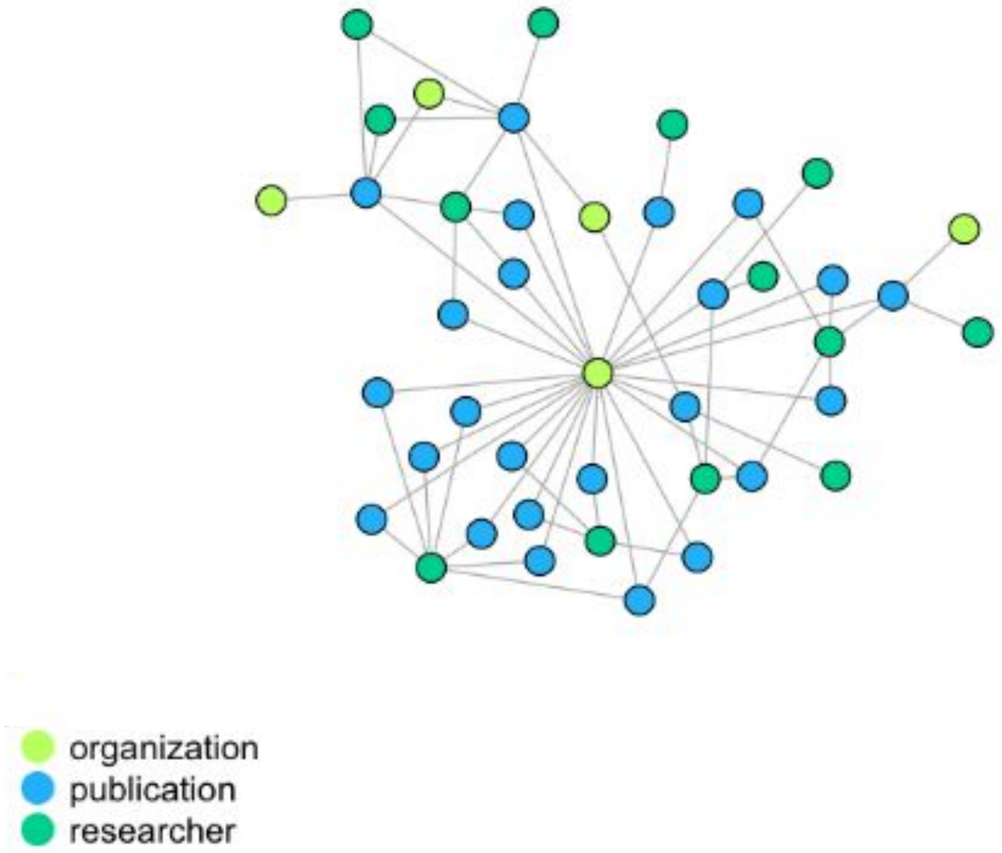


Better questions

Graph of all the publications associated with DataCite, plus all the researchers and organizations associated with those publications.

Answers the question :

With whom has DataCite collaborated on their publications?



So . . . it’s conceptual?

Mostly.

As of now, FREYA has done a lot of work to flesh out PIDs and metadata so that there’s anything to search for and link to in the first place. The infrastructure and data are there.

But it still requires a bit of developer knowledge to get this data out and use it. There’s no “magic button”.

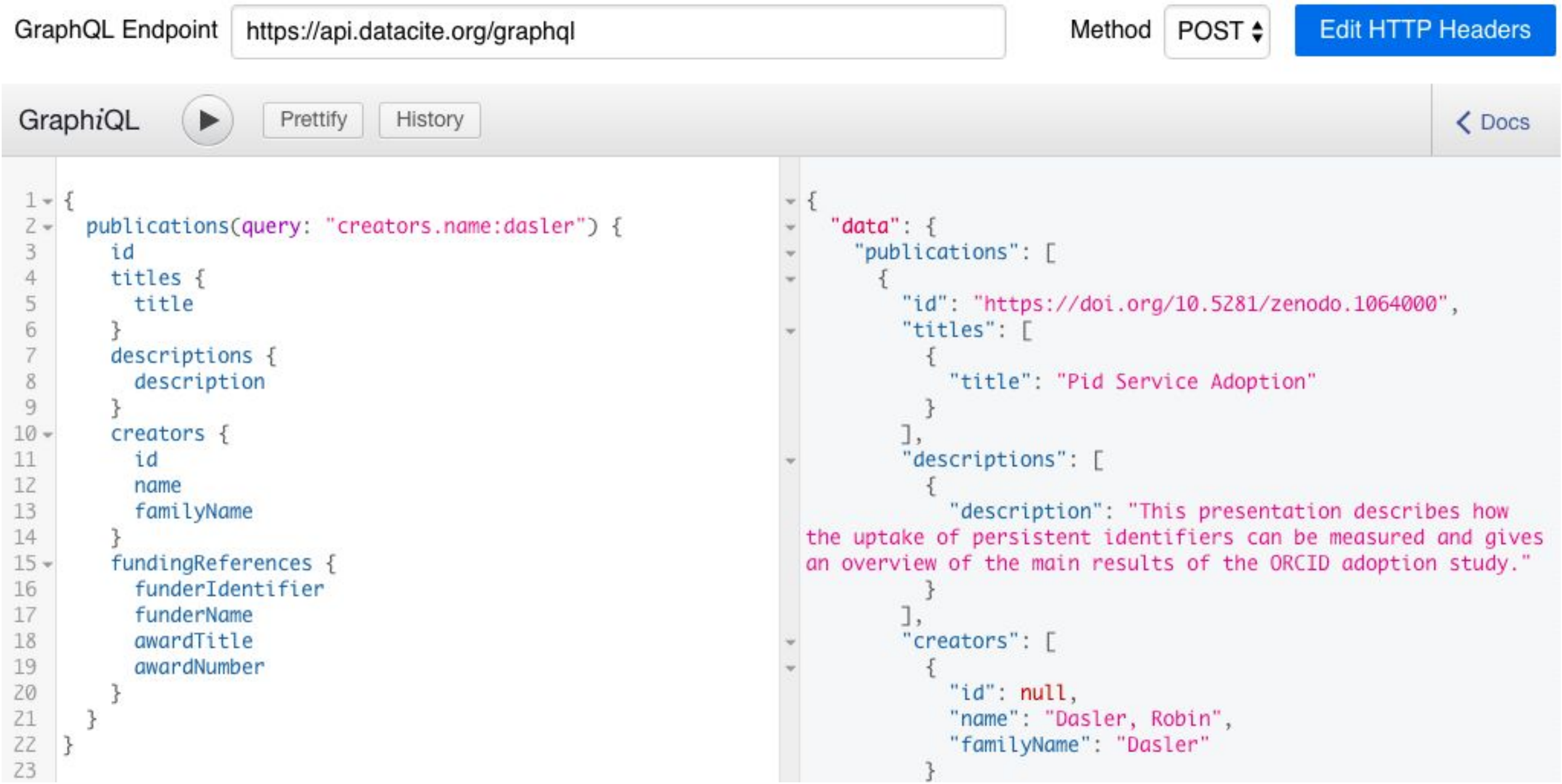
This is what the GraphQL API is for.

What is GraphQL?

GraphQL is a query language specifically for making queries across a graph. It came originally from Facebook, where they know a thing or two about graphs.

With a “normal” REST API, you have to provide the query in a specified way in order to get the results the API provider wants to give you.

With GraphQL, you can specify the information you want to receive. Want to find all the connections to a DOI, but without any authors? Ask that way, and that’s what you’ll get.



GraphQL Endpoint https://api.datacite.org/graphql

Method POST

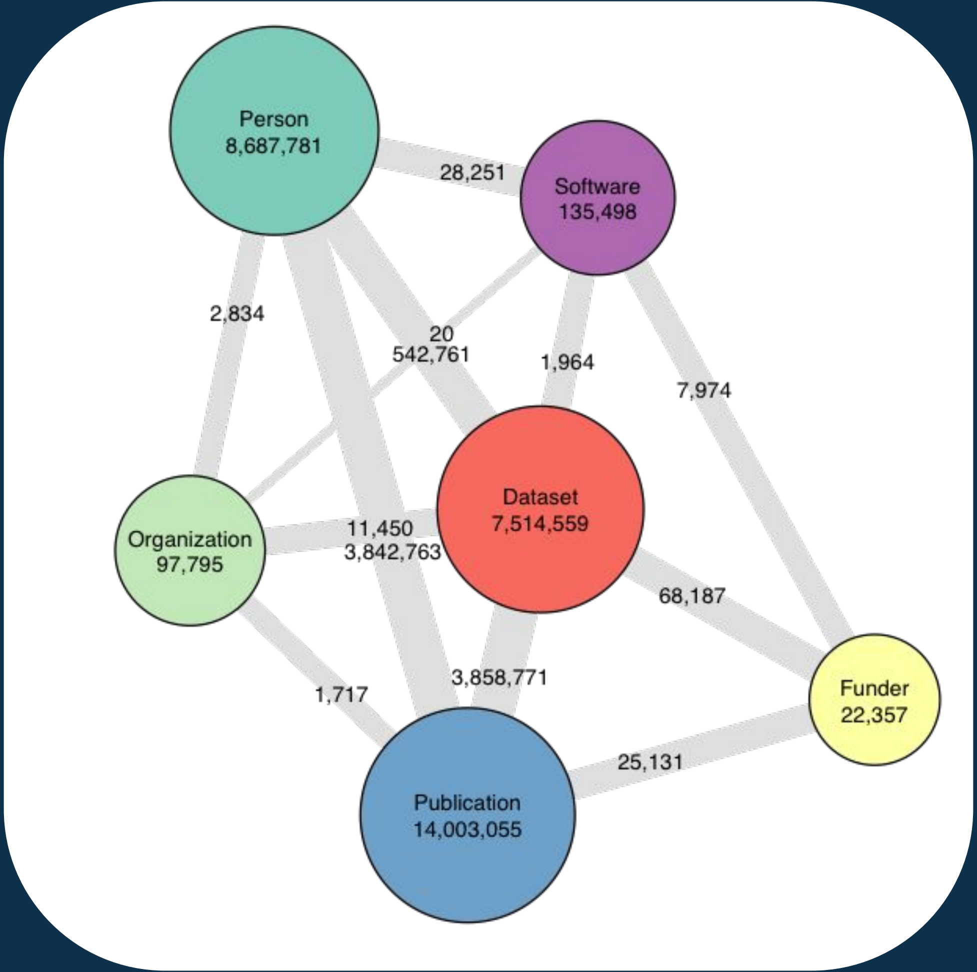
```
{
  publications(query: "creators.name:dasler") {
    id
    titles {
      title
    }
    descriptions {
      description
    }
    creators {
      id
      name
      familyName
    }
    fundingReferences {
      funderIdentifier
      funderName
      awardTitle
      awardNumber
    }
  }
}
```

```
{
  "data": {
    "publications": [
      {
        "id": "https://doi.org/10.5281/zenodo.1064000",
        "titles": [
          {
            "title": "Pid Service Adoption"
          }
        ],
        "descriptions": [
          {
            "description": "This presentation describes how
the uptake of persistent identifiers can be measured and gives
an overview of the main results of the ORCID adoption study."
          }
        ],
        "creators": [
          {
            "id": null,
            "name": "Dasler, Robin",
            "familyName": "Dasler"
          }
```

What can I ask for?

As of the latest release (May 6), the GraphQL API contains:

- All Datacite DOIs
- ~9 million Crossref DOIs
- All ORCID iDs
- All ROR IDs
- All Crossref Funder IDs
- All re3data records



Resources and links available via the GraphQL API as of May 4, 2020.

For more info

Support documentation (from DataCite):
https://support.datacite.org/docs/datacite-graphql-api-guide

Video tutorial (from DataCite):
https://www.youtube.com/watch?v=efvxGfU_oVM

About GraphQL generally: https://graphql.org

Jupyter notebooks demonstrating FREYA use cases:
https://github.com/datacite/notebooks

PID Forum for all PID Graph related questions:
https://www.pidforum.org/c/pid-graph