Metadata Catalogue Release

Document Control Information

Settings	Value
Document Identifier:	MS12
Project Title:	ExPaNDS
Work Package:	WP3
Document Author(s):	Carlo Minotti (PSI), S. da Graca Ramos (Diamond), Alun Ashton (PSI), Stephan Egli (PSI), Fredrik Bolmsten (ESS), Henrik Johansson (ESS), Massimiliano Novelli (ESS), Alejandra Gonzalez-Beltran (STFC), Stuart Pullinger (STFC)
Document Reviewer(s):	N/A
Doc. Issue:	1.0
Dissemination level:	Public
Date:	24/08/2021

Abstract

We present the milestone achieved for a *metadata catalogue release* in the domain of photon and neutron (PaN) science. With the primary goal of supporting PaN FAIR data catalogue services, we have developed a self-contained, stand-alone metadata catalogue release that facilities can download to test/try and play with.

The work represents the achievement of milestone MS12 of the Horizon 2020 ExPaNDS project.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

Executive Summary

We report on the design and development of a self-contained, stand-alone data catalogue (later addressed also as: *reference implementation* or *metadata catalogue release*) to support Photon and Neutron (PaN) facilities to get started with the adoption of a data catalogue. The exploitation of a data catalogue can be hampered by the complex initial configuration and interaction between components. The code development should also partially cover the documentation aspect, as it provides an example of a reference implementation of a data catalogue. This paper describes the architecture and development of a *reference implementation* of a data catalogue consisting of several components, in particular providing a database where to store metadata, some predefined reference metadata, a rest API access layer to the database, a GUI for this access layer and the *panosc-search-API* [1], all at their latest version. Crucially, each component is supported by a community maintenance process to allow a managed and agreed approach to modification and extensions in future development. We present Version 1.0 of the *metadata catalogue release* as a viable starting point for long-term use. The initial version is very simple and designed with the data catalogue use-case firmly in mind. Future versions are likely to include more components, for example, an authentication layer. It is anticipated that future developments will allow increasing integration with the wider semantic web.

The implementation described in Section 3 is specific to SciCat, which is one of the two major data catalogues in the PaN community: ICAT (https://icatproject.org/) and SciCat (https://scicatproject.github.io/). The ICAT metadata catalogue has been used in production systems in multiple facilities for over 10 years, and the ICAT collaboration is continuously working to evolve the different components to address FAIR data, including a solution for PaN ontologies [2]. This report includes a description of the latest improvements related to continuous deployment and support for FAIR data in ICAT (Section 4.2).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

Table of Contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

1. Background and Purpose of the metadata catalogue release

1.1 Background

The work outlined in this document is part of the ExPaNDS project, carried out in close communication with the PaNOSC project (https://www.panosc.eu/), thus representing the majority of European Photon and Neutron sources in coordinated activity to drive forward Findable, Accessible, Interoperable and Reusable (FAIR) facility data and EOSC services.

The specific task for this deliverable is as follows:

> *Task 3.3: Implement ontologies in metadata catalogues*
>
> *The defined ontologies will be implemented in different data catalogues (e.g. ICAT at UKRI and SciCat at PSI). To foster the federation of local services a reference implementation will be provided on the basis of the NeXus format. These will result in a European standard with international impact, and as such provides the basis for APIs and interoperability. The latter activities will be aligned with the PaNOSC initiative.*
>
> *Task 3.4: Coordinate the integration of metadata catalogues to manage the data lifecycle at EU national Photon and Neutron RIs*
>
> *The defined ontologies will be implemented in different data catalogues (e.g. ICAT at UKRI and SciCat at PSI). To foster the federation of local services a reference implementation will be provided on the basis of the NeXus format. These will result in a European standard with international impact, and as such provides the basis for APIs and interoperability. The latter activities will be aligned with the PaNOSC initiative.*

Milestone related to this task:

> M3.1: Metadata catalogue release

During the installation of a data catalogue in a facility, it is often required to follow steps aimed to integrate it with existing infrastructure and eventually installing more than one component covering different functionalities. Despite being fairly automated and documented, this process still holds some complexity and is prone to errors. To facilitate

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

even more the installation of a data catalogue, this milestone delivers an easy installation of a data catalogue, including the most common components. The first step in such an endeavour is therefore to gain an understanding of the immediate and future potential uses of the local deployment.

1.2 Purpose

The main purposes of the *metadata catalogue release* alluded to in the original proposal and fleshed out by consultation with PaN community representatives, include:

- To provide an easy solution to install a local instance of a data catalogue.
- To enable the end-user to interact with some reference metadata through the interfaces enabled by the *metadata catalogue release*.
- To provide a sample set of metadata, showing the capabilities of the data catalogue and providing an example of a possible metadata structure.
- To provide an example of how to use the *panosc-search-API* in conjunction with metadata following the terminology defined by PaNET ontology [2].

The specific deliverable of this component of the ExPaNDS project, 'ExPaNDS: metadata catalogue release', should be viewed as a 'demonstrator' which aims to: provide an interactive example to facilities; help and drive a subsequent and more specific data catalogue adoption by each facility, tailoring its installation to each existing needs and infrastructure.

2. Stakeholder Engagement

Active engagement with the European and global photon and neutron science community has played an essential role in this project. A summary of some of the main engagement activities is given below.https://github.com/icatproject/icat.server/issues

● *Engagement with ICAT and SciCat community.*

ICAT and SciCat are currently the two major metadata catalogues in use at the PaN facilities. To provide a one-click installation of a metadata catalogue (*reference implementation*), a consultation has taken place among ICAT and SciCat colleagues, driving to an agreement to deliver the first version - V1.0 - based on SciCat.

● *Engagement with ExPaNDS WP3 in particular task 3.2.*

Under the field *technique* of the metadata, this milestone includes an example of PaNET ontology [2], retrieved directly from *NCBO BioPortal* (https://bioportal.bioontology.org/). The user will be able to search for data by using the *panosc-search-API* in conjunction with PaNET terms [2].

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

● *Coordination of metadata catalogue release with PaNOSC through regular meetings.*

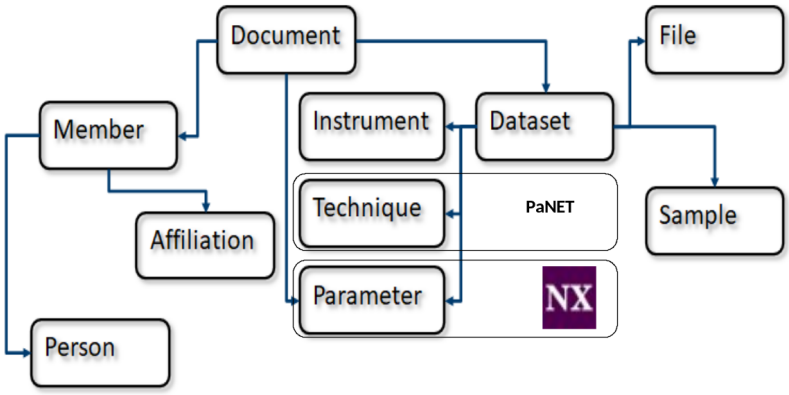

Figure 1: PaN search API Data Model

As the *reference implementation* will be implemented on one of the main data catalogues, close collaboration with the PaNOSC partners is important. Figure 1 shows the PaN search API data model (PaNOSC deliverable D3.1 [1]) and the anticipated role of the experimental techniques (PaNET) and Nexus ontologies within that API.

● *Presentation on the metadata catalogue release implementation at ExPaNDS WP3* meeting, held in June 2021.

The solution and architecture suggested (figure 2) was discussed and agreed on during the meeting.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

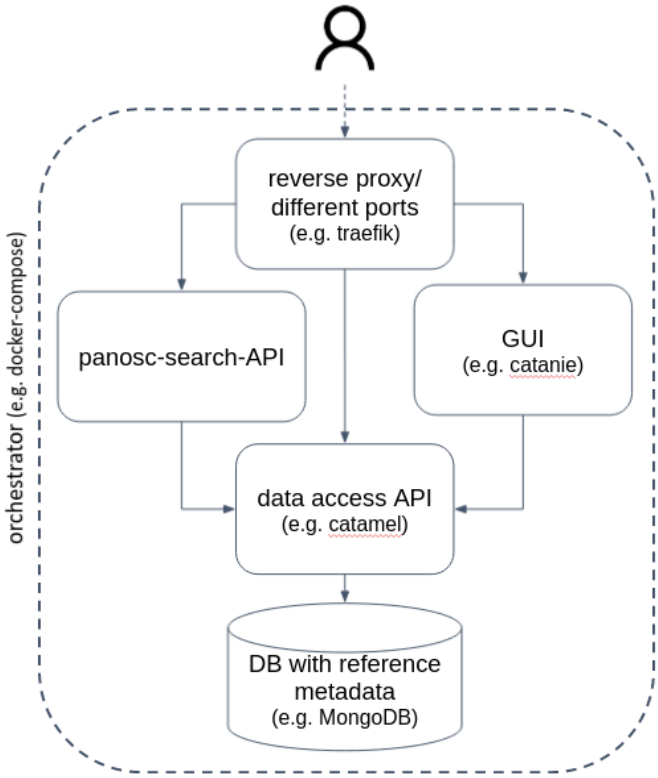

Figure 2: *reference implementation* architecture

● *Coordination of metadata catalogue release with ExPaNDS D4.2.*

The example metadata in the *metadata catalogue release* comes from ExPaNDS deliverable 4.2 “Photon and Neutron reference data sets” [3] and contains links to open access data.

3. Metadata catalogue release

3.1 Availability

The current approach to the milestone and its *reference implementation* of a metadata catalogue can be found on Github (https://github.com/) at:

- https://github.com/SciCatProject/scicatlive.

Its content has been tested by ExPaNDS and PaNOSC members.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 857641.

3.2 Purpose

The primary purpose of the *metadata catalogue release* described in this paper is to support ExPaNDS Work Package 3, in particular with the adoption of a data catalogue by the PaN facilities, along with the parallel PaNOSC work package, and to provide an interactive example of the use of PaNET [2]. Specifically, the programme of developing a common interface to facility data catalogues requires facilities to have in the first place a data catalogue, and the *reference implementation* provides an easy solution to experiment with its functionalities, possibly driving its subsequent adoption.

The *reference implementation* also includes a reference to the PaNET ontology [2], the main purpose of which is to provide a common set of standard technique names with global persistent identifiers (along with common alternate names (labels), human-readable annotations and a rudimentary formal semantic description). All this information is stored in the reference metadata of the data catalogue and can later be queried through the *panosc-search-API*. In V1.0, we exploit the connection to PaNET [2], by including in the reference metadata the name of the technique and its PID from *NCBO Bioportal*.

3.3 Design Principles

The approach to the *metadata catalogue release* was mainly determined by the need of the facilities to install a data catalogue with the fewest manual steps possible. This solution is scalable as it does not depend on the number of facilities using it and will assist in onboarding new facilities adopting a data catalogue.
Conceptually, the solution consists of a set of access layers (applications) to the data catalogue that is installed all together via a single command. From the many potential solutions, we have adopted *docker-compose* (https://docs.docker.com/compose/) as a solution for the orchestration, because of its minimum requirements and ease of use.

Here, we outline the design principles of the V1.0 *metadata catalogue release*. Possibilities for future development are addressed at the end of this section. The key design decisions are as follows:

- Each service accessing data is available from a URL. There are three such services, and traffic is adequately directed to each using a reverse proxy.
- Each of the eleven reference publications from ExPaNDS deliverable 4.2 [3] have a corresponding entry in the publication section of the data catalogue.
- Each publication is linked to one or more datasets. From the data catalogue, the end-user will be able to see related metadata and be redirected to the landing pages of these publications.
- Almost every dataset contains a non-empty field technique. The content of this field comes from ExPaNDS deliverable 3.2 [2]. Relation to sample and instrument is also exploited.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

- Each technique is represented as a string where the full hierarchy is made explicit and nodes are separated by a dot, e.g. for *technique1: grand_parent.parent.technique1*. In the case of multi-inheritance, different streams are stored and separated by a semicolon: *grand_parent.parent1.technique1;grand_parent.parent2.technique1*.
  E.g. *optical spectroscopy* (see figure 3) would be represented as: *photon_and_neutron_technique.defined_by_experimental_probe.photon_probe.UV_visible_photon_probe.visible_photon_probe.optical_spectroscopy;photon_and_neutron_technique.defined_by_functional_dependence.spectroscopy.optical_spectroscopy;photon_and_neutron_technique.defined_by_functional_dependence.spectroscopy.versus_energy.optical_spectroscopy*.
- Almost every dataset has a link to *origDatablock*, storing the information of files created during the dataset collection.
- As an example, one dataset has a link to its proposal.
- The functionalities requiring users to be authenticated can be exploited using some predefined accounts.
- The *panosc-search-API*, a layer that translates data catalogue specific metadata to a common PaN terminology, does not support authentication, so it can retrieve public data only.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

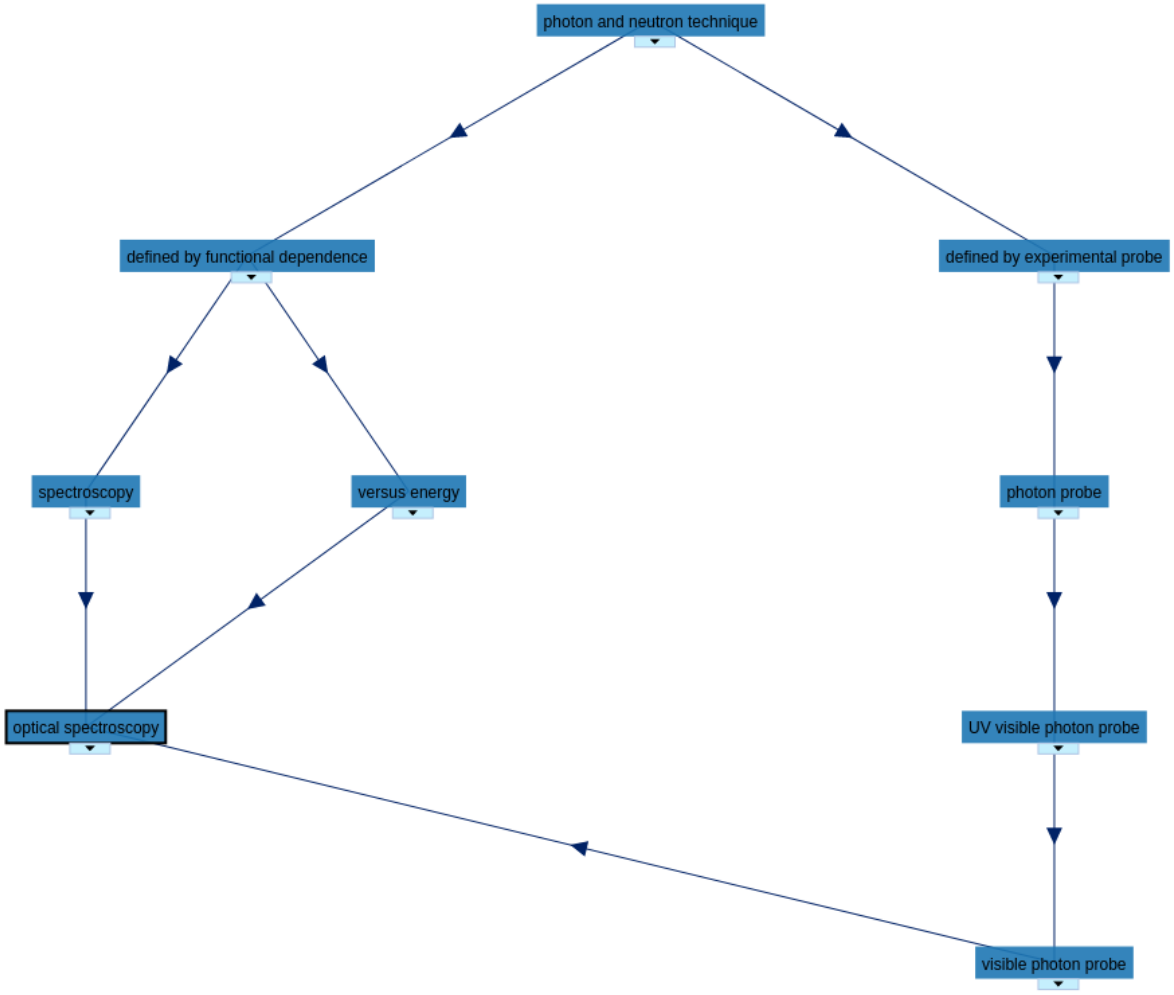

Figure 3: A graphical representation of the subclasses of the technique 'optical spectroscopy' produced by NCBO BioPortal (see [2] for more details).

3.4 Implementation

The *metadata catalogue release* is based on container (https://www.docker.com/resources/what-container) orchestration. It conceptually consists of two blocks, one gathering the containers and the other running their orchestration, such that links between different components are fulfilled. Each of the components described in the previous section is a container that comes from two distinct sources. The first is made of well known and established containers, developed by the web community, and which are ready to use. To this group belong the *Mongo database* (https://hub.docker.com/r/bitnami/mongodb) and the *traefik reverse proxy* (https://hub.docker.com/_/traefik), all at their latest, but hardcoded, version. The second source is made of containers developed in a collaboration including ExPaNDS and PaNOSC communities, which are part of the data catalogue implementation. These are *catamel* (https://hub.docker.com/r/dacat/catamel), *catanie* (https://hub.docker.com/r/dacat/catanie)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

and the *panosc-search-API* (https://hub.docker.com/r/dacat/panosc-search-api), all coming from the dacat (https://hub.docker.com/u/dacat) community organisation on Docker Hub (https://hub.docker.com/). As before, they are downloaded in their latest version, but unlike before their version is not hardcoded in the *docker-compose* file. This is done to always retrieve the latest *dacat* versions when installing the data catalogue *reference implementation*.
The *docker-compose* file is now responsible for downloading such containers from Docker Hub and making sure they are up and running at any point in time, managing their relative dependencies.

Here follows a brief description of the components:

- *Mongo database*: it is the database where metadata is stored. Being a document-based DB, no fixed metadata structure is mandatory.
- *Catamel*: it is the backend of SciCat, namely it consists of APIs and REST endpoints to run operations on the database.
- *Catanie*: it is a single-page application that creates the GUI for the end-user. This application makes use of the *catamel* endpoints to interact with the database.
- *Panosc-search-API*: it can be described as a translation layer from the SciCat specific metadata to a PaN community terminology, which the end-user can use to query.
- *Traefik reverse proxy*: it makes the components aforementioned reachable from the host network, redirecting to the right application depending on the path in the URL relative to *localhost*. The end-user can then, depending on the path in the URL, land on the right application. For example: *localhost → catanie, localhost/api → catamel, localhost/panosc-api → panosc-search-API*.
- *Docker-compose file*: contains the commands to deploy each of the aforementioned applications and some configuration settings, for example, the path-based routing.

In addition, the Mongo database contains a set of initial metadata, covering publications of ExPaNDS deliverable 4.2 “Photon and Neutron reference data sets” [3], that are stored into the database using mongo import functionalities and that can be used according to the PaNET ontology definitions [2].

The resultant implementation and all input files are maintained on a public GitHub repository: https://github.com/SciCatProject/scicatlive.

3.5 Usage

For all the three applications developed by the data catalogue community, namely *catamel*, *catanie* and *panosc-search-API*, the user can find documentation about their usage, respectively on:

- *Catamel* and *catanie*: https://scicatproject.github.io/documentation/
- *Panosc-search-api*: https://github.com/panosc-eu/search-api/tree/master/doc

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 857641.

Both *catamel* and *panosc-search-API* are based on the *loopback* framework (https://loopback.io/), so the end-user can exploit its functionalities, when interacting with the application's endpoints, through the *loopback query* syntax (https://loopback.io/doc/en/lb3/Querying-data.html), i.e. including the query in the *HTTP query string* (http://en.wikipedia.org/wiki/Query_string).
This can be achieved either by calling the application endpoints directly or by using the *explorer* (https://loopback.io/doc/en/lb3/Use-API-Explorer.html) functionalities, which is an interface where to test endpoints. The *explorers* of *catamel and panosc-search-API* are available respectively at *localhost/explorer* and *localhost/panosc-explorer*, after the deployment of the *metadata catalogue release*.

Catanie is a single page application and its usage, aside from what is described in the documentation, can be explored by the end-user at *localhost*, after having deployed the *metadata catalogue release*. To use the metadata editing functionalities, the end-user needs to be logged in with the *admin* account.
For the *metadata catalogue release* deployment, the only step required is described in the README.md file in https://github.com/SciCatProject/scicatlive/. The documentation also describes how to modify data ingested during the database creation.

Official documentation for *Traefik reverse proxy* and *Mongo DB* can be found on their web pages, respectively: https://doc.traefik.io/traefik/ and https://docs.mongodb.com/manual/tutorial/getting-started/.

3.6 Examples

In this section, we provide some usage examples of the *APIs* of *catamel* and *panosc-search-API*. For the latter, we analyse how one user can tie back to PaNET [2] and how this implementation makes it easy to find all datasets using a particular technique no matter where the term used by the end-user during the search lives in the PaNET structure [2].
For *catanie* we think the examples in the official documentation should suffice.

As it will be shown in the examples, and as mentioned previously, the loopback filters are added in the *HTTP query string*.

Catamel:

- Get all datasets where dataset type = raw:
  - Loopback filter:

```
{
 "where": {
   "type": "raw"
 }
}
```

  - Request URL:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

```
http://localhost/api/v3/Datasets?filter={"where":{"type":"raw"}}
```

- Get all samples and their associated datasets:
  - Loopback filter:

```
{
 "include": "datasets"
}
```

  - Request URL:

```
http://localhost/api/v3/Samples?filter={"include": "datasets"}
```

- Get at most 10 samples having associated datasets scientificMetadata.beamEnergy.value = 21 and scientificMetadata.beamEnergy.unit = keV:
  - Loopback filter:

```
{
 "include": {
   "relation": "datasets",
   "scope": {
     "where": {
       "and": [
         {
           "scientificMetadata.beamEnergy.value": 21
         },
         {
           "scientificMetadata.beamEnergy.unit": "keV"
         }
       ]
     }
   }
 },
 "limit": "10"
}
```

  - Request URL:

```
http://localhost/api/v3/Samples?filter={"include":{"relation":"datasets","scope":{"where":{"and":[{"scientificMetadata.beamEnergy.value":21},{"scientificMetadata.beamEnergy.unit":"keV"}]}}}, "limit":"10"}
```

These are just a few examples of *loopback queries* and more details and functionalities can be found in the official documentation: https://loopback.io/doc/en/lb3/Querying-data.html.

Panosc-search-API:

The most common use cases of the *panosc-search-API* can be found in the documentation (https://github.com/panosc-eu/search-api/tree/master/doc), with suitable examples.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

It is worth depicting the example of a user searching for datasets using a particular technique, as this is not yet in the official documentation.

- Get all datasets with techniqueId = PaNET01168:
  - Loopback filter:

```
{
 "include": [
   {
     "relation": "techniques",
     "scope": {
       "where": {
         "pid": "PaNET01168"
       }
     }
   }
 ]
}
```

  - Request URL:

```
http://localhost/panosc-api/Datasets?filter={"include":[{"relation":"techniques", "scope":{"where":{"pid":"PaNET01168"}}}]}
```

- Get all datasets whose technique is a subclass of scattering_technique:
  - Loopback filter:

```
{
 "include": [
   {
     "relation": "techniques",
     "scope": {
       "where": {
         "name": {
           "like": "scattering_technique"
         }
       }
     }
   }
 ]
}
```

  - Request URL:

```
http://localhost/panosc-api/Datasets?filter={"include":[{"relation":"techniques","scope":{"where":{"name":{"like":"scattering_technique"}}}}]}
```

The last example shows that it is possible to query on any depth in the PaNET tree [2] and get all the datasets whose technique is a subclass of the term in the query.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

3.7 Update and maintenance workflow

For the V1.0 *metadata catalogue release,* we have included all components strictly required to have a stand-alone data catalogue instance plus the *panosc-search-API*.

All future updates, either from this community or from external contributors, will follow a standard Github workflow, i.e the contributor, after having implemented a new feature, will open a *pull request* and changes will be reviewed and eventually accepted. These changes will very likely involve the addition of new components in the *docker-compose* file, for example adding an authentication service.

For what concerns the updates from *dacat* containers (*catanie*, *catamel and panosc-search-API)*, these follow a continuous deployment approach, which is such that every time a change will be approved by the community, a new release will be published on *dacat* at Docker Hub and this new version will be automatically downloaded by the *metadata catalogue release*. On the contrary, this does not happen for *traefik reverse proxy* and the *Mongo DB*, because, since we don't control their development, we decided to stick to the latest working version, at the time of this report, hardcoding it in the *docker-compose* file. This can still be updated, by following the Github workflow aforementioned.

4. Future developments

A *metadata catalogue release* can generally have many more components, which are not strictly required for its basic functionalities. This version implements the ones which can be considered mandatory when setting up a metadata catalogue in a PaN facility. Many more can be added and are likely to be part of future developments.
To name a few:

- Authentication service
- Message queue
- OAI-PMH
- Landing page server
- And many more...

4.1 SciCat

All these components (https://github.com/orgs/SciCatProject/repositories) can be part of the *docker-compose* file previously described, contributing to the idea of having an evolving one-click installation of a metadata catalogue, with an increasing amount of functionalities over time.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

4.2 ICAT

ICAT is a tool ecosystem to support Photon and Neutron data management that has been used in production systems in large-scale facilities across Europe for over 10 years. The community behind ICAT development is the ICAT collaboration and it includes partners such as Diamond Light Source (UK), the European Synchrotron Radiation Facility (France, PaNOSC), Helmholtz-Zentrum Berlin für Materialien und Energie (Germany), ISIS Neutron and Muon Source (UK), ALBA (Spain), Central Laser Facility (UK). ICAT handles large-scale volumes of data. For example, Diamond's data archive hosted over 27 PetaBytes and 3.109 billion files of data back in February 2021.

ICAT is designed in a modular fashion with an ICAT instance composed of several components installed into a JavaEE application server. Component implementations exist for authenticator services covering different authentication types (for example LDAP: https://github.com/icatproject/authn.ldap and OpenID Connect: https://github.com/icatproject/authn.oidc), an OAI-PMH interface (https://github.com/icatproject/icat.oaipmh) and SOAP & REST interfaces (SOAP: https://repo.icatproject.org/site/icat/server/4.9.1/soap.html and REST: https://repo.icatproject.org/site/icat/server/4.9.1/miredot/index.html and https://github.com/ral-facilities/datagateway-api). There are several options for user interfaces that allow users to browse, search, share and download their data (eg. Topcat: https://github.com/icatproject/topcat and DataGateway: https://github.com/ral-facilities/datagateway), together with landing pages for published datasets identified with Digital Object Identifiers. The latest RESTful API to ICAT (https://github.com/ral-facilities/datagateway-api) will be extended to provide the interface required for the common Search API across Photon and Neutron Facilities (see Figure 1 and PaNOSC deliverable D3.1 [1]).

The ICAT collaboration continues to make progress on the required changes to the schema to support the PaNET ontology and extensions to support FAIR data, and the discussions can be followed in the icat.server issue tracker (https://github.com/icatproject/icat.server/issues?q=is%3Aissue+is%3Aopen+label%3Aschema).

The current installation and deployment process is documented per each component and the main entry point to the documentation is at https://icatproject.org/installation.

The ICAT collaboration has been working on a continuous deployment based on containerising each of the required ICAT components. A container including a full-stack of ICAT components in a single image has been contributed by a member of the ICAT collaboration and is available here: (https://hub.docker.com/r/rkrahl/icat). The work to containerise each of the ICAT components individually and orchestrate them for continuous deployment is ongoing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.

References

[1] https://www.panosc.eu/wp-content/uploads/2020/12/D3.1_API-definition.pdf

[2] Collins, Steve P., da Graça Ramos, Silvia, Iyayi, Daniel, Görzig, Heike, González Beltrán, Alejandra, Ashton, Alun, Egli, Stefan, & Minotti, Carlo. (2021). ExPaNDS ontologies v1.0. Zenodo. https://doi.org/10.5281/zenodo.4806026

[3] Ashton, Alun, Barty, Anton, Fuhrmann, Patrick, Konrad, Uwe, Lang, Franz, Matej, Zdenek, Ounsy, Majid, Reynolds, Christopher, & Servan, Sophie. (2021). Photon and Neutron reference data sets. Zenodo. https://doi.org/10.5281/zenodo.4558708

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857641.