PaNOSC

Photon and Neutron Open Science Cloud

H2020-INFRAEOSC-04-2018

Grant Agreement Number: 823852

Deliverable: API Definition (common search API)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Project Deliverable Information Sheet

Project Reference No.	823852
Project acronym:	PaNOSC
Project full name:	Photon and Neutron Open Science Cloud
H2020 Call:	INFRAEOSC-04-2018
Project Coordinator	Andy Götz (andy.gotz@esrf.fr)
Coordinating Organization:	ESRF
Project Website:	www.panosc.eu
Deliverable No:	D3.1
Deliverable Type:	Report
Dissemination Level	Public
Contractual Delivery Date:	2020-05-31
Actual Delivery Date:	
EC project Officer:	René Martins

Document Control Sheet

Document	Title: API definition
	Version: 1
	Available at: https://github.com/panosc-eu/panosc/tree/master/Submitted%20Deliverables
	Files:
Authorship	Written by: Tobias Richter (ESS)
	Contributors: Gareth Murphy (ESS), Fredrik Bolmsten (ESS),
	Lajos Schrettner (ELI-DC), Stuart Caunt (ILL), Jamie Hall (ILL), William Turner (ILL), Sandor Brockhauser (EuXFEL), Henrik Johansson (ESS), Alejandro de Maria Antolinos (ESRF), Emiliano Coghetto (CERIC), Alessandro Olivo (CERIC), Silvia da Graca Ramos (ExPaNDS - DLS), Alun Ashton (ExPaNDS - PSI)
	Reviewed by: Andy Götz
	Approved: Jordi Bodera Sempere

List of participants

Participant No.	Participant organisation name	Country
1	European Synchrotron Radiation Facility (ESRF)	France
2	Institut Laue-Langevin (ILL)	France
3	European XFEL (XFEL.EU)	Germany
4	The European Spallation Source (ESS)	Sweden
5	Extreme Light Infrastructure Delivery Consortium (ELI-DC)	Belgium
6	Central European Research Infrastructure Consortium (CERIC-ERIC)	Italy
7	EGI Foundation (EGI.eu)	The Netherlands

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Table of Content

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 823852

Executive Summary

Overall, the work package is advancing according to plan. This document marks the delivery of the first version of a common search API[1] for open data. This API will be implemented by all PaNOSC[2] partner sites and ExPaNDS[3] (https://expands.eu) and any further facilities that want to join this effort as part of the EOSC[4]. A meeting was held jointly with the cataloguing work package of the ExPaNDS project and there is agreement that the API can be used as a common target by both projects to begin the implementation. This search API allows to find datasets and data publications based on relevant domain specific metadata and can be used by third parties to find data that has been released from any facility-imposed embargo period, as well as by the original researchers. This will be an important entry point for anyone to use the EOSC visualisation, processing or data transfer services that are being developed in other PaNOSC Work Packages (WPs).

Introduction

This document summarises the development of a common search API achieved in the PaNOSC Work Package 3. It explains the rationale behind the development and many of the decisions made as well as the general process that was followed.

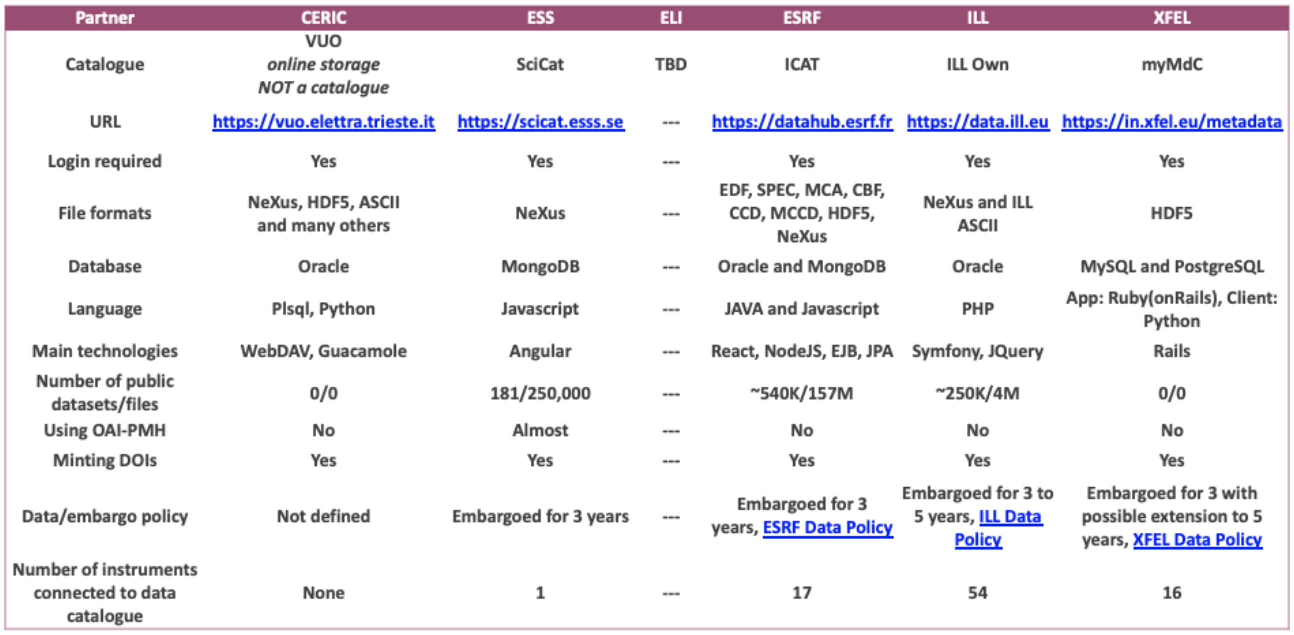
One of the main objectives of PaNOSC is to open up the large repositories of experimental data collected from measurements and experiments at photon and neutron European Research Infrastructures (RIs) to the scientific communities at large under the FAIR[5] principles. Due to the large number and volume of the dataset, being able to identify what data is and is not relevant before processing or transferring them is a strong requirement. To this end the PaNOSC facilities have deployed data catalogues that store metadata and support domain specific searches. Table 1 shows the status before this project was started, with a number of different catalogues, based on different technologies, and that all data catalogues (where they existed), required a login.

In order to get relevant information about the datasets held to the user communities, the project aims to deploy two different mechanisms. One is to open a service for harvesting metadata on all open access data to the EOSC aggregating repositories, namely OpenAIRE[6] and B2Find[7].
Alternatively, the common search API will provide a uniform way via a web portal or computer program to interrogate all data catalogues with specific searches in order to find data of interest by an individual.

[1] Application Programming Interface
[2] Photon and Neutron Open Science Cloud
[3] EOSC Photon and Neutron Data Service
[4] European Open Science Cloud
[5] Findable, Accessible, Interoperable and Reusable
[6] https://www.openaire.eu/
[7] https://www.eudat.eu/services/b2find

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Table 1: Overview of Data Catalogues deployed at the PaNOSC facilities and some related information (Status 2019)



Partner	CERIC	ESS	ELI	ESRF	ILL	XFEL
Catalogue	VUO *online storage* *NOT a catalogue*	SciCat	TBD	ICAT	ILL Own	myMdC
URL	https://vuo.elettra.trieste.it	https://scicat.esss.se	---	https://datahub.esrf.fr	https://data.ill.eu	https://in.xfel.eu/metadata
Login required	Yes	Yes	---	Yes	Yes	Yes
File formats	NeXus, HDF5, ASCII and many others	NeXus	---	EDF, SPEC, MCA, CBF, CCD, MCCD, HDF5, NeXus	NeXus and ILL ASCII	HDF5
Database	Oracle	MongoDB	---	Oracle and MongoDB	Oracle	MySQL and PostgreSQL
Language	Plsql, Python	Javascript	---	JAVA and Javascript	PHP	App: Ruby(onRails), Client: Python
Main technologies	WebDAV, Guacamole	Angular	---	React, NodeJS, EJB, JPA	Symfony, JQuery	Rails
Number of public datasets/files	0/0	181/250,000	---	~540K/157M	~250K/4M	0/0
Using OAI-PMH	No	Almost	---	No	No	No
Minting DOIs	Yes	Yes	---	Yes	Yes	Yes
Data/embargo policy	Not defined	Embargoed for 3 years	---	Embargoed for 3 years, ESRF Data Policy	Embargoed for 3 to 5 years, ILL Data Policy	Embargoed for 3 with possible extension to 5 years, XFEL Data Policy
Number of instruments connected to data catalogue	None	1	---	17	54	16

All the documentation about the search API is open and available on the PaNOSC confluence platform https://confluence.panosc.eu. Confluence is a content collaboration tool used to help teams collaborate and share knowledge therefore it is well suited to act as a documentation repository for the common search API and more generally for the PaNOSC project. The search API calls are described in the OpenAPI format (https://www.openapis.org/) and a tool called Swagger UI is used to visualize and interact with the API's resources without having any of the implementation logic in place (https://confluence.panosc.eu/x/TwGm). The search API calls are based on a well-defined and common data model whose entity-relationship schema can be found at https://confluence.panosc.eu/x/IQD8.

In the future ExPaNDS, the INFRAEOSC-05b project, will expand, accelerate and support the data management and data services provided through the EOSC for an additional ten national Photon and Neutron Research Infrastructures (PaN RIs). This will be achieved by working closely with PaNOSC on common objectives (e.g. a domain specific search API) and expand where their additional expertise the national facilities can bring added value to the endeavour.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Purpose of the API

The main goal of the search API is to provide a unified way for scientists to find data using parameters like the facility, instrument or experiment technique used to collect it or its main experiment parameters including source (X-ray or neutron beam) characteristics, sample information or detector details, or the name of investigators related to it, that originate from the different institutes in PaNOSC.

To present the data via the search API, the idea is to implement the API at each facility that will retrieve the necessary metadata and data for each search. A federated search engine can then browse the catalogues of each facility. In contrast to harvesting data from each facility and making it accessible from a central location, here we provide a possibility for centralised data queries which then directs the user to the facility storing the data.

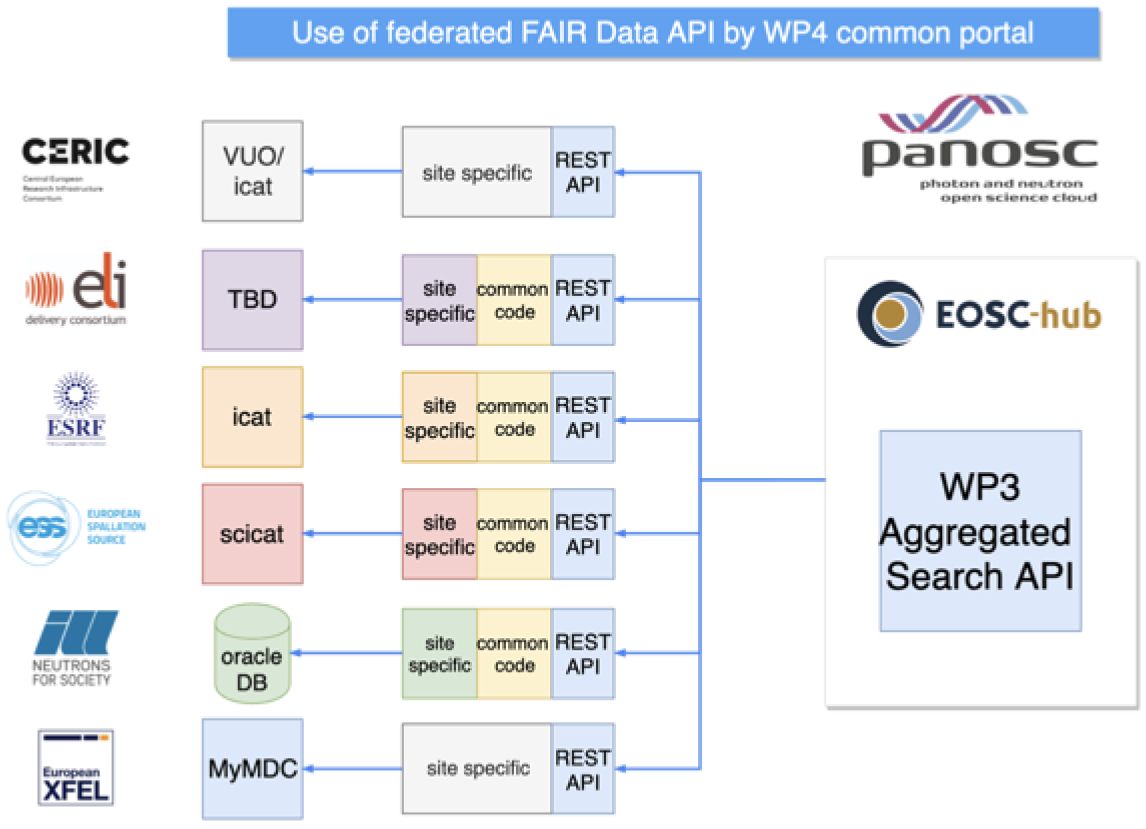


Figure 1: Overview of targeted deployment at partner sites and use of search API.

Since some data are under embargo, viewing their metadata, downloading or their analysis does require authentication. Being integrated with data analysis services prepared by WP4 (see figure 1), the search API helps not only in finding the data of interests, but also in being provided by a relevant data analysis environment where the chosen dataset can immediately be investigated. While WP6's authentication system can be integrated to support this, the main focus of the search API for this release is to enable open data to be integrated into EOSC/b2find/OpenAIRE services. Later revisions will bring improvements and clarify the integration of authentication.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Development Process

Early work in the WP

The WP participants regularly meet on monthly video conferencing to review and discuss the ongoing WP activities. Since the very start of the ExPaNDS project the WP meetings have seen a growing involvement of ExPaNDS people and are now regularly held together since both projects can benefit one another and the photon and neutron community as a whole.

Early work started out with the collection of potential actors and use cases. The full list of actors and use cases is available at https://confluence.panosc.eu/x/JIDS. In the process, particular attention has been given to involve some of the scientists operating at the PaNOSC institutes since they obviously represent the majority of the future PaNOSC users. Among other things, the inputs collected allowed the WP to clarify what the interesting data and the relevant metadata for the search including sample, chemical formula and scientific technique just to name a few.

Later a significant amount of work has been dedicated to a series of show and tell where every facility described their sources of metadata and current NeXus integration and usage so that commonalities could be identified and leveraged when defining the common search API. The full series of show and tell is available at https://confluence.panosc.eu/x/IwCm. Some of the facilities have shared some data files at https://confluence.panosc.eu/x/UACm.

All facilities agreed that NeXus usage would be beneficial and all are open to its usage. Currently, only a subset of every facility beamlines is generating NeXus files. However, despite Nexus being the data format that one would like to see in PaNOSC, the common search API is independent from any data format.

GitHub, the software development platform, has been chosen as the repository for both the code and the documentation. The WP GitHub repositories can be found at https://github.com/panosc-eu/search-api and https://github.com/panosc-eu/harvest-api.

For some types of resources GitHub is not particularly easy to use, so PaNOSC deployed Confluence. Confluence is a collaborative workspace document oriented and therefore is more suited to be used as a repository documentation and particularly as a tool for drawing and sharing sketches and class diagrams. The WP3 home page on Confluence can be found at https://confluence.panosc.eu/x/BQAW.

Summary of Meetings

The following sections describe the output from the meetings which shaped the search API in chronological order. The following meetings were where many discussions and decisions took place.

Kick off kick off meeting for WP in May 2019 (Copenhagen)

The agenda to the meeting can found on: https://indico.esss.lu.se/event/1233/
Topic discussed include:

- Fair data API
- Use cases for fair data API
- Federated search with EOSC hub
- Catalogue integration
- Ontologies

The WP core goal is to make PaNOSC data Findable, Accessible, Interoperable and Reusable (FAIR). As the main objective we will provide data to be harvested/pulled by EOSC/b2find/OpenAire and develop an API, focusing on open data. Authentication will be required for data analysis/WP4 case, where the intention is to

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

access embargoed data. The API will allow an optional integration of a suitable authentication solution from WP6 for searches. Access to the data, for download, processing or other purposes, will be possible via a link or identifier returned from the search (public or not). This scope is mostly covered by WPs 4, 5 and 6 in PaNOSC.

The FAIR data API presentation showed a roadmap to start providing open data to EOSC Hub with OAI-PMH[8] by providing an OAI-PMH endpoint and schema mapper to map individual database schema to shared PaNOSC schema. We can provide our metadata to EOSC data discovery services such as b2find, OpenAire to be harvested. Initially this can be limited to the Dublin Core metadata schema (which lacks the desired domain specific extensions we aim for). Each partner will mint DOI's[9] via Datacite, this in turn will provide an OAI-PMH endpoint that can be used for harvesting. Hitting this mark is a simple but important step that not all partners have taken. The next step is to address domain specific needs, either in the query endpoints or on the payload.

ILL Grenoble September 2019

The agenda to the meeting can found here: https://workshops.ill.fr/event/218/

The principal objective of the ILL workshop was to advance the specifications of the search API. It was also the first workshop with participants from the ExPaNDS project. The principal topics concerning the search API were:

- Data model (API endpoints and data structure)
- Search and query formulation
- API Validators
- Result aggregation from multiple facilities
- Federated Search Demonstrator

Presentations from this workshop are available on the PaNOSC confluence server: https://confluence.panosc.eu/x/aQCm. Notes taken during the meeting are also available: https://confluence.panosc.eu/x/YACm

Data model (API endpoints and data structure)

One of the main outcomes of the meeting was a first version of the data model (https://confluence.panosc.eu/x/WwCm). Initially a proposition was made to have a model that followed the lifecycle of experiments at an institute:

- Proposals
- Schedules
- Datasets (experiment)
- Datafiles

In other words, a Proposal would have a number of Schedules (planned experimental beamtimes, also called visits in some places) which in turn would be used to perform a number of experiments. Each experiment corresponds to one or more datasets which are composed of a number of datafiles.

However, this structure was not adapted to all facilities or use cases. For example:

- not all facilities had the notion of a schedule or were willing to publish them
- the above model does not allow for grouping of arbitrary datasets
- it does not allow for the notion of a scientific publication which can use data from multiple experiments

[8] https://www.openarchives.org/pmh/
[9] Data Object Idenfitier

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

An agreement was made on a more general model, including namely:

- Documents
- Datasets
- Datafiles

In this respect a Document can be *typed* to have a more specific meaning (i.e. proposal, publication, schedule or dataset group). As a result it was decided that both Document and Dataset are the main endpoints for the API (datafiles being related to, and therefore accessible by, a dataset). This generic approach was also aimed to meet the expectations of the scientist from other fields, that is not so much concerned with the operational workings of a photon or neutron facility.

Search and query formulation

It was decided that the API needs to be capable of performing complex search queries involving:

- multiple parameter queries (instrument, sample properties, scientific technique, dates, experimental team, etc)
- comparison (>, <, =, etc) and logical operators (and, or)
- querying metadata parameters from both proposals and datafiles

This requires a specific query language sent in an API request and to be processed by the FAIR Data implementation. During discussions it was proposed to use an existing query language, such as that of Loopback (https://loopback.io/doc/en/lb3/Querying-data.html).

API Validators

It was decided that a common set of tests would be required to ensure that each facility implemented the API correctly. The test harness will be executable against a given site catalogue service and result in a report summarising the status towards compliance. Some of the frameworks that were proposed were: Chakram, Tavern, Rest-assured, Apache J-meter and Karate.

Result aggregation from multiple facilities

There was a lot of discussion about how to aggregate search results across the facility sites and how the results should be ordered in terms of relevance for the given search query. There was overall agreement to defer the decisions on the specific provisions in the API for aggregation were better left for the time when problems could be addressed in practice, at a later point in the project.

Federated Search Demonstrator

It was decided that before a federated search demonstrator could be implemented, each site would have to implement the search API that connects to their data catalogue. The search API proposes the same structure for requests and responses. Facilities use different database backends for querying their data, therefore the search API provides an abstraction to provide access to these different storage options.

Meeting in Trieste at PaNOSC General Assembly, November 2019

The agenda can be found at https://confluence.panosc.eu/display/wp3/Draft+agenda

Presentations from the workshop are available on these links:

- Search API Hackathon with Loopback - https://tinyurl.com/y8ldut8j
- PaNOSC GA WP3 status update - https://tinyurl.com/ydcz33a5
- Consideration and suggestions on Search API - https://tinyurl.com/yb5qs9wh
- Notes: https://tinyurl.com/panosc-ceric-wp3

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

The meeting was a satellite meeting held right after the General Assembly and mainly focused on reviewing the milestones 3.1 and 3.2 and showcasing the Loopback implementation of the common search API.

The main topics of the workshop were:

- Reports on status and upcoming WP3 activities from each site
- Joint discussions with WP4 and WP6
- Hackathon on search API and loopback handson
- ResourceSync upgrade from OAI-PMH
- Come back to the points left opened during the Grenoble meeting

The six PaNOSC catalogues will make available their respective metadata to be harvested by EOSC providers including, but not limited to, OpenAIRE and B2find. The existing OAI-PMH has been chosen as the metadata provision protocol and in order to connect with OpenAIRE. All institutes will register with re3data.org as a research data repository and will roll out metadata to EOSC by registering with OpenAIRE as a research data provider.

All facilities agreed that NeXus usage would be beneficial and all are open to its usage. Currently, only a subset of beamlines are generating NeXus files, which means that significant effort will be required on the standardisation of the existing systems to cover their needs in encoding information in NeXus.

Each site reported on progress made since the last meeting. Considerable progress was made on the metadata harvesting side: almost all sites had successfully deployed OAI-PMH, had registered to re3data.org and can be harvested by OpenAIRE. The scheduled presentation of ResourceSync as successor of OAI-PMH has been postponed to the future. As a general idea the possible upgrade to ResourceSynch has been foreseen for 2021. The set of ResourceSynch specifications can be found on http://www.openarchives.org/rs/toc

Principal discussions that came out from the joint workshop with WP3, WP4, WP6:

- Portal:
    - Deployment (locally execution and federated service)
    - Services offered like remote desktop and/or Jupyter notebook
    - User experience
    - Two pilots (one more advanced)
    - Training and support
- How to move the data
- Working with EGI
    - For data transfer from RIs to compute centres
    - Notebook
- Work packages responsibilities
- How to expose services
- Authentication and Authorization solutions:
    - Integration with EOSC
    - UmbrellaID (provides single sign-on) - https://www.umbrellaid.org/
    - SAML - Security Assertion Markup Language
    - Keycloak - https://www.keycloak.org/
    - OpenAuth2 - https://oauth.net/2/
    - AAI Authentication and Authorisation Infrastructure - https://wiki.eosc-hub.eu/display/EOSC/Authentication+and+Authorization+Infrastructure+-+AAI
    - ORCiD

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

- openEDU Connect
- EA Hash

ESS showcased the first version of the Loopback implementation of the search API (https://github.com/panosc-eu/search-api). The session started with an overview of the application architecture and of the API endpoints. The rest of the session has been dedicated to a hackathon aimed at modifying and testing parts of the application. Participants were encouraged to attach the search API to a database or a fake data provider and implement PaNOSC use cases as unit tests.

Meeting of Cataloguing Work Packages with ExPaNDS in Lund, February 2020

An agenda to the meeting can found here: https://indico.esss.lu.se/event/1373/
Presentations from this workshop are available on the event page: https://indico.esss.lu.se/event/1373/
Notes taken during the meeting are also available at
https://confluence.panosc.eu/pages/viewpage.action?pageId=18186323

The main objective of the workshop was to advance the specifications of the FAIR Data API and increase maturity of the design in preparation for submitting the first deliverable. It was also an opportunity to have participants from the ExPaNDS project. The main topics of the meeting were:

- Review different catalogue systems in use by partner institutions
- Search API and harvest API discussions, OAI-PMH status
- API data model, search and query formulation, units conversion and handling
- Integration with analysis service work packages
- Finalisation of API, preparation for deliverable

The meeting began with a general overview of the work package and outlined the topics expected to be covered during the meeting. Numerous participants from ExPaNDS took part in a joint meeting the first time, they were introduced to the other participants and to the work that had been carried out up to the meeting. Their presence ensured that activities between PaNOSC and ExPaNDS are aligned as much as reasonably possible.
Alun Ashton gave an introduction to WP3 in ExPaNDS, which must cover most of the topics WP3 covers in PaNOSC, with a special emphasis on developing an ontology in Task 3.2.

Review different catalogue systems in use by partner institutions

Luke Gorman (PSI) gave a presentation about SciCat, a metadata catalogue used at e.g. PSI, ESS, and MAX IV, giving details about the design principles, aims, and the architecture of the system. SciCat intends to support the whole data lifecycle from user proposal to long term storage of the data on tape. The tools available for users include an integrated electronic logbook called SciChat. The catalogue architecture applies so called ingestors to collect data from the beamlines and stores them persistently. Various standard ingestors exist, as well as new ones can be developed to extend the systems to support additional beamlines. This feature is of particular interest to institutions planning to adopt SciCat as their catalogue system. The preferred deployment method of SciCat is via a Kubernetes cluster.
Stuart Pullinger (UKRI/STFC) gave a presentation about ICAT, a cataloguing system in use at several institutions, e.g. STFC, Diamond, HZB, and ESRF. He gave details about the design concepts, internal architecture, the data model, and roadmap for further developments. The two most important use scenarios are pushing data to the catalogue / file store and searching the catalogue to find files the user wants to download. Emphasis was put on the metadata scheme that is based on CSMD (Core Scientific Metadata Model). Site-specific data entities need to be mapped to ICAT concepts so that ICAT functions can operate on them. The system is highly customizable, several institutions put effort into developing their own extensions. One example is ICAT+, developed at ESRF, was presented by Maxime Chaillet. ICAT+ implements new services on top of ICAT like an e-logbook, a gallery feature, a search feature based on Elasticsearch. In the future other services will be added e.g. sample tracking.
Luís Maia (XFEL.EU) presented MyMdC, the catalogue system of XFEL.EU, which exhibits most of the same

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

concepts as the other catalogue systems. In particular it aims to organize and manage data in a coherent way, handle large volumes of data, and provide the possibility of inspecting data during experiments.

Search API and harvest API discussions

Once again, the differences and similarities in the use cases between the two APIs were discussed. The most important conclusions were the following: Facilities need to agree what level of granularity is necessary in order to publish data. A coarser granularity model is easier to define and implement, but may not provide the details that users expect when searching. On the other hand the granularity should not be site specific and still give common properties the same terms, in order to fully open the data to non-domain (or site) experts. The WP will start with a coarser definition for metadata and work towards standardising more and more search terms during the course of the project. In a similar way to what is planned for the harvesting metadata schema. Certain technology aspects need further attention, such as encoding of requests and answers, JSON support for encoding, the inclusion of MD5 hash into the datasets, for example.

API data model, search and query formulation, units conversion and handling

The data model defined and refined during earlier meetings was discussed, several additions/modifications were discussed at length and accepted. In particular, the issue of supporting queries which want to reference multiple levels of the model was brought up. The query syntax needs more documentation, also design decisions on how to be able to formulate the multilevel queries.
Also of interest is the issue of units handling: how measured quantities stored and/or searched for using various units can be handled. There are some existing software packages aiming to solve similar problems, these need to be investigated.
A list of scientific techniques need to be assembled and advertised in the ontology tasks, so that users can refer to them while formulating searches.

Integration with analysis service work packages

There is an integration aspect with WP4 (Data Analysis Services) in that search results are a natural input to data analysis services: the user searches for some data, then would like to proceed with analysis/visualisation of the data he or she found. On top of ensuring the functionality above, it is a possible requirement from WP4 that search results be previewed or visualised on the user interface before proper data analysis is attempted. For this to be supported, pre-generated image files would need to be attached to the dataset and/or datafiles.

Finalisation of API, preparation for deliverable

Regarding this topic the delivery format was discussed. An endpoint definition exists, but needs to be completed. The data model needs to be updated, as well as its documentation. Contributors were assigned to review different parts and Gareth Murphy (ESS) to finalize changes and documents.
The API is to be disseminated with sister project ExPaNDS. Issues and pull requests need to be created in the search API repository to track actions from the meeting minutes, implementation should cover more use cases. Definition of dictionaries were left for Task 3.5.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Documentation of the API

General Overview and Considerations

The REST framework Loopback was chosen for the implementation of the demonstrator search API. Loopback is a mature Open Source project that is backed and maintained by IBM, this should give it a level of sustainability for the future. It is following the REST architectural style, it also has a number of features built in such as JSON filters, ordering, pagination and authentication. The subset of features from Loopback that has been used has been documented in detail below, this enables other REST frameworks to be used for the local implementation (adaption to the facility data catalogue), as long as they adhere to the specified documentation. In addition to documenting the API a test harness has been developed that can be used to verify any implementing API.

The search API repository is available on GitHub at https://github.com/panosc-eu/search-api. Anyone trying to use or implement it can raise issues and submit pull requests there. If a facility develops an interface to a metadata catalogue currently not supported this can then be used by other facilities. The Search API can be run either locally by using node package manager or by building the Docker file found in the repository root. Setting up with a container is as simple as running *"docker build . -t search-api"* followed by *"docker run -it seach-api -p 3000:3000"* this will expose the service on localhost:3000.

Data Model

The description of the data structures used to compose a search query is obviously a defining part of the API. Results of the search are returned in the same data model. Figure 2 shows the status of the model as used by the search API released with this document. Classes and their dependencies are shown, with the respective requirements. Classes marked with an asterisk (*) are optional, but can be present multiple times. What is marked with a plus sign (+) is required at least once. Numbers (1) or ranges (0..1) indicate a single required instance or zero or one instance respectively.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

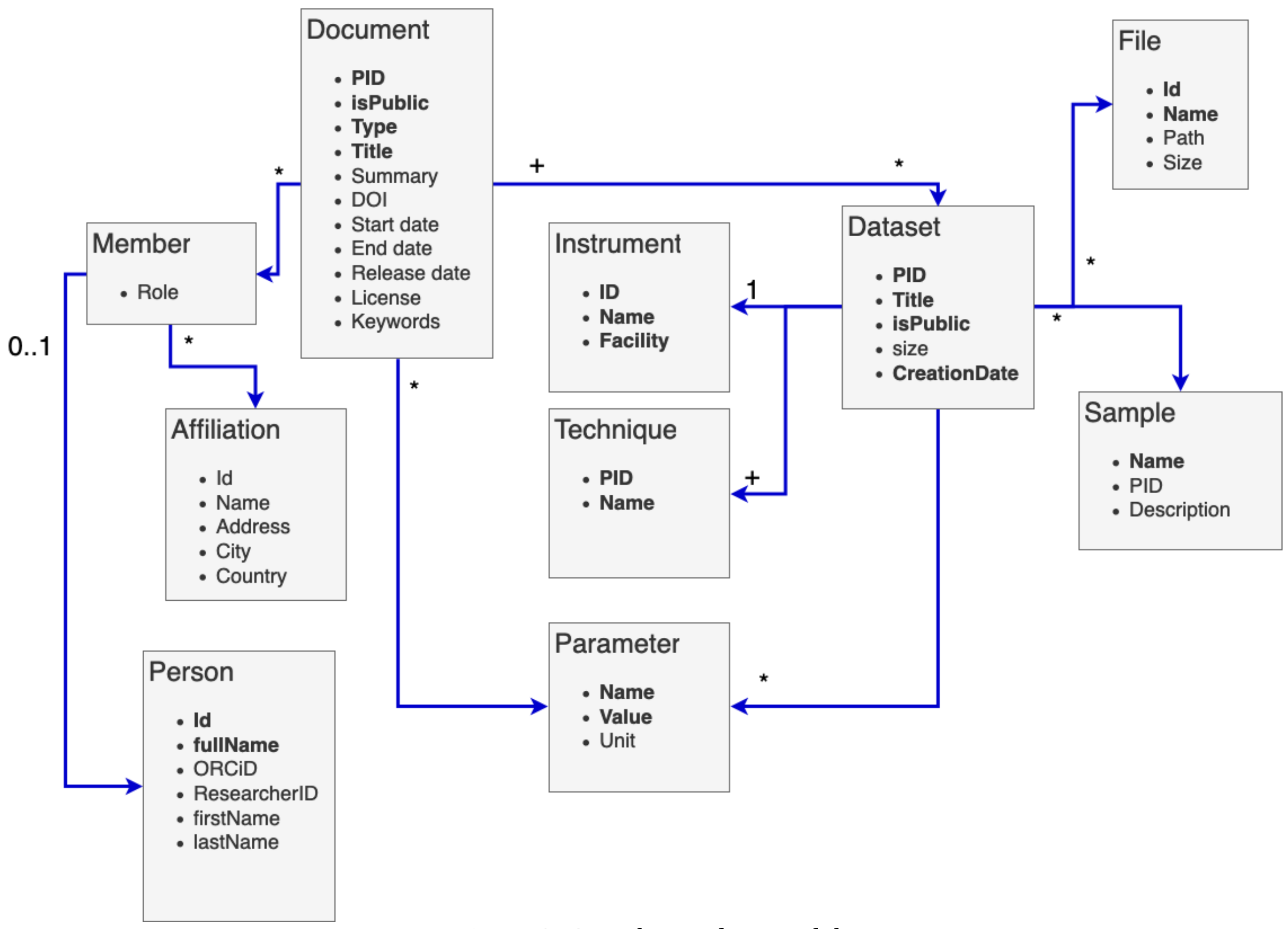


Figure 2: Search API data model

Classes that may be returned by API calls have an id property allowing reference to them in subsequent calls like GET /datasets/{id}. This id may be an internal identifier of the local metadata catalogue. It should be considered ephemeral and should not be retained by the client beyond the current session. Some amount of effort should be taken by the server to make this property globally unique to prevent clashes. The value should be restricted to the characters 0-9A-Za-z_.~-. Future versions of the API may pose more stringent restrictions on this property for federation purposes.

Some classes have a PID property. This is a persistent identifier that is supposed to not change over time and may be stored in the client for later referral. It also allows cross references to objects in remote repositories. The value should be a well established persistent identifier such as a DOI, a Handle, an ORCID-iD, or a ROR. If such a PID is not available for the object, a locally assigned identifier in the metadata catalogue is acceptable, as long as it is guaranteed to be stable.

Document

Represents a proposal, beamtime, measurement campaign, a (data) publication, or groups dataset for a specific sample. A curated list of document types may come out of the ontology task in work package 3.

Dataset

A dataset combines information about an experimental run, including optional File, Sample, Instrument and Technique. The granularity is so that this should be normally the smallest unit that can be analysed sensibly. That may require including multiple files, for example multiple images from a tomography run.

File

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Reference to a data file with data, which can be used for further PaNOSC services, like analysis, visualisation, transfer etc.

Instrument

Experimental station where experiment took place. In order to provide a consistent view of the data, the choice was to only allow a single entity per dataset. Some facilities wish to express finer granularity and could distinguish between end stations, beamlines, sample environment equipment or detectors. But from the user point of view a consistent level of detail in the information returned is desirable.

Technique

The experimental method used. At least one value is required. The list currently being created in the ontology task https://confluence.panosc.eu/display/wp3/PaNOSC+Scientific+Techniques will include sufficient generic choices, like "neutron" or "scattering", that should match any data from a particular instrument or beamline. This way also legacy data, where the specific intentions of the experiment may not have been recorded, can receive a technique label that still provides the user benefit over having no such information.
The aim of the ontology task is to come up with a hierarchical or inclusive scheme that would allow matching related techniques, for example recognise "absorption spectroscopy" as a "spectroscopy" technique.

Sample

Substance, material or object probed by neutrons or photons in the experiment.

Parameter

The sample and the technique will be some of the most frequently used search terms, according to the use cases that have been sampled. In addition the dataset, instrument, file or sample can have a number of parameters that may be useful to further filter on. How these parameters are associated in the individual data catalogues at partner sites depends on the choices made there. For simplicity this API attaches parameters exclusively to datasets and documents, as these are the main search endpoints (see below).
The ontology task will curate a list of parameter keywords that the partners will then map onto what is stored in their catalogues (now and in future). Typical examples would be:

- sample temperature
- sample size or thickness
- photon energy
- neutron wavelength
- total number of counts

In the ontology discussions these will receive a single unique name. What is currently in use by the API demonstrator and test cases is for illustration only, like for the technique, roles, etc.
Parameter values are scalar measurement values with units. Strings are also permitted. We rely on JSON using double quotes for strings, for example

```
	{ "name": "detector1_name", "value": "incoming_beam" }
```

versus

```
	{ "name": "detector1_data", "units": "A", "value": 3.38e-05 })
```

to distinguish either.

Member

An individual associated with the data in a role defined by the role property, for example the principal investigator of the experiment, or a person involved in the data analysis. The allowed values of the role property also come out of the ontology task.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Person

An individual associated with the data in a role defined by the Member class, for example the principal investigator of the experiment, or a person involved in the data analysis.

Affiliation

Home institution of a member.

API Endpoints

As the purpose of the API is to find metadata of documents (currently publications and proposals) and the datasets that are part of these documents, a decision has been made to only expose endpoints to access documents, datasets and instruments.

Dataset

Get a single dataset

Call
GET /datasets/{pid}

Path parameters
pid : the PID of the dataset

Query parameters
filter: a query

Response
A dataset JSON object as defined in the data model.

```
{
    "pid": "20.500.12269/panosc-dataset1",
    "title": "PaNOSC Test Dataset 1",
    "isPublic": true,
    "creationDate": "2020-05-05T15:01:02.341Z",
    "documentId": "10.5072/panosc-document1",
    "instrumentId": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71"
}
```

Get files for a dataset

Call
GET /datasets/{pid}/files

Path parameters
pid : the pid of the dataset

Query parameters
filter: a query

Response

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

An array of file JSON objects as defined in the data model.

```
[
   {
      "id": 1,
      "name": "panosc-file1.hdf",
      "datasetId": "20.500.12269/panosc-dataset1"
   }
]
```

Get number of files for a dataset

Call
GET /datasets/{pid}/files/count

Path parameters
pid : the pid of the dataset

Query parameters
where: a where query

Response
A JSON object.

```
{
  "count": 1
}
```

Search for datasets

Call
GET /datasets

Query parameters
filter : a query

Response
An array of dataset JSON objects as defined in the data model.

```
[
   {
      "pid": "20.500.12269/panosc-dataset1",
      "title": "PaNOSC Test Dataset 1",
      "isPublic": true,
      "creationDate": "2020-05-05T15:01:02.341Z",
      "score": 0,
      "documentId": "10.5072/panosc-document1",
     "instrumentId": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71"
   },
   {
      "pid": "20.500.12269/panosc-dataset2",
      "title": "PaNOSC Test Dataset 2",
      "isPublic": true,
      "score": 0,
      "creationDate": "2020-05-05T15:01:02.341Z",
      "documentId": "10.5072/panosc-document1",
      "instrumentId": "20.500.12269/125e8172-d0f4-4547-98be-a9db903a6269"
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
    }
]
```

Get number of datasets

Call
GET /datasets/count

Query parameters
where : a where query

Response
A JSON object.

```
{
  "count": 4
}
```

Documents

Get a single document

Call
GET /documents/{pid}

Path parameters
pid : the pid of the document

Query parameters
filter : a query

Response
A document JSON object as defined in the data model.

```
{
    "pid": "10.5072/panosc-document1",
    "isPublic": true,
    "type": "publication",
    "title": "PaNOSC Test Publication"
}
```

Search documents

Call
GET /documents

Query parameters
filter : a query

Response
An array of document JSON objects as defined in the data model.

```
[
    {
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
        "pid": "10.5072/panosc-document1",
        "isPublic": true,
        "type": "publication",
        "title": "PaNOSC Test Publication",
        "score": 0
    },
    {
        "pid": "10.5072/panosc-document2",
        "isPublic": true,
        "type": "proposal",
        "title": "PaNOSC Test Proposal",
        "score": 0
    }
]
```

Count documents

Call
GET /documents/count

Query parameters
where : a where query

Response
A JSON object.

```
{
  "count": 2
}
```

Instruments

Get a single instrument.

Call
GET /instruments/{pid}

Path parameters
pid : the pid of the instrument

Response
An instrument JSON object as defined in the data model.

```
{
    "pid": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71",
    "name": "LoKI",
    "facility": "ESS"
}
```

Search instruments

Call
GET /instruments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Path parameters
filter : a query

Path parameters
filter : a query

Response
An array of instrument JSON objects as defined in the data model.

```
[
    {
        "pid": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71",
        "name": "LoKI",
        "facility": "ESS",
        "score": 0
    },
    {
        "pid": "20.500.12269/125e8172-d0f4-4547-98be-a9db903a6269",
        "name": "ODIN",
        "facility": "ESS",
        "score": 0
    }
]
```

Count instruments

Get number of instruments

Call
GET /instruments/count

Path parameters
where : a where query

Response
A JSON object. The example below is the response expected without any filter applied.

```
{
  "count": 15
}
```

Units and Conversions

Measurement values and their associated units must be first class citizens in a data catalogue and a search API. The adequate choice of units for a particular quantify, experiment and sample is important to the user carrying out the experiment. Users would like to see the units displayed back to them how they were entered. Storing everything in SI units is not acceptable. That does have implications on the search. Just matching values in the database engine would be easy, but there is no universally accepted choice of units. For these reasons' conversions need to be performed on the fly.

The choice of which units to be supported in the search was created based on observations from current metadata catalogues and domain knowledge. It is given in table 2, with the accepted prefixes in table 3.

When supplying units in a parameter query, the quantity will be converted for comparison with the value stored

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

in the database. Before returning the results, the relevant quantity is converted to the unit supplied by the user in the query. For example when querying a parameter using *keV*, the keV quantity will be converted and compared to the value stored in the database. Results will also be returned in the same unit that the user provided in the query, in this case *keV*. This enables the user to easily compare the results of the search with their filter, but also to sort results by that parameter. The functionality for this is provided by the *math.js* library (https://mathjs.org/), which also includes support for the chosen units and prefixes.

Table 2: Supported units. The notation in brackets represent alternative ways to specify the unit in the API.

Base	**Unit**
Length	meter (m), angstrom
Angles	rad (radian), deg (degree), grad (gradian), arcsec (arcsecond), arcmin (arcminute)
Time	second (s, secs, seconds), minute (mins, minutes), hour (h, hr, hrs, hours), day (days)
Frequency	hertz (Hz)
Mass	gram (g)
Electric Current	ampere (A)
Temperature	kelvin (K), celsius (degC), fahrenheit (degF)
Amount of Substance	mole (mol)
Luminous Intensity	candela (cd)
Force	newton (N)
Energy	joule (J), erg, Wh, electronvolt (eV)
Power	watt (W)
Pressure	Pa, psi, atm, torr, bar
Electricity and Magnetism	ampere (A), coulomb (C), watt (W), volt (V), ohm, farad (F), weber (Wb), tesla (T), henry (H), siemens (S), electronvolt (eV)
Binary	bits (b), bytes (B)

Table 3: Supported decimal prefixes.

Name	**Abbreviation**	**Value**
yocto	y	1e-24
zepto	z	1e-21

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

atto	a	1e-18
femto	f	1e-15
pico	p	1e-12
nano	n	1e-9
micro	u	1e-6
milli	m	1e-3
centi	c	1e-2
deci	d	1e-1
deca	da	1e1
hecto	h	1e2
kilo	k	1e3
mega	M	1e6
giga	G	1e9
tera	T	1e12
peta	P	1e15
exa	E	1e18
zetta	Z	1e21
yotta	Y	1e24

Support for searching ranges of a value is included, this will enable users to specify a specific range a parameter needs to reside between. This is essential as filtering metadata values after conversions with exact matches with only work in a few cases. How this is specified in the API is borrowed from loopback and illustrated in

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

the example use cases below.

Examples from Use Cases

Like for the description of the end points above, the demonstrator search API on GitHub has been primed with a few datasets that illustrate search use cases we intended to address. Test cases on the demonstrator API ensure the correct results are returned. This test harness can also be run against other implementations for verification.

Query datasets from X-Ray Absorption experiments

Endpoint: GET /datasets

Filter

```
{
    "include": [
        {
            "relation": "techniques",
            "scope": {
                "where": {
                    "name": "x-ray absorption"
                }
            }
        }
    ]
}
```

Response

```
[
    {
        "pid": "20.500.12269/panosc-dataset3",
        "title": "PaNOSC Test Dataset 3",
        "isPublic": true,
        "creationDate": "2020-05-05T15:01:02.341Z",
        "score": 0,
        "documentId": "10.5072/panosc-document2",
        "instrumentId": "20.500.12269/f0637030-9f89-4398-8f01-09211145efa1",
        "techniques": [
            {
                "pid": "20.500.12269/panosc-tech2",
                "name": "x-ray absorption"
            }
        ]
    },
    {
        "pid": "20.500.12269/panosc-dataset4",
        "title": "PaNOSC Test Dataset 4",
        "isPublic": true,
        "creationDate": "2020-05-05T15:01:02.341Z",
        "score": 0,
        "documentId": "10.5072/panosc-document2",
        "instrumentId": "20.500.12269/d3dd2880-637a-40b5-9815-990453817f0e",
        "techniques": [
            {
                "pid": "20.500.12269/panosc-tech2",
                "name": "x-ray absorption"
            }
        ]
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
    }
]
```

Query datasets with a photon energy between 880 and 990 eV

Endpoint: GET /datasets

Filter

```
{
    "include": [
        {
            "relation": "parameters",
            "scope": {
                "where": {
                    "and": [
                        {
                            "name": "photon_energy"
                        },
                        {
                            "value": {
                                "between": [880, 990]
                            }
                        },
                        {
                            "unit": "eV"
                        }
                    ]
                }
            }
        }
    ]
}
```

Response

```
[
    {
        "pid": "20.500.12269/panosc-dataset2",
        "title": "PaNOSC Test Dataset 2",
        "isPublic": true,
        "creationDate": "2020-05-05T15:01:02.341Z",
        "score": 0,
        "documentId": "10.5072/panosc-document1",
        "instrumentId": "20.500.12269/125e8172-d0f4-4547-98be-a9db903a6269",
        "parameters": [
            {
                "id": 3,
                "name": "photon_energy",
                "value": 930,
                "unit": "eV",
                "datasetId": "20.500.12269/panosc-dataset2"
            }
        ]
    }
]
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Query datasets with files matching a string using full text search

Endpoint: GET /datasets

Filter

```
{
    "include": [
        {
            "relation": "files",
            "scope": {
                "where": {
                    "text": "file1"
                }
            }
        }
    ]
}
```

Response

```
[
    {
        "pid": "20.500.12269/panosc-dataset1",
        "title": "PaNOSC Test Dataset 1",
        "isPublic": true,
        "creationDate": "2020-05-05T15:01:02.341Z",
        "score": 0,
        "documentId": "10.5072/panosc-document1",
        "instrumentId": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71",
        "files": [
            {
                "id": 1,
                "name": "panosc-file1.hdf",
                "datasetId": "20.500.12269/panosc-dataset1"
            }
        ]
    }
]
```

Query documents where wavelength is between 1000-1100 nm

Endpoint: GET /documents

Filter

```
{
    "include": [
        {
            "relation": "parameters",
            "scope": {
                "where": {
                    "and": [
                        {
                            "name":"wavelength"
                        },
                        {
                            "value": {
                                "between": [1000, 1100]
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
                        }
                    },
                    {
                        "unit": "nm"
                    }
                ]
            }
        }
    }
  ]
}
```

Response

```
[
    {
        "pid": "10.5072/panosc-document1",
        "isPublic": true,
        "type": "publication",
        "title": "PaNOSC Test Publication",
        "score": 0,
        "parameters": [
            {
                "id": 6,
                "name": "wavelength",
                "value": 1064,
                "unit": "nm",
                "documentId": "10.5072/panosc-document1"
            }
        ]
    }
]
```

Query documents investigating a particular sample using a certain technique

Endpoint: GET /documents

Filter

```
{
    "include": [
        {
            "relation": "datasets",
            "scope": {
                "include": [
                    {
                        "relation": "samples",
                        "scope": {
                            "where": {
                                "name": "solid copper cylinder"
                            }
                        }
                    },
                    {
                        "relation": "techniques",
                        "scope": {
                            "where": {
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
                        "name": "x-ray absorption"
                    }
                }
            }
        ]
    }
  }
 ]
}
```

Response

```
[
   {
       "pid": "10.5072/panosc-document2",
       "isPublic": true,
       "type": "proposal",
       "title": "PaNOSC Test Proposal",
       "score": 0,
       "datasets": [
           {
               "pid": "20.500.12269/panosc-dataset3",
               "title": "PaNOSC Test Dataset 3",
               "isPublic": true,
               "creationDate": "2020-05-05T15:01:02.341Z",
               "documentId": "10.5072/panosc-document2",
                "instrumentId": "20.500.12269/f0637030-9f89-4398-8f01-09211145efa1",
               "samples": [
                   {
                       "pid": "20.500.12269/panosc-sample1",
                       "name": "solid copper cylinder"
                   }
               ],
               "techniques": [
                   {
                       "pid": "20.500.12269/panosc-tech2",
                       "name": "x-ray absorption"
                   }
               ]
           },
           {
               "pid": "20.500.12269/panosc-dataset4",
               "title": "PaNOSC Test Dataset 4",
               "isPublic": true,
               "creationDate": "2020-05-05T15:01:02.341Z",
               "documentId": "10.5072/panosc-document2",
               "instrumentId": "20.500.12269/d3dd2880-637a-40b5-9815-990453817f0e",
               "samples": [
                   {
                       "pid": "20.500.12269/panosc-sample1",
                       "name": "solid copper cylinder"
                   }
               ],
               "techniques": [
                   {
                       "pid": "20.500.12269/panosc-tech2",
                       "name": "x-ray absorption"
                   }
               ]
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
            }
        ]
    }
]
```

Query instruments by name

Endpoint: GET /instruments

Filter

```
{
    "where": {
        "name": "LoKI"
    }
}
```

Response

```
[
    {
        "pid": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71",
        "name": "LoKI",
        "facility": "ESS",
        "score": 0
    }
]
```

Query instruments at a certain facility with pagination

Endpoint: GET /instruments

Filter

```
{
    "where": {
        "facility": "ESS"
    },
    "skip": 0,
    "limit": 3
}
```

To get the next three results, set "skip": 3, then "skip": 6, and so on.

Response

```
[
    {
        "pid": "20.500.12269/07297dd4-557f-4ef6-974f-5c1eec610b9e",
        "name": "ESTIA",
        "facility": "ESS",
        "score": 0
    },
    {
        "pid": "20.500.12269/0f98fcf2-7bd7-430e-ad20-d47031ca8f71",
        "name": "LoKI",
        "facility": "ESS",
        "score": 0
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

```
    },
    {
        "pid": "20.500.12269/125e8172-d0f4-4547-98be-a9db903a6269",
        "name": "ODIN",
        "facility": "ESS",
        "score": 0
    }
]
```

Signalling of Errors

Upon invalid inputs, the API responds with an error in JSON format, while a stack trace is logged to the server console. The error JSON object contains the following fields:

- "statusCode": An HTTP status code of either 4xx or 5xx
- "name": The name of the error
- "message": A descriptive error message, to aid the user in troubleshooting

Example of error response:

```
{
    "error": {
            "statusCode": 400,
            "name": "Error",
            "message": "Cannot parse JSON-encoded object value."
    }
}
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Next Steps

With this baseline version of the search API agreed, between the PaNOSC ESRFI[10] research infrastructures, and the ExPaNDS national research infrastructures, the obvious next step in the plan is to start implementing this API and connect it up to the local data catalogues. In this process, undoubtedly some issues will arise that were unforeseen. The same is likely to happen, when the demonstrator, or an early implementation at a facility, gets connected up to the prototype data portal from work package 4. Both may lead to new releases of this API. However, we expect mainly minor modifications, that will not jeopardise the overall design or would result in larger redesigns of the local implementations.

In order to enable wider testing and exposure of the API it has been suggested to create a Python interface for using the API. This would allow simple programmatic as well as interactive use inside Jupyter notebooks. Via node.js npm packages this may be a relatively easy target.

The two outstanding issues to be resolved within the project are the federation and authentication. For the authentication, work package 6 is preparing important groundwork. Once this is available in normal operations, adapting this to a specific web service should be straightforward. For the federation and aggregation of results there are a number of expected issues like scoring and sorting of results, pagination, local caching of aggregated results in the client, etc. Before these can be investigated in detail, it is necessary to ensure there are a number of compliant implementations of the current baseline API.
While the implementation work is being started, both cataloging work packages (ExPaNDS and PaNOSC) expect to make good progress on the ontologies, namely the list of parameters, their mappings with unit conversions, the hierarchy of experimental techniques as well as a list of roles for dataset members. Early draft versions can be issued relatively soon and would allow for speedy testing of the API with some use cases on data that has already been released out of embargo periods.

[10] European Strategy Forum on Research Infrastructures

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852