ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

EXTENDING THE ICAT METADATA CATALOGUE TO NEW SCIENTIFIC USE CASES

A. Götz*, A. de Maria, M. Bodin, M. Gaonach, ESRF, Grenoble, France
A. Gonzalez Beltran, P. Austin, K. Phipps, V. Bozhinov, L. Davies, STFC, Harwell, United Kingdom
R. Krahl, HZB, Berlin, Germany
M. Al Mohammed, S. Ali Matalgah, SESAME, Allan, Jordan
R. Cabezas Quirós, ALBA, Barcelona, Spain
K. Syder, Diamond Light Source, Harwell, United Kingdom
A. Pinto, Brazilian Synchrotron Light Laboratory, Campinas, Sao Paulo, Brazil

Abstract

The ICAT metadata catalogue is a flexible solution for managing scientific metadata and data from a wide variety of domains following the FAIR data principles. This paper will present an update of recent developments of the ICAT metadata catalogue and the latest status of the ICAT collaboration. ICAT was originally developed by UK Science and Technology Facilities Council (STFC) to manage the scientific data of ISIS Neutron and Muon Source and Diamond Light Source. They have since been joined by a number of other institutes including ESRF, HZB, SESAME, ALBA and SIRIUS who together now form the ICAT Collaboration[1]. ICAT has been used to manage petabytes (12.4 PBs for ESRF and over 50 PBs for DLS) of scientific data for ISIS, DLS, ESRF, HZB, and in the future SESAME and ALBA and make these data FAIR. The latest version of the ICAT core as well as the new user interfaces, DataGateway and DataHub, and extensions to ICAT for implementing free text searching, a common search interface across Photon and Neutron catalogues, a protocol-based interface that allows making the metadata findable, electronic logbooks, sample tracking, and web-based data and domain specific viewers developed by the community will be presented.

INTRODUCTION

Science needs data in order to make new discoveries and verify existing theories. Data are therefore essential to science [1] and need to be managed for the short and long term i.e. during and after an experiment. Managing data during the experiment helps provide feedback to the experimentalists, while in the long-term, a record of the data needs to be kept so that scientists can get back to their own data after the experiment, and others can verify and eventually reproduce the results. Preserving data requires having a metadata catalogue for storing all metadata and references to data. The metadata catalogue supports browsing, searching and displaying of relevant metadata, as well as providing access to the data themselves. Over a thousand scientific data repositories exist today built on diverse solutions ranging from bespoke closed source data software to common open source metadata catalogue software. The choice of which solution to use will depend on the needs of the scientific domains being considered, as well as the technical preferences and available resources of the software engineers. This paper describes the ICAT solution, its architecture, implementation and which sites have adopted ICAT and how they use it.

ICAT ARCHITECTURE

Data Model

The metadata stored in ICAT follows a schema [2] based on the Core Scientific Meta-Data Model (CSMD), originally conceived in 2003 [3] and most recently updated in 2013 [4]. CSMD captures high level information about scientific studies and the data that they produce, and contains 27 entities, which have remained fairly stable since their inception. At the core of the ICAT schema (see Fig. 1), includes:

- **Investigations**, which set high level and scientific context. They may correspond to a proposal or visit to a facility.
- **Datasets**, which define context for creating data. They may correspond to a single experiment, measurement or simulation. A single **Investigation** can contain many **Datasets**.
- **Datafiles**, which represent the actual files that make up the data. A single **Dataset** can contain many **Datafiles**.

Users are associated with data at the **Investigation** level. Other aspects of the metadata are represented with their own entities and related to these core entities, such as at which **Instrument** an **Investigation** was carried out. It is also worth noting that these entities and their fields can be used to suit the needs of the **Facility** using ICAT. Many are optional which allows the schema to be applied flexibly.

In particular, **Types**, **Parameters** and **Technique** allow a highly configurable method for categorising data and capturing the experimental conditions. Each of the three main entities and the **Sample** is required to have a named **Type** (with the exception of the **Datafile** which has the optional **DatafileFormat**, intended to explicitly capture the file format). All four can possess any number of **Parameters**, which in turn must have a defined **ParameterType** (as shown in Fig 2). The former records a value, the latter what it is measuring

* andy.gotz@esrf.fr
[1] https://icatproject.org

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

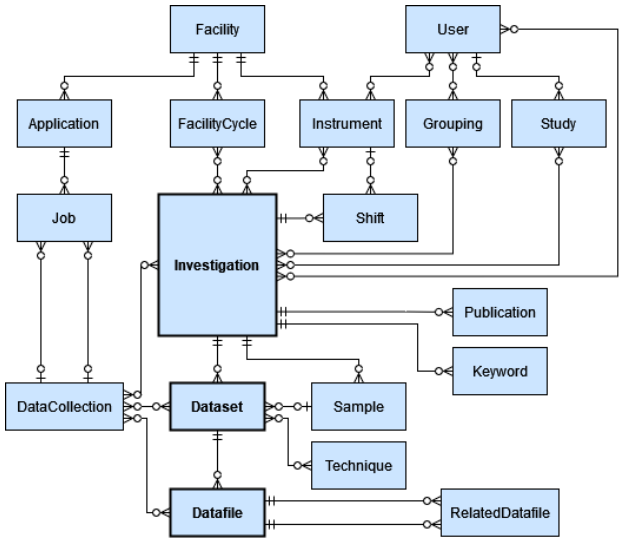

Figure 1: ICAT schema entities, based around the CSMD. Note that entity fields and relational entities for one:many relationships have not been included for simplicity, as have entities shown separately in figure.

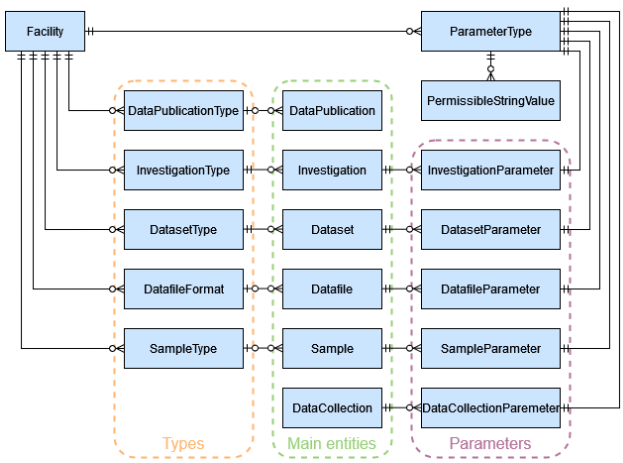

Figure 2: ICAT schema entities for defining parameters and types, across the data hierarchy.

- including units if relevant. **Parameters** can be a single numeric value, an upper/lower value specifying a range, a data/time or a string which can be controlled to only allow certain values. As noted before this approach allows **Facilities** to define the scientific metadata which is meaningful, without the ICAT schema having to accommodate all possible properties which may be of interest or relying on a generic representation that cannot easily be included in queries to the ICAT server. Thus, **Parameters** provide an extension point, allowing for any values, or key-value pairs, to be added to the entities and making the schema very flexible.

ICAT extends this with a further 12 entities, primarily centred around **DataPublications** which can be used to group any number and combination of the core entities for the purpose of assigning them to a single DOI. The entities and fields related to **DataPublications** are based on the DataCite Metadata Schema [5], where equivalent fields in ICAT did not already exist.

Core Services

The ICAT catalogue as a whole is comprised of a number of individual software components, as shown in Fig. 3. Some of these components are optional, providing additional functionality for ICAT (such as **icat.lucene**), and in other cases multiple options exist for a single aspect of the service (such as the DataGateway and DataHub front-ends). In this way, ICAT can be customised or extended to meet the needs of the facility running it. Brief descriptions of the core components and their roles in ICAT follow.

icat.server The majority of the business logic for the ICAT service is contained in this component [6]. It handles requests from front-end components either directly or via one of the other intermediary API layers (see sub-section *Extensions*) and interfaces with the underlying relational database in which the metadata is stored.

In addition to dealing with incoming queries, **icat.server** handles permissions for both the metadata and underlying data catalogued in ICAT. As can be seen from Fig. 3, even the IDS component (which does not need to query or return metadata, but rather the data itself) contacts ICAT server in order to perform authentication and authorization for download requests. While **Users** or a **Grouping** of users can be associated with an **Investigation** (see Fig. 1), by itself this does not provide access to data. Access is provided by creating a **Rule** which can provide access to:

- All users, including those accessing open data anonymously
- A specific **Grouping** of **Users**, such as admins of the instance
- **Users** that meet some arbitrary criteria, defined with JPQL (Java Persistence Query Language) which is evaluated dynamically when attempting to access the data

Each **Rule** can allow any combination of Create, Read, Update or Delete (CRUD) permission, and can apply to either all entities in a particular table, or only those entities that match the JPQL criteria described above. This approach allows for highly configurable permissions that are relevant to the facility running ICAT, without mandating that any particular approach be followed. It also means any new policies can be implemented immediately, as a new **Rule** can be created (or existing one modified) without requiring changes to source code or configuration files (as the **Rules** are stored in the ICAT database itself). Examples of common use cases are:

- Making high level metadata readable to all, such as **Facility** and **Instrument** names

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

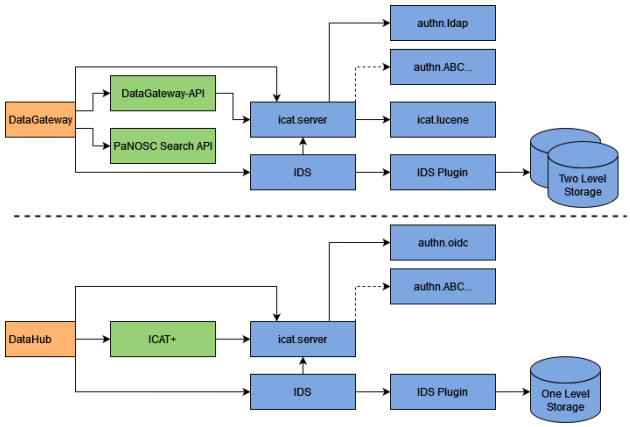

Figure 3: Two examples of possible ICAT instances. Different frontends (orange) and extensions (green) can be run against the backend of core ICAT components (blue). This can also be configured with optional components and run against different storage solutions. Arrows show the flow of control, not data.

- Giving users associated with the experiment access to their data via the **User <-> Investigation** or **User <-> Instrument <-> Investigation** relationships shown in Fig. 1
- Making data readable to all a set period of time after it has been collected (note that this does not require any action on the "release date" of the data, as this can be compared against the current date dynamically)
- Only making certain **Datasets** in an **Investigation** editable to the Principal Investigator based on fields such as the name, type or date

This final example is where the benefit of each facility being able to define its own **Rules** is most apparent - as their policy on who can edit what data will vary considerably, as will how information is recorded in the metadata (and therefore upon which fields the criteria will depend).

Authentication components In addition to the **Rule** system of ICAT making authorization highly configurable, multiple methods of authentication are supported as well. Login requests to icat.server should include the desired method in addition to the user's credentials, and the request is then handled by the corresponding component. There is a standard set of authenticators available including:

- an anonymous authenticator for users browsing open data, for example
- an LDAP authenticator for authenticating via accounts stored in an organsiation's LDAP or Active Directory server
- database and simple authenticators typically used for functional accounts used by data ingest tools etc.
- an OpenID Connect authenticator for verifying tokens issued by an external Identity Provider

This system can be (and has been) extended to create customised authenticators that implement the same endpoints and follow the same pattern.

icat.lucene In addition to JPQL style querying supported by **icat.server**, the optional **icat.lucene** component enables users to perform searches on the metadata entities using a single free text field via their frontend of choice. In the backend this uses the Apache Lucene library [7] to build and search inverted indices for **Investigations**, **Datasets** and **Datafiles**. This acts as a complementary tool to JPQL querying, which is suited to queries where the user has a good understanding of both the metadata schema and the specific values their (meta)data contains. In the frontend, browsing for data is often characterised by hierarchically "drilling down" (e.g. from **Instrument** to **Investigation** to **Dataset**) and filtering on specific fields of those entities. By contrast, searches with the **icat.lucene** component can be performed across one more indices in their entirety by specifying keywords - without needing to know where in the metadata they may be found or to which high level parent entities they "belong" as with hierarchical based browsing. This can aid data discovery, particularly for open data (when an understanding of the data's format cannot be assumed).

icat.oaipmh The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) [8] provides a mean for repositories to disseminate their metadata. External services, such as metadata aggregators like B2FIND and OpenAire may use it to harvest the metadata regularly, for instance, to provide some cross repository search services to their users.

icat.oaipmh provides an OAI-PMH endpoint to answer those harvesting requests. It connects to **icat.server** as a client in order to search the metadata and to compile the responses. There is deliberately no access control built into **icat.oaipmh**. It will disseminate all data it has access to. The access can be controlled with standard ICAT access rules instead, by making sure that the user that **icat.oaipmh** connects **icat.server** as, is only allowed to see what should be disseminated. For instance, most facilities would want to disseminate only public data via OAI-PMH. This can be achieved by configuring **icat.oaipmh** to connect to **icat.server** using the anonymous authenticator.

Frontends

ICAT's modular architecture supports having multiple frontends. There are currently two main frontends and one bespoke frontend implemented for ICAT.

DataGateway DataGateway is the current front-end for access to data stored in ICAT at both ISIS and Diamond. It is used both by internal facilities users to retrieve their own data and at ISIS it also provides open data for anyone to access. It provides the ability to browse through the ICAT

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

catalogue in hierarchies specific to the facilities, e.g. ISIS uses an **Instrument** -> **FacilityCycle** -> **Investigation** hierarchy, whereas Diamond focuses just on **Investigation** downwards. It can display entities in two formats, a tabular view and a "card" view more similar to a search engine results page, with options for the user to expand information of any specific item for more detailed metadata. Users can select **Investigations**, **Datasets** and **Datafiles** and these will be added to a "cart" - a selection of items which is stored on a server for the user to action on later. This cart can then be submitted to **ids.server**, where it will process the request and it allows the user to monitor the progress of the download through all the steps of the two-level storage retrieval process. Additionally, it also provides a free-text search interface which uses **icat.lucene** to help users discover data when they do not know much about the metadata.

Some upcoming features include:

- **Datafile** previewing in the browser for ISIS (since all their data is available in single-level storage)
- More detailed progress bars for data retrieval from Diamond
- Improved search interface
- Replacing existing service for ISIS DOI landing pages with DataGateway providing the landing pages
- Ability for users to generate DOIs for custom selections of data

DataHub It is the front-end used at the ESRF. Its development started in 2017. The motivation was to include the extended functionality that ICAT+ supports like the electronic logbook. Since then, it has been enhanced with features and tools needed for implementing FAIR data in the context of a large research facility.

Its main features are:

- **Electronic logbook** allows users and software to annotate the experiments. The electronic logbook is made available within the metadata and helps to understand the decision-making of the experiment.
- **Sample tracking** A considerable number of experiments are performed remotely nowadays. Users send the samples to the facility and this features allows to keep track of the parcels and the content of the parcels.
- **Custom display** It allows to change the metadata displayed depending on the technique of the dataset. By making custom views of the list of datasets it makes easier for users to understand and follow their experiment
- **H5Viewer** is integrated in DataHub and allows to browse the content of HDF5 files.
- **User management** DataHub allows to add collaborators to an investigation as well as to manage the permissions of the staff.
- **Statistics** allows to monitor the current status by providing statistics in real time of the ingestion of data, tracking of the parcels and the logbook.
- **DOI Minting** It allows participants of an experiment to mint a DOI by selecting one or multiple datasets.

Currently, a new release of DataHub is being developed with a module federation based architecture. Besides the current features supported by DataHub the new version will enhance the display of (meta)data with special emphasis in the processed data and will provide users with the possibility to reprocess data.

Human Organ Atlas A third example of a frontend is the Human Organ Atlas (see Fig. 4) which is a bespoke frontend based on the PaNOSC search API implemented on top of ICAT and a dedicated frontend for data of human organs curated in the ESRF ICAT instance. The web interface, based on react.js, demonstrates how it is possible to provide a tailored user experience for a specific scientific domain. The users see the data in ICAT through the bespoke frontend.

Extensions

The core of ICAT as described above is a mature well tested solution for curating metadata and data. The modularity of ICAT makes it possible to implement extensions on top which leverage the features of ICAT like the user access rights to data. This section describes the most evolved and used ICAT extensions currently available.

python-icat python-icat [9] is a client library package that provides a collection of modules for writing Python programs that access an ICAT service. It provides clients to talk to **icat.server** and **ids.server**. It defines native Python classes to represent the entity object types from the ICAT schema. The package includes a module to read configuration from various sources, such as command line arguments, environment variables, and configuration files. A query builder allows to build complex JPQL search expressions programmatically. Furthermore there are modules to export ICAT content to a flat file and to ingest it from a file into ICAT.

Some facilities use python-icat to implement their internal management workflows for ICAT, such as importing proposal metadata from the user office or ingesting data from the experiments.

DataGateway API & PaNOSC Search API The DataGateway API is a RESTful web service that was designed and developed using the lightweight Python framework, Flask RESTful [10]. It utilises the functionality provided by the Python ICAT library [9] to interact and retrieve data from

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

an ICAT metadata catalogue. DataGateway API serves two distinct use cases, each with its own set of endpoints and functionalities. The first use case serves the DataGateway user portal, where it provides endpoints essential for tasks such as data browsing, discovery, and retrieval.

The second use case focuses on the implementation of the Search API which was required to be developed and deployed by the ICAT facilities as part of the European Open Science Cloud Photon and Neutron Data Service (ExPaNDS) [11] and The Photon and Neutron Open Science Cloud (PaNOSC) [12] EU-funded projects. The Search API is a reduced functionality version of the DataGateway API in terms of endpoints, schema relationships, and query filters available. It adheres more closely with the Loopback REST query framework [13] and reuses DataGateway API code wherever possible. Notably, it eliminates the need for user authentication, allowing easy access to publicly available data stored in an ICAT metadata catalogue.

The DataGateway API is fully configurable, allowing users to configure and operate both use cases within a single API instance. Alternatively, it can be set up to run only one of these use cases, depending on specific requirements.

SCIENCE USE CASES

The ICAT generic data model (CSMD) supports modelling data from many different scientific techniques. These range from the generic case supporting curation of data with only basic metadata to highly specific scientific use cases with rich domain specific metadata.

Generic Data

The generic case stores only minimum metadata required by the CSMD which consists of date, authors, title of investigation, list of data files associated with the investigation. These metadata allow investigations to be searched for and the data for an investigation to be downloaded. However searching via technique specific metadata is not supported as the metadata are not stored in ICAT but instead in the data files. The generic approach allows sites to start storing data in ICAT with minimal effort but has its drawbacks for encouraging data reuse because they are difficult to find and search for, i.e. they are not FAIR.

Processed Data

ICAT was designed to follow the entire life cycle of data from the sample declaration to the publication of the results. One of the main steps to producing scientific results is processing raw data to produce results. ICAT can be used to store the processed results using enriched parameter metadata to link processed results to raw data.

FAIR Data

The ultimate goal of curating data and metadata are that they follow the FAIR guidelines i.e. that they are Findable, Accesible, Interoperable and Reusable. ICAT provides all the necessary support to make data FAIR. The Human Organ Atlas is an example of FAIR data managed in ICAT (see Fig. 4). A similar repository of FAIR data for paleontology is under development at the ESRF.

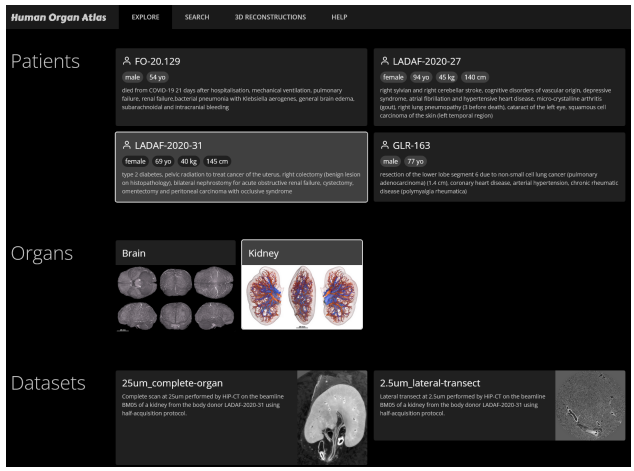

Figure 4: Example of FAIR data from the Human Organ Atlas.

ICAT SITES

ICAT was originally deployed at ISIS and DLS but has since been adopted by other sites. This section summarises the situation at the different sites using ICAT.

ISIS

ISIS Neutron and Muon Source facility is a world-leading centre for research in the physical and life sciences at the STFC Rutherford Appleton Laboratory near Oxford in the United Kingdom. It is host to a suite of neutron and muon instruments that gives unique insights into the properties of materials on the atomic scale. ISIS has been using ICAT since 2007 to catalogue raw data produced from all instruments and all data produced at ISIS since 1984, which comprises of over 150,000 experiments and 20,000,000 files, has been catalogued and is available online via DataGateway. According to the ISIS data policy [14], all data is automatically made open 3 years after the end of an experiment, and DOIs are automatically minted for open experiments via DataCite.

DLS

Diamond Light Source is the UK's national synchrotron. Over 14,000 researchers from across life and physical sciences both from academia and industry use Diamond to conduct experiments, assisted by approximately 700 staff. Diamond has been using ICAT to catalogue the experiment metadata, datasets and datafiles since around 2008 and now has data for over 50,000 experiments catalogued totalling nearly 5,000,000,000 files and over 50 petabytes of data. The files are stored on tape and users have the option to either download them, restore them to a Globus server ready for transfer, or have them restored to a data analysis cluster. Users can find their data via DataGateway.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

ESRF

The ESRF, the European Synchrotron, based in Grenoble, FRANCE, has been accepting users since 1992. In 2020 the ESRF underwent a major upgrade to become the first 4th generation storage ring providing hard xray synchrotron radiation to the photon user community. ESRF chose to use ICAT as their metadata catalogue in 2012 after comparing it with 14 other metadata catalogues [15]. ICAT was installed at ESRF in 2013 and went into production cataloging metadata and data in 2015. In the beginning only a few beamlines were connected to ICAT but since then all beamlines have been configured to catalogue metadata and data according to the ESRF data policy [16]. ESRF has extended ICAT significantly developing the DataHub web interface together with the ICAT+ backend as frontend to ICAT. In order to make data as FAIR as possible new components for electronic logbooks, DMPs, sample definition and tracking have been developed and integrated into the DataHub. ESRF ingests data automatically immediately after it has been acquired into ICAT via queues. The ESRF has so far catalogued more than 25,000 experiments, 2,000,000 datasets comprised of 665,000,000 files and a total volume of 12.4 PB. ESRF is the first ICAT repository to be certified by CoreTrustSeal, a self-certification process for scientific data repositories [17].

HZB

Helmholtz-Zentrum Berlin für Materialien und Energie (HZB) strives to achieve a climate neutral society through science and innovation. Scientists are developing and optimising efficient and cost-effective materials amongst others for solar cells, batteries and catalysts. Therefore, we run state-of-the art labs and BESSY II light source, which delivers intensely bright light, soft X-rays in particular. Researchers are using this special light to study the structure and function of energy and quantum materials. BESSY II has an annual average of 2700 user visits from 28 countries. HZB scientists are also conducting research on new concepts for accelerator-based light sources.

ICAT is deployed at HZB in a simple setup based on Docker containers. IDS is deployed as a two level storage. The storage is using a Hierarchical Storage Management (HSM) system that is composed of tape libraries and online disks for the archive storage. As a result, most of the data will reside on tapes, but the IDS plugin may still access the storage as a normal file system that happens to have a very high latency on read, though. Proposal data are automatically imported from the HZB user office portal GATE into ICAT as Investigations. The integration of BESSY instruments and the ingestion of data is still work in progress. Our goal is to create the data from the instruments using the NeXus format, whenever possible.

ALBA

ALBA is a 3rd generation Synchrotron Light facility located in Spain. ALBA received its first users in 2012 and serves over 2,000 scientists annually. ALBA has ten beamlines in operation, which will increase to fourteen in the following years. In parallel, the facility is starting to leap from the 3rd to the 4th generation by upgrading the accelerator, the existing instrumentation and adding a new and fully optimized beamlines portfolio, giving birth to the ALBA II. ALBA began cataloguing raw data using ICAT in 2020 with its catalogue based on the ESRF's implementation of DataHub and ICAT+. With a major overhaul of its catalogue beginning in 2023, effort has gone into the convergence towards the NeXus data format in the current beamlines and the migration of the whole internal architecture to a high-availability platform. ALBA's catalogue will open to the public by the end of 2023.

SESAME

SESAME (Synchrotron-light for Experimental Science and Applications in the Middle East) is a "third-generation" synchrotron light source that was officially opened in Allan (Jordan) on 16 May 2017. It is the first synchrotron light source in the Middle East and neighbouring countries, and also the region's first major international center of excellence. In June of 2020, the SESAME Council adopted its "Experimental Data Management Policy" [18], and this was harmonized following the community best practices and standardization. Accordingly on 2022, the Scientific computing team made a community-driven decision and started the implementation of ICAT as a metadata catalogue solution, and to handle the required integrations with environment subsystems on the research infrastructure, data acquisition system, User office and proposal submission systems. Currently, ICAT is in the development and validation phase, with plans to transition into production mode by 2024. ICAT4.8 is built on the Rocky Linux 8.4 operating system and utilizes a technology stack that includes MariaDB, Glassfish 4.1 IDS, IDS plugins, Java 8, ActiveMQ, MongoDB, as well as Python 2.7 and Python 3.8. Metadata for users and proposals are sourced from the SESAME User Portal (SUP) and systematically imported into the designated ICAT database using a combination of Python and ActiveMQ. The structure and delivery of SESAME experimental data (SED) formats are managed by data acquisition tools to ensure standardization across the produced data. These formats can vary and encompass options such as HDF5, DX for BEATS beamline, and XDI for HESEB, MS, and XAFS beamlines. The SESAME main storage architecture comprises two distinct levels: short-term storage (STS) and long-term storage (LTS). Both LTS and STS are managed and maintained by the ICAT services, enabling users to access and download their SESAME experimental data (SED) conveniently. This integration ensures that users can retrieve their data from both short-term and long-term storage seamlessly through the ICAT services provided by SESAME.

SIRIUS

Sirius is a fourth-generation synchrotron light source designed to offer to the scientific community a wide range of X-ray experimental techniques. Currently, Sirius is in

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

its first phase of operation with 14 experimental stations, of which six are already fully operational and work with different experiment control systems, five are in scientific commissioning, and two are under construction. The Sirius is maintained by The Brazilian Synchrotron Light Laboratory (LNLS), which is part of the Brazilian Center for Research in Energy and Materials (CNPEM), in Brazil, a private non-profit organization under the supervision of the Brazilian Ministry of Science, Technology, and Innovations (MCTI). Sirius has recently started the evaluation of ICAT as a standard solution for cataloging the data and metadata produced during the experiments by taking the current ESRF's implementation of DataHub web interface alongside with the icatplus. This initiative is part of a major effort at the LNLS for fully automated managing data and metadata following the FAIR data principles at the Sirius. The main reasons that make ICAT attractive to the LNLS-Sirius are: (i) the stability and maturity of the project since this solution is operating in production mode for a significant amount of time in several large-scale facilities; (ii) the extensions available for the core solution, which are essential for gathering a plenty of metadata associated to the sample and experiments; (iii) the license of the core software and its components as an open source software; and (iv) the support of an active software development community aligned with the nowadays synchrotron needs. All these reasons makes the ICAT a suitable solution for efficiently managing scientific data and metadata of large-scale facilities, following FAIR principles, and data governance took place at the Sirius.

COLLABORATION

The ICAT collaboration has been active for almost 20 years now. The original members were STFC, ISIS and DLS. Today the collaboration has been extended to include ESRF, HZB, ALBA, SESAME and recently SIRIUS is evaluating ICAT. The way the current collaboration is organised follows the best practices of open source software which include: (1) all code in the core components of ICAT is on GitHub in a single project `https://github.com/icatproject` under an open source licence and accessible by anyone (2) site specific developments are in local institute repositories accessible by all (3) remote meetings are held monthly to discuss progress and site deployment issues (4) face-to-face workshops are held on an annual basis (roughly) (5) meetings on special topic are held when required (6) on-site experience sharing and help are provided on request (7) a website at `https://icatproject.org` (8) a mailing list on google groups (9) a Slack workspace `icatcollaboration.slack.com`. The collaboration is essential to sustaining ICAT and making it easy to use.

NEXT STEPS

There are ongoing improvements and new developments being made in the various components that make up the ICAT service. Major changes to the icat.lucene component are scheduled for release, both to improve performance for facilities with large amounts of data (several billion **Datafiles**) and add new features, including:

- Improving search syntax to support targeting of specific metadata fields and increasing the number of searchable fields
- Providing sorting on specific metadata fields alongside the default (relevancy to search term)
- Synonym injection to ensure searches using abbreviations of scientific terms and techniques return the same results as non-abbreviated searches
- Unit conversion for numeric parameters
- Ability to perform facet filtering of results on specific categorical or numeric fields, such as:
  - Investigation/DatasetTypes
  - Sample names
  - Parameters (types and values)

An extension to leverage the **DataPublication** entity is under development: an API that allows Principal Investigators to mint DOIs for their data in ICAT, and support the automatic minting of high level metadata by facilities without user input. In addition to communicating with other ICAT components, it will send requests to the DataCite API [5] to mint the DOIs with all recommended metadata provided. Frontend integration is planned for DataGateway, and allow access to the data in question via the ICAT **Rule** system.

CONCLUSION

The flexible CSMD data model as well as the maturity and stability of ICAT for managing large data makes it an attractive solution for data repositories. The core team maintaining ICAT keep updating it to support the latest version of the runtime platform. The number of contributions to ICAT has grown over the years e.g. python-icat, oai-pmh, as well as the extensions e.g. icatplus, e-logbook, HDF5 viewer, sample tracking etc. The number of ICAT sites is growing and there is an vibrant community of users and developers who meet monthly to improve and extend ICAT. The scientific use cases for ICAT have grown recently and it is now being used to serve raw, processed and FAIR data for a variety of experimental techniques. ICAT is being used to successfully manage the research data of a number of large and small facilities - it just works!

ACKNOWLEDGEMENTS

Part of the work described in this paper has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreements No 823852 (PaNOSC), No 870313 (STREAMLINE), and No 857641 (ExPaNDS).

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE3BC007

REFERENCES

[1] John Helliwell *et al.*, "The science is in the data", *Int. Union of Crystallography J.*, vol. 4, no. 6, pp. 714-722, Oct. 2017. doi:10.1107/S2052252517013690

[2] ICAT 6.0.0 Schema, https://repo.icatproject.org/site/icat/server/6.0.0/schema.html

[3] B Matthews *et al.*, "Using a Core Scientific Metadata Model in Large-Scale Facilities", *Int. J. Digital Curation*, vol. 5, no 1, Jul. 2010. doi:10.2218/ijdc.v5i1.146

[4] Erica Yang, Brian Matthews, Michael Wilson, "Enhancing the core scientific metadata model to incorporate derived data", *Future Generation Computer Systems*, vol. 29, no. 2, pp. 612–623, Feb. 2013. doi:10.1016/j.future.2011.08.003

[5] DataCite Metadata Working Group. "DataCite Metadata Schema Documentation for the Publication and Citation of Research Data and Other Research Outputs", version 4.4, 2021. doi:10.14454/3w3z-sa82

[6] ICAT Server 6.0.0, https://repo.icatproject.org/site/icat/server/6.0.0/

[7] Apache Lucene, https://lucene.apache.org/

[8] Carl Lagoze, Herbert Van de Sompel, Michael Nelson, Simeon Warner, "The Open Archives Initiative Protocol for Metadata Harvesting", version 2.0, 2002, https://www.openarchives.org/OAI/openarchivesprotocol.html

[9] Rolf Krahl, The ICAT project, "python-icat – Python interface to ICAT and IDS", version 1.1.0, 2023. doi:10.5281/zenodo.8099311

[10] Flask-RESTful, https://flask-restful.readthedocs.io/en/latest/

[11] ExPaNDS, https://www.panosc.eu/expands-project/

[12] PaNOSC, https://www.panosc.eu/panosc-project/

[13] LoopBack, https://loopback.io/doc/en/lb3/

[14] ISIS Data Policy, https://www.isis.stfc.ac.uk/Pages/Data-Policy.aspx

[15] Nicolas Bessone *et al.*, "MS6 – Proposed Metadata Catalogue Architecture Document", CRISP FP7 Project Deliverable, 2012, https://www.docslides.com/sistertive/crisp-wp-17-1-2-proposed-metadata-catalogue/-architecture-document

[16] ESRF Data Policy, https://www.esrf.fr/datapolicy

[17] CoreTrustSeal, https://www.coretrustseal.org/

[18] SESAME Experimental Data Management Policy, https://www.sesame.org.jo/for-users/user-guide/sesame-experimental-data-management-policy