Title: ICAT Interfaces Overview

Author: Shoaib Sufi

Company: Devigo

Version: 1.0

Related Task Id: 26

Date: 15/08/2008

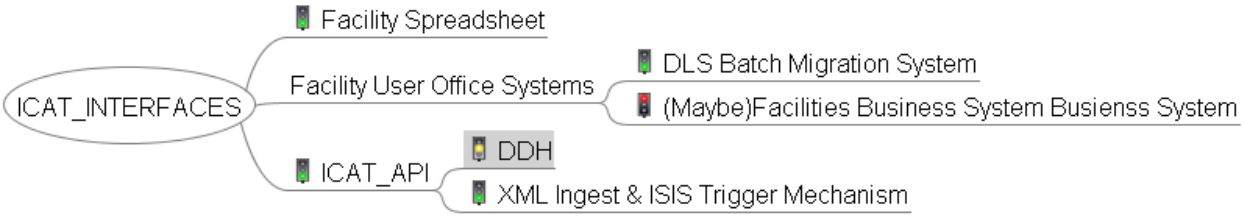
1 Introduction

This is an overview document giving a short introduction to the ICAT interfaces.

The ICAT interfaces are a process or subsystem which is used to write data into the ICAT, this covers systems which read and write data into the ICAT. Systems which only read data from the ICAT are not termed interfaces in the proper sense.

By ICAT system it is meant the ICAT schema, supporting database procedures, triggers that maintain the ICAT data model and the ICAT API.

2 The Interfaces



2.1 Facility User Office System

These designate the system which feed in information from the Facilities User office system.

The User office system takes contains cleaned up data from the proposal system relating to proposals and other experiments which are actually scheduled to run at the facility.

The User office of the facility cleans up the data and adds vital information such as the instrument/beamline that the experiment is actually scheduled to take place on and who actually will do the experiment which is often radically different then who proposed to do the experiment.

2.1.1 Diamond User Office System DUO Desk

DUO Desk[1] is the name of the User Office system used at Diamond.

It contains a vast amount of information pertaining to the users coming to do experiments at Diamond. Some of the information is relevant for the dual archival and operations role that ICAT plays at DLS such as the beamline being used, who is on the experimental team and when the experiment is scheduled to run at a particular instrument amongst many other things. Other data in the DUO Desk such as who has spent the most money pertaining to their allowances during their visit to Diamond is of less use for the archival purposes of experiments and data collected at Diamond and is therefore not a candidate for data which should be stored in ICAT.

The ICAT 3.3 specification for information that should be copied from DUO Desk to ICAT was outlined in the deliverable to task 18[2].[3]

1 For further information on Duo Desk please contact Bill Pulford of Diamond – b.pulford@diamond.ac.uk
2 https://esc-cvs.dl.ac.uk/svn/dl/metadata/icat/branches/3.3.1/documentation/interface-overview/duodesk-to-icat3.3-link.pdf
3 For information on the implementation please contact Keir Hawker of e-Science – k.hawker@rl.ac.uk

2.1.2 Facilities Business System – Business System (FBU-BS)

This is the STFC User Office system Interface. The STFC User office covers the ISIS[4] and CLF[5] facilities.

There is a 'push interface' specified and agreed by the person in charge of the User Office (at the time of writing this is Richard Browning). However this was specified before the December the 14th announcement at Daresbury Lab and the ensuing premature end of the e-Science Facilities programme. It was also not covered in the Devigo contract. However it would be a very useful system to put in place to bring ISIS and potentially CLF (if they choose to use ICAT) in line with with the clean data feed as used at Diamond.

The 'push interface' specifies a minimal set of useful data which the User Office system would periodically push into a database schema about scheduled experiments and related information.

The most up to date specification of the interface is available at https://esc-cvs.dl.ac.uk/svn/dl/metadata/icat/branches/3.3.1/jdeveloper/fbu-bs/specification/FBU_Business_system_and_ICAT_Interface_Document.doc. It is a recommended read, the introductory information is particularly useful giving context to why the work is necessary.

4 http://www.isis.rl.ac.uk/userOffice/

5 http://www.clf.rl.ac.uk/reports/userguide/index.htm#contact

2.2 ICAT API

ICAT API version 3.3 is the Grid aware software infrastructure that enables applications to exploit the capabilities of the ICAT system based on the requirements of the STFC facilities, with potential wider applicability to other research groups and institutions.

It is a web services API built using the Java2 Enterprise Edition software stack. Allowing any software platform that can talk web services (e.g. Perl, Python, C#, Java) to interact with the ICAT schema. The API Shields the user from amongst other things the internal representations pertaining to audit, the rules of authorisation and ability to maintain data integrity with multiple users registering metadata into the ICAT system.

Software application which interact with the ICAT schema should go via the ICAT API.

You can read more on the ICAT API and it's design and purpose with the example of an application build upon it (DataPortal) by reading the following design document https://esc-cvs.dl.ac.uk/svn/dl/dataportal/modules/tags/v3.3/documentation/dataportal-icatapi-design.pdf. [6]

2.2.1 The Diamond Data Handler (DDH)

Is the metadata registration system in use at Diamond.

It interacts directly with the ICAT API registering files with the Diamond ICAT for files that are newly created at the Beamline machines of Diamond prior to their ingestion into storage system at Diamond.[7]

2.2.2 ISIS Trigger Mechanism

Is the metadata registration system in use at ISIS.

It takes a different approach to DDH and was developed independently. It was developed with the special need to process a massive amount of ISIS's back catalog files as well as new experiments. It reads files from ISIS's data archive and then puts together an XML document representing all the metadata for a particular investigation including users, datasets, datafiles, parameters and keywords. A service called XML Ingest written by ISIS is then called, this invokes methods in ICAT API to register files and their associated context.[8]

Aside: The accuracy of cataloging was vastly improved by enforcing the input of a valid experiment number in the ISIS data acquisition system SECI[9], this data was then written to the data file and used by the registration process for correct cataloging.

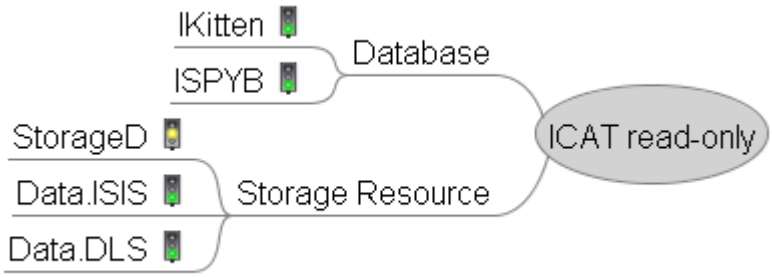
6 For further information on ICAT and the ICAT API please contact Damian Flannery – d.w.flannery@rl.ac.uk or for consultancy please contact www.metataxa.com

7 For further information please contact Tobias Richter of Diamond – tobias.richter@diamond.ac.uk or Alun Ashton of Diamond - *alun.ashton*@diamond.ac.uk

8 For further information please contact Tom Griffin of ISIS – t.a.n.griffin@rl.ac.uk

9 For further information on this change please contact Matt Clarke of ISIS - m.j.clarke@rl.ac.uk

3 Aside: Read Only interaction with ICAT



These are system which only read data from ICAT. These are not interfaces in the way defined in the introduction but serve as useful examples of how ICAT is used as a data source.

3.1 Database

3.1.1 IKitten

IKittens are versions of ICAT which only have the information about investigations pertaining to a particular instrument. They do not contain information about registered datafiles.

They are a way of allowing programmatic access at the beamline machines to have access to information from ICAT while being subject to less network issues.

This has allowed the data acquisition system at the beamline (GDA) to move beyond just authenticating users but also to check which scheduled experiment a user is authorised to be a part of. Further this information is then written into data files collected at the beamline such that they can be perfectly correlated and cataloged along side their parent ICAT record.[10]

3.1.2 ISPyB

The ISPyB (Information System for Protein CrystallographY Beamlines) project is a joint development between ESRF/spine and BM14/MRC/eHTPX. The aim is to build a LIMS dedicated to protein crystallography experiments on synchrotron beamlines[11].

Rather then go directly to the DUO Desk business system at Diamond. It uses the minimal clean set of data in ICAT as a feed for the visit number, proposal information and information about experimenters.[12]

10 For further information about this work please contact Steve Kinder of SR – s.h.kinder@dl.ac.uk or Karen Ackroyd or SR – k.s.ackroyd@dl.ac.uk or Alun Ashton of Diamond – alun.ashton@diamond.ac.uk

11 http://www.esrf.eu/UsersAndScience/Experiments/MX/Software/ispyb/

12 For further information please contact Karl Levik of Diamond – karl.levik@diamond.ac.uk or Alun Ashton of Diamond - alun.ashton@diamond.ac.uk

3.2 Storage Resource

Data is accessible via the ICAT API using a HTTP interface to the facility's data archives.

The advantage of this approach is that the user/application is shielded from the complexities of the underlying data archive system and only has to use the URLs provided by the ICAT API to download the data. This can then be used programmatically or through a browser to download and stream the data back to the user/application.

3.2.1 Data.ISIS

ISIS has a HTTP server installed in front of their Windows DFS storage system. This takes a request in the form of a HTTP GET for datafiles or datasets. The HTTP interface then check using the ICAT API that the user is logged in and has authorisation to download the files before providing a link from which the user can then (using HTTP) download the compressed data archive of files and datasets requested.[13]

3.2.2 Data.DLS

Diamond also have a HTTP server installed infront of their Storage Resource Broker (SRB) system. This takes a request in the form of a HTTP GET for datafiles or datasets. The HTTP interface then check using the ICAT API that the user is logged in and has authorisation to download the files before providing a link from which the user can then (using HTTP) download the compressed data archive of files and datasets requested[14].

3.2.3 StorageD

Is the process which ingests data from the Diamond beamlines into the Storage Resource Broker. In the version 3.1 ICAT it used a view in ICAT called V_FEDERAL_INVESTIGATION to find which people were associated with which experiment and set appropriate authorisation in the storage system allowing the storage system to be ready to ingest data once it was collected.

13 For further information please contact Freddie Ackeroyd of ISIS – f.a.akeroyd@rl.ac.uk
14 For further information please contact Roger Downing of e-Science – r.downing@dl.ac.uk