ICAT

# data discovery and interoperability

Steve Fisher <dr.s.m.fisher@gmail.com>

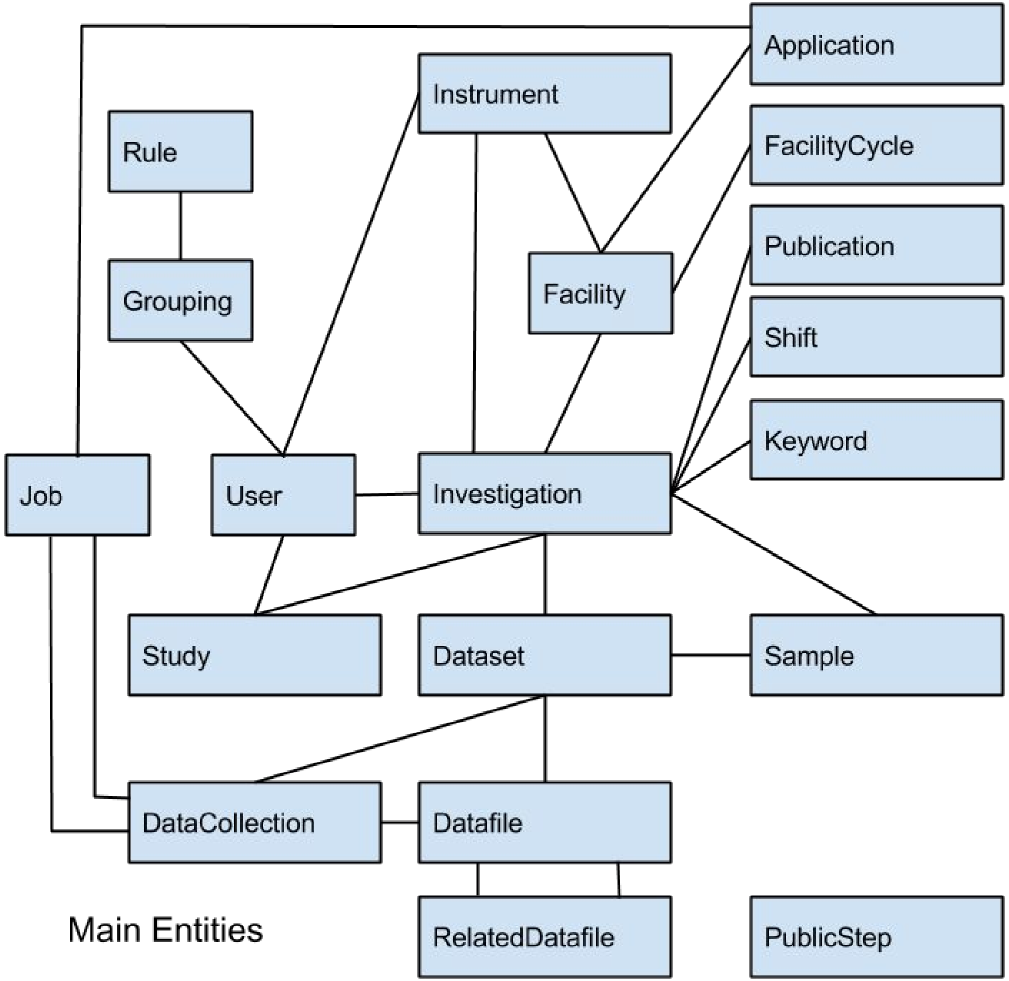
Please do interrupt with any questions/thoughts

# Overview of talk

- Summary of the ICAT Family
- Cover the seven points in the google doc (though not entirely in order):
    1. Data discovery
    2. Authn (and authz)
    3. Server queries
    4. Transfer protocols
    5. APIs
    6. Reproducibility
    7. Controlled vocabularies

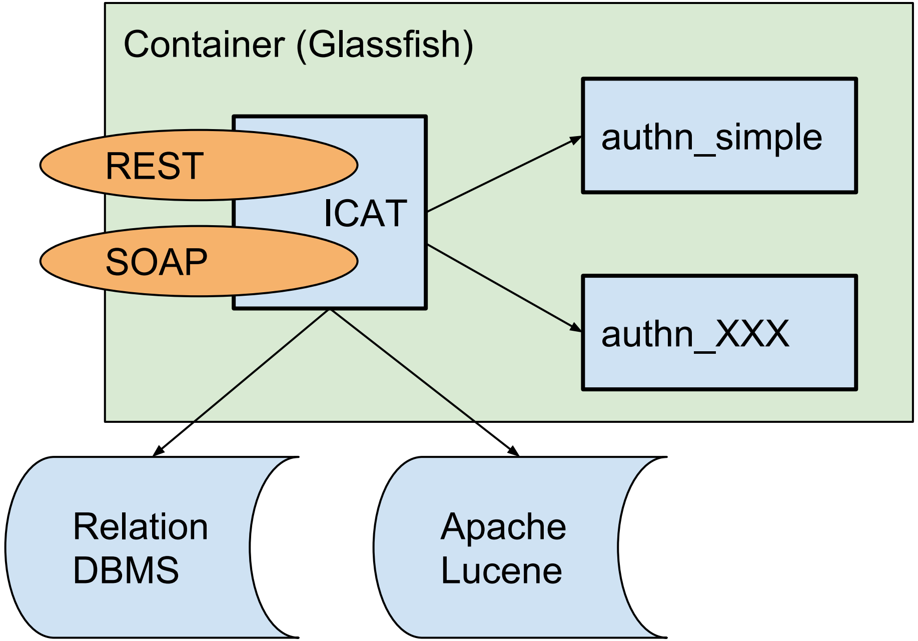
# The model

- Diagram only shows high level view.
- The actual model not relevant to talk.



# ICAT Server

- Java EE application inside container
- REST and SOAP interfaces
- Pluggable authenticators
- RDBMS and Lucene
- Rule based authorization



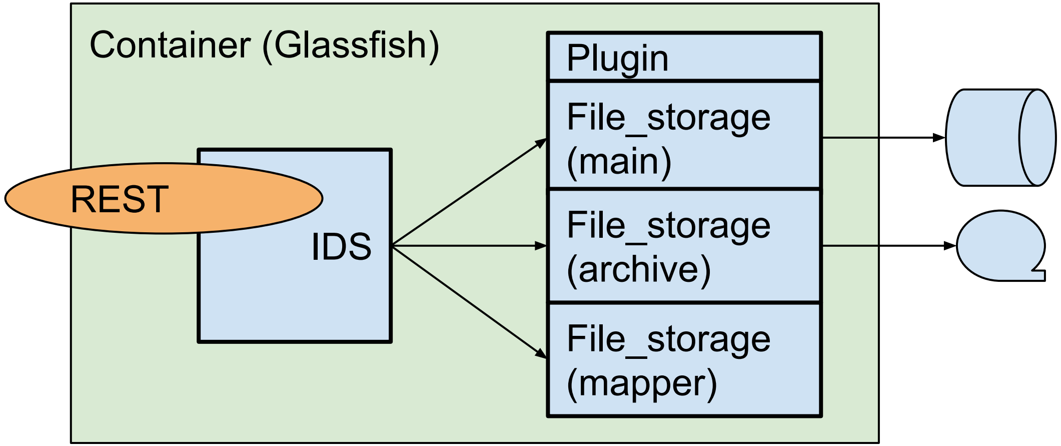
- Generic calls to:
  - Write
  - Update
  - Search
  - Delete

# Rule Based Authorization

- Rules to implement a policy
- Such as:
    - All data is public after n days
    - All investigation records are public
    - Those users related to an investigation can read all Datasets and their related Datafiles and Parameters.
- JPQL SELECT statements define a View.
- Can also define permissions for those in a “group”

# IDS Server

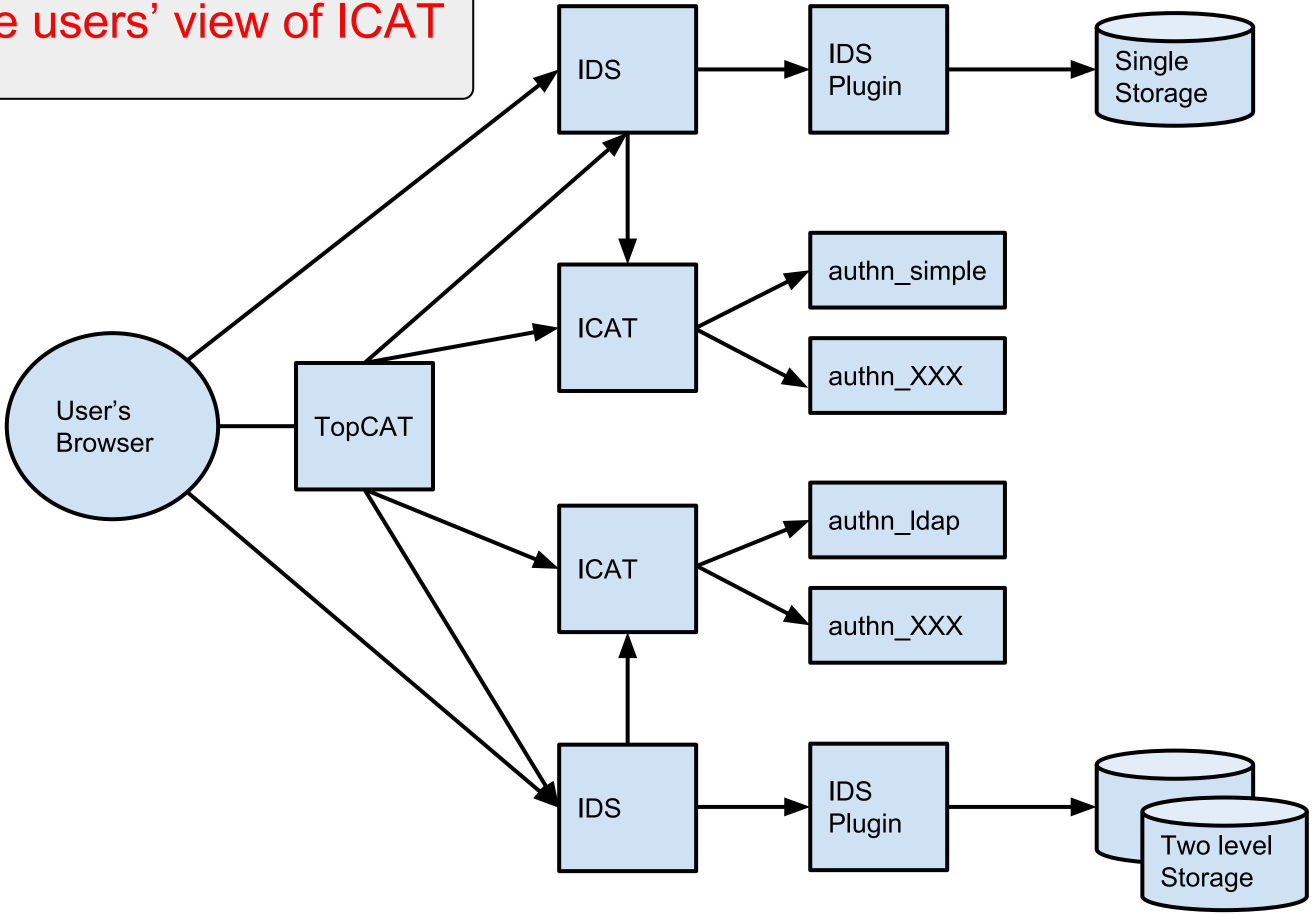
- Conceptually simple
  - Store a file and catalogue it
  - Retrieve a file or files
- Multiple download mechanisms:
  - http
  - globus (gridFTP based)
  - smart client



# TopCAT

The users' view of ICAT

- Interface to multiple ICAT and IDS servers
- Highly configurable
- Facility dependent view
- Makes use of lucene search
- Download mechanisms:
    - http(s)
    - smartclient
    - various PollCATs
- Pluggable

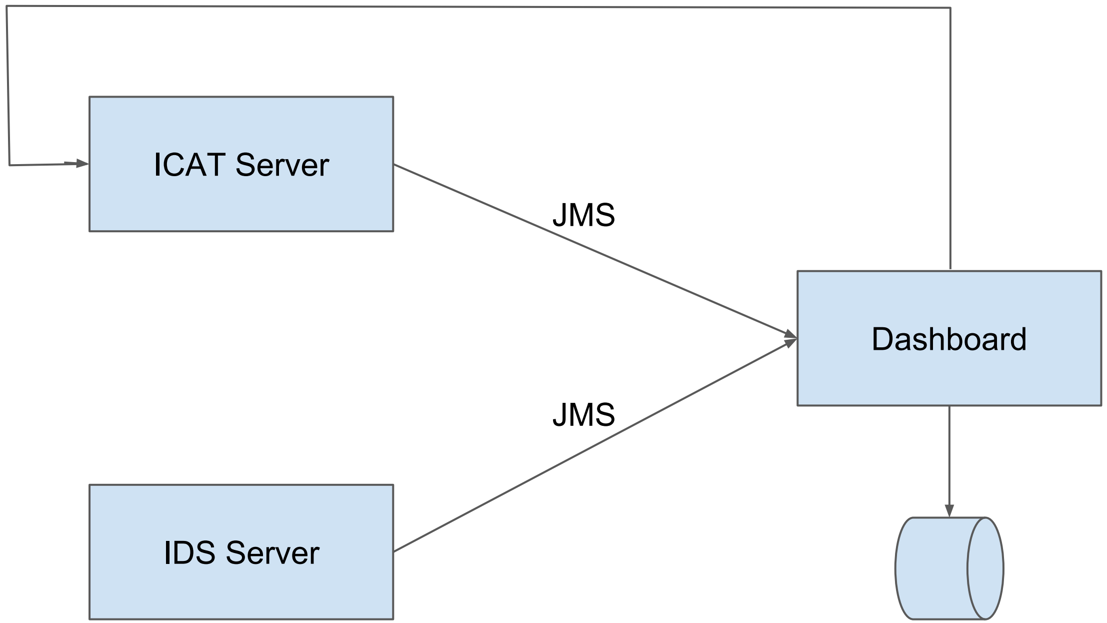


# IJP

- ICAT Job Portal
- Was a standalone component
- Now implemented as a TopCAT plugin so feels like TopCAT

# Dashboard

- Web based GUI (Angular JS) to give overview of ICAT Usage



# 1 - Data Discovery

- An OAI-PMH component is being built for ICAT as a front-end.
- TopCAT is able to search across multiples ICATs - but the facility has never been used to my knowledge
    - Not really wanted?
    - Doesn’t work?
- The lack of a controlled vocabulary is a problem.
    - Even comparing 0.5V and 600mV is inefficient making it easy to miss things.
    - DLS now have more than $10^8$ data files so inefficient solutions are not an option.
    - The lucene index linked to ICAT makes text searches work well but not numeric parameters.

# Digression on OAI-PMH

Wikipedia says:

> *The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) is a protocol developed for harvesting (or collecting) metadata descriptions of records in an archive so that services can be built using metadata from many archives.*

- The interface is not suitable for discovery but only for bulk downloads.
- No support for data - only the metadata

# 2a - Authentication

- ICAT has no built in authentication but uses one or more authentication plugins:
  - LDAP
  - Database
  - “Simple”
  - Anon
  - An umbrella plugin also exists but I don’t think it is used much
  - Some facilities write their own
- You are then identified as something like:
  - ldap/smf98
  - anon/anon

# 2b - Authorization

ICAT has rule based authentication for any object in the database

Rules allow a data policy to be defined - for example:

- All data is public after 180 days
- All investigation records are public (good for discovery - bad for secrecy)
- Those users related to an investigation can read all Datasets and their related Datafiles and Parameters.

IDS (Icat Data Server) applies the same authz rule to the data file as to the Datafile metadata object in ICAT.

# 3 - Server Queries

icat.server provides SOAP and Restful operations where actual queries are in JPQL (almost all JPQL is allowed and there are some extensions)

Icat.server provides Restful search calls making use of the lucene index. For example finding a Datafile based on the occurrence of a word in the datafile description needs a text database. Can search for Investigations, Datasets and Datafiles.

Ids.server provides Restful interface to:

- store (and catalogue) datafiles
- download arbitrary collections of Investigations, Dataset and Datafiles
- manage store data

# 4 - Transfer protocols

The IDS currently only supports http(s) for data transfer but

TopCAT can make use of direct http(s), SmartClient or pollcat to deliver the data by various plugins:

- Globus (GridFTP)
- Local HPC facility

# 5 - APIs

All components have APIs

We try to avoid introducing incompatibilities between versions

The ICAT changes very infrequently however the model does evolve so a query may stop working.

# 6 - Reproducibility

ICAT has entities to describe Jobs that have been run on data along with the input and output data collections. This in principle provides what is need for reproducibility and is exploited by the IJP.

# 7 - Controlled vocabularies

Users don’t like being controlled so in ICAT some facilities have a huge number of parameter types - making them effectively unsearchable.

I think that facilities should take this very seriously when storing metadata and choose a minimal set of meaningful parameter types.