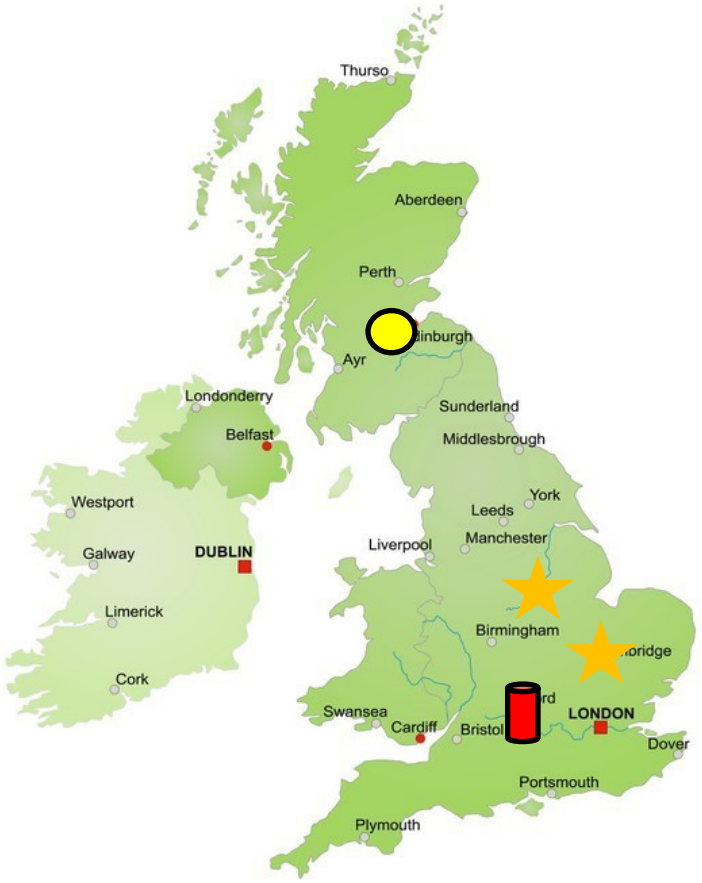
DiRAC-3 Data Curation Service: Rucio Evaluation

Alastair Dewhurst

Rucio setup

- Rucio could be configured to manage several Storage Elements
  - ★ DIRAC Sites
  - ▮ Data Archive
  - ● External User Site
- Each site needs to provide an externally facing transfer service.
  - GriFTP, WebDav, etc.



Types of data

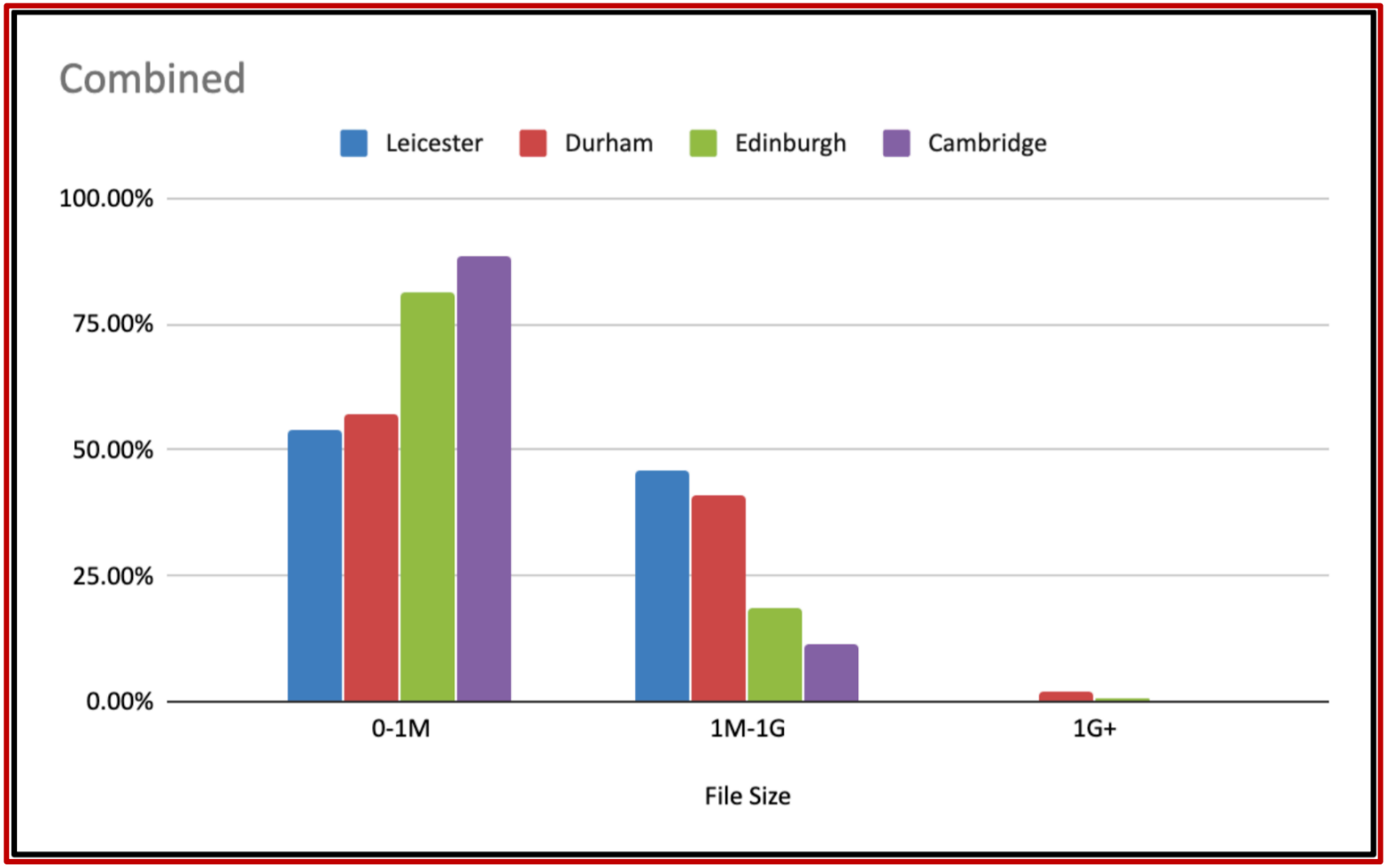
1. Types of Data

Broadly speaking, DiRAC simulations produce four "levels" of output data:

- **Level-4: Raw data.** These are the raw data produced by the simulations performed on the DiRAC systems.
- **Level-3: Reconstructed data**. These data are derived from the raw data through analytics and post-processing activities.
- **Level-2: Data to be used for outreach and education**.
- **Level-1: Published results**. These are the final results of the research and are generally published in journals and conference proceedings.

- The rule based approach to data management allows for easy management of different types of data.
  - E.g. For Level-4, two copies of the data, one of which must be on tape.

File Size


Combined
Leicester
Durham
Edinburgh
Cambridge
100.00%
75.00%
50.00%
25.00%
0.00%
0-1M
1M-1G
1G+
File Size

File size (2)

- DIRAC has a lot of small files.
  - At 100PB level I would estimate ~10 Billion files.
- The Rucio database would need to have 10 Billion entries, which is possible but requires significant optimization.
- Storage endpoints and data transfer services will struggle to efficiently cope with that many files.
- Rucio is not designed as either a backup service or software management system.

Managing small files

- Containers and tarfiles which can be managed by Rucio are ways of packaging up small files.
  - At RAL DLS use a service called iCAT, which tars up their small files before writing them to tape.
- Services like CVMFS can manage and distribute software which has vast numbers of small files that are similar.
- There are many backup tools, that will create a few large files that could then be managed by Rucio.
- Really small files could be stored as meta data in a database?

Software and Metadata

3. Software and Metadata

In early 2016, the DiRAC facility decided that all data should be tagged with metadata that conforms to the appropriate standards of the relevant research communities. However, implementation of this policy was subject to DiRAC-3 funding.

As part of the DiRAC-3 upgrade all data products requiring long-term preservation will be labelled using standard metadata containing the necessary minimum of the following:

Project Code; Project PI; Date of Production; Name of DiRAC System; Physical Location of compressed Data OR the source code, inputs and/or a VM image of the code and inputs and runtime environment; Research Area; Associated Subject metadata that describes each data format.

The Facility will have to ensure, subject to funds awarded for data management purposes, that the infrastructure exists to tag data with these metadata. These will be in the form of appropriate APIs and appropriate software to write and read metadata.

- Rucio automatically stores metadata about the date the data was produced and its location.
- Rucio can store arbitrary metadata about the datasets.
- Rucio can group data.
  - E.g. This data goes with these log files.

Metadata

Metadata Catalogue and Data Publishing: An appropriate metadata catalogue will be deployed and installed, if funds allow. In the DiRAC-3 context, the Data Curation Service must include

- hosting services for those projects that wish to publish their metadata and data products.
- An Analytics and Visualisation environment in which analytics can be run on stored data and new data generated and labelled with updated metadata.

- These are outside the scope of "Core" Rucio.
- ATLAS offer similar services to those requested.
  - They pull data from Rucio but are standalone.

Software and APIs

Software and APIs

The Software and API environment needed to manage the data will include:

- Suitable database technologies to hold catalogue data;
- Suitable APIs to allow the creation of metadata and its interrogation;
- Suitable backup and restore software.

- Rucio can use industry standard relational databases.
  - RAL uses PostgreSQL.
- Rucio provides a CLI, API and Restful interfaces.
- All data necessary to restore the service is held in the database.
  - This is backed up at RAL (see backup).

Summary of Requirements

Data volume: The current volume of data with long-term value across DiRAC sites is estimated to be about 9PB, with an estimated growth rate of about 8PB per year, increasing to 16PB per year once DiRAC-3 comes online. The initial scale of the Data Curation Service storage requirement is therefore 100PB, to allow for 5 years of data ingest.

- For ATLAS:
  - Rucio is currently managing over 200PB of data on Disk and 400PB on Tape.
  - Annual growth rate of 46PB.

Data formats: By data volume, the primary data format is binary, with HDF5 being most common (typically compressed). Other formats, e.g. NetCDF, VTK, FITS and custom formats are also common. Large numbers of small files are text (ascii or UTF-8), which includes XML documents.

- There are no restriction on the kind of data formats Rucio can manage.

Summary of Requirements

Data users: The data that are expected to have value will almost exclusively be used by researchers, with only a small fraction being used by the general public. However, many of these researchers will be outside the scope of DiRAC, and will not have access to DiRAC facilities, so the ability to transfer large amounts of data off the system will be necessary.

- Rucio (e.g. Robot x.509 certificates) will need to have read/write access to the relevant DiRAC storage endpoints.
- Rucio can set ACL to restrict access where necessary.
- Users could:
  - Download files to their laptop using rucio-get.
  - Request a bulk transfer.
- Rucio wouldn't be used by the general public to access data.
  - Rucio would manage the data into a space that was accessible to the general public.

Summary of Requirements

Access methods: The preferred method for accessing and transferring data to the facility is via the command line, using ssh (including scp, sftp and rsync). A web interface for browsing and downloading data is also desirable, and a small number of projects would like database access, or GlobusOnline. Background tasks are desirable when transferring data between facilities, i.e. once a task has been initiated, the user no longer needs to monitor.

- Users would be strongly encouraged to use the Rucio CLI such as rucio-get, rucio-upload.
  - Users are not prevented from using other commands to access data.
- A web interface for browsing and downloading data could easily be made.
- I am not sure what the requirements "database access" and "Globus Online" mean.
- Rucio manages vast numbers of transfers between sites, and automatically handles common failure modes.

Data Preservation

4. How long does data need to be preserved?

There is currently no upper time limit foreseen for the retention of DiRAC data. OpenData requirements from UKRI are likely to mandate a minimum retention period of 10 years from data creation. Results from the DiRAC PI survey indicate that 25% of DiRAC projects already require retention periods greater than 10 years, and 9% of projects require data to be retained for at least 15 years.

- ATLAS will need data management for the lifetime of the LHC.
  - This is currently planned out until at least 2036.
- Rucio already manages over half an Exabyte of data.
  - Any change in APIs etc. will not be made lightly!

FAIR

5. **Which data will have value to others and should be shared?**

A significant fraction of current DiRAC raw data cannot be interpreted by third parties without them having a very detailed knowledge of the experimental reconstruction software. Derived data may be more easily usable by third parties, and provision is currently made to make this available upon request on a case-by-case basis. In the near future, UKRI is expected to mandate adherence to FAIR[1] data principles and the Data Curation Service should therefore reflect these requirements.

- Rucio can facilitate adherence to FAIR data principles.
- It can manage data that is either publicly accessible or accessible to external researchers.
- You will still need a system to process these requests and decide what data to be made available.

Data Sharing

6. How will data be shared?

Data is currently shared by the individual projects. In some cases, individual projects and systems have also taken the initiative to develop or engage with value added open data services using resources leveraged from STFC and other Facilities. Examples of this are Virgo and PLANCK, and we expect similar leveraging of resources from projects/missions such as LSST, SKA, LIGO, CTA, JUNO, JUICE, GAIA and Euclid.

However, as part of the DiRAC-3 upgrade the DiRAC Facility will also offer data sharing facilities to all of its users through the Data Curation Service.

Data will be made available in an externally intelligible format as specified by the DiRAC Project using a suitable metadata standard and file format. The software required to read the data will also be made available on a similar basis, along with appropriate documentation.

- If all projects are using Rucio, it is easy to allow them to access (some of) each others data.
- Rucio also provides data dumps:

Replicas per RSE

Retrieve a tab-separated, bz2 compressed, list of replicas at an RSE.
URL: https://rucio-hadoop.cern.ch/replica_dumps
URL params: rse, date (optional)
Example: https://rucio-hadoop.cern.ch/replica_dumps?rse=CERN-PROD_DATADISK&date=21-01-2015
Format: RSE, scope, name, checksum, size, creation date, path, update date, state, last accessed date, tombstone
Notabene: if no date provided, the latest available dump will be taken

Key features

Key features: Based on the requirements from the Key features for the new data curation service include:

- Ease of upload/download
- Citability (e.g. deterministic URLs to retrieve datasets, preferably human readable)
- A stable interface and API (so that scripts will remain usable for the lifetime of the data)
- Efficient, comprehensive metadata search capabilities to aid discovery and re-use of data
- Rapid availability
- Access controls
- Proximity to compute resource and access to visualisation tools

- Yes.
  - Metadata search doesn't scale indefinitely.

RDI Requirements

- RDIs should reflect a sector-appropriate implementation of FAIR principles - this requires the ability to define appropriate metadata and search tools to ensure that DiRAC data products can be analysed, discovered, combined, reused and repurposed;
- RDIs should be inclusive, interoperable (both with UKRI and non-academic database management systems), federated and sustainable in order to ingest, link and blend new data sources - this requires the DiRAC Data Curation Service to be able to support access from other services outside DiRAC's control;
- RDIs should work with Jisc to develop principles for negotiation of commercial interoperable open research data infrastructure, allowing data ownership to be retained and reducing the risk of commercial "lock-in";
- Shared APIs between different communities should be defined and supported where possible.
- RDIs should have the ability to hold and exchange sensitive data securely to allow greater re-use of data;
- Where possible and appropriate, data processing and storage capability should be co-located, aggregated and shared;
- RDI services should be user friendly, whether they are generic or sector-specific.

Security

- Rucio uses X.509 security to authenticate and authorize.
  - Rucio is adding IAM support.
- It does not:
  - Secure the facilities.
  - Encrypt the data.
  - Secure the transfers.

Rucio

A genuinely secure system

CLI / Scripted Upload

Example: CLI / Scripted Upload

A researcher is coming to the end of a project and wishes to upload their data products to the Curation Service for long-term archival. The researcher is familiar with command-line interfaces and wishes to perform the upload this way from a data-transfer-node on their institution's cluster.

The researcher has a list of files and directories to archive, which they feed to the CLI tool along with arguments specifying the metadata to attach to this data.

The tool issues the necessary data-transfer commands to the curation service, which begins the job for the researcher. The researcher is given a job identifier they are able to use with the tool in future to query the status of the job, to get an idea of its progress and when it completes.

The output of the tool can be formatted in a machine-friendly format which the researcher uses in a small script to periodically query the job's status and to launch a further archival request when it notices the original job has reached completion.

The researcher also visits the Curation Service website and authenticates to browse their archive and check the progress of ongoing jobs.

CLI / Scripted Upload

- Create the dataset and define where the data will be stored.

```
$ rucio add-dataset user.dewhurst:myPhDAnalysis
$ rucio add-rule user.dewhurst:myPhDAnalysis copies RSE_expression
```

- Upload data, metadata and attach to dataset

```
For File in Files # some kind of loop over the files
$ rucio upload --rse Site File
$ rucio add-did-meta --did File --key KEY --value VALUE
$ rucio attach user.dewhurst:myPhDAnalysis File
```

- Query status of transfer

```
$ rucio rule-info [--examine] [--estimate-ttc] rule_id
```

Metadata search and Bulk Download

Example: Metadata search and Bulk Download

A new researcher in a particular group has a novel idea and wants to perform a new analysis of an archived dataset. This dataset was previously moved into the DiRAC curation service from another institution's HPC storage system, and was tagged with a particular piece of metadata, identifying the study that produced the dataset.

The researcher opens the DiRAC curation service web-UI and is able to authenticate, and be authorized as a member of the research group to which they belong, which enables them to have permission to view the multiple archived datasets belonging to the PI of their research group.

The researcher is able to perform a search using the study metadata they were given, which shows them a listing of the data tagged with this metadata.

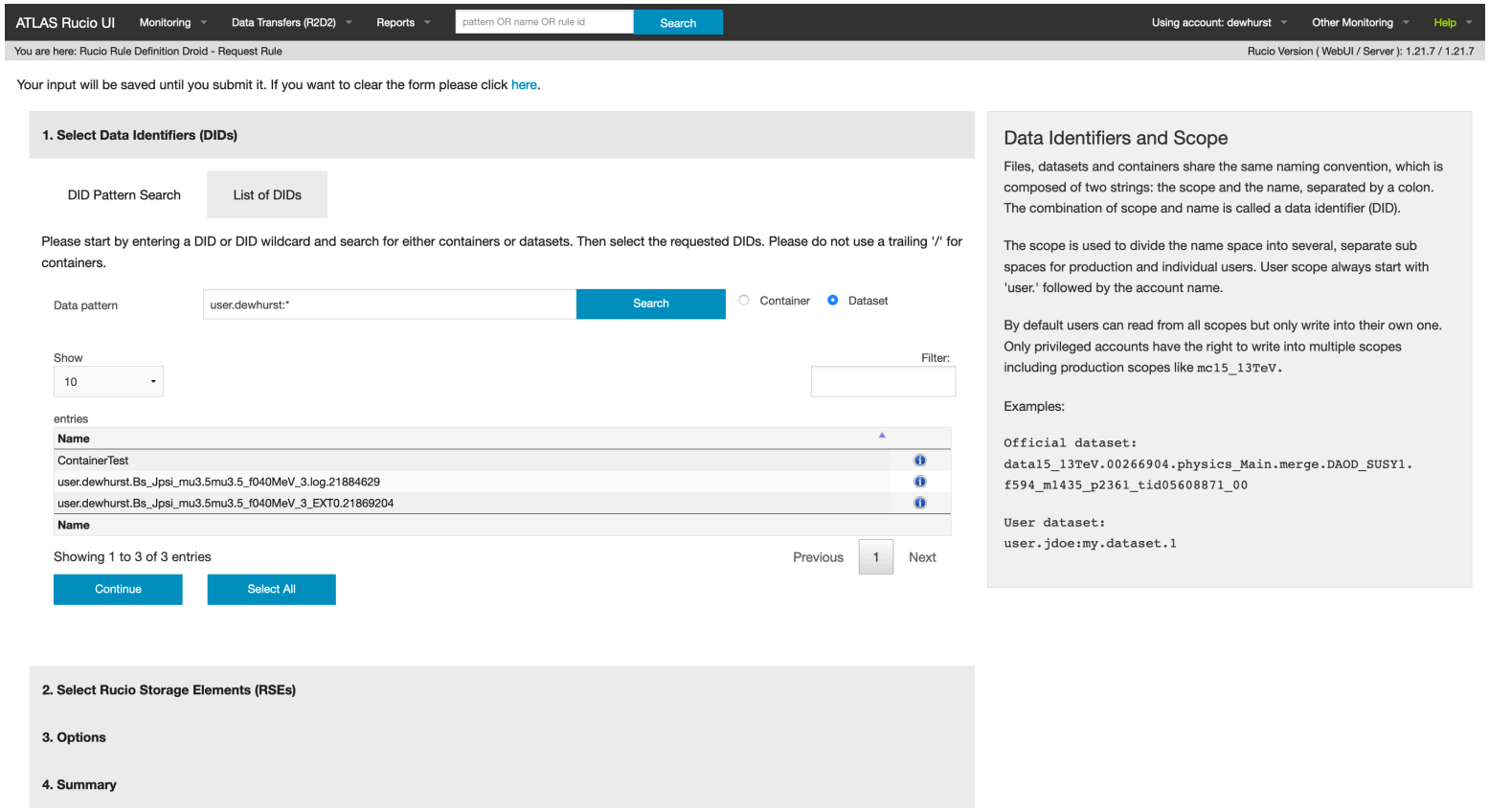
The researcher requests a download of this data from the web UI of the curation service, and is able to request a copy of this data to be delivered to another of the DiRAC institutes where their project happens to currently have an allocation on a HPC storage system on which to store the dataset to be processed.

The researcher selects the institute and the destination directory for the data to be downloaded to, and begins the transfer which happens asynchronously in the background by the Curation Service.

The researcher is able to check on the status of the download from the Web UI, and is notified by email when the download is completed.

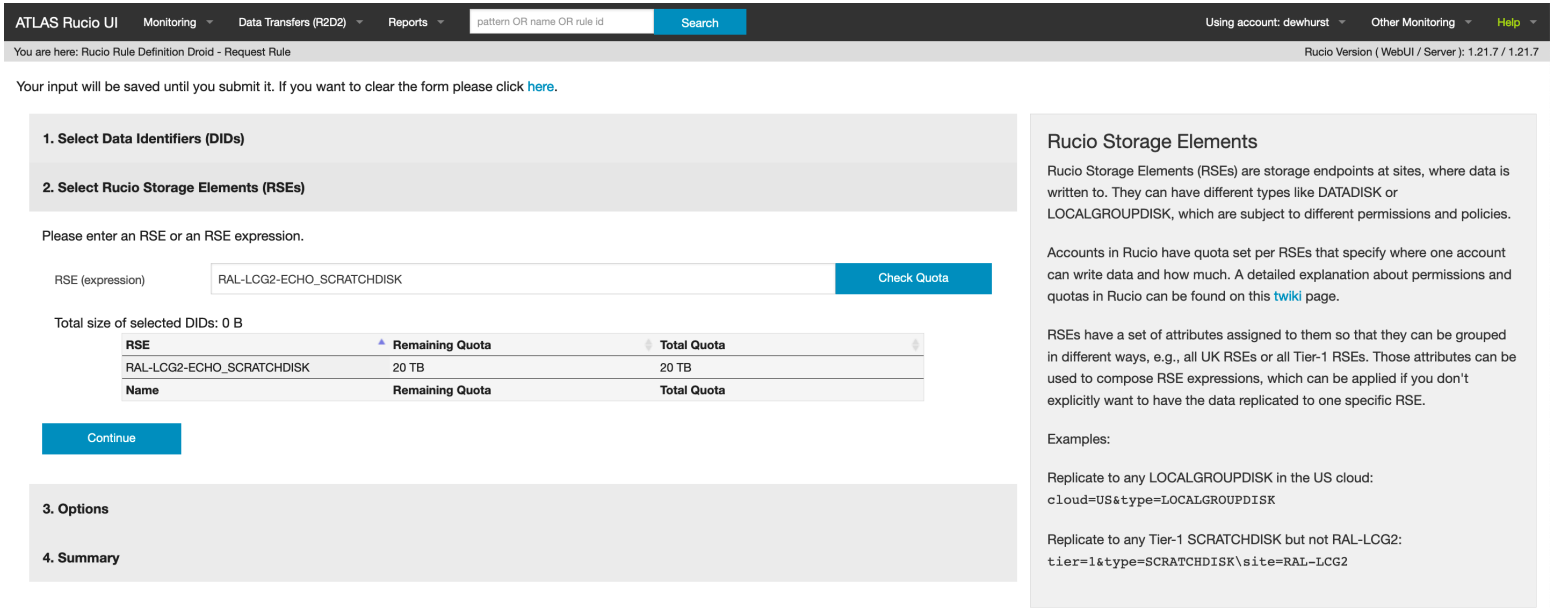
User Requests (1)

ATLAS User can submit bulk transfer requests via a CLI or via a Web UI

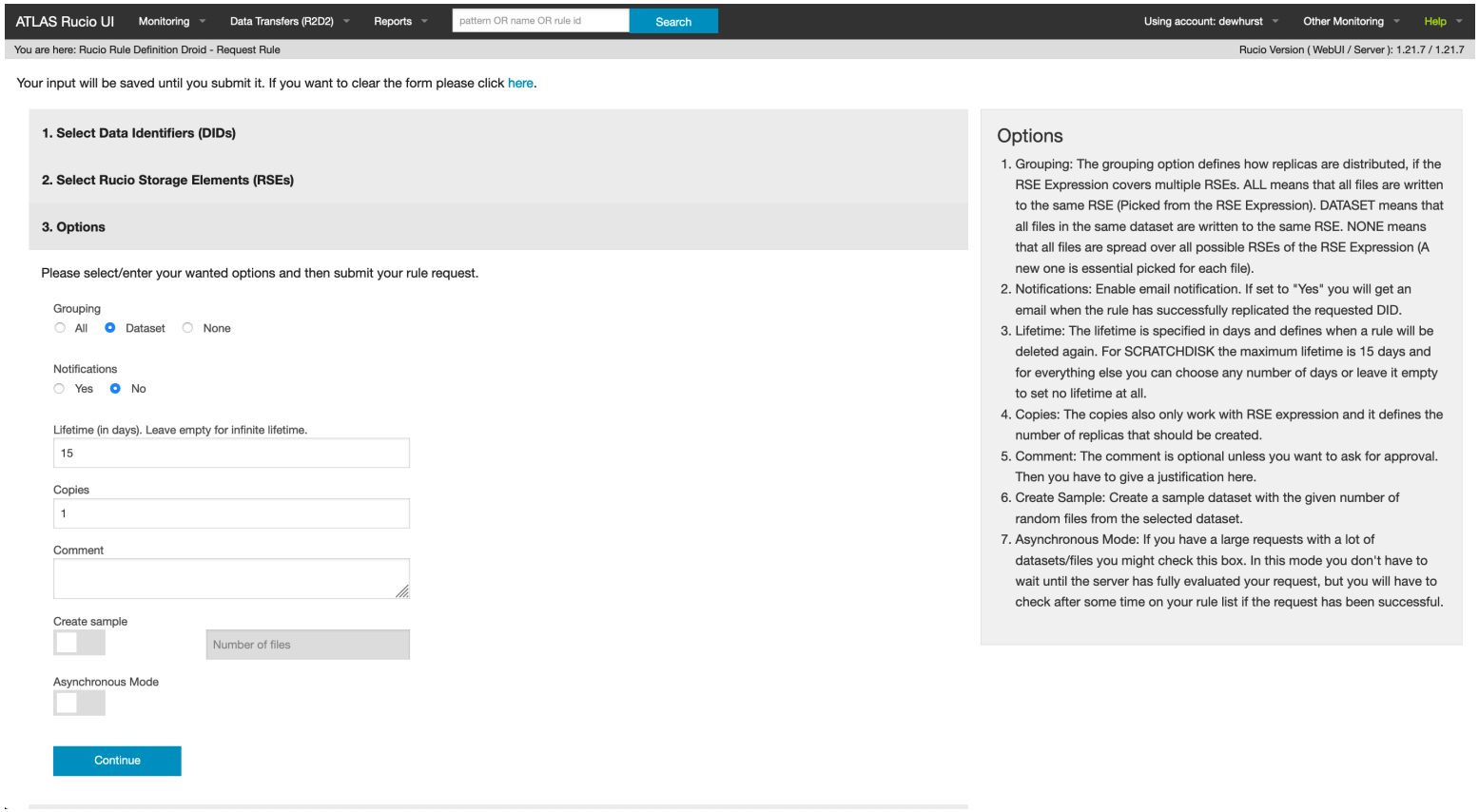


Note: This Web UI does not come out the box with Rucio.

User Requests (2)



User Requests (3)

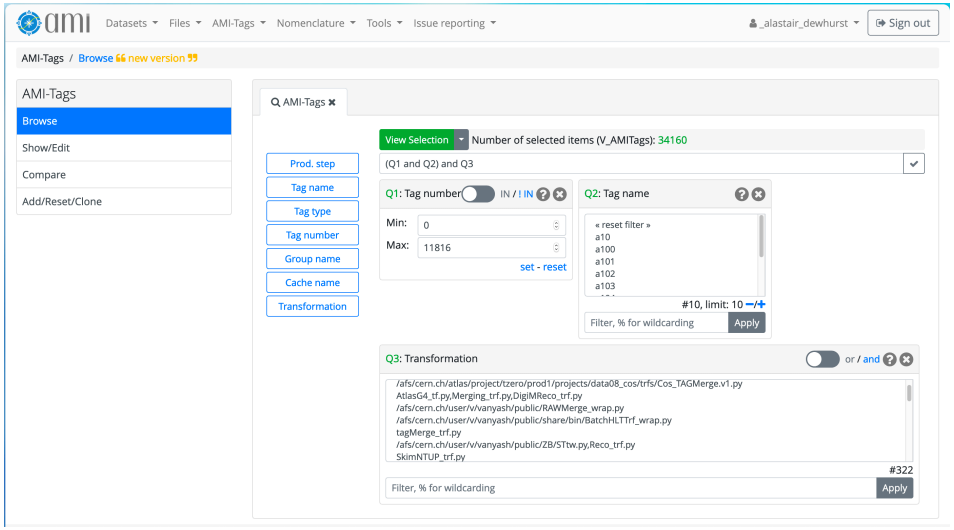


Metadata

- The database behind Rucio is an industry standard relational database:
  - Oracle at CERN.
  - PostgreSQL at RAL.
- Adding $O(10^1)$ well defined (so they can be indexed) pieces of metadata to each dataset $O(10^6)$ should not cause a problem.
  - Optimizations may been needed at DB level.
- Adding arbitrary metadata $O(10^3)$ to each file $O(10^9)$ will not work in practice.
  - You would need a separate service.

AMI

- The ATLAS Metadata Interface[1] (AMI) is a separate service to Rucio.
  - AMI talks to Ruico using the API.
  - It provides a generic metadata service.
  - Also used by nEDM and SuperNemo.



ATLAS has an awful lot of metadata and users who are constantly adding more…

[1] https://iopscience.iop.org/article/10.1088/1742-6596/898/6/062001/pdf

Transfers between DiRAC services

Example: Transfer between DiRAC services

A DiRAC project has finished some simulations which have produced a large amount of data on the DiRAC Memory Intensive service. This project has further compute and storage allocations on the Data Intensive service at Leicester, where post-processing and analysis will be performed.

A member of this project authenticates with the Curation Service website and initiates a transfer between these two DiRAC sites. Although the data volume is large (petabyte scale), and contains a large number of small files (in addition to large files), the Curation Service is able to move the data efficiently, and the transfer completes in a significantly shorter time than if the user had used rsync to transfer it.

File metadata, such as striping information, is intelligently migrated with the data, so that files at the destination are striped in a way suitable for the destination storage (which has different numbers of LUNs). The project members have identical user names at each site, though their UIDs are different. Ownership of the data is preserved correctly.

Once the transfer has completed the project member is notified by email.

Data Transfers

- Rucio submits transfer requests to the FTS service.
- FTS (File Transfer Service) manages (and optimizes) the many transfers between sites.
- You could write a bespoke transfer plugin that optimized how the data was written to a specific site.
  - This doesn't scale with the number of sites.
  - Probably better if the site handles optimization of its underlying storage.

Summary

- I have some concerns over the number of small files and the amount of metadata required.
- Otherwise Rucio looks like it fits your use case well.
- You want a lot of features which ATLAS use in production but aren't the standard setup.
  - Work would be required, but it should be straight forward when requirements are finalized.

Backup

Identity and Access Management

Flexible authentication support

- (SAML, X.509, OpenID Connect, username/password, …)

Account linking

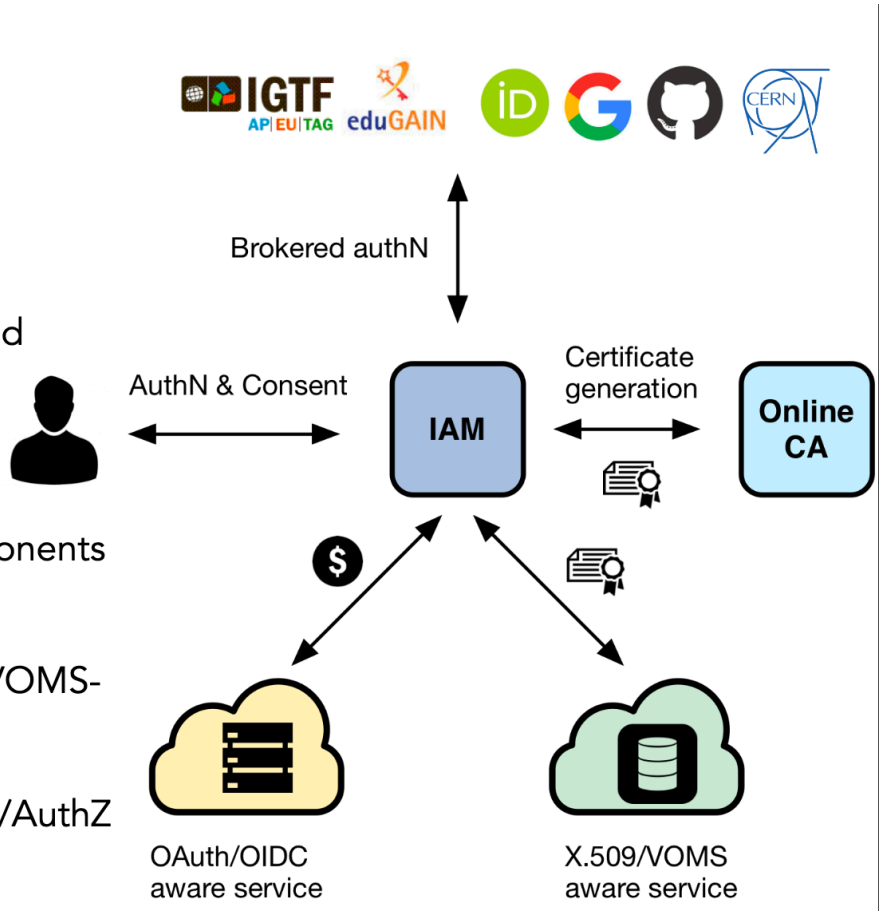
Registration service for moderated and automatic user enrollment

Enforcement of AUP acceptance

Easy integration in off-the-shelf components thanks to **OpenID Connect/OAuth**

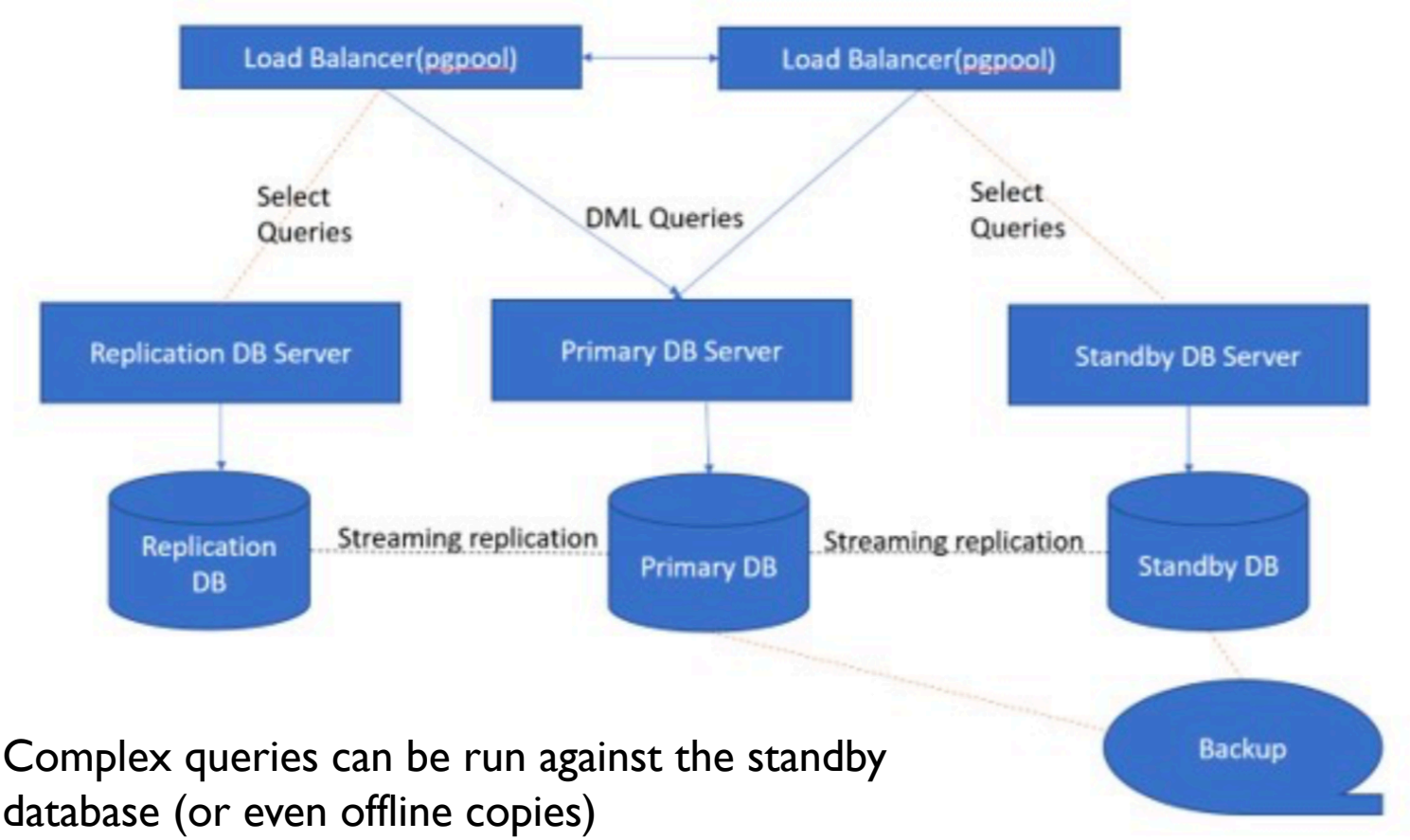
VOMS support, to integrate existing VOMS-aware services

Self-contained, comprehensive AuthN/AuthZ solution



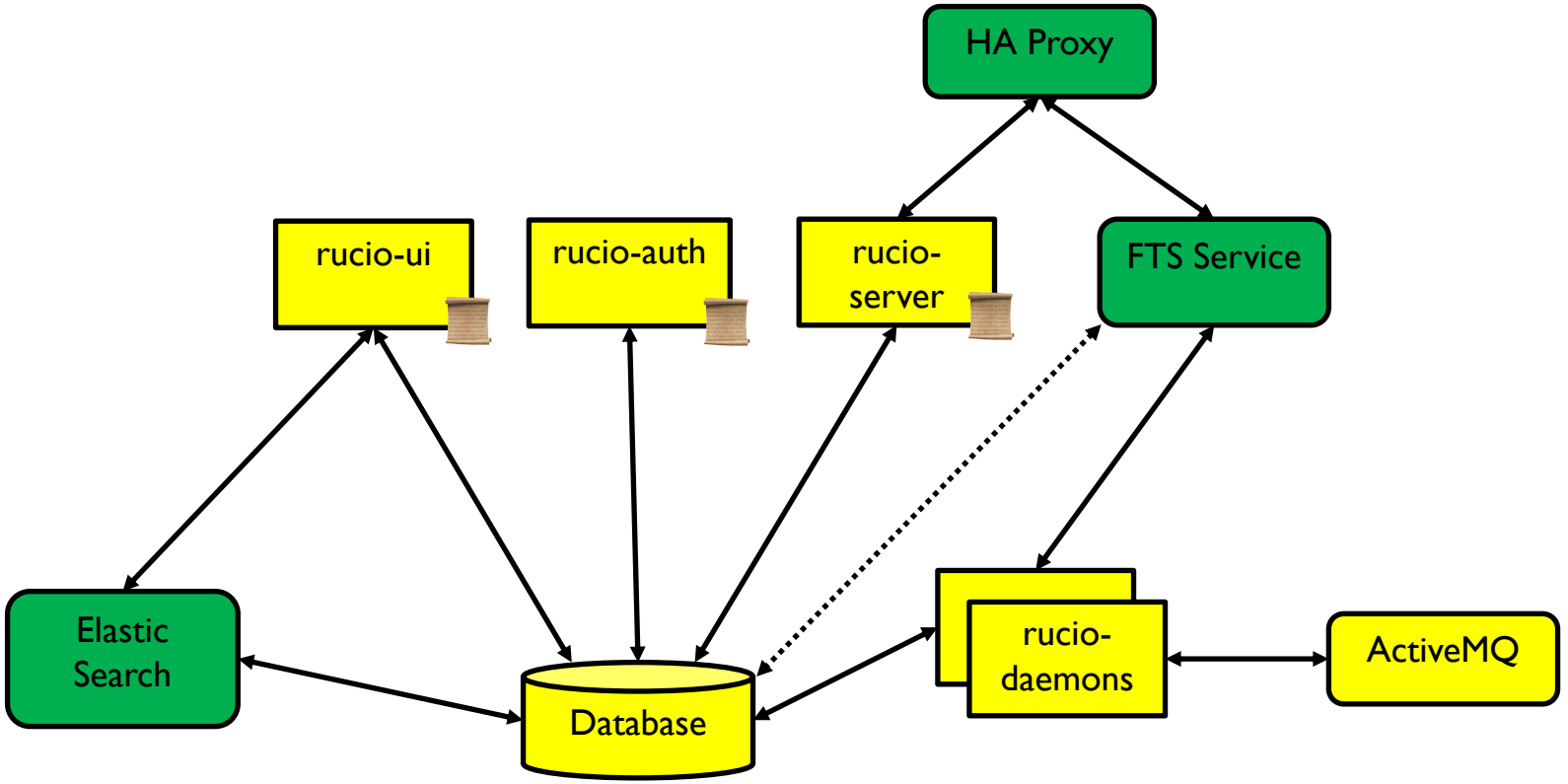
https://indico.cern.ch/event/651348/sessions/286989/attachments/1758213/2873793/IAM-PreGDB-11-12-2018.pdf

Database Architechure



Complex queries can be run against the standby database (or even offline copies)

Rucio@RAL


HA Proxy
rucio-ui
rucio-auth
rucio-server
FTS Service
Elastic Search
Database
rucio-daemons
ActiveMQ