DUNE UK Computing

Andrew McNab

University of Manchester

Overview

- UK computing organisation in DUNE
- Computing capacity delivered to DUNE
- Roadmap away from SAM
- MetaCat/Rucio/justIN transition

DUNE UK Computing organisation

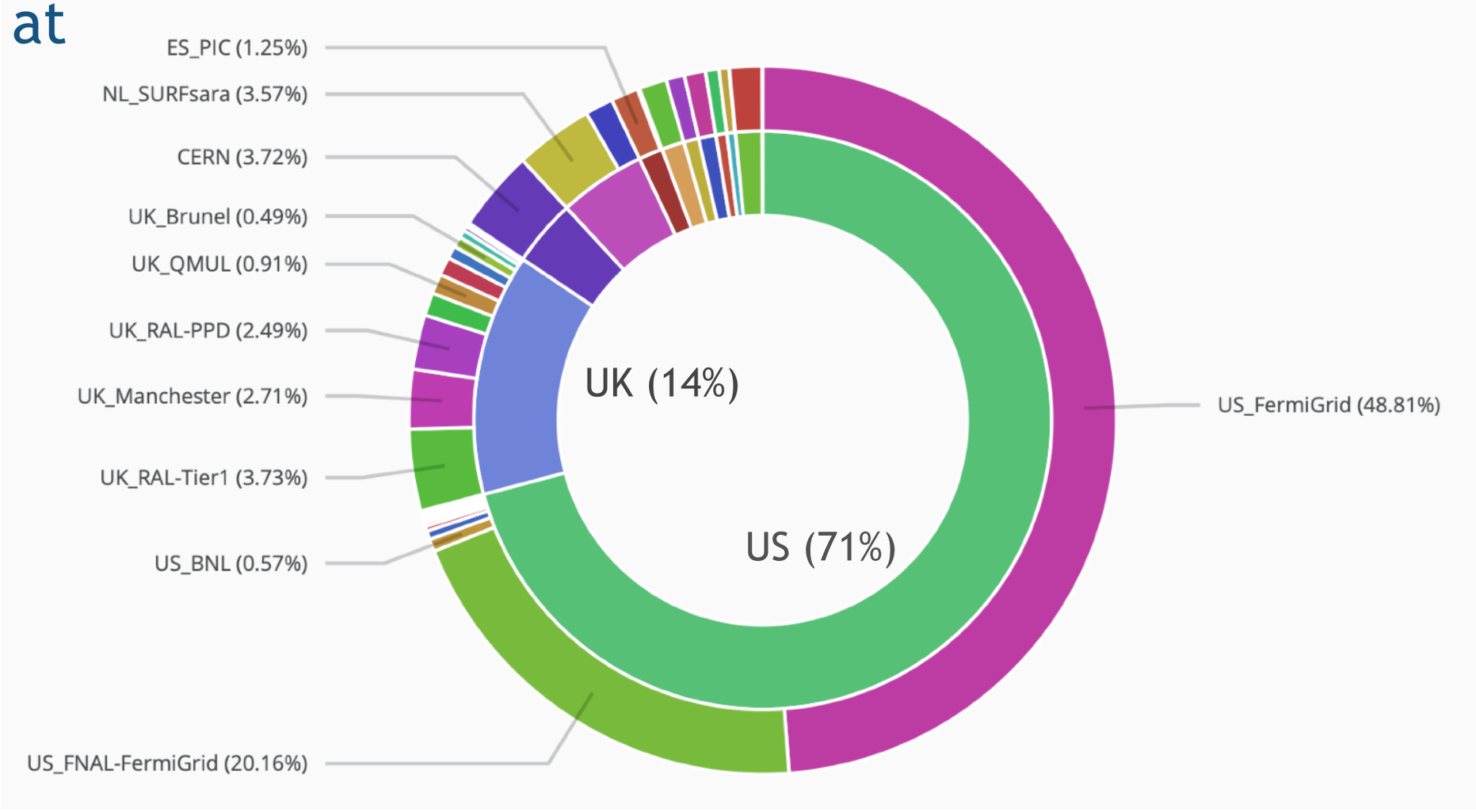
- Computing = hardware at sites and middleware which connects it
  - So not physics software. Not LArSoft etc.
- Development funded as part of WP1 of the UK construction project
  - At Edinburgh, Manchester, RAL-PPD
  - Non-funded institutes (Bristol, Imperial, Lancaster, QMUL, …) also provide computing capacity and support, attend ops meetings etc
- Monthly meetings to
  - Report status of development work
  - Review use of UK CPU and storage capacity by DUNE
- UK also contributes
  - DUNE Computing Contributions Board chair (Peter Clarke)
  - DUNE Computing International Technical Lead (Andrew McNab)

Computing capacity delivered

DUNE jobs for July-June 22/23

Still a strong bias towards running at Fermilab by default

But for production, the picture is changing as new tools are being used …



Fraction of total slot hours per site or country

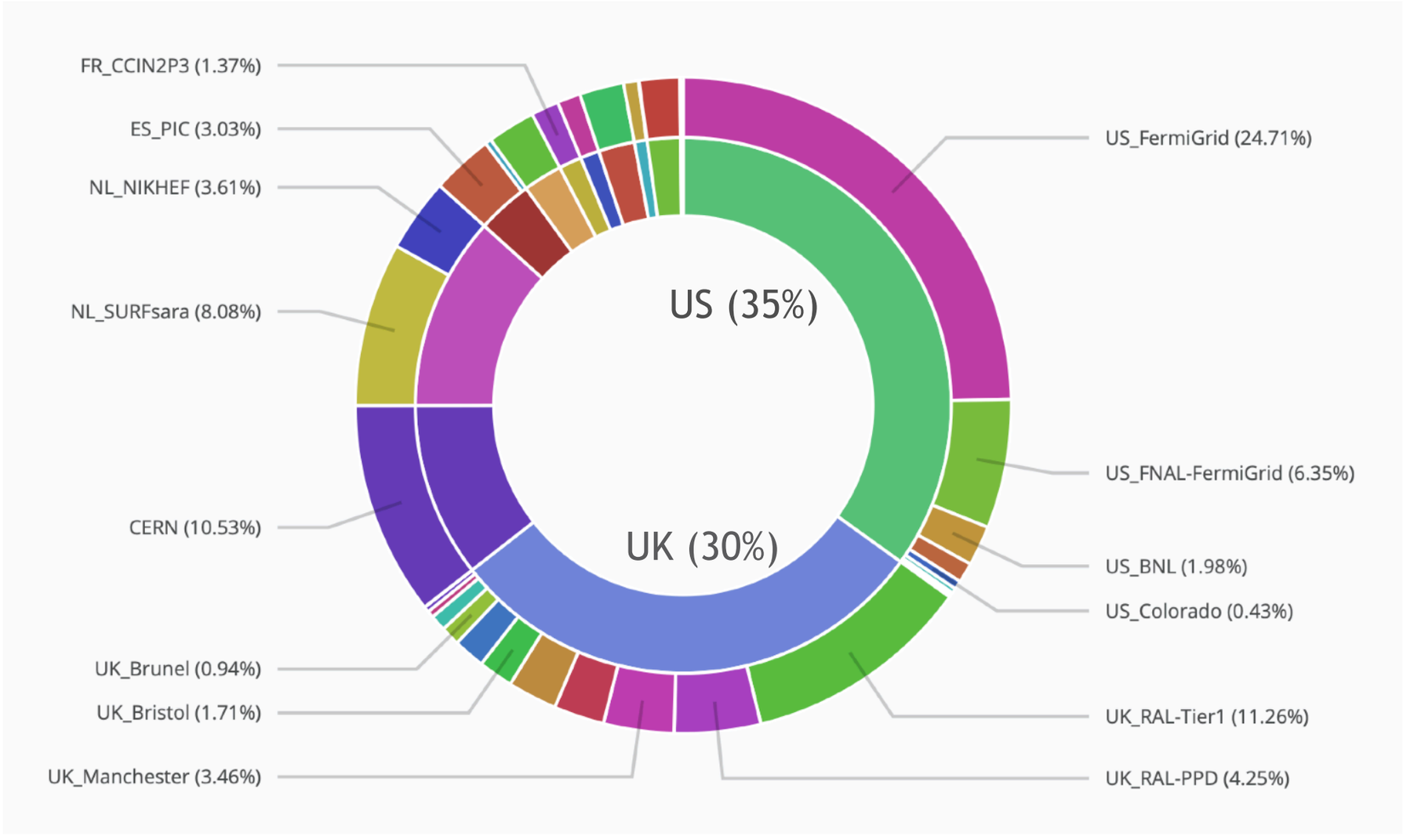
US_FermiGrid and US_FNAL-FermiGrid just due to a renaming

Production July-June 22/23

Larger memory requirements enabled at UK sites to attract jobs

Broader set of sites being targeted

2022 Data Challenge in Q4 tested systems with automated matching of jobs to best site



Fraction of total slot hours per site or country

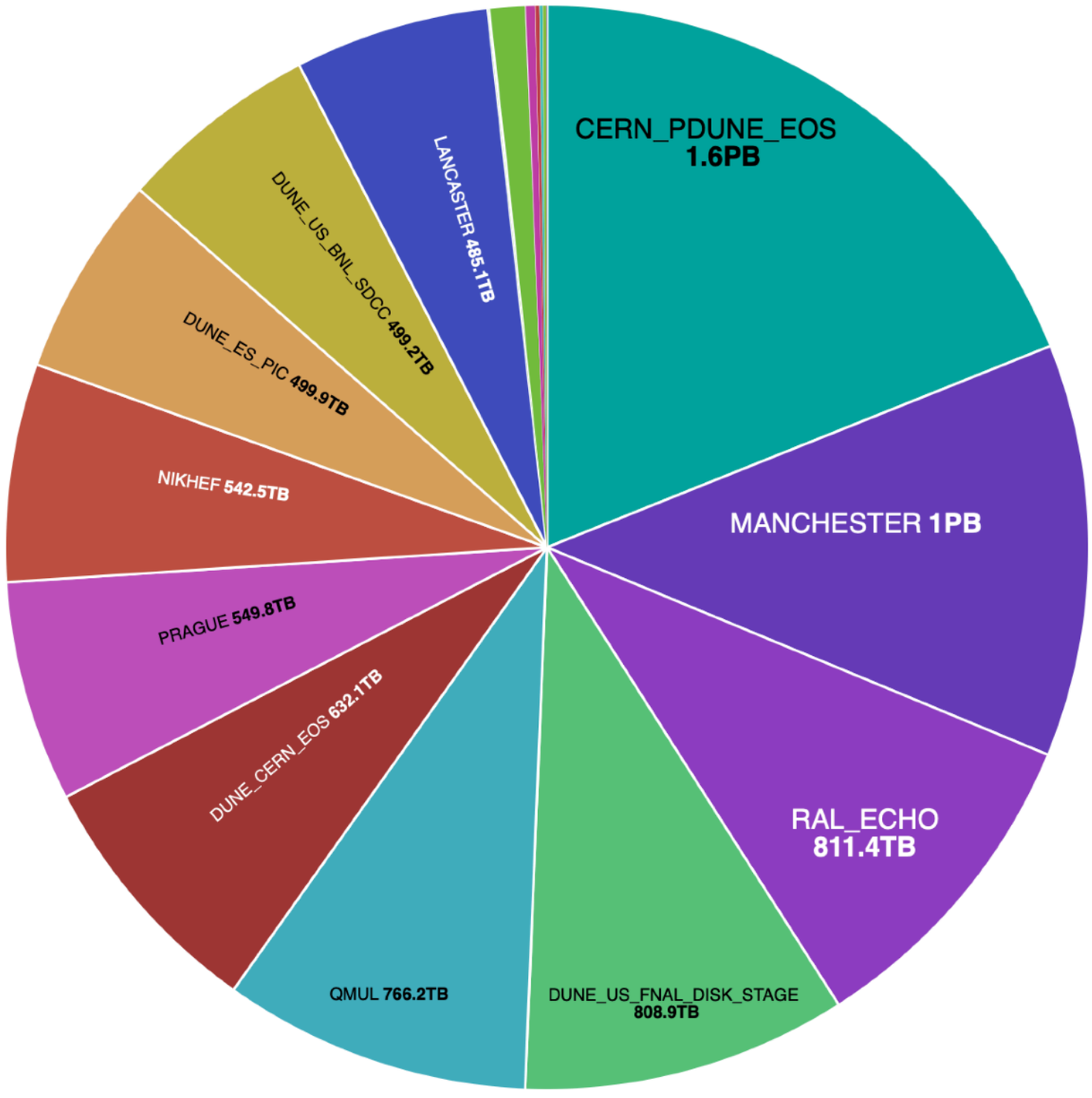
DUNE disk usage status

Also 10.6PB of dCache/ tape in use at Fermilab and 5.5PB on tape at CERN

All of this is Rucio-managed storage. We know what is where and can reclaim it when no longer needed.

Screenshot of the storage dashboard developed by Edinburgh as part of DUNE UK

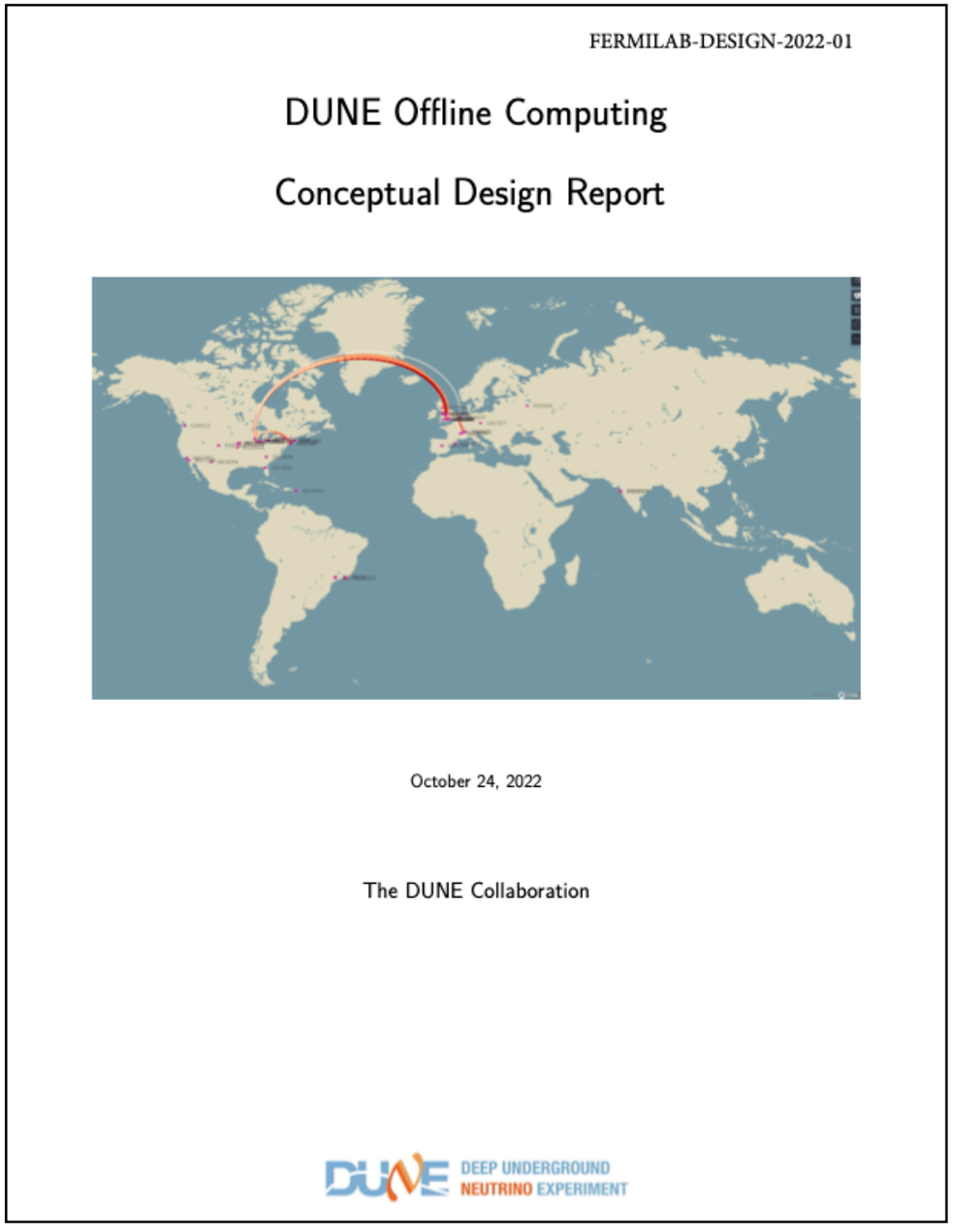
Snapshot of usage NOT space provided. So skewed by any mass deletions



Roadmap away from SAM

Roadmap away from SAM

Roadmap away from SAM



FERMILAB-DESIGN-2022-01

DUNE Offline Computing

Conceptual Design Report

October 24, 2022

The DUNE Collaboration

DUNE DEEP UNDERGROUND NEUTRINO EXPERIMENT

Replacing SAM

- SAM plays several roles which are now being shared out
- MetaCat is our new metadata catalogue

  - stores the metadata describing each file
  - handles complex searches for lists of types of file
- Rucio is our new replica catalogue

  - knows where copies of files are
  - can make additional copies elsewhere following rules
  - Edinburgh is extending Rucio and creating DUNE add-ons
- justIN keeps track of processing requests

  - decides where best to process each file
  - keeps track of which files still need to be processed
  - created and maintained by Manchester and RAL-PPD

But first, jobsub and Condor …

- This spring we've also seen Fermilab CSAID replace old jobsub with new jobsub_lite
  - Very similar interface but uses SciTokens/WLCG Tokens instead of X.509 certificates and proxies
- At the same time the dedicated DUNE HTCondor Pool has gone into production
  - By default you still submit to the shared FIFE HTCondor Pool
  - But the DUNE pool gives us more control of configuration, and scheduling between users/groups
  - And allows DUNE to run extra schedulers outside the Fermilab site firewall
    - RAL-PPD is running one for the justIN workflow system

MetaCat

- Central service and database at Fermilab
  - Plus command line client and a Python API package
- For each data file in DUNE's managed disk and tape storage
  - We must have a MetaCat entry so we know what the file is
  - So you submit a JSON file to MetaCat with the key/value metadata in the required format
- You can then query the MetaCat database using MetaCat Query Language (MQL) expressions:

```
files from MyScope:MyDataset
    where params.x > 0.5 and params.x < 1.5
            and data.run = 123
            and ( data.type="MC" or data.type="Data" )
```

  - In return you get a list of matching files

Rucio replica management

- Rucio was originally the ATLAS file replica catalogue but has since become an independent product
- We use it to record which storages have replicas of each file
  - Also has rules that can trigger the replication of the file to more sites (“one replica on each side of the Atlantic” etc)
- Edinburgh contributed in a number of Rucio areas for DUNE
  - Core Rucio development, including support for “policy packages” (plugins) and simplifying the Rucio client dependencies
  - DUNE specific extensions including the DUNE policy package which enforces a MetaCat entry for each file
  - DUNE’s storage monitoring dashboard which gets its info from Rucio

justIN workflow management

- The UK leads the Workflow area in DUNE computing
- Design work for a successor to SAM led to justIN which uses a just-in-time model
  - Implemented and maintained by Manchester and RAL-PPD
- User defines a processing request involving one or more stages
- justIN finds the files and manages the jobs that process them
- Relies on MetaCat to supply the initial list of input files using a MQL query
- Relies on Rucio to say where the copies of the files are
- Uses GlideInWMS/HTCondor to deliver jobs to compute sites
  - But justIN takes the decisions about what request jobs will work on and which file(s) each job works on

justIN developments in 2023

- justIN validated in 2022 Data Challenge, exceeding minimal requirements for protoDUNE data processing
- Further work this year so far
  - All user interaction now relies on SciTokens/WLCG Tokens (same as new jobsub_lite) - no X.509 certificates
  - Jobscripts supplied by users now run inside Apptainer/Singularity containers which controls what users can do to DUNE data files
  - justIN uses new DUNE Global HTCondor pool, with a dedicated scheduler at RAL
- Design and prototyping underway to embed justIN just-in-time decisions about where to run processing requests in GlideInWMS/ HTCondor matching
  - This will make core of justIN available to POMS users too
  - … but without write access to Rucio managed bulk storage

Two stages from the roadmap

Outline of a future workflow

- Imagine you want to process some data files on DUNE's Rucio managed storage and save outputs
- You submit a request to justIN which specifies
  - the MetaCat MQL query to find the right input files
  - how much memory, processors, hours, etc each job will need
  - the jobscript you want to run (in Bash or Python)
  - what output files to create
- justIN gets the list of input files from MetaCat and the file replica locations from Rucio
- justIN runs the jobs at suitable sites, and gives your job the replica URLs to access the files
- justIN finds the output files your jobscript created and uploads them to nearby Rucio managed storage, after some sanity checking

DUNE permissions model

- Mostly “who can upload/delete files on Rucio-managed storage”
- We are organising Rucio storage into scopes rather than by directory
- Rucio Data Identifiers (DID) are scope:filename

```
dc4-vd-coldbox-bottom:dc4_np02bde_np02_bde_coldbox_run012352_20211215T220536.hdf5
```

- Current *proposal* is to limit who can create files in each scope
  - Can then impose quotas based on space used by each scope
  - Scopes can be owned by working groups who are then able to
    - manage their own usage
    - add/remove people responsible for working group productions
- justIN already implements this model for central production
  - Only people in the `/dune/production` group can create files in scopes owned by the production account `dunepro` in Rucio

Summary

- The UK is making a significant and visible contribution to the CPU and storage supplied to DUNE
  - Underpinned by STFC funding for hardware and staff via IRIS and GridPP
- We are active members of the Computing Consortium,
  - providing a Technical Lead (Andrew McNab) and the Chair of the Computing Contributions Board (Peter Clarke)
- As part of the construction project, we are providing development effort and leadership in Data Management and in Workflow
- The stages on the roadmap away from the initial jobsub/SAM ecosystem are becoming visible to users — more to follow
- All of this is key to providing the volume of computing needed to meet the experiment's goals

Workflow System components