Rucio functionality and components

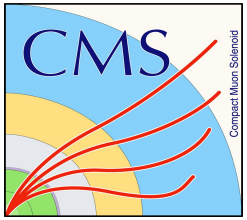
Cédric Serfon

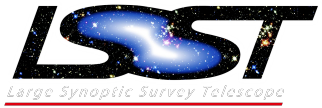
BROOKHAVEN NATIONAL LABORATORY

Rucio in a nutshell

- Rucio is an advanced Distributed Data Management initially developed for the ATLAS experiment
- Its development started in 2012 and it was fully put into production in December 2014, before the start of LHC run 2
- Rucio is replacing a previous DDM system called DQ2 that used LFC as replica catalog. It addresses the issues that were identified in DQ2
  - Scalability
  - Dependency on external services
  - No support of multiple protocols
  - Limited policy replication tools
  - And many more
- Rucio is now evaluated or used by a large community

Rucio community





Rucio main functionalities

- Provides many features (can be enabled selectively)
  - File and dataset catalog (logical definition and replicas)
  - Transfers between sites and staging capabilities
  - Web Interface and Command Line Interface to discover/download/upload/transfer data
  - Extensive monitoring
  - Powerful policy engines (rules and subscriptions)
  - Bad file identification and recovery
  - Dataset popularity based replication
  - …

More advanced features

- Rucio can be integrated with Workload and Workflow Management System, in particular Panda. The two systems evolved in symbiosis over the last years

Some non-exhaustive advanced features

- Rules and subscriptions :
  - Data Management tool to implement replication policy (e.g. place one copy of all RAW data to one site matching certain criteria according to a defined share)
- Smart space usage (only clean data once needed based on LRU algorithm)
- Very well integration with Panda and Information System (CRIC)
- Generic metadata (being evaluated by Belle II)
- Coming soon : Rucio is involved in almost all the Data Organization, Management and Access (DOMA) activities from WLCG. Some new developments are currently in the pipeline (e.g. storage QoS)
- As mentioned previously, Rucio is modular and you are free to choose which feature you enable

Experience with Rucio at BNL

- I’m member of the Rucio core team
- ATLAS BNL is leading many projects involving Rucio (tape carousel, GCS)
- BNL has been responsible for the migration of Belle II to Rucio.
  - NPPS was responsible of developing the new features needed for this migration that was done in collaboration with SDCC.
  - SDCC is running the Rucio instance of Belle II at BNL and the File Transfer Service (FTS) that is used to move the data
- Belle II instance :
  - Running under PostGreSQL and hosts around 107M file replicas using a dedicated machine for the DB
  - 4 nodes (2 servers + 2 daemons) are able to serve Belle II needs
- BNL also hosts different monitoring dashboards for transfers or space accounting

Conclusion

- A lot of effort was done by the Rucio core team to help the adoption of Rucio by other communities
- This was a success. Now more and more requirements and features come from non-ATLAS or even non LHC community
- ATLAS still plans to use Rucio for run 3 and beyond → Long term support
- To learn more:
  - Some resources are attached to the agenda if you want to learn more
  - For any other questions, you can also ask rucio-dev@cern.ch or me directly. If there is some interest, we can setup a Rucio Mattermost channel at BNL

More information

Website		http://rucio.cern.ch
Documentation		https://rucio.readthedocs.io
Repository		https://github.com/rucio/
Images		https://hub.docker.com/r/rucio/
Online support		https://rucio.slack.com/messages/#support/
Developer contact		rucio-dev@cern.ch
Publications		https://rucio.cern.ch/publications.html
Twitter		https://twitter.com/RucioData