Rucio data management

Recap of PUNCHLunch

Andreas Redelbach

28/07/2022

Overview

PUNCHLunch on June 26:

RUCIO - Large scale data management for high energy physics experiments and beyond

→ Indico https://indico.desy.de/event/35097/

→Discussion of needs and options for SKAO

→Slides available

→Conclusions for TA5 for offline and online workflows to be discussed

Rucio in a nutshell

Input from Martin Barisits et al.

Horizontally scalable catalog for files, collections, and metadata

Many interfaces available, including CLI, web, FUSE, and REST API

Not a distributed file system, it connects existing storage infrastructure over network

Data centres are free to choose which storage system suits them best

All data stored in Rucio is identified by a **Data IDentifier (DID)**

Rucio supports different kinds of **metadata backends** via a plugin approach

Column-based stores, generic json-based metadata, mongodb, ...

Plugin approach is adjustable to proprietary (experiment internal) **metadata stores**

Searchable via name and metadata

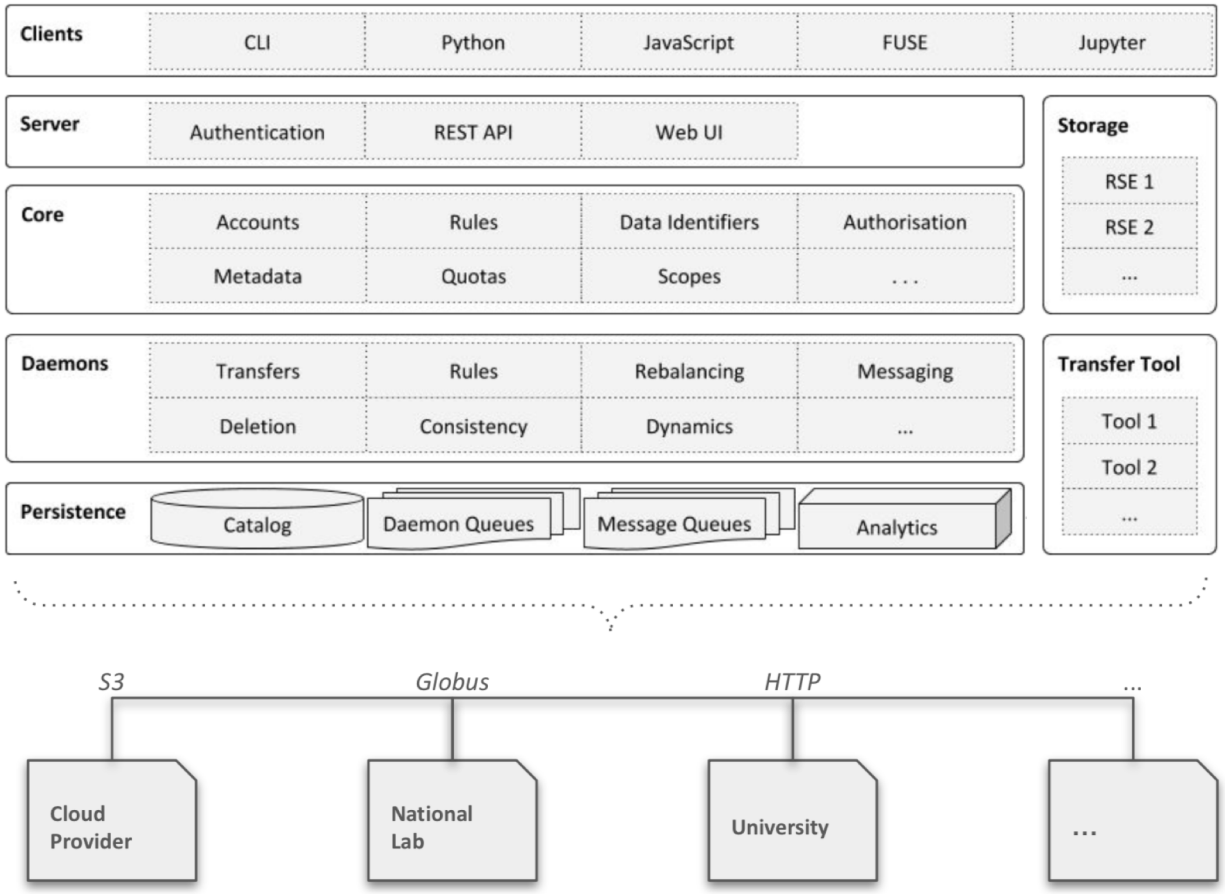
Rucio Storage Elements (RSEs) are logical entities of space

RSEs collect all necessary metadata for a storage system

RSEs can be assigned meta data

Belle II (interaction rate 100 Hz) Rucio instance with synchronization agents

Rucio: High-level architecture



Horizontally scalable component-based architecture

Servers interact with users
- HTTP API using REST/JSON
- Strong security (X.509, SSH, GSS, OAuth2, …)
- Many client interfaces available

Daemons orchestrate the collaborative work
- Transfers, deletion, recovery, policy, …
- Self-adapting based on workload

Messaging support for easy integration
- STOMP / ActiveMQ-compatible protocol

Persistence layer
- Oracle, PostgreSQL, MySQL/MariaDB, SQLite
- Analytics with Hadoop and Spark

Middleware
- Connects to well-established products, e.g., FTS3, XRootD, dCache, EOS, Globus, …
- Connects commercial clouds (S3, GCS, AWS)

SKA workflows using Rucio prototype

ESCAPE WP2 DIOS collaboration - CERN as lead, but developing real interest from several Astroparticle / HEP Experiments CTAO, KM3NET, LOFAR, SKAO, FAIR

SKAO Rucio prototype

Judicious re-use of existing stack from ESCAPE (eg FTS, storage, IAM)

Well suited to centralised Operations model for data management

Performed long-haul transfers, Rucio stress tests, subscriptions (via our automated test framework)

Aim to integrate storage from national SRC efforts to increase understanding and inform assessment

Input from Rohini Joshi et al.

See also: R. Bolton et al. „**ESCAPE Data Lake**“ , EPJ Web of Conferences **251**, 02056 (2021)
ESACPE Data Lake: [D2.2 Assessment and analysis of performance of the first pilot data lake Report]

SKA metadata

Metadata: Recent work done (Filtering engine)

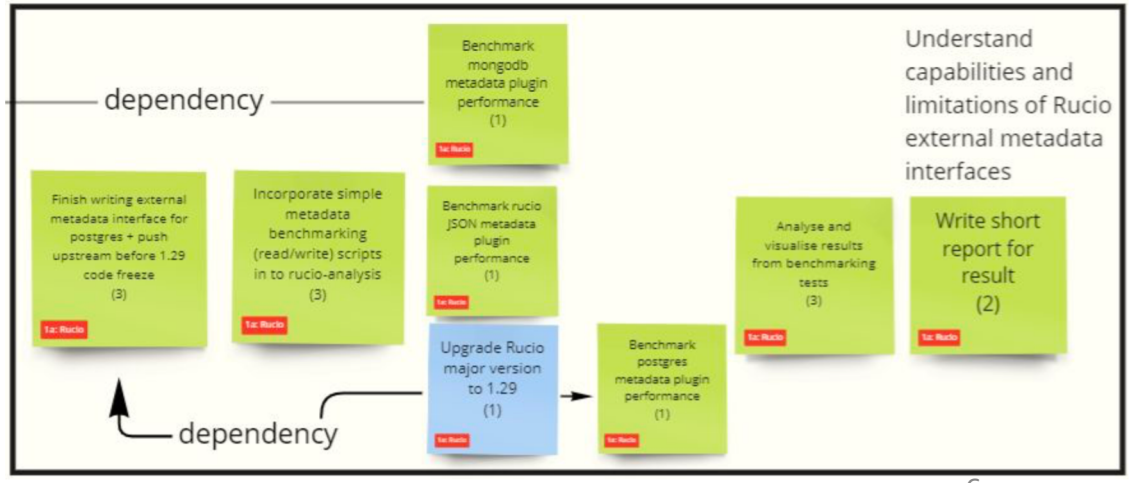
Engine supports an extended syntax including inequalities and wildcards

Plugin system allowing one to write external metadata interfaces by overriding base functions for how to get/set/delete/query metadata

Another metadata plugin to interface to an (external) postgres database will be available

Adding an external RDBMS such as postgres may help with integrations, e.g. TAP access (table access protocol, Virtual Observatory)

SKAO has plans to **test e.g. how performant searching is, and how this scales with number of files and complexity of query** (restrictions from project roadmap)



Summary / open points

→ Rucio system is well supported
→ Suited for offline data handling of distributed data
→ Results from stress-tests and data lifecycles to be completed/followed up

Developments:

- Flexible metadata
- Token-based authorisation in development (relevant for TA2)
- First results from SKA stress tests will be interesting

To be discussed:

- Not for online / time-critical workflows
- Handling of large-scale metadata to be tested
- Coupling of online data to offline data bases (seminar planned with expert from ALICE)