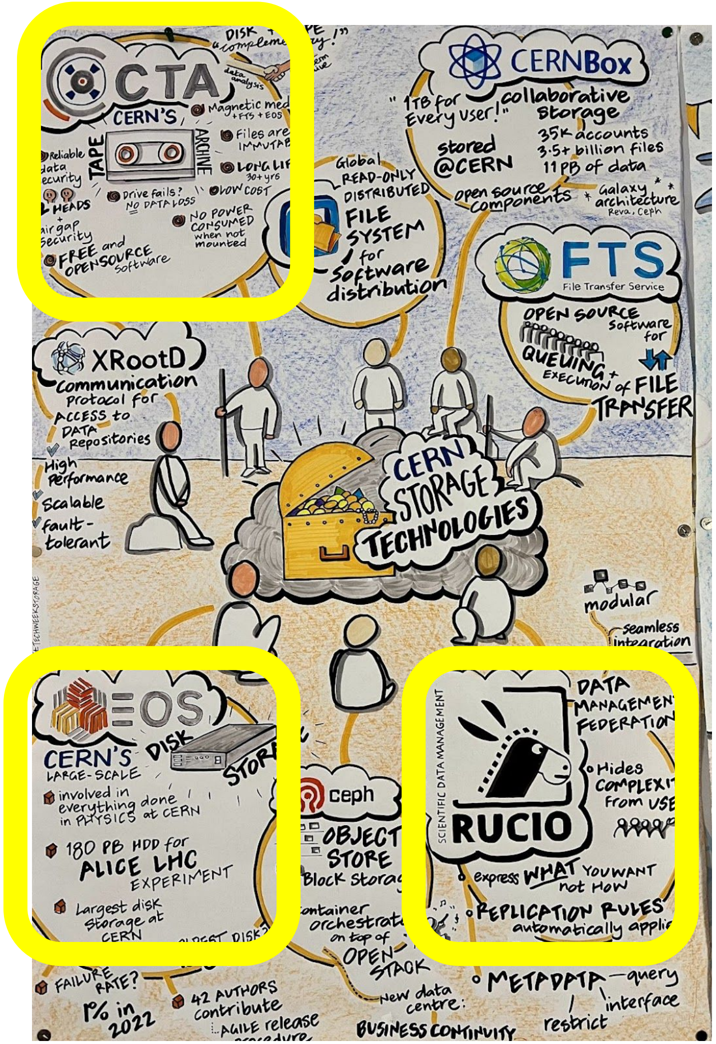
Advancing Large Scale Scientific Collaborations with Rucio

Hugo González Labrador on behalf of the Rucio project

CERN | IT Department

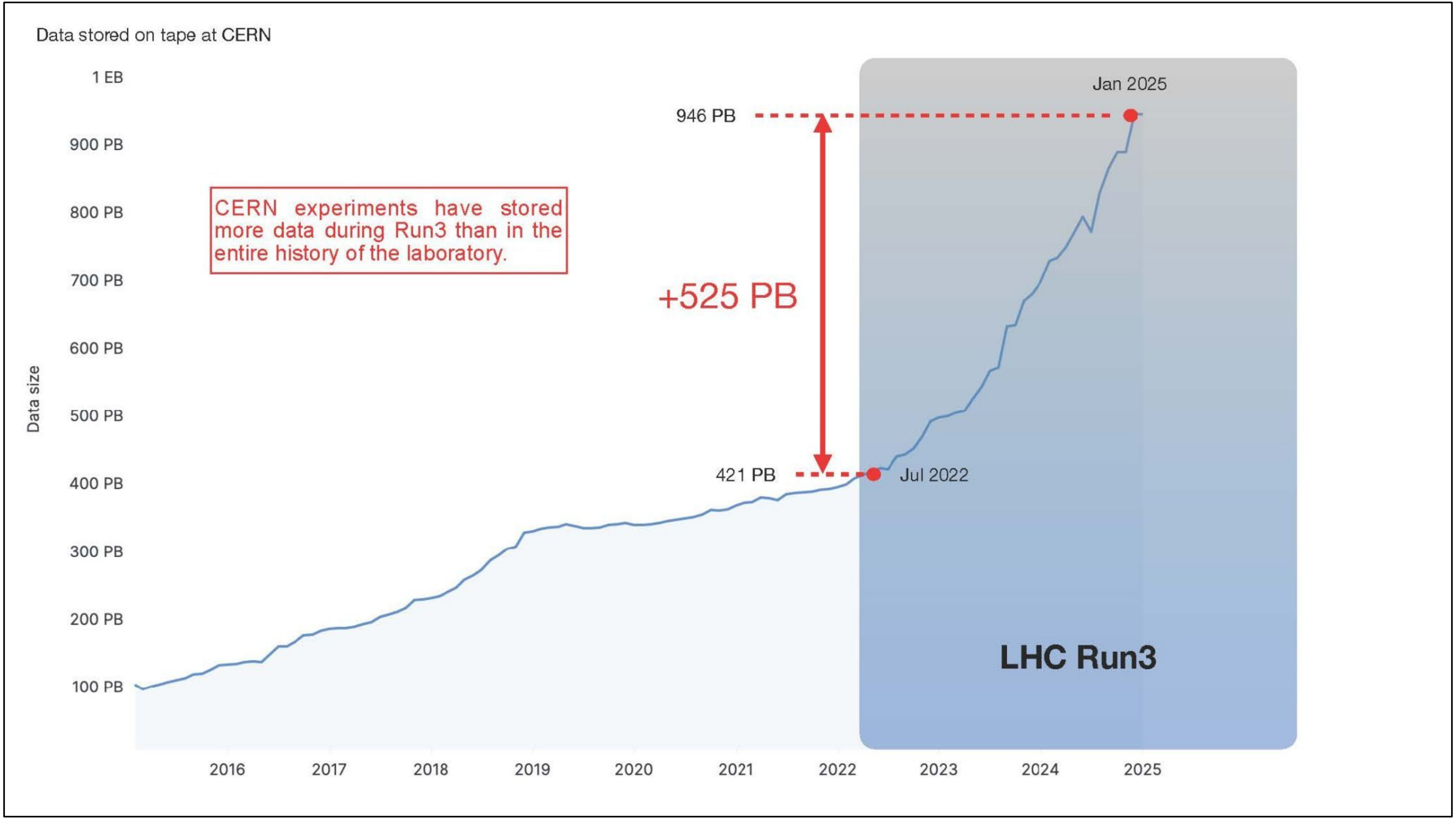
FOSDEM 2025| 1st February 2025



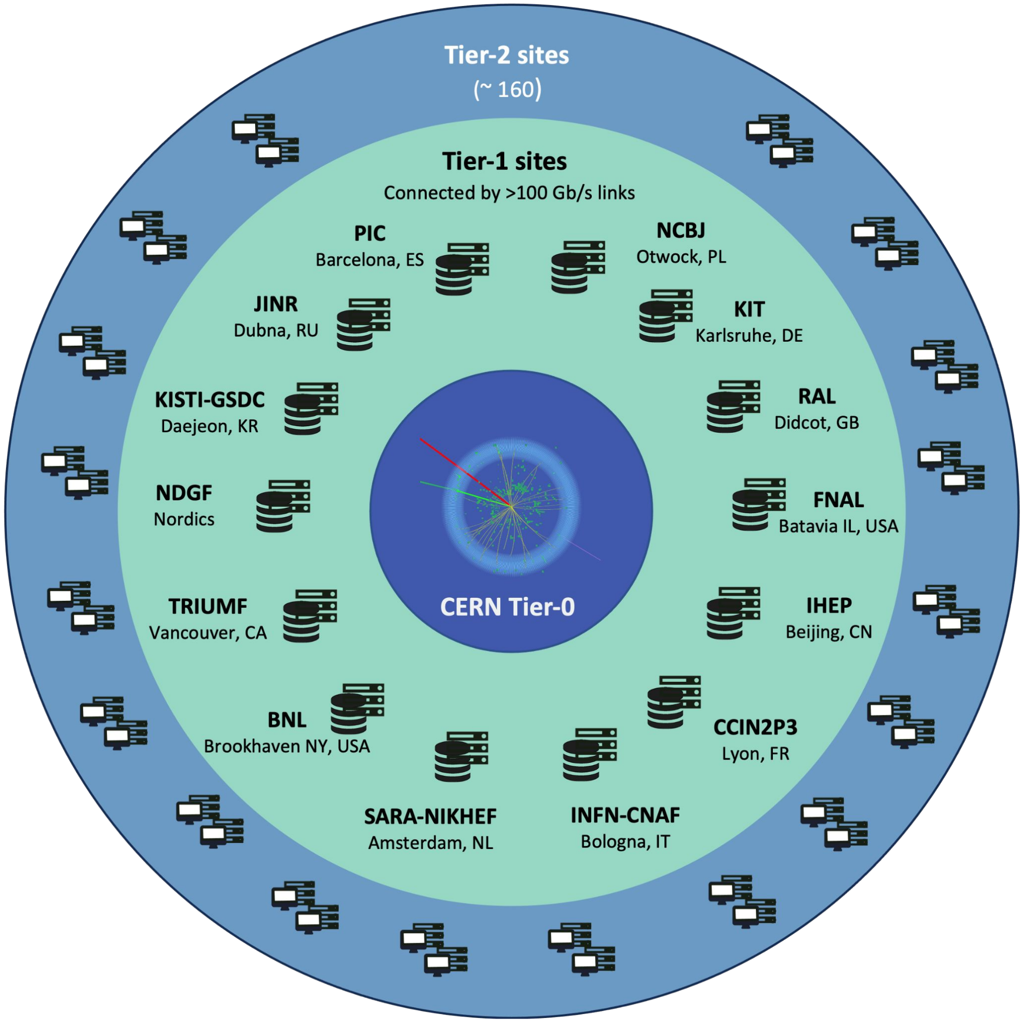
CERN develops many Open Source Storage Systems or contribute to upstream projects (Ceph, SAMBA, …)

Not time for all of them today

Focus on disk (EOS), tape (CTA) and **data distribution (Rucio/FTS)**


Data stored on tape at CERN
1 EB
900 PB
800 PB
700 PB
600 PB
500 PB
400 PB
300 PB
200 PB
100 PB
Data size
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
CERN experiments have stored more data during Run3 than in the entire history of the laboratory.
946 PB
Jan 2025
+525 PB
421 PB
Jul 2022
LHC Run3

Worldwide LHC Computing Grid (WLCG)



The largest computing grid in the world
42 countries
170 computing centres
Used by more than 40K people

User request

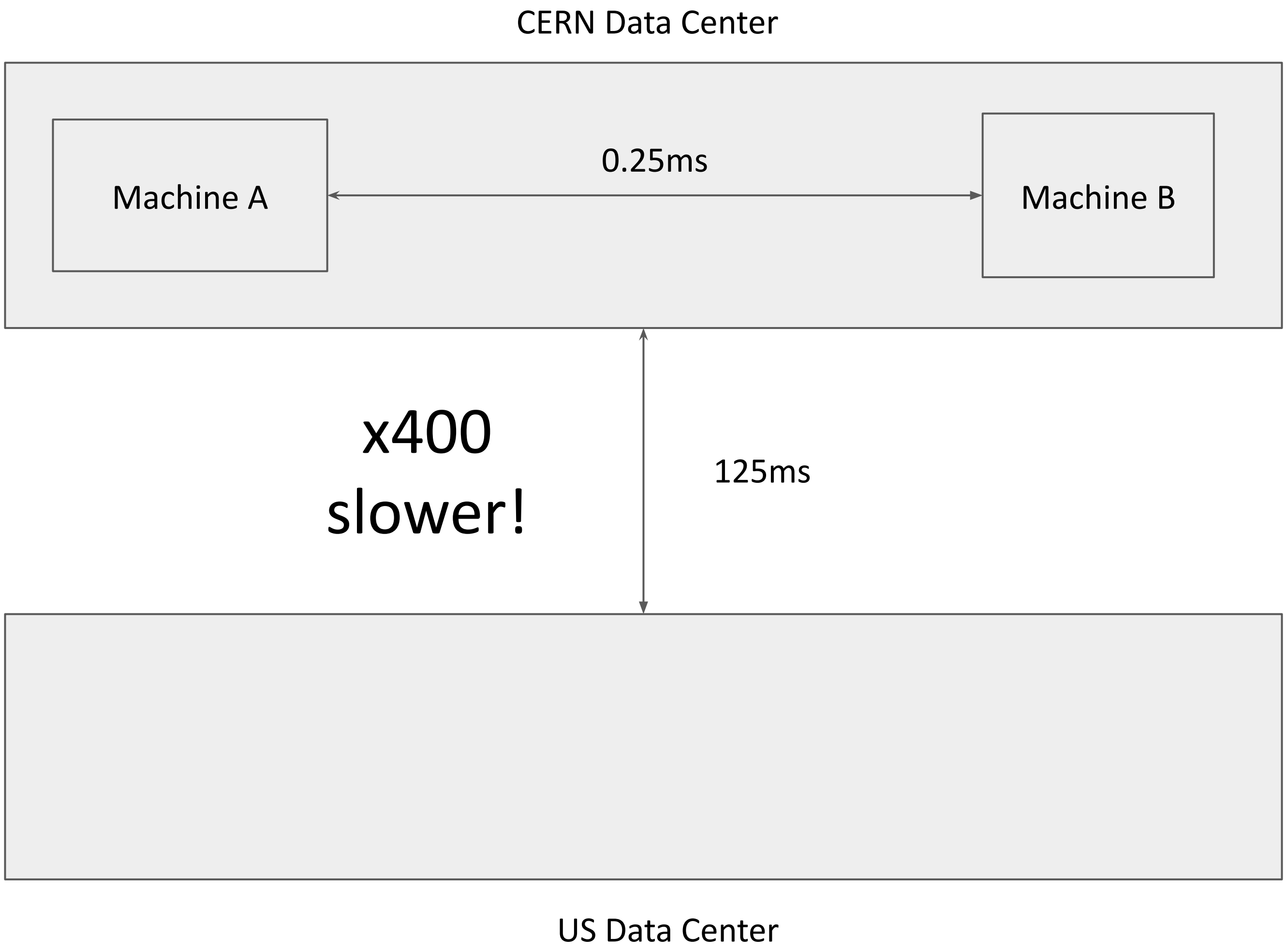
My analysis at CERN (T0) runs very fast but is very slow when doing it from a machine in the US?

Can you make your storage system faster?

What does it means to us?

Can you optimize a local POSIX IO workflow* to read/write remote data over shared transatlantic network link over FUSE?

* usually written by a physicist


CERN Data Center
Machine A
0.25ms
Machine B
x400
slower!
125ms
US Data Center

```
strace -c -e trace=read,newfstatat,write,open,close go run analyze.go
The word 'your' appears 0 times in the file.
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 65.86    0.012441          99       125        29 newfstatat
 24.81    0.004687          11       418         1 read
  9.32    0.001761           8       202           close
------ ----------- ----------- --------- --------- ----------------
100.00    0.018889          25       745        30 total
```

So …

The most performant way to run local POSIX workflows is to run them on metal (physical hw) or over fast networks (local network)

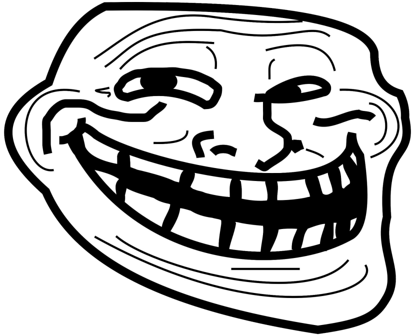
Meaning ..

We need a way to move the data from CERN to the other 170 computer centres across the globe

How?

Let’s scp 2 petabytes containing 1 billions file over a WAN link …

What can go wrong?



What can go wrong?

- Network failures
- Temporary transfer errors
- ISP blocking connecting from country X
- Did we transfer all the data?
- Did some data get corrupted in transit?
- Etc …

We need a robust and structured way to manage data at scale

Introducing Rucio

Rucio in a nutshell

CERN

Rucio is free and open-source software licenced under *Apache v2.0: github.com/rucio*

Rucio provides a mature and modular scientific **data management federation**

- **Seamless integration** of **scientific and commercial** storage and their network systems
- Data is stored in **global single namespace** and can contain **any potential payload**
- Facilities can be **distributed at multiple locations** belonging to **different administrative domains**
- Designed with **more than a decade of operational experience** in very large-scale data management

Rucio is location-aware and manages data in a heterogeneous distributed environment

- Creation, location, transfer, deletion, annotation, and access
- **Orchestration of dataflows** with both low-level and high-level policies

open source initiative Approved License

Open community-driven development process

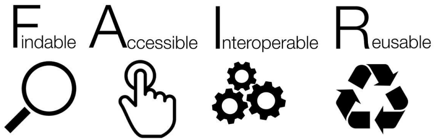
docker

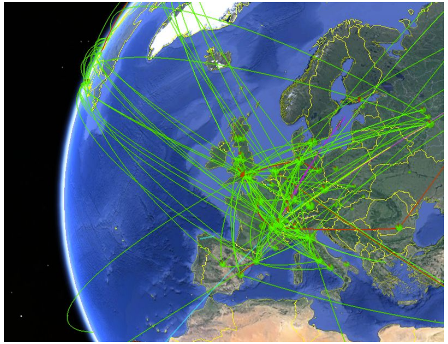
Rucio main functionalities

Provides many features that can be enabled selectively

More advanced features

- **Horizontally scalable catalog** for files, collections, and metadata
- Transfers between facilities including **disk, tapes, clouds, HPCs**
- **Authentication and authorisation** for users and groups
- **Many interfaces** available, including CLI, WebUI and REST API
- **Extensive monitoring** for all dataflows
- Expressive **policy engine** with rules, subscriptions, and quotas
- Automated **corruption identification and recovery**
- Transparent support for **multihop, caches, and CDN dataflows**
- **Data-analytics based flow control**



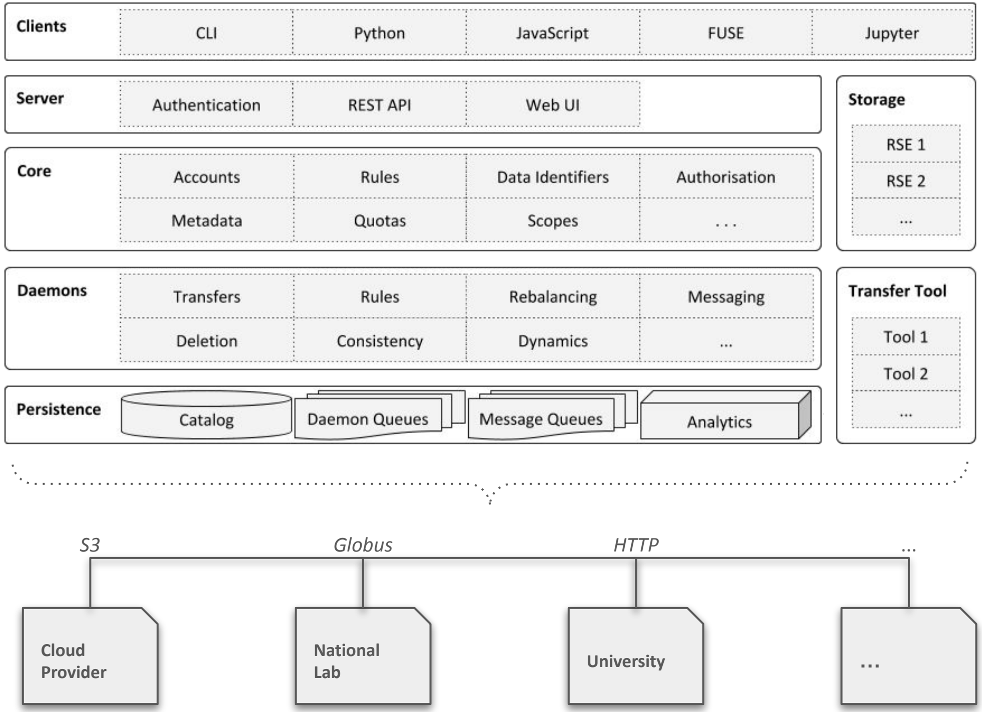


Rucio is not a distributed file system, it connects existing storage infrastructure over the network

No Rucio software needs to run at the data centres

Data centres are free to choose which storage system suits them best - avoids vendor lock-in

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)

Declarative data management

Express what you want, not how you want it

e.g., *"Three copies of this dataset, distributed across MULTIPLE CONTINENTS, with at least one copy on TAPE"*

e.g., *"One copy of this file ANYWHERE, as long as it is a very fast DISK"*

Replication rules

Rules can be **dynamically added and removed** by all users, some pending **authorisation**

Evaluation **engine resolves all rules** and tries to satisfy them by requesting transfers and deletions

Lock data against deletion in particular places for a given lifetime

Cached replicas are **dynamically created replicas** based on traced usage over time

Workflow system can drive rules automatically, e.g., **job to data flows** or vice-versa

Subscriptions

Automatically generate rules for newly registered data matching a **set of filters or metadata**

e.g., *"All derived products from this physics channel must have a copy on TAPE"*

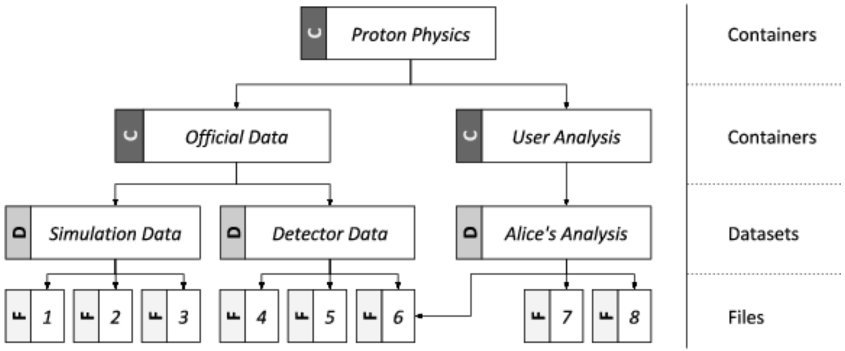
Rucio concepts - Namespace

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

There are different types of DIDs

- Files
- Datasets   Collection of files
- Container   Collection of dataset and/or container

Each DID is uniquely identified and composed of a scope and name, e.g.:



`detector_raw.run34:observation_123.root`

scope   name

Rucio concepts - Metadata

Rucio supports storage and querying of metadata

- Generic metadata that can be set by the users
- Up to the community to define the schema
- Searchable via name and metadata, aggregation based on metadata searches

Metadata interfaces

- Per default, generic metadata stored “within” Rucio (json data types)
- Metadata interfaces enable communities to connect other metadata backends (mongodb, science specific metadata stores, …)
- Metadata queries against Rucio are internally relayed to the matching backend and aggregated

Generic metadata can be restricted

- Enforcement possible by types and schemas
- Naming convention enforcement and automatic metadata extraction

Operations model

Objective was to minimise the amount of human intervention necessary

Large-scale and repetitive operational tasks can be automated

- Bulk migrating/deleting/rebalancing data across facilities at multiple institutions
- Popularity driven replication and deletion based on data access patterns
- Management of disk spaces and data lifetime
- Identification of lost data and automatic consistency recovery

Administrators at the sites are not operating any Rucio service

- Sites only operate their storage exposed via protocols (POSIX, ROOT, HTTP, WebDAV, S3, gsiftp, …)
- Users have transparent access to all data in a federated way

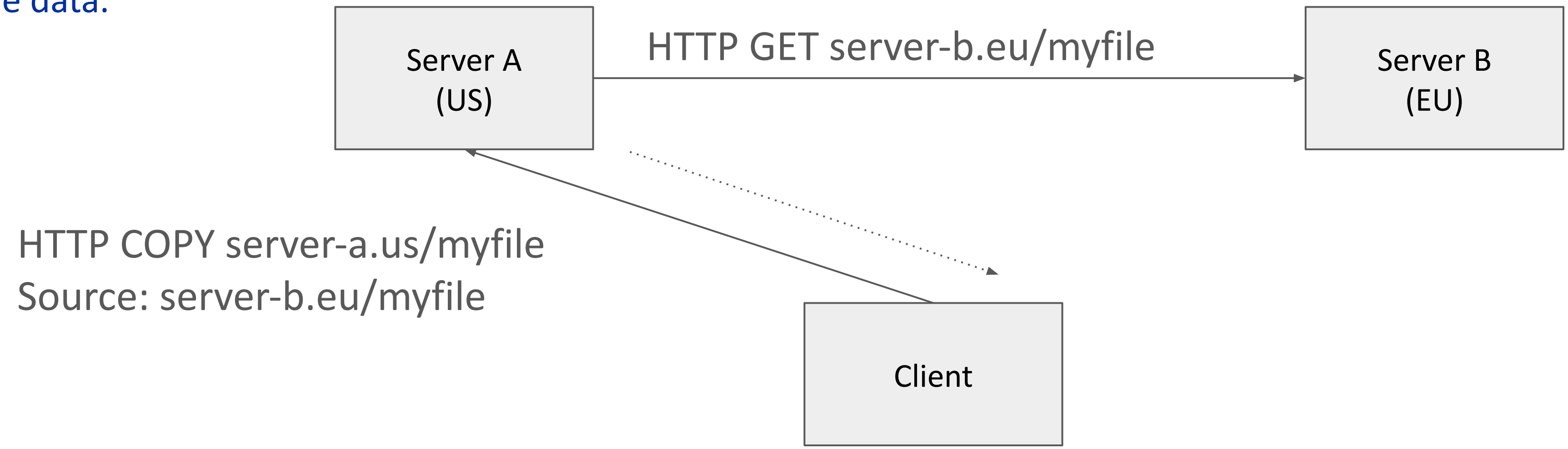
Easy to deploy

- PIP packages, Docker containers, helm-charts, Kubernetes

Transfers: Third Party Copy (TPC)

Rucio does not proxy data between servers, it relies on a tool named FTS3 (open source) to drive transfers point to point.

TPC is an extension to WebDAV that allows peer to peer pull/push data without middleware proxying the data.



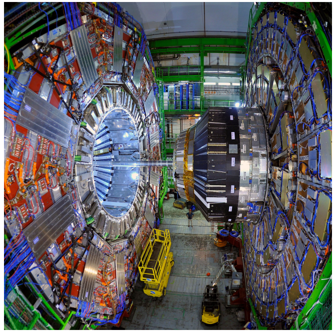
Concepts

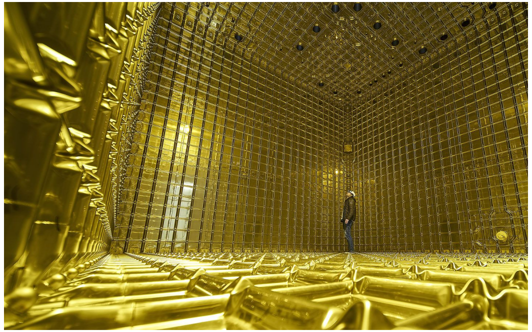
Rucio community experiences

Summary

Community experiences

- Rucio has become the de-facto standard for open scientific data management
  - Used by CERN-based experiments — AMS, ATLAS, CMS
  - And non-CERN experiments — Belle II, CTAO, LBNF/DUNE, SBN/ICARUS, KIS Solar, LIGO/VIRGO/KAGRA, SKA, Vera Rubin Observatory, XENON, …
  - Under evaluation by many others — EIC/ePIC, KM3NeT, …
  - Used by several EU projects — ESCAPE, InterTwin, DaFab, RI-SCALE





Summary

Rucio is an open, reliable, and efficient data management system

- Supporting the world's largest scientific experiments, but also a good match for smaller sciences
- Extended continuously for the growing needs and requirements of the sciences

Strong cooperation between physics and multiple other fields

- Diverse communities have joined, incl. astronomy, atmospheric, environmental, …
- Community-driven innovations to enlarge functionality and address common needs

Benefit from advances in both scientific computing and industry

- Lower the barriers-to-entry by keeping control of data in scientist hands
- Seamless integrations with scientific infrastructures and commercial entities
- Detailed monitoring capabilities and easy deployment have proven crucial

Additional information

Website	http://rucio.cern.ch
Documentation	https://rucio.cern.ch/documentation
Repository	https://github.com/rucio/
Images	https://hub.docker.com/r/rucio/
Online support	http://rucio.cern.ch/doc../join_rucio_mattermost/
Developer contact	rucio-dev@cern.ch
Journal article	https://doi.org/10.1007/s41781-019-0026-3
Twitter	https://twitter.com/RucioData