Highlight of the Joint Xrootd & FTS Workshop

HEPiX Fall 2024 Workshop

Horst Severini (Oklahoma University)
Wei Yang (SLAC)

The Workshop, 9-13 Sep. 2024

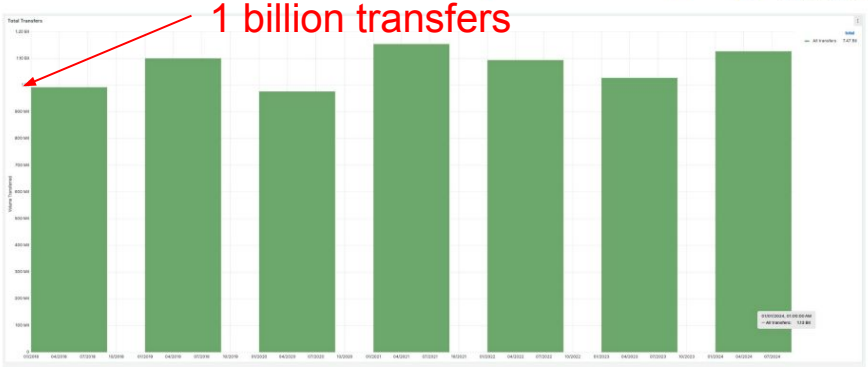
The 2nd Joint Xrootd and FTS Workshop

- https://indico.cern.ch/event/1386888/
- Thanks for the local organizers at STFC/RAL
  - The 1st joint workshop was at Ljubljana in 2023
  - Will be an annual workshop
- 1st 2.5day on FTS & 2nd 2.5 day on Xrootd
- 62 registered participates (including 5-10 remote participants)
- 51 presentations
- 1 white board session (on CERN DMC client)
- 1 live demo (Xrootd github/CI/CD)
- Reached way beyond LHC and HEP

Highlight on the FTS Session

The CERN FTS team is also responsible for the DMC clients (gFAL, Davix, etc.)

FTS - Number of Transfers (2018-2024)



FTS v3.13 is now available in Alma Linux 9

FTS & DC'24

The good

- FTS upheld its contract to saturate StorageEndpoints (according to configured limits)
- FTS successfully pioneered WLCG transfers with token support in a high stress environment

The bad

- Saturating StorageEndpoints does not correlate with maximizing throughput
- No way to prioritize faster links (T0→T1) over slower ones (T0→T2)

FTS & DC'24

The ugly

- Problems with congested database queries
  (→ token queries further refined post-DC'24)
- FTS3-ATLAS instance overloaded
  (→ must impose limits and refuse submissions)
- FTS3-ATLAS Optimizer malfunctioned
  - Optimizer would never finish a cycle through the database
  - No link decision would ever be updated

FTS and Tokens

DC 24 is a perfect time to test the WLCG token

- Experiments uses tokens in different ways.
- Both ATLAS and CMS see scalability issues
- This issue is related to how tokens are used.
  - How large a FTS (and DB) deployment do we need?
  - How large an IAM cluster do we need?

In X509 world, this is O(10)

ATLAS Token Testing

- Long-lived (9 days) per-file (`storage.modify:<full-path>`) tokens
- FTS will not manage the lifecycle for these tokens (lacking `offline_access` scope)
  - FTS won't request a refresh token (token-exchange), nor try to refresh them
- Details in Dimitrios' presentation

- Small amount of development needed to skip token lifecycle management
- No operational concerns or effects on production traffic (according to FTS monitoring)
- Sporadic measurements showed up to 600k access tokens in the FTS database

(→need for constant monitoring of # of tokens in database)

CMS Token Testing

- Pragmatic combination of audiences + scopes per dataset
- FTS will perform full token lifecycle management
- Details in Rahul's presentation

- Slightly more involved in debugging certain transfer failures
  → uncovered certain shortcomings once tokens are considered "no longer used"
  → would only address with the just-in-time refresh
- Large token submission "incident" (28k tokens for exchange vs average of ~20-50 / min)
  → behaved surprisingly well, but only ran at 10Hz (50 threads on FTS side) / ~45m
- Sporadic measurements showed up to 100k tokens in the database

Token Reflections

- Tokens are more secure
  - Tokens will leak, no matter what *(FTS blamed in the past for EGI-SVG-2024-02)*
  - Are we ready to mitigate this? (i.e.: how wide should the scope be?)
- Tokens will simplify things
  - ATLAS & CMS take different approaches
  - FTS had to integrate with: INDIGO-IAM, EGI CheckIn, CILogon
  - No guarantee a future TokenProvider will work out of the box *(in fact, not likely)*
- Tokens are an industry standard
  - Perhaps, but is refreshing the right way to go?

  (operational experience shows refresh tokens are difficult to deal with)

- Tokens are flexible
  - Too much room also for interpretation *(why can't we have a VO field?)*
  - Profile definitions can change, but software cannot as easily

This is what was promised

This is what we actually see

Token scalability is not just the responsibility of FTS and IAM (and other middle wares)

- Experiments need to think of realistic ways of using tokens
- Not just following the token "ideologies".

FTS 4 - the future version, and major changes

The plan – The steps to get to “FTS4”

1) Move from MySQL to PostgreSQL ← **This is the top priority**

2) Implement a dedicated FTS scheduler daemon in Python

3) Use DB in a scalable way

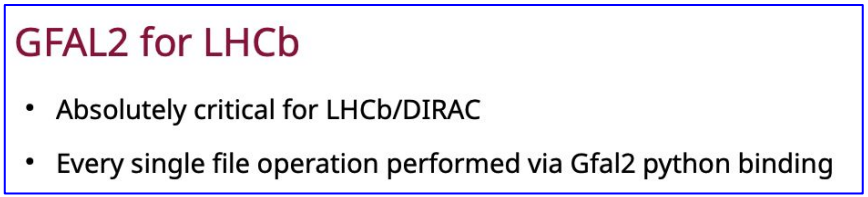
4) Add new functionalities

5) Migrate as much code as possible to Python and reduce C++ code

CERN Data Management Client (DMC) Update

Whiteboard session on gFAL and Davix

- gFAL
  - gFAL is an important tool for all experiments and should continue to be supported



  - Interface (CLI and Python API) are important and should not change
    - Not sure about C++ API (who use it?)
  - Protocol support: want to keep HTTP and Xroot protocols, and drop all others
  - Want to change the under the hood implementation, to make it more supportable
    - make gFAL a thin layer and directly using Davix (for HTTP) and Xrootd tools and APIs.
    - Discussed possible routes, for example, implement via fsspec?
- Davix
  - Both API and packaging makes it convenient to use.
  - Will make Davix 1st class citizen in DMC.
  - Will clean the code, keep S3 and Webdav support, drop unneeded stuffs.

Highlight of the Xrootd Session

What's Happened Since 2023

- 27-March-23 **XRootD** Workshop @ JSI
  - 08-May-23 Patch Release 5.5.5
  - **30-June-23** **Feature Release 5.6.0**
  - 11-July-23 Patch Release 5.6.1
  - 15-September-23 Patch Release 5.6.2
  - 27-October-23 Patch Release 5.6.3
  - 11-December-23 Patch Release 5.6.4
  - 22-January-24 Patch Release 5.6.5
  - 25-January-24 Patch Release 5.6.6 (Oops)
  - 06-February-24 Patch Release 5.6.7
  - 23-February-24 Patch Release 5.6.8
  - 08-March-24 Patch Release 5.6.9
  - **01-July-24** **Feature Release 5.7.0**
  - **04-July-24** **Patch Release 5.7.1**
- 09-September-24 **XRootD** Workshop @ RAL

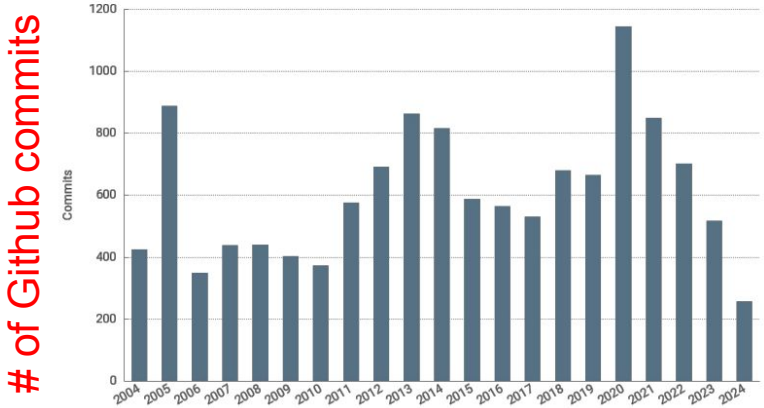
The largest number of patch releases for a feature release

SLAC NATIONAL ACCELERATOR LABORATORY

XRootD in HEP Community

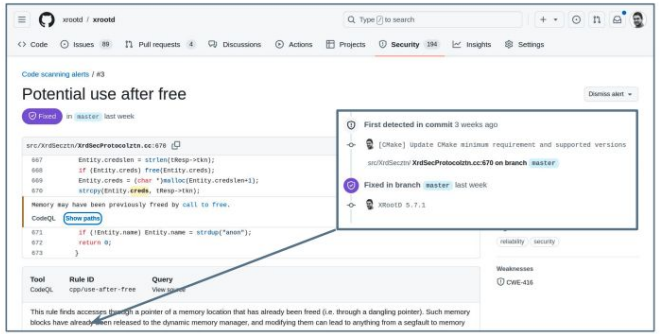
- Core component of HEP software ecosystem
  - Depended on by CTA, FTS, EOS, ROOT, Rucio, experiment frameworks, etc
- Exabytes of data processed each year

20 Years of XRootD Commit Activity



Active in GitHub and heavily utilized GitHub

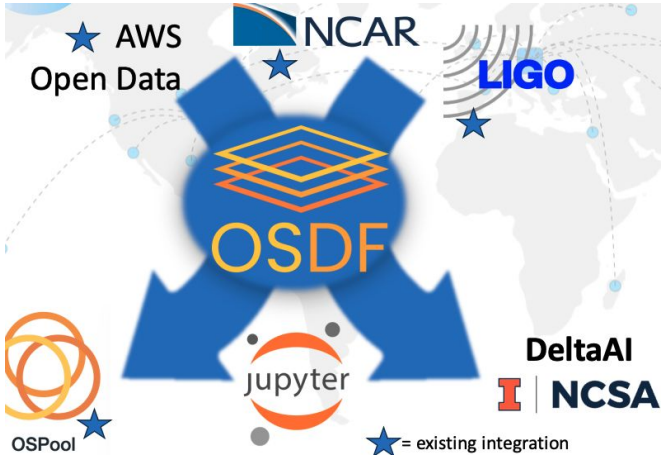
Code Scanning with CodeQL on GitHub



Highlight of New Features in Xrootd (release 6.0)

1. Rucio aware dataset backup plugin
2. Improved error message propagation (especially for the curl error reporting)
3. Kernel level TLS
4. RDMA support in the Xroot protocol
5. Improvement on monitoring
6. Drop support of CentOS 7
7. Drop support of python2
8. Plus lots of improvements in Xcache (incremental, not waiting for release 6)

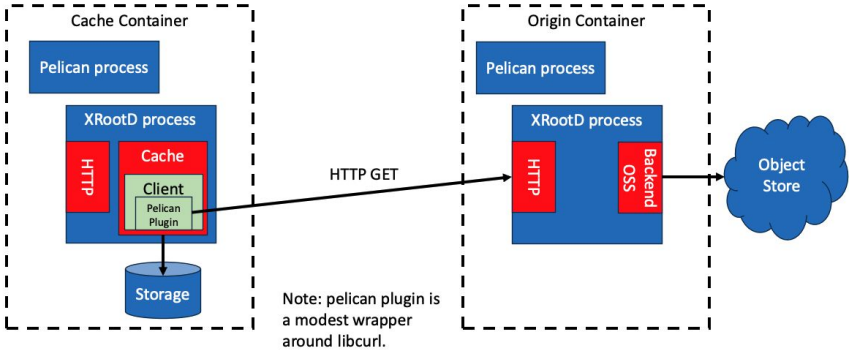
OSDF, Pelican and Xrootd



Pelican platform is the technology that supports OSDF

Centered around OSDF operated regional caches (Xcache with HTTP protocol)

A slide for the XRootD people out there…



OSDF long term vision: an “all-science” Content Delivery Network.

Pelican and non-Posix storage

Pelican is developing support for several HTTP type storages

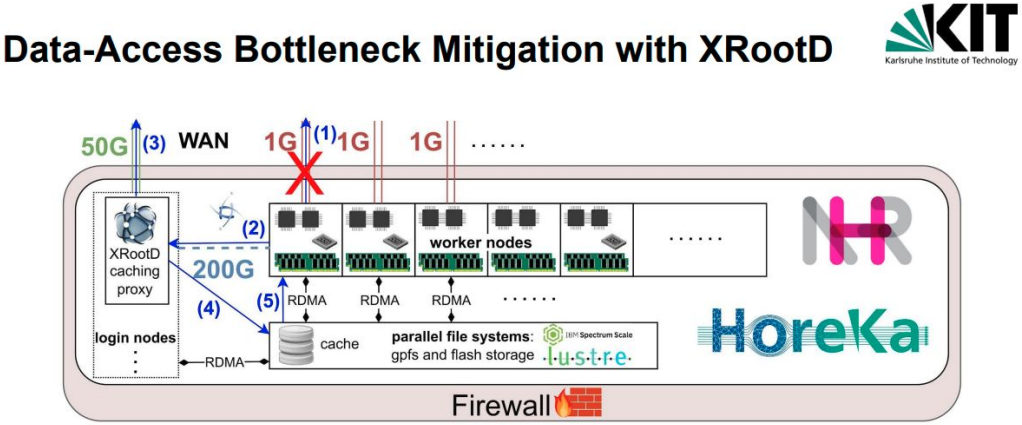
- Via Xrootd storage plugin or remote client plugin
  - This is different from the existing XrdClHttp in Xrootd (to support S3 storage backend).
  - Also different from XrdCEPH in Xrootd (a RAL contribution)
- Support several HTTP/S3/Globus type storages, tailed for Pelican's (caching) need
  - Current phase focus on readonly to storage
  - Support for Globus is interesting:
    - Put a Globus endpoint behind Xrootd. This is an alternative way to integrate Globus into the WLCG Rucio/FTS/Xrootd/dCache ecosystem.
    - Does not depend on the integration of Globus service and Rucio

Xrootd and Xcache in German HPCs

Transition from German dedicated T2 resources at universities to shares on national HPC centers

Caching proxy at KIT's login node:

- It provide sufficient bandwidth (50Gbps) to the outside
- IPoIB to distribute data to worker nodes
- Looking forward for Xroot protocol over RDMA



Other Xrootd Community Effort and R&D

- Xrootd and CEPH at UK
  - RAL Tier 1 Echo system: CEPH core storage. Integrate with WLCG via Xrootd
  - UK Tier 2s: CephFS + Xrootd is a popular model.
    - Also used by the University of Oklahoma ATLAS Tier 2
  - Xcache may be used for diskless sites (and ATLAS Virtual Placement)
- StorageD@RAL to support Light Source and Environmental Science communities
- GridPP effort on building Xrootd testing environment in Kubernetes for HEP and AstroPhysics.
- Alternative load balancing algorithm.
- Optimization of Xrootd's Posix vector read to RAL CEPH object store
- (HTTP web) server side processing of ROOT files
- Monitoring, and more monitoring…