CERN
Tape Archive

CTA Day at the 7th EOS Workshop Challenges and Roadmap

Dr. Michael Davis for the CTA Team

CERN, IT Department
Storage and Data Management Group
Tape Archive and Backups Section

CTA Service : Plans for 2023

- Global HTTP rollout
  - HTTP transfers for archival
  - HTTP REST API for recall
  - ATLAS switching from XRootD to HTTP/WebDAV
- Deployment of XRootD 5 and EOS 5
- Full Run–3 production workload
- Full-chain data management (Rucio+FTS+EOS+CTA)
  - Archival: protecting archival of raw data
  - Retrieval: backpressure and disk buffer management

CTA Software : Building and Packaging

- Separate CTA Catalogue shared libraries
  - Oracle
  - PostgreSQL
- Refactor CI/CD scripts
  - Bash → Python
  - Common code for system tests
- Improve static analysis (Sonarcloud)
- Improve upgrade procedure between CTA Public releases
  - DB schema changes

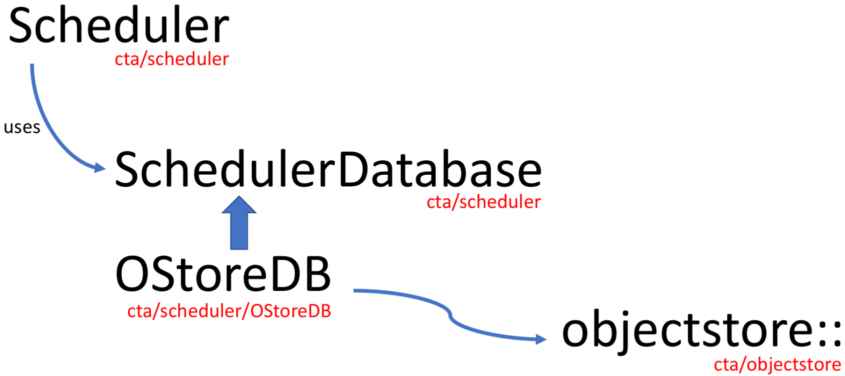
CTA Software : Scheduler Database

- The CTA Scheduler controls the workflow and lifecycle of Archive, Retrieve and Repack requests
  - Enqueue requests in the Frontend
  - Select next tape to mount
  - Data transfers to/from tape (pop batch from queue, RAO, …)
  - Error handling and retries
  - Reporting of success or failure
- The transient data on which the Scheduler works is stored in the Scheduler Database
- The current Scheduler Database implementation is Ceph RADOS Objectstore

Scheduler Database : Objectstore

- Efficient and works well for FIFO queuing operations (archive/retrieve)
- Requires workarounds for non-FIFO operations (delete, priority queues)
- Limitations of the objectstore
  - Constraint on CTA software development
  - Operational issues: difficult to change schema, trace problems, clean up
  - Additional software dependency
  - Additional technology for new team members to learn

Replacing the SchedulerDB component



Scheduler

cta/scheduler

uses

SchedulerDatabase

cta/scheduler

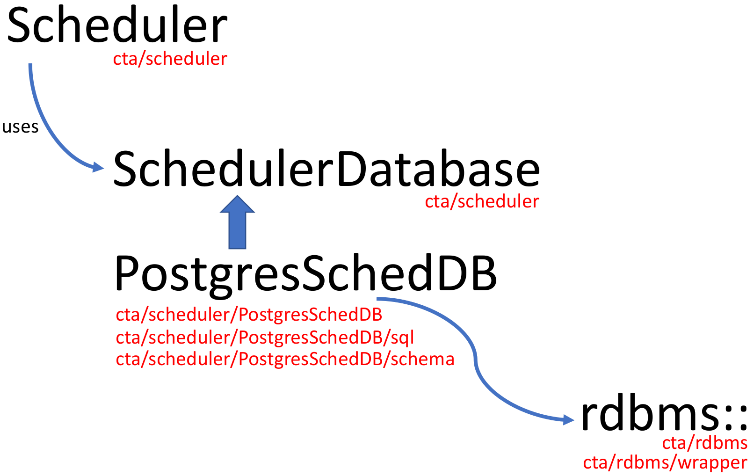
OStoreDB

cta/scheduler/OStoreDB

objectstore::

cta/objectstore

Replacing the SchedulerDB component



Scheduler

cta/scheduler

uses

SchedulerDatabase

cta/scheduler

PostgresSchedDB

cta/scheduler/PostgresSchedDB

cta/scheduler/PostgresSchedDB/sql

cta/scheduler/PostgresSchedDB/schema

rdbms::

cta/rdbms

cta/rdbms/wrapper

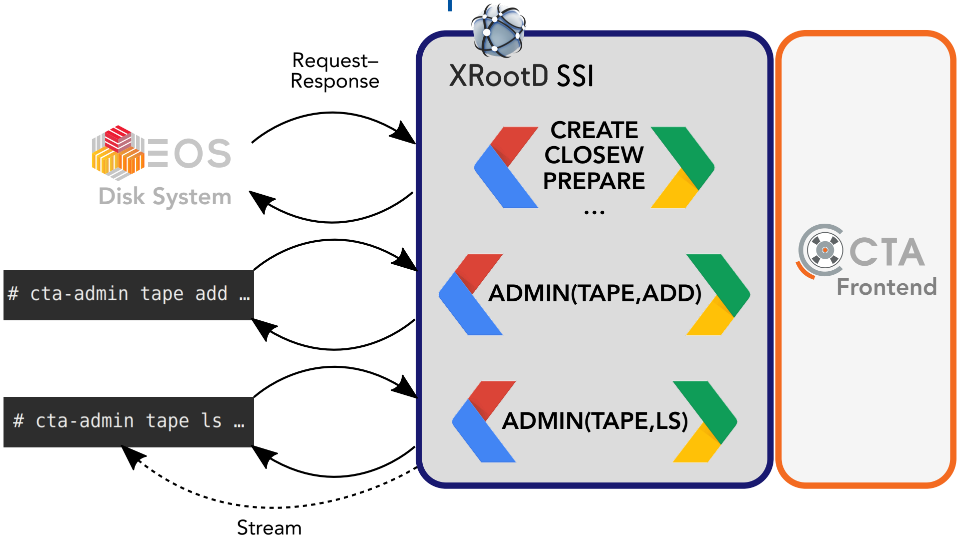
Postgres Scheduler Database Status

- Cleaned up objectstore references outside the SchedulerDB code
- Removed non-queuing operations (Drive Status) from objectstore to the CTA Catalogue DB
- Created PostgreSQL tables and views to replace objectstore request/queue objects
  - `Archive_Job_Queue`, `Retrieve_Job_Queue`
  - `Archive_Job_Summary`, `Retrieve_Job_Summary`

Postgres Scheduler Database Roadmap

- New PostgresSchedDB class to replace OStoreDB class
  - Archive methods mostly done
  - Retrieve methods in progress
- Additional functionality To Do
  - Repack
  - Reporting
- Goal is to begin testing in 2H 2023
- Repack as initial production use case

CTA Frontend Transport Protocol



Request–Response

EOS Disk System

```
# cta-admin tape add …
```

```
# cta-admin tape ls …
```

Stream

XRootD SSI

CREATE
CLOSEW
PREPARE
…

ADMIN(TAPE,ADD)

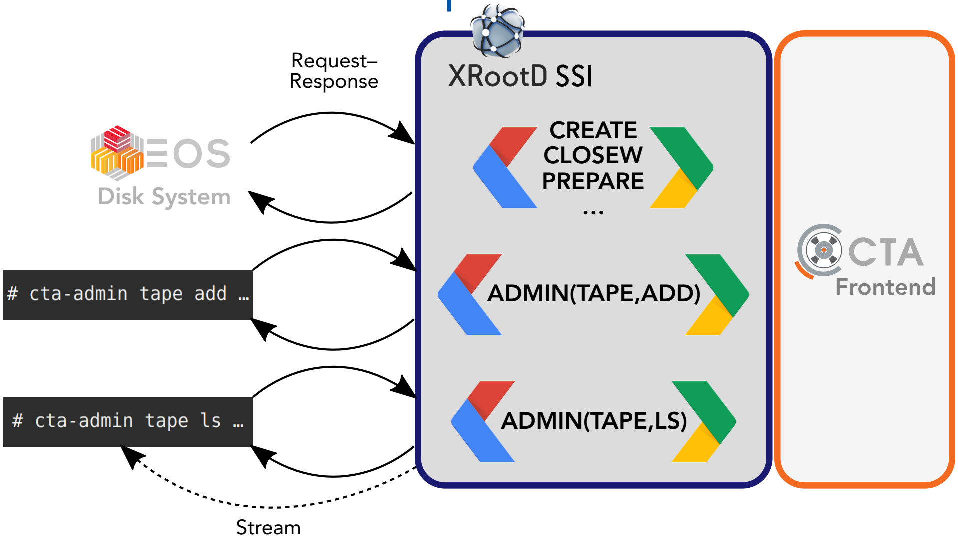
ADMIN(TAPE,LS)

CTA Frontend

CTA Frontend Transport Protocol

- Client request messages to CTA Frontend are serialised in Google Protocol Buffers
- Transport protocol is XRootD Scalable Service Interface (SSI). This works well, but:
  - SSI extensions not supported by dCache client
  - SSI not widely used; additional (non-standard) dependency
  - gRPC is the native transport for protobuf
- gRPC Frontend implementation/proof-of-concept contributed by dCache team

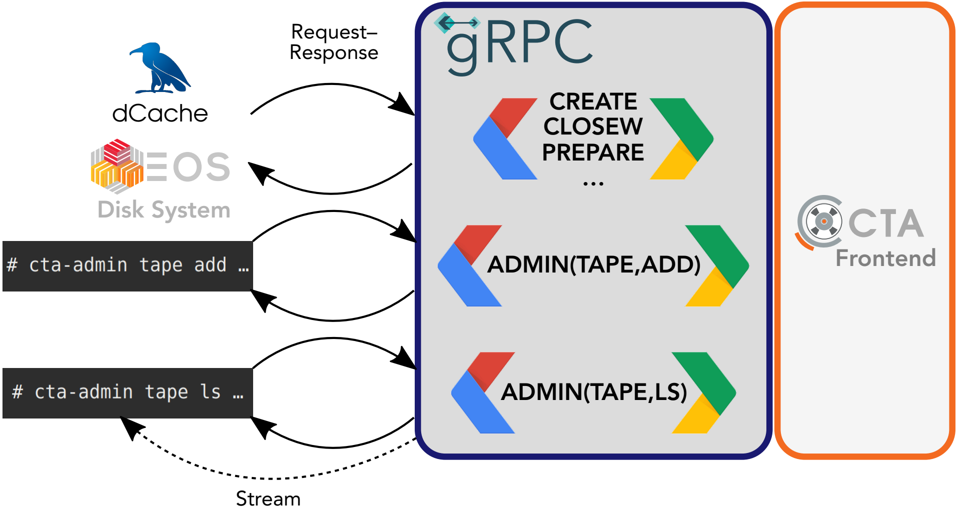
CTA Frontend Transport Protocol



```
# cta-admin tape add …
```

```
# cta-admin tape ls …
```

CTA Frontend Transport Protocol



Request–Response

dCache

EOS Disk System

```
# cta-admin tape add …
```

```
# cta-admin tape ls …
```

Stream

gRPC

CREATE
CLOSEW
PREPARE
…

ADMIN(TAPE,ADD)

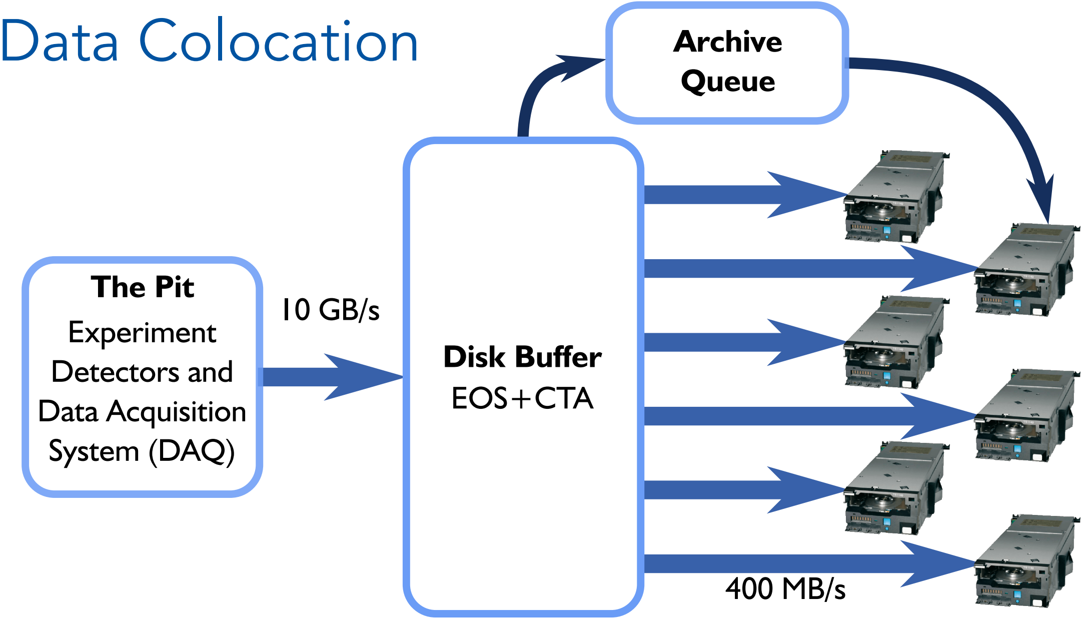
ADMIN(TAPE,LS)

CTA Frontend

gRPC Frontend Roadmap (3Q 2023)

- Disk workflow events are implemented
- Refactor SSI and gRPC Frontend implementations to share request message processing code
- Complete implementation of "streaming" admin commands: `cta-admin … ls`
- Ensure authentication (gRPC token, Kerberos) works as expected
- Add system tests in CI

Data Colocation



Archive Queue

The Pit

Experiment Detectors and Data Acquisition System (DAQ)

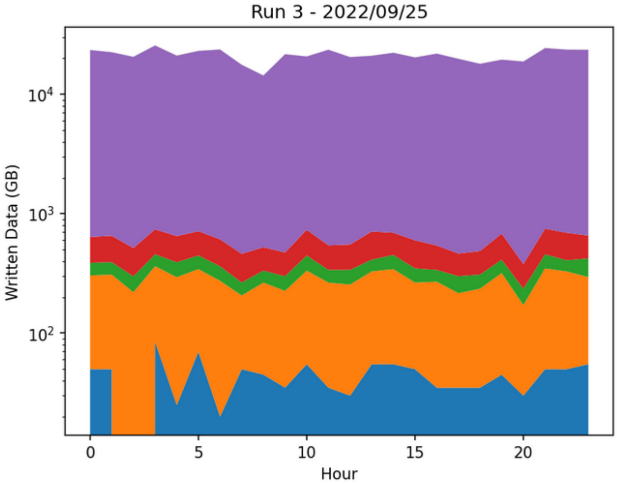
10 GB/s

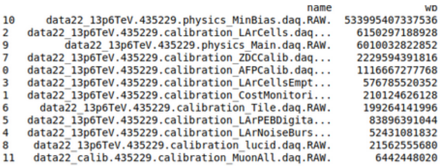
Disk Buffer

EOS+CTA

400 MB/s

Data Colocation : Multiple Streams





```
                                           name               wb
10     data22_13p6TeV.435229.physics_MinBias.daq.RAW.  533995407337536
2   data22_13p6TeV.435229.calibration_LArCells.daq...    6150297188928
9         data22_13p6TeV.435229.physics_Main.daq.RAW.    6010032822852
7   data22_13p6TeV.435229.calibration_ZDCCalib.daq...    2229594391816
0   data22_13p6TeV.435229.calibration_AFPCalib.daq...    1116667277768
3   data22_13p6TeV.435229.calibration_LArCellsEmpt...     576785520352
1   data22_13p6TeV.435229.calibration_CostMonitori...     210124626128
6     data22_13p6TeV.435229.calibration_Tile.daq.RAW.     199264141996
5   data22_13p6TeV.435229.calibration_LArPEBDigita...      83896391044
4   data22_13p6TeV.435229.calibration_LArNoiseBurs...      52431081832
8    data22_13p6TeV.435229.calibration_lucid.daq.RAW.      21562555680
11   data22_calib.435229.calibration_MuonAll.daq.RAW.       6442448020
```

Raw dataset:
1.8 billion events, 1.3 PB

12 data streams in parallel

Data Colocation : The Problem

- CTA workflow is optimised for efficient archival of raw data coming from the detectors
- **BUT** this results in organisation on tape which is not optimal for recall:
  - Multiple different streams intermixed on the same tape
  - Individual streams scattered across many tapes (dataset fragmentation)
  - Problem exacerbated during tape repacking operations

Data Colocation: HTTP Archive Metadata

- *activity* and *priority* are for scheduling, not data colocation
  - *activity* is a share name for external scheduler (Rucio/FTS)
  - *priority* defines latency within the *activity* lane
- New *archive metadata* for data colocation
  - Files tagged with the dataset they belong to
- Measure the problem
  - Analyse dataset fragmentation across multiple tapes
  - Monitoring and analysis of tape mounts for recalls
- Use archive metadata as a data colocation hint

CBACK : CERN Backup project

- Developed to backup CERNBox and CephFS to CERN S3 disk
  - Uses open-source Restic backup software
  - Provides CLI/REST API interface for backup agents
- Extend CBACK to offload data to cold storage (tape)
  - Reduce size of the online CBACK storage pool
  - Immutable source suitable for disaster recovery
- Compressed Restic archives are stored in CTA
  - Transparent to CTA: Backup archives are like any other file
  - CBACK keeps track of whether data is hot (disk) or cold (tape)

CBACK Status and Roadmap

- Restic hot/cold metadata interface debugged
- CBACK archiver agent implemented
- CBACK-created Restic archives can be stored and retrieved from CTA
- Next steps
  - CBACK performance benchmarking
  - Restic optimisations; full/incremental archiving

CTA Roadmap : Summary

- Important development tasks for 2023
  - New SchedulerDB back-end
  - gRPC Frontend
- In the pipeline
  - Archive metadata, data colocation R&D
  - Backup to CTA with CBACK/Restic
- CTA Website : Source Code, Documentation, Presentations and Publications
- CTA Community on Discourse

home.cern