CERN Tape Archive Run 3 Production Experience

CTA Tier-0 service performance during the start of the LHC Run 3 and the various lessons learnt

Julien Leduc[1], *João* Afonso[1], *Richard* Bachmann[1], *Vladimír* Bahyl[1], *Jorge* Camarero Vera[1], *Michael* Davis[1], *Pablo* Oliver Cortés[1], *Fons* Rademakers[1], *Lasse* Wardenær[1], and *Volodymyr* Yurchenko[1]

[1]CERN, Esplanade des Particules 1, 1211 Geneva 23, Switzerland

Abstract. The *EOS disk + CERN Tape Archive* (EOSCTA) service is CERN's primary physics data long-term storage and archival solution for LHC Run 3. It entered production at CERN during summer 2020 and has since been serving all the LHC and non-LHC workflows involving archival to- and retrieval from tape.

The CTA system is a complete redesign of the previous tape software, tape cache and tape workflows, which will need to scale to the data rate requirements of the present LHC activity period, as well as the one after. At the time of writing it has already set new records for monthly tape archival volume at CERN and reached write efficiencies equalling those demonstrated during earlier data challenges.

1 Introduction

The EOSCTA service at CERN[1] provides data archival storage for all collaborating experiments. The infrastructure must be able to cope with indefinite archival of the new data, while at the same time also supporting frequent massive recall campaigns.

An EOSCTA deployment is composed of at least one instance of an EOS disk system in combination with the CTA tape system. EOS is a disk-based storage software and acts as CERN's strategic data storage platform for all user-facing namespaces hosted by the IT Storage Group. CTA is a pure tape-based storage system, which handles interactions with the physical tape hardware in order to open tape as a strategic archive medium which delivers cost efficient long-term capacity.

Early in LHC Run 3[1], we optimised various operations processes in order to limit their impact on production. These include tasks such as consolidating testing procedures and deployment of new CTA software using the *Rundeck*[2] job automation platform. We also modified the software in order to have it distinguish between user and operator accesses to the data on tapes, with separate logic for both cases, which enabled the development of an automatic data repacking [2] engine. Lastly, we extended support for external CTA community by publishing an open source suite of operational tools which supplements the core software.

[1] Run 3 refers to the third cycle of Large Hadron Collider activity, where active periods are separated by a period dedicated to upgrades. Each cycle spans multiple years.

[2] Repack is the CERN/CTA term for transferring data from one tape cartridge to another. There may be many motivations for doing this, such as moving data to a newer media type, creating replicas of the files in question, or recovering data from corrupted media. Repacking an intact tape allows it to be re-used for future write operations.

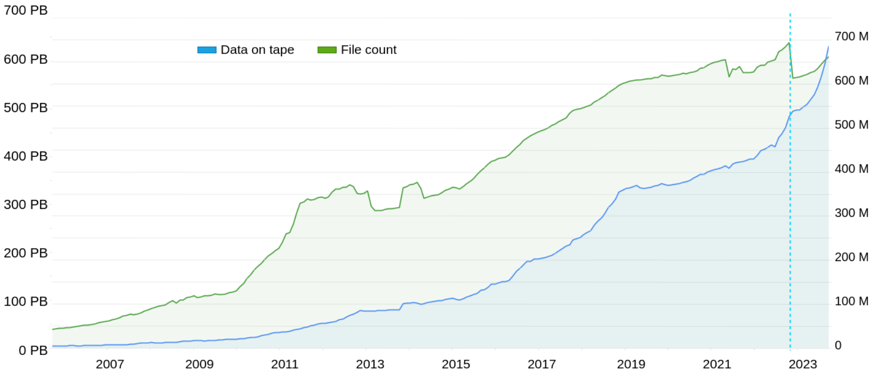

Figure 1. Tier-0 data storage evolution over the past 18 year. The dotted blue vertical line marks the CASTOR service's end of life and the completed migration of data to CTA.

As of May 2023, the amount of data stored in the archive reached around 513 PB with a peak of over 25 PB of new data arriving in one month. The archive capacity is expected to grow exponentially as new data significantly overtakes the existing quantities from earlier eras.

2 Run 3 tape infrastructure

2.1 Tape hardware

For Run 3, most of the existing IBM enterprise tape technology was upgraded to the latest library and drive models. Additionally, we have replaced an obsolete IBM TS3500 enterprise tape library (having 15000 cartridge slots), with the new 22000 slot IBM TS4500 model, which is *Linear Tape Open* (LTO)-based. The Oracle StorageTek enterprise tape technology used during Run 2 was completely replaced with an LTO-based solution. This approach was taken as a consequence of market evolution and with the intention of maintaining diversification as a safeguard against unexpected (technical, financial or other) issues.

In total, the number of installed data cartridges increased from 30 000 to around 40 000, in order to accomodate 150 PB of new data per year. By virtue of the increase of per-cartridge data capacity, we were still able to shrink the number of tape libraries from 7 (used during Run 2) to 5 in total. On the other hand, the total number of tape drives has more than doubled to around 180 (LTO9 and IBM TS1160 combined), such that we can accomodate the expected incoming data rates of up to 40 GB/s. This hardware infrastructure setup proved sufficient for the start of Run 3, offering one redundant library on top of CTA SLAs[3] while still leaving room for further upgrades if necessary.

A complete inventory of the Run 3 tape hardware setup at CERN is described in a dedicated presentation[3].

2.2 Dedicated tape buffer for efficiency

The CTA software architecture is designed to allow us to focus on tape-specifics, such as streamlining data paths, hardware interaction and infrastructure. However, operating tape drives at full speed full efficiently requires a disk buffer on the data path, ideally one which is

[3] Service Level Agreement, a contract between a service provider and a customer, defining the types and standards of services to be offered.

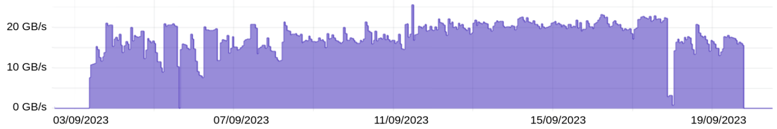

Figure 2. Best effort[4]archive rates for the ALICE experiment in September 2023. The ALICE experiment is archiving over the 10GB/s SLA in Best Effort mode just before the Heavy Ion (HI) Run: Up to 73 tape drives were running to reach a sustained peak performance of 21GB/s (24PB archived in 17 days).

SSD-based. For this purpose an EOS instance is used to manage the disk storage dedicated to each experiment. It is this instance which also provides the user-facing file namespace.

A file is only considered as safely stored in CTA when it has been written on tape, not while it is only present in the disk buffer. Write operation data integrity at the final step is ensured by recomputing the checksum from the tape drives and comparing it with the checksum originally sent by the user. Following this rule, intermediate data integrity validation in the tape buffer is not necessary anymore. As the service does not have to ensure it, the tape buffer does not need more than one replica per file, allowing to further improve overall tape buffer efficiency.

This rule was a decisive design change for Run 3 and was integrated in FTS[4] code as *checkOnTape*[5]: When a user set an *archive_timeout* value for a file transfer to a tape storage endpoint, FTS waits up to *archive_timeout* seconds for the file to be safe on tape otherwise failing the transfer job. All FTS based experiment workflows have been reviewed before Run 3 to integrate the *checkOnTape* feature for all transfers to the Tier-0 CTA storage endpoint. Another positive aspect of this evolution is that end users can now measure tape queue times to Tier-0 CTA in FTS service monitoring.

The combination of these rules allowed us to dimension the hardware needed for the tape buffer for the duration of Run 3 at an early stage: The entirety of the buffer hardware was ordered in 2020. Each buffer server can write and read full duplex on its internal 16 × 2TB SSDs at 2.5 GB/s, yielding up to 3.5 hours worth of data buffering at this rate.

The buffer efficiency gains are expected to allow us to operate the four LHC CTA buffer instances at up to 20GB/s of incoming archive throughput when enough tape drives are available in the shared tape infrastructure (approximately 70 tape drives are needed for a single experiment at this speed) (Fig. 2).

2.3 HTTP Tape REST API

The HTTP Tape REST API[6] was deployed in production during Q1 2023. It is currently actively used in by the LHCb experiment, for a tiny fraction of its traffic between Tier-0 tape and Tier-1s, and the ATLAS experiment, which has moved all its traffic to HTTP.

With regards to protocols it is in the CTA service's interest to consolidate the implemented data transfer protocols to only xrootd and HTTP, while simultaneously improving transfer efficiency. HTTP is replacing gridftp traffic on the *Worldwide LHC Computing Grid* (WLCG)[7] and will allow the CTA service to remove its last two gridftp gateways. The gateway removal will in turn allow for more efficient and more direct transfers. Similarly, in the case of LHCb, transfers between Tier-0 and Tier-1s required xrootd gateways for Third Party transfers with delegation. The HTTP *redirect* directive offers a more efficient and direct alternative in this case.

Production deployment and additional details have been reported in a dedicated article[6].

[4] Best effort service is here considered to be transfer rates that opportunistically exceed the guaranteed minimum.

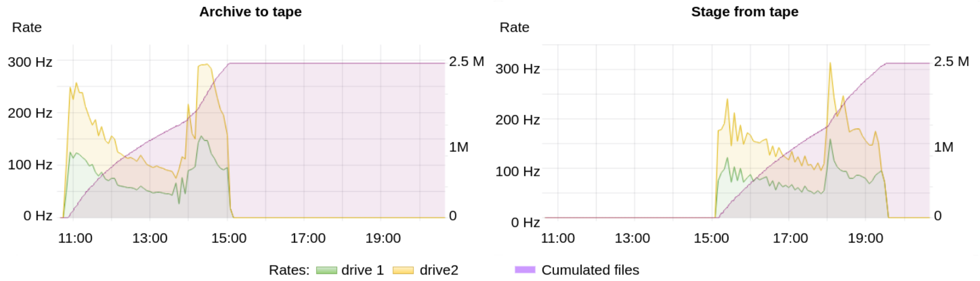

Figure 3. IO rates during the CTA 4.10.0-1 stress test. Archive and stage rates in Hz for the 2 tape drives - drive1 and drive2 - are stacked. The purple accumulation line measures the cumulated number of files archived and then staged.

3 Operational Best Practices

3.1 Testing releases for production

The CTA project's Continuous Integration procedures in Gitlab perform a series of standard functional tests on every commit, including unit and system tests.

Before tagging a CTA release, the candidate commit is also *stress-tested* on a dedicated `kubernetes` single node instance. This instance is composed of an `mhvtl`[8] backend, a `quarkdb`[9] database, an EOS namespace backend, as well as `eos` disk-server storage backends, each of which has has its own dedicated 1TB SSD. It is specifically built such that the hardware mimics the production environment hardware, and utilises a similarly performant database and `ceph` cluster. The test scenario is quite simple: First archive, and then stage, 2.5 million files of 100kB each over approximately 9 hours (Fig. 3). This stress test targets metadata rates in the release candidate software stack and must for this reason run over a long enough period to catch possible regressions. The *Stress Test* has become a standard benchmark for CTA software where execution failure, or reduced performance, means that the release will not move forward in the present state.

3.2 Production Deployments

Upgrading the CTA production instance at CERN implies the installation of the new software on the two CTA frontend instances, and then on the >180 tape servers which run `cta-taped` while minimizing impact on the service users. This would not be possible without clear CTA software release procedures, which have been regularly refined over time to in order to enable smooth software upgrades. The main rule is that two consecutive CTA versions must be compatible with the same backends. In other words, objects in the shared object store must work for both, and catalogue version in the shared database tables must be compatible. This was required to ensure a smooth development workflow: Release and deploy often with confidence (Fig. 4).

3.2.1 Upgrading CTA software to the next version

As the next version of CTA software will be compatible with the current one, operators only need to upgrade the software for `cta-frontend` or `cta-taped` between user requests to avoid any service disruption.

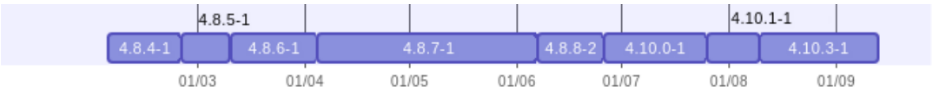

Figure 4. CTA software versions deployed in production. Between February and September 2023 8 versions of CTA were transparently deployed in production with no impact on tape user activity.

Upgrading the `cta-frontend` may fail a few queuing requests. This issue has not been solved at the time of writing, as there is no high availability solution in place for the system as a whole.

Upgrading `cta-taped` is by far more complicated and requires additional properties on the software, especially tape drive management. The tape drive's *desiredStatus* is set to *DOWN* and as soon as the current tape transfer session is over and the tape is dismounted, the tape drive status is automatically set to be *DOWN*. In this state it won't look for more tape mounts and can be safely taken away from the set of production machines. The drive state's *reason* field may be used to keep track of the upgrade progress.

A Rundeck job with the next CTA version as an argument is then launched on all the production tape servers in parallel performing these steps:

1. Check if CTA software needs to be upgraded (ensuring idempotence)
2. Ensure that all tape drives connected to this server are *DOWN*
3. Upgrade CTA software
4. Put the drive back in production if the previous steps succeeded

This procedure is idempotent and operators may retry it on all *failed nodes* until all tape servers are successfully upgraded.

A user transparent upgrade of a CTA instance lasts for as long as the longest user tape mount that was present when the upgrade process was initiated. In practive 80% of T0 CTA production instance tape servers are transparently upgraded in less than 5 hours.

3.2.2 Upgrading CTA catalogue version

Once CTA software has been upgraded on **all** `cta-frontend` and `cta-taped` services of a CTA instance to the next CTA software version, CTA operations may have to upgrade the CTA catalogue by deploying the next schema version.

CTA convention is that the CTA catalogue can only be upgraded in specific software *pivot versions*, which accept the previous and the current CTA catalogue version. These can be identified by a `0` in the third position of the semantic version number. For instance, CTA version 4.10.0-1 accepts *catalogueVersion* 12.0 or 14.0 and is an upgrade from CTA 4.8.7-1 that only allows *catalogueVersion* 12.0. The next CTA release in 4.10 will only run on *catalogueVersion* 14.0.

CTA catalogue upgrade procedures have largely been automated with `liquibase`, and the standard procedure has been published in the form of a public container, which is available from CERN CTA public container registry[10].

3.3 Tape Lifecycle

The new CTA tape state machine separates user and operator access to tapes. In particular, not mixing user and operator tape queues and mounts allowed for the simplification and automation of the tape lifecycle management.

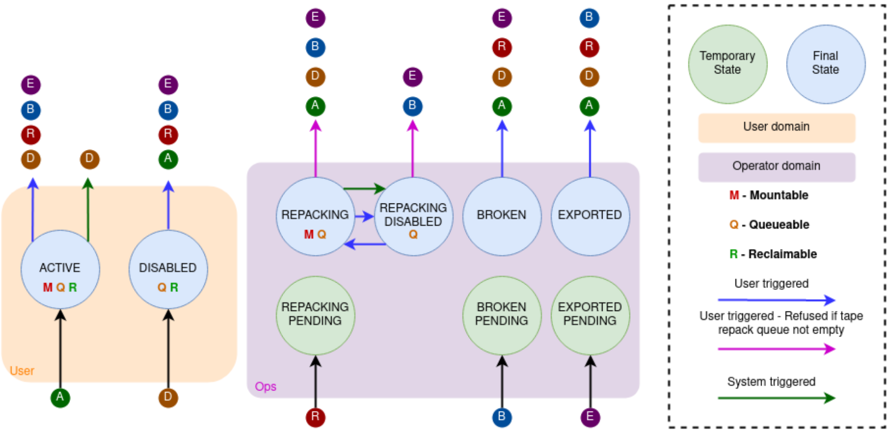

Figure 5. CTA's new *tape state* state machine. It separates tape capabilities *Mountable/Queueable/Reclaimable* between User or Operator domains depending on the present tape state.

3.3.1 CTA tape states

Initially, CTA inherited tape states from CASTOR classification, where there were only 2 main options:

- **full**, which is set to *True* if the tape has reached capacity and cannot be written to anymore
- **disabled**, which is set to *True* if requests cannot be queued on it anymore for one reason or another

Combining these flags allowed tape operations to define some local conventions for different scenarios, such as repack, but the lack of enforced convention lead to various misunderstandings. In addition to this, leaving user recall requests in the same queue as repack jobs which were about to move the data onto another tape caused complications.

After thorough discussions between operators and developers, as well as several tests on dedicated release candidates, the following state machine was adopted (Fig. 5).

The *full* flag was kept, but in order to represent additional tape states, a dedicated field was introduced.

By default the tape is ready for user access, which is referred to as *ACTIVE* state. In the event that a problem involving that tape occurs, it is set to *DISABLED*. It is expected that this is a temporary state during which the operator investigates the issue. The DISABLED tape is not mountable to prevent other tape servers from trying to access it, but the already queued user jobs for that particular tape are kept in the system, and it is also possible to queue new jobs. Once the operator decides that the tape is healthy, it is put back into the ACTIVE state, such that all user jobs can continue normally.

On the other hand, if the operator decides that the tape is not suitable for use, the tape is set into *REPACKING*, *BROKEN* or *EXPORTED* status. In all these 3 states, the existing user jobs are removed from the system and the user receives an error message. This job removal is done during the transition through the internal *REPACKING_PENDING*, *BROKEN_PENDING* or *EXPORTED_PENDING* states.

If the operator set the tape to the REPACKING state, the tape can now be submitted for repack using the `cta-admin repack add` command. In case of further problems with the tape, the monitoring system will recognise that the tape is undergoing repack and will set the tape state to *REPACKING_DISABLED*. This will pause the repack, giving the operator another possibility to investigate the issue and unblock it. The system iterates between those two states until the source tape has been fully repacked. Only when there is no running repack is it possible to change the state to something different than REPACKING.

The states mentioned thus far are internal to the CTA software, and do not capture the complete lifecycle of the physical tapes, which may begin and end outside of CTA. For instance, it may be desirable for a tape undergoing a repack to be put into an extended state of quarantine between the CTA-side repack operation itself and the subsequent reuse of the media. Such supplemental logic is implemented in the CTA operator tools[11], which are now being released as free and open source software for the benefit of other CTA sites[12]. For the above case, ATRESYS — Automated Tape REpacking SYStem[13] manages the higher-level state machine that builds on top of the one in CTA.

4 Future work

4.1 Archive Metadata

After focusing development on write efficiency, the next step is to improve staging efficiency by logically grouping retrieve requests. In order to achieve such a grouping, additional information about the transferred data is required. The *Archive Metadata* (AM) proposal is a first step toward standardizing a small common set of critical metadata to be passed along a file during its transfer to a tape storage endpoint. AM discussions for tape collocation and scheduling hints started between CTA, dCache[14] developers and CTA operations, as well as the LHC experiments and T1 tape sites.

Implementing AM-based scheduling is entirely optional for tape sites, so as not to constrain a site's selected software stack. In the CTA Tier-0 case, AM will be first stored as a tape file property in the catalogue, in order to better refine AM definition with the experiments, and also model and evaluate possible gains before starting any implementation at the scheduler level.

Nonetheless, AM (Fig. 6) passed by the experiments must provide enough information to satisfy all tape storage services:

- `collocation hints` is a 4 level hierarchical grouping that translate any experiment logical file grouping into metrics tape storage can use to improve file collocation independently from the experiment namespace.
- `scheduling hints` tell how important it is for the experiment to have this file moved on tape with `archive_priority` integer between 0 (lowest) and 100 (highest).
- `optional hints` are an additional set of optional hints that can help tape sites by providing the total size of the set of files a given file is part of, how many files are in this set, the FTS *activity* for this transfer, etc.

We are continuing discussions with the various stakeholders and more details will be published on this topic later. The FTS service already accepts AM for submitted jobs. Experiments now need to feed these metadata to allow tape storage endpoints to collect and analyse them.

5 Conclusion

During its short history, the CTA Service demonstrated not only impressive archival data rates, but also that its linearly scaling performance was predictable and could be sustained over long periods. The observed service stability and performance allows the CTA team to focus on other problems like tape data placement/scheduling and to share its software

```
archive_metadata = {
    "scheduling_hints": {
        "archive_priority": "100"            # highest priority
    },
    "collocation_hints": {
        "0": "data23_13p6TeV",                                  # project
        "1": "RAW",                                             # datatype
        "2": "00452799",                                        # runnumber
        "3": "data23_13p6TeV.00452799.physics_Main.daq.RAW",    # dataset
    },
    "optional_hints": {
        "activity": "T0 Tape",  # Tier-0/DAQ
        "3": {                  # dataset level
           "length": "19123",   # total number of files at specified level
           "bytes": "80020799318456" # total size of files at specified level
        }
    }
}
```

Figure 6. *Archive Metadata* example hand-crafted on one ATLAS file. This is a `data23_13p6TeV` project `RAW` data file generated during run `00452799` dataset `data23_13p6TeV.00452799.physics_main.daq.RAW`. This file transfer has the highest possible priority and there are 19123 files for 80 TB in total in this logical dataset.

stack, operations tools and best practices with a broader audience inside and outside the HEP community. While development efforts thus far have focused on archival efficiency, the upcoming Archive Metadata support will ensure future performance gains for data retrieval operations.

References

[1] J. Leduc, *The CERN Tape Archive (CTA) - running Tier 0 tape*, https://indico.cern.ch/event/1123214/contributions/4821950/ (2022), HEPiX

[2] *Rundeck*, https://docs.rundeck.com

[3] R. Bachmann, *CERN's Run 3 Tape Infrastructure*, https://indico.cern.ch/event/1123214/contributions/4821966/ (2022), HEPiX

[4] *FTS*, https://fts.web.cern.ch/fts

[5] M. Patrascoiu, *Fts: Towards tokens, qos, archive monitoring and beyond*, https://indico.cern.ch/event/898285/contributions/4034143/ (2020), HEPiX

[6] J. Afonso, C. Caffy, M. Patrascoiu, J. Leduc, M. Davis, S. Murray, P.O. Cortés, *An HTTP REST API for Tape-backed Storage* (2024), to be published in proceedings of CHEP 2023

[7] I. Bird, Annual Review of Nuclear and Particle Science **61**, 99 (2011), `https://doi.org/10.1146/annurev-nucl-102010-130059`

[8] M. Harvey, *mhVTL*, http://www.mhvtl.com/

[9] E. Sindrilaru, *QuarkDB*, https://quarkdb.web.cern.ch/quarkdb/docs/master/

[10] *CTA Container Registry*, https://gitlab.cern.ch/cta/public_registry

[11] *CTA Operations Utilities*, https://gitlab.cern.ch/cta/cta-operations-utilities/-/wikis/home

[12] M. Davis, J. Afonso, R. Bachmann, V. Bahyl, J.C. Vera, J. Leduc, P.O. Cortés, F. Rademakers, L.W. r, V. Yurchenko, *The CERN Tape Archive Beyond CERN — an Open Source Data Archival System for HEP* (2024), to be published in proceedings of CHEP 2023

[13] V. Bahyl, *ATRESYS—Automated Tape REpacking System, a tool for managing CTA repacks and tape lifecycle*, https://indico.cern.ch/event/1227241/contributions/5366313/ (2023), 7th EOS Workshop

[14] *dCache*, https://www.dcache.org