

LHC Data Storage: Preparing for the Challenges of Run-3

Maria Arsuaga-Rios^{1,}, Vladimír Bahyl¹, Manuel Batalha¹, Cédric Caffy¹, Eric Cano¹, Niccolò Capitoni¹, Cristian Contescu¹, Michael Davis¹, David Fernandez Alvarez¹, Jaroslav Guenther¹, Edouardos Karavakis¹, Oliver Keeble¹, Julien Leduc¹, Fabio Luchetti¹, Luca Mascetti¹, Steven Murray¹, Mihai Patrascioiu¹, Andreas Peters¹, Michal Kamil Simon¹, Elvin Sindrilaru¹, and Rainer Toebbicke¹*

¹CERN European Laboratory for Particle Physics 1211 Geneva (Switzerland)

Abstract. The CERN IT Storage Group ensures the symbiotic development and operations of storage and data transfer services for all CERN physics data, in particular the data generated by the four LHC experiments (ALICE, ATLAS, CMS and LHCb).

In order to accomplish the objectives of the next run of the LHC (Run-3), the Storage Group has undertaken a thorough analysis of the experiments' requirements, matching them to the appropriate storage and data transfer solutions, and undergoing a rigorous programme of testing to identify and solve any issues before the start of Run-3.

In this paper, we present the main challenges presented by each of the four LHC experiments. We describe their workflows, in particular how they communicate with and use the key components provided by the Storage Group: the EOS disk storage system; its archival back-end, the CERN Tape Archive (CTA); and the File Transfer Service (FTS). We also describe the validation and commissioning tests that have been undertaken and challenges overcome: the ATLAS stress tests to push their DAQ system to its limits; the CMS migration from PhEDEx to Rucio, followed by large-scale tests between EOS and CTA with the new FTS "archive monitoring" feature; the LHCb Tier-0 to Tier-1 staging tests and XRootD Third Party Copy (TPC) validation; and the erasure coding performance in ALICE.

1 Introduction

The CERN IT Storage group (IT-ST) provides many services to the physics community. In this introduction, we will describe the three key components used by the four LHC experiments: the EOS disk storage system [1], its archival back-end, the CERN Tape Archive (CTA) [2], and the File Transfer Service (FTS) [3], which are essential for LHC data taking, data processing, data distribution, and analysis.

1.1 EOS: Open Storage

EOS is a distributed disk storage solution for central data recording, analysis, and processing, designed to store large amounts of physics data and user files. Its high-capacity, low-latency,

*e-mail: maria.arsuaga.rios@cern.ch

and reliable design allow over 350 Petabytes (PB) to be currently stored at CERN, and more than 2.5 Exabytes (EB) of physics data has been served during 2020. EOS supports accesses from thousands of concurrent clients with random remote I/O patterns with flexible multi-protocol support, including WebDAV, CIFS, FUSE, XRootD, and GRPC. According to the experiments' needs, EOS has also expanded its offer with a wide variety of authentication methods, such as Krb5, X.509, OIDC, JWT, and proprietary token authorization. Furthermore, it became a common integrator in the CERN storage solutions by providing different functionalities for other CERN services, for example the Sync & Share functionalities for the CERNBox front-end, to offer easy collaborations between CERN users; and its buffer capabilities for the CERN Tape Archive (CTA) software. All these functionalities consolidate EOS as the main storage solution for the LHC and non-LHC experiments at CERN and, therefore, the key component to evaluate during the Long Shutdown (LS2) period.

1.2 CTA: CERN Tape Archive

The CERN Tape Archive (CTA) is the tape back-end to EOS, as well as being the evolution of and replacement for its predecessor, CASTOR. The primary purpose of CTA is to provide reliable, long-term archival storage of the custodial copy of the data from all of the physics experiments at CERN. In terms of functionality, CTA controls the physical tape infrastructure (tape drives, cassettes and robotic libraries); maintains the tape catalogue (which files are stored on which cassette); and offers a high-performance queuing system for archival and retrieval requests. One of CTA's design goals was simplicity. As EOS is the *de facto* storage system for physics analysis at CERN, CTA delegates all disk operations—namespace operations and staging files to and from tape—to EOS. This left the CTA developers free to focus their efforts on designing a system which makes efficient use of the tape drives and which can handle high throughput of archival and retrieval requests as well as the high data rates anticipated during LHC Run-3. During 2020, the tape storage for the ATLAS, ALICE and CMS experiments was migrated from CASTOR to CTA; LHCb was migrated early in 2021.

1.3 FTS: File Transfer Service

The File Transfer Service (FTS) distributes the majority of the Large Hadron Collider (LHC) data across the Worldwide LHC Computing Grid (WLCG) infrastructure. It is integrated with experiment frameworks such as Rucio, PhEDEx and DIRAC and is used by more than 35 experiments at CERN and in other data-intensive sciences inside and outside the High Energy Physics (HEP) domain. In 2020, the centrally-monitored FTS instances transferred more than 1 billion files and a total of 1 EB of data. FTS is a low-level data management service, responsible for scheduling the reliable bulk transfer of files from one site to another, while allowing participating sites to control their network resource usage. It can be accessed through a REST API or using command-line tools. FTS provides easy user interaction for submitting transfers; a WebFTS portal, which is a web-based file transfer and management solution that allows users to invoke reliable, managed data transfers on distributed infrastructures from within their browser; real-time monitoring that is rich in content; and a Web Admin interface to be able to modify the internal settings of the service such as configuring access rights and placing limits on storage and links.

2 LHC workflows for RUN 3

The LS2 period is a good opportunity to evaluate and understand the next data-taking workflows of each of the four LHC experiments. This allows the Storage Group to offer and adapt the best solutions to achieve the goals of Run-3.

2.1 ATLAS workflow

ATLAS is one of the four major experiments of the Large Hadron Collider (LHC). It is a general-purpose particle physics experiment run by an international collaboration and it is designed to exploit the full discovery potential and the huge range of physics opportunities that the LHC provides. The physics program of ATLAS is thus very diverse and requires a flexible data management workflow. The ATLAS workflow for Run-3 is shown in Figure 1,

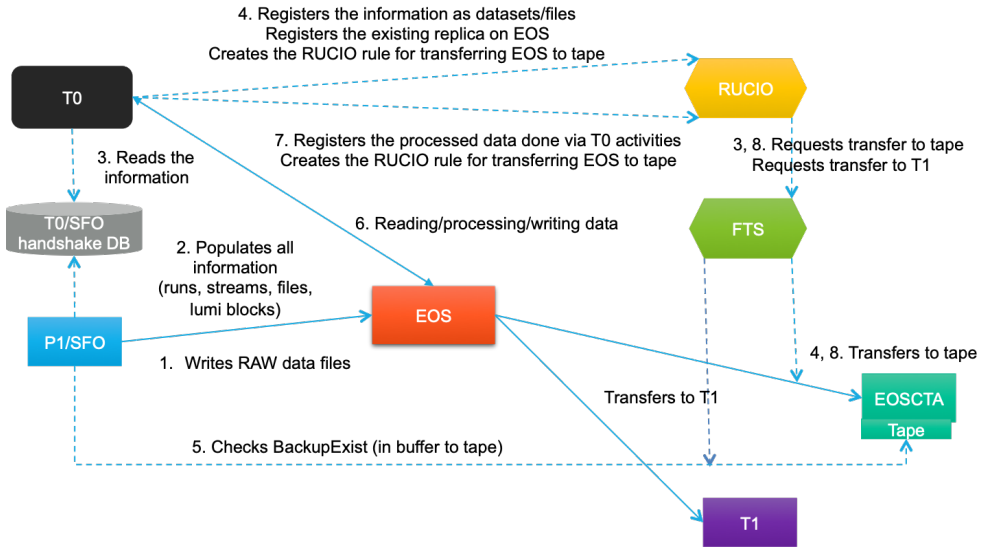


Figure 1: ATLAS workflow for Run-3. Dashed lines indicate metadata operations.

with its data flow operations and metadata operations (dashed lines). The data logger, or Sub-Farm Output (SFO), writes the RAW data files to EOS. Then it populates the T0/SFO handshake database with all necessary information about runs, “streams”, “lumi blocks” and files. Tier-0 (T0) reads all this information, and registers it as datasets/files in Rucio. T0 registers the existing replica on EOS and creates the Rucio rule for transferring files from EOS to CTA. Rucio is the scientific data management framework that interacts with the File Transfer Service for injecting the transfer requests: EOS to CTA and EOS to Tier-1s (T1s), according to pre-defined subscriptions. Finally, SFO checks if a tape copy has been successfully archived before deleting the file from its buffers in LHC Point 1 (experimental area where the ATLAS detector is located). During this process, T0 launches offline activities. T0 reads the data from EOS for processing (reconstruction, merging, *etc.*) and storing the EOS results again. Then, T0 registers the processed data in Rucio in the same way as the RAW data.

2.2 CMS workflow

The Compact Muon Solenoid (CMS) experiment is also a general-purpose detector, designed to exploit a large spectrum of the physics opportunities presented by the LHC. CMS’s workflow is following a similar data flow as ATLAS. During LS2, CMS migrated from PhEDEx, the data management system used in previous RUNS, to the same scientific data management framework as ATLAS, Rucio. As shown in Figure 2, the CMS Data Acquisition system (DAQ), located in LHC Point 5, transfers the data to EOS and notifies Tier-0 (T0) using a

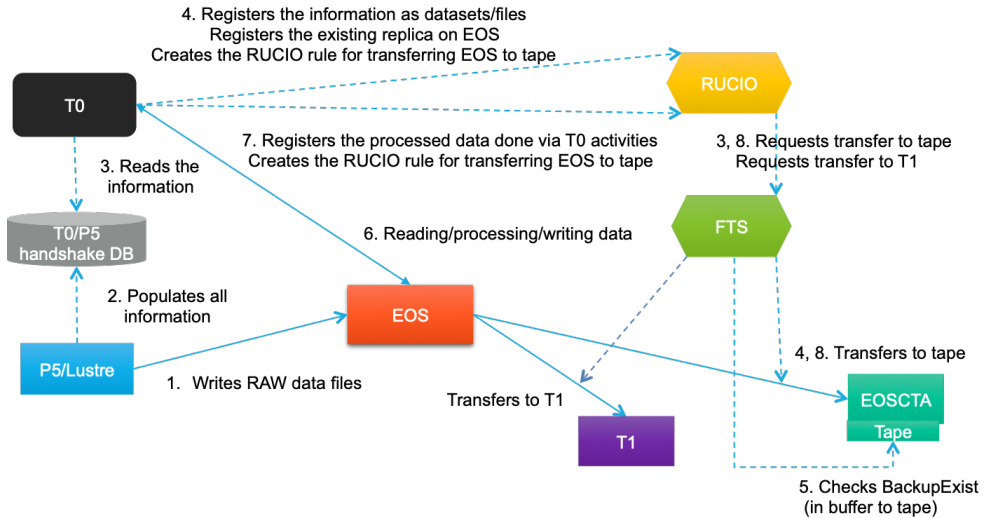


Figure 2: CMS workflow for RUN-3. Dashed lines indicate metadata operations.

database and a handshake protocol. Then, T0 reads all the information needed and registers it with Rucio, who triggers the transfers to CTA and Tier-1s (T1s), using the File Transfer Service (FTS). One of CMS’ main requirements was that FTS provides a new “Archive Monitoring” feature, which checks if and when the file has been successfully archived to tape. This check ensures the existence of a tape copy before CMS deletes the data from its DAQ buffers. T0 also performs offline activities, direct communication with EOS and the corresponding registration in Rucio for transferring data out to EOS.

2.3 LHCb workflow

The Large Hadron Collider beauty (LHCb) experiment is a specialized detector to investigate the slight differences between matter and antimatter by studying a type of particle called the “beauty quark” (“b-quark”). Figure 3 shows the LHCb data flow. LHCb starts in the same way as the previous experiments, by sending the data from their DAQ system to their EOS instance, triggered by DIRAC, the LHCb data management system. DIRAC relies on FTS for triggering transfers between EOS to CTA and EOS to Tier-1s. Something to point out from the LHCb workflow is that DIRAC (and FTS) also manages the export and staging in between CTA and Tier-1s for online and offline activities. LHCb, like CMS, will depend on the new FTS Archive Monitoring feature.

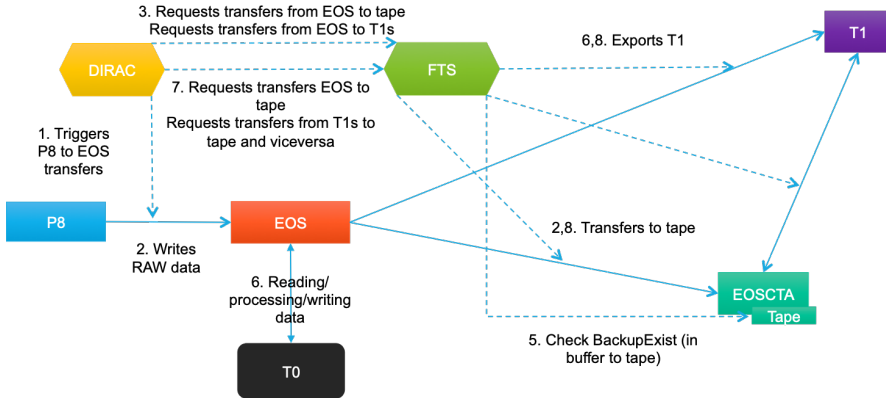


Figure 3: LHCb workflow for RUN 3. Dashed lines indicate metadata operations.

2.4 ALICE workflow

ALICE (A Large Ion Collider Experiment) is a detector dedicated to heavy-ion physics at the LHC. It is designed to study the physics of strongly interacting matter at extreme energy densities, where a phase of matter called quark-gluon plasma forms. In the ALICE work-

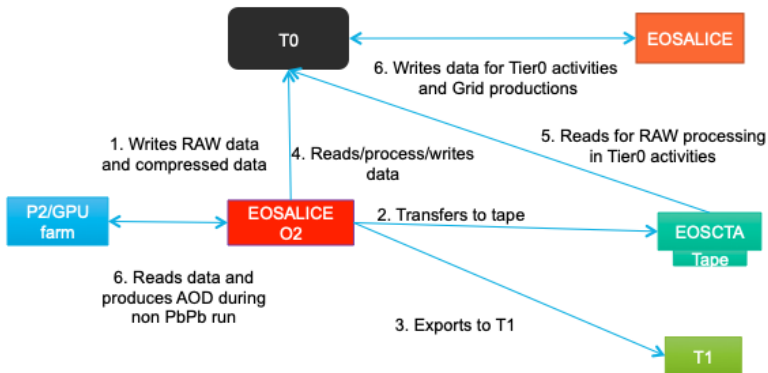


Figure 4: ALICE workflow for RUN 3.

flow (Figure 4), the DAQ system sends the data to the dedicated EOSALICEO2 instance. EOSALICEO2 is, *de facto*, an extension of the ALICE Online farm, and acts as a cache to cope with the high data rate and allow high-performance processing. EOSALICEO2 exports the data to Tier-1s and CTA, and it is used for T0 activities which process data and store them in another EOS instance, EOSALICE.

3 Challenges and tests

During the LS2 period, planning and communication activities became more challenging due to the consequences of COVID-19: multiple delays in hardware procurement and only virtual communications. To overcome these new challenges, we created a standard framework definition for planning and communication, to be followed by all the experiments. This framework helped us to approve and validate the goals, dates, and blocking factors for individual tests, per component and per experiment. Data challenge tests are also included, which implies more than two experiments interacting together. This section presents the tests which have been carried out to date, and the challenges for each experiment.

3.1 ATLAS: Pushing DAQ system to its limits

The purpose of this test was to handle the accumulated data in SFOs, due to possible issues such as a broken network connection between LHC Point 1 (P1) and the computer center, or a short EOS unavailability during a run. The test scenario is as follows: P1 generates data without transferring to EOS, thus accumulating it on the SFOs. Once the SFOs are full, P1 stops the data generation and starts moving data to EOS. Data is deleted from the SFO as soon as it reaches EOS. No migration to tape is taken into account for this test. The traffic is tagged at P1 with an EOS-specific attribute. This last point helps IT-ST to have more control over the monitoring system. The total amount of data generated is 250 TB with an average file size of 500 MB. SFO counts on six servers achieving 15 GB/s. The goal of this test is that EOS should handle peaks close to 15 GB/s. As shown in Figure 5, data was transferred

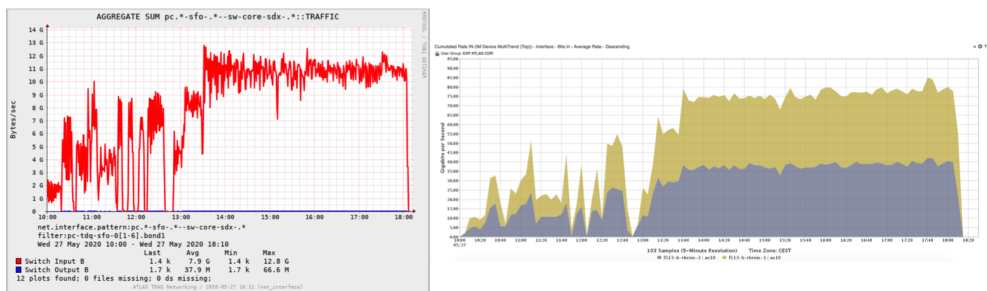


Figure 5: SFO plot with a 30-second average (left side) and network plot with router information with 5-minute average (right side)

to EOSATLAS at 9.5 GB/s on average for more than four hours, with data rate peaks of up to 12.8 GB/s, double the peak data rate achieved in 2018. The final configuration starting at 13:00 was considering 100 transfers per server. We discovered that the bottleneck was actually the DB updates executed for registering the transfer status. During this four-hour period, EOS handled this traffic without any problem. There were only 303 transfer failures (which were later retried) from a total of more than 200,000 file transfers.

3.2 CMS double migration to Rucio and CTA

During the first round of Run-3 preparation, the main challenge for CMS was the migration of their data management system from PhEDEx to Rucio. During this process, follow up by the storage team was crucial, as CMS needed to create rules in Rucio and trigger transfers

with the help of FTS. In this preparation, we designed several functional tests to check that their migration uses FTS successfully with full performance transfers from EOS to CTA. Figure 6 shows an extensive test from EOS to CTA using Rucio in production. In this test,



Figure 6: Large test from EOS to CTA (675 TB) with Rucio. CTA reading and writing throughput between the tape buffer and tape (behind) and FTS efficiency (in front)

CMS created the corresponding rules in their Rucio instance to send 675 TB from EOS to CTA with 166,510 files from a CMS dataset. According to the CTA setup for this test, the expected throughput was a maximum of 3 GB/s. This was achieved and sustained. We detected a drop in efficiency thanks to the FTS monitoring, which alerted us to the fact that CMS’s proxy certificate had expired. CMS was informed and reacted quickly, fixing the issue for future transfers.

Another important aspect of the CMS migration to CTA was the disk buffer setup in front of the tape system. CTA uses an SSD pool for the disk buffer, maximizing tape IO at the cost of disk buffer space. However, storage space constraints require that files do not remain on the buffer after they have been successfully written to tape. The FTS Archive Monitoring feature helped with this: a transfer is reported as finished only after the file was successfully archived to tape. FTS executes the transfer and follows the file’s archival progress. Once the transfer is reported as FINISHED, the experiment can safely discard the file from the disk buffer, as it is guaranteed that a tape copy exists. The state of archiving to tape implies longer queue residency for FTS jobs, often measured in days. The FTS Archive Monitoring mechanics follow a similar design to the staging process. A series of tests were undertaken before they were released. Of particular note were the stress tests, which used 100,000 files using a highly compressible file format to maximize tape space. The goal was to test the FTS archiving queues and metadata operations. Incremental tests of 1k, 10k and 100k archive operations were executed, reaching peaks of 37k operations queued per FTS node. Due to the way the load is divided between the nodes, this amount of active archiving operations towards a storage endpoint per node is more than what’s expected to commonly see in production and a good outcome over all. After the successful conclusion of the stress test, the next step involved a shared test with CMS, archiving 10 TB of data in a “production-like” scenario. Having encountered no problems during the testing stage, the Archive Monitoring feature was released and is currently in production.

3.3 Xrootd/HTTP Third Party Copy (TPC) validation between Tier-0 and Tier-1s in LHCb

One of the main challenges for the LHCb computing team is to deprecate the SRM protocol. This protocol is not supported by CTA. However, many of the LHCb Tier-1s (T1s) use dCache, and they were still using space tokens, an SRM feature that defines a concrete

instance of a space reservation. The first step to follow up with them was migrating those T1s (Gridka, PIC, RAL, IN2P3, and SARA) to deprecate the space tokens consistently. After that, IT-ST helped LHCb to deploy and configure XRootD or HTTP Third Party Copy (TPC) with the corresponding tests against our LHCb EOS instance. Our EOS instances at CERN were the first to support both TPC protocols (XrootD and HTTP). Therefore, we were able to validate all the T1s and debug any problems with their configuration. One example was a missing configuration in CNAF where they were using unsupported credentials, which prevented them from using HTTP-TPC. In Figure 7, we see how HTTP-TPC for T1s works 100% on both sides against EOSLHCb. With this validation, it was easy to extrapolate to the entire matrix, allowing LHCb to run the tests via DIRAC using FTS against all Tier-1s. Notice that only RAL was pending for HTTP-TPC (RAL is, however, functionally working with XrootD-TPC). Currently, Gridka, PIC, RAL, and IN2P3 work with both TPC protocols.

Source	Destination	Source	Destination
+ https://grid05.lal.in2p3.fr	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://se.cis.gov.pl
+ https://lhcbwebdav-kit.gridka.de	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://lhcbwebdav-kit.gridka.de
+ https://storage01.lcg.cs.cscs.ch	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://ccsrm.ihep.ac.cn
+ https://ccdavlhcb.in2p3.fr	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://hepgrid11.ph.liv.ac.uk
+ https://hepgrid11.ph.liv.ac.uk	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://webdav-lhcbt1.pic.es
+ https://webdav.grid.sara.nl	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://lpsnel.in2p3.fr
+ https://svr018.gla.scotgrid.ac.uk	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://ccdavlhcb.in2p3.fr
+ https://marsedpm.in2p3.fr	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://bohr3226.tier2.hep.manches
+ https://se.cis.gov.pl	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://gfe02.grid.hep.ph.ic.ac.uk
+ https://gfe02.grid.hep.ph.ic.ac.uk	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://storage01.lcg.cs.cscs.ch
+ https://bohr3226.tier2.hep.manche	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://marsedpm.in2p3.fr
+ https://lhcbadm.tl.grid.kiae.ru	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://se0003.m45.ihep.su
+ https://lpsnel.in2p3.fr	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://tbit00.nipne.ro
+ https://ccsrm.ihep.ac.cn	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://xfer-lhcb.cr.cnaf.infn.it
+ https://xfer-lhcb.cr.cnaf.infn.it	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://lhcbadm.tl.grid.kiae.ru
+ https://se03.cat.cbpf.br	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://svr018.gla.scotgrid.ac.uk
+ https://se0003.m45.ihep.su	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://grid05.lal.in2p3.fr
+ https://webdav-lhcbt1.pic.es	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://se03.cat.cbpf.br
+ https://tbit00.nipne.ro	https://eoslhcb.cern.ch	+ https://eoslhcb.cern.ch	https://webdav.grid.sara.nl

Figure 7: Successful HTTP TPC transfers between all LHCb T1s and EOSLHCb

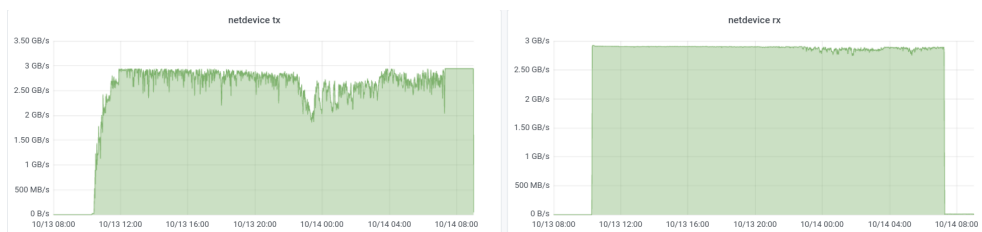


Figure 8: 200 TB test transfer from EOS to CTA

Another critical challenge for LHCb is to migrate to CTA. A validation test of 200 TB from EOS to CTA was carried out, as shown in Figure 8. Currently, XRootD must be used to stage files from tape to the disk buffer, whereas for transfers, the plan is to employ HTTP-TPC from T0 towards T1s, as previously detailed. FTS will act as the glue between CTA (XRootD-based tape staging) and HTTP-TPC transfers. To do so, it provides a composite, or “multihop”, job, which defines multiple steps for a given file transfer. To fit LHCb’s needs, changes were made to allow the first step to be a stage-only operation. Using multihop submissions, the first part is a stage-only operation using the XRootD protocol, while the second step is the actual transfer via HTTP-TPC from CTA to the T1s. This new feature was tested together with DIRAC, to make sure it fits LHCb’s requirements. After DIRAC integration with this new submission model, LHCb will be able to effect T0 to T1 transfers via HTTP-TPC.

3.4 Erasure coding performance in ALICE

Currently, the EOS services at CERN are configured to store two replicas of a file to provide data redundancy. The downside of this approach is the double space requirement and the limitation to a single I/O stream by the sequential read/write performance of hard disks. Erasure Coding allows scaling single stream performance with the number of data disks. The number of parity disks and the ratio of data to parity disks are selected to yield the desired redundancy level and storage volume overhead. The ALICE O2 use case requires high-performance streams due to limited transfer time budget and minimal storage costs with a nominal performance of up to 100 GB/s. Erasure Coding is a perfect match for these requirements. The native implementation of Erasure Coding in EOS follows a gateway model. A client contacts only one storage server and downloads or uploads a file to via a single server (FST). In case of upload, the so-called entry server (gateway FST) splits the stream into equal-sized blocks and computes for m data blocks k parity blocks using the Reed-Solomon algorithm. The default block size is 1 MB. Out of $m+k$ blocks, only one block is stored locally at the entry server. All other blocks are sent to $m+k-1$ remote servers. In case of downloads, the entry server has to access m available data or parity blocks to deliver the original or reconstructed data to a client. If all data blocks are available, no reconstruction is required. The maximum tolerated number of unavailable blocks is k . Due to cost and performance considerations, the envisaged configuration for ALICE O2 is to create two parity blocks for every ten data blocks. The usual notation for this Erasure Coding algorithm is $RS(12,10)$. For data writing of $RS(12,10)$ 10 data chunks enter a gateway FST. One chunk is stored locally on this FST and 9 data and 2 parity chunks have to be copied to remote FSTs. When reading $RS(12,10)$ 9 out of 10 chunks are read remote on the a gateway FST, one is read locally. This results in a traffic amplification of 2.1x of incoming traffic for writing and 1.9x of outgoing traffic for reading. During testing with ten 100GE disk servers with 96 hard disks and 2 GB files, we observed a maximum write performance of 30 GB/s with 1,000 streams. The performance is relatively constant for various EC configurations. The EC configuration defines the resulting data upload performance. For an $RS(8,2)$ configuration, we observed an upload performance of 24 GB/s. The final installation of ALICE O2 is planned to have 75 disk servers. The extrapolated write performance for such a system would be 215 GB/s, which is more than sufficient. A possible problem for the ALICE O2 use case is tails in the upload time distribution. If we run 1000 streams as fast as possible, we observe a wide distribution of upload times with a long tail, where some of them exceed the available upload time window. The tail effect can be significantly reduced by limiting the upload bandwidth, e.g. at 100 MB/s. The interest of the use case is to upload files as fast as possible to free the nominal 2 GB files from memory as soon as possible. To achieve this, we are testing a client-driven EC plug-in,

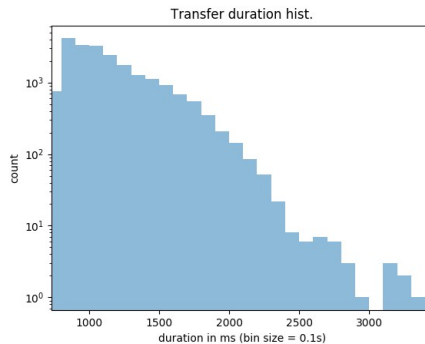


Figure 9: ALICE Erasure Coding (EC): Upload time distribution for RS(12,10) using $3 \times 100\text{GE}$ disk servers with 96 hard disks and max. 300 streams

which creates data and parity blocks client-side and uploads them to data and parity server locations without going through a storage gateway. The advantage of this model is a bisection in network traffic for writing and reading, and a notable reduction of long transfer time tails. A test configuration with only three server nodes achieves an average file transfer speeds of 1.8 GB/s, 15 GB/s in total and not a single transfer exceeds the defined upload window (see Figure-9). The estimated write bandwidth for the complete ALICE O2 pool would be 375 GB/s based on the performed measurements.

4 Conclusions and future work

During this first part of the RUN 3 preparation, our main objective was to understand all the LHC workflows, goals, and challenges. Therefore the first steps were to unblock the various aspects that could prevent running the final data challenges:

- obtain the throughput objectives of the DAQ systems
- validate the migration to a new data management system (Rucio in case of CMS)
- validate the CTA migrations with extensive tests
- confirm correct setups for Third Party Copy protocols
- verify the new Archive Monitoring feature of FTS

As LS2 is not over, the CERN IT storage group is still in the planning process for individual data challenges per experiment, which means testing the whole workflow as it would be during data taking in Run-3. Together with the experiment computing teams, IT-ST are currently planning multiple collective data challenges with more than two experiments simultaneously.

References

- [1] Andreas J Peters and Lukasz Janyst, Exabyte Scale Storage at CERN, 2011 J. Phys.: Conf. Ser. **331** 052015 doi:10.1088/1742-6596/331/5/052015
- [2] S. Murray, V. Bahyl, G. Cancio, E. Cano, V. Kotlyar, D.F. Kruse and J. Leduc., An Efficient, Modular and Simple Tape Archiving Solution for LHC Run-3, 2017 J. Phys.: Conf. Ser. **898** 062013 doi:10.1088/1742-6596/898/6/062013
- [3] E. Karavakis et al, FTS improvements for LHC Run-3 and beyond, 2020 EPJ Web of Conferences **245** 04016 doi:10.1051/epjconf/202024504016