ATLAS DQ2 to Rucio renaming infrastructure

C. Serfon\textsuperscript{1}, M. Barisits\textsuperscript{1,2}, T. Beermann\textsuperscript{1}, V. Garonne\textsuperscript{1}, L. Goossens\textsuperscript{1}, M. Lassnig\textsuperscript{1}, A. Molfetas\textsuperscript{1,3}, A. Nairz\textsuperscript{1}, G. Stewart\textsuperscript{1}, R. Vigne\textsuperscript{1} on behalf of the ATLAS Collaboration

\textsuperscript{1} CERN PH-ADP-CO/DDM, 1211 Geneva, Switzerland
\textsuperscript{2} University of Innsbruck, Innsbruck, Austria
\textsuperscript{3} University of Melbourne, Melbourne, Australia

E-mail: cedric.serfon@cern.ch

Abstract. To prepare the migration to the new ATLAS Data Management system called Rucio, a renaming campaign of all the physical files produced by ATLAS is needed. It represents around 300 million files split between \(\sim\)120 sites with 6 different storage technologies. It must be done in a transparent way in order not to disrupt the ongoing computing activities. An infrastructure to perform this renaming has been developed and is presented in this paper as well as its performance.

1. Introduction

Rucio \cite{1} is the new evolution of the ATLAS Data Management system that will replace the current one called DQ2 \cite{2}. This new system has many improvements, but it breaks the compatibility with DQ2. One of the biggest changes is the fact that no external replica catalog like the LCG File Catalog (LFC) is being used in Rucio, whereas it was one of the central components in DQ2. In Rucio, the physical path of a file can be simply extracted from the Logical File Name via a deterministic function. Therefore, all file replicas produced by ATLAS have to be renamed before migrating to Rucio. This amounts to around 300 million files split between \(\sim\)120 sites with 6 different storage technologies. An infrastructure to perform this is needed. After shortly presenting the new naming convention, we expose in the second part the constraints for this massive renaming and describe the infrastructure that has been implemented as well as the technologies that were used. Finally, we show the performance for the sites where the files were already renamed.

2. The Rucio naming convention

The old naming convention (also known as DQ2 naming convention) for the Physical File Name (PFN) was based on the filename and the dataset name. Since a file can belong to many distinct datasets, it can have many different PFNs, and thus there is no deterministic way to go from a filename to a PFN. Moreover, the convention changed more than 3 times during the last 6 years. That’s the reason why an external file catalog like the LFC is mandatory for DQ2.

The new naming convention (also known as Rucio naming convention) is based on a deterministic function that allows to transform every Logical File Name (LFN) into a PFN. A LFN in Rucio is composed of 2 parts: a scope and a name. The scope allows partitioning
of the namespace for different users. It consists of a string of at most 30 characters (e.g. data12_8TeV, user.jdoe, group.phys-susy...). The name is a string of at most 255 characters (e.g. myanalysis.xyz.root.1, AOD.491965. 0042.pool.root.1) and is unique within a scope. A LFN is the combination of a scope and a filename, for instance user.jdoe:myanalysis.xyz.root.1 or data11_7TeV:AOD.491965. 0042.pool.root.1. The PFN can be easily constructed from the LFN with the following algorithm:

- The first part is constructed with the scope, where the dots delimit the subdirectories. In our examples: user.jdoe → user/jdoe, group.phys-susy → group/phys-susy and data12_8TeV → data12_8TeV
- The second part is composed of the 2 first bytes (in hexadecimal) of the md5 sum of the LFN that are used to create a 2 level directory structure: md5(user.jdoe:myanalysis.xyz.root.1) → 68a2ea3ea370b43f437d62955fba1435 → 68/a2.
- The last part is the filename.

Therefore LFN=user.jdoe:myanalysis.xyz.root.1 will be translated into the following PFN: user/jdoe/68/a2/myanalysis.xyz.root.1. The PFN is prefixed by the protocol and the site specific part to give a Uniform Resource Identifier (URI). With this convention, there is no need of an external file catalog: the information of the internal Rucio catalog and the deterministic function are enough to build the URI. The drawback is that all physical files registered in DQ2 (more than 300 million replicas) will have to be renamed according to this new convention. To perform this work, a renaming infrastructure is crucial.

3. Renaming infrastructure

3.1. Constraints on the renaming

There are many constraints that must be satisfied by the renaming infrastructure. First of all, it must be fully automatised and require as little work as possible from the sites. Secondly, the renaming must be a transparent process that doesn’t interfere with the ongoing computing activities, as Monte Carlo simulation and user analysis. Moreover, during the transition phase to Rucio, we still want to be able to use the old DQ2 system and therefore the renaming must include both the physical renaming of the files and the renaming in the LFC. The infrastructure must be able to interact with most of the different storage technologies used by the sites supporting ATLAS. It must be fault tolerant, and in case of major failure of the site, it must be able to stop and to recover where it stopped. Finally, it must be fast enough to allow the renaming of 300 million files in one year (which means a renaming rate of 10 Hz sustained over a year).

3.2. Description of the renaming infrastructure

The renaming infrastructure that has been developed is based on gearman [4]. It consists of different workers that are dedicated to specific tasks (DQ2 lookup, LFC lookup...). Each worker is an independent process and gets payload from a gearman server, that is fed by a main process which we call supervisor, that controls the workflow (see Figure 1). Thanks to this approach, the load can be spread across multiple machines if needed. The workflow consists of 6 steps:

1. The supervisor queries the DQ2 catalog that stores the dataset replicas to get a list of datasets on a given site (this site has been defined manually in a configuration file, one

---

1 All alphanumerical characters and “.”, “-”, “_” are authorized. scopes with more than 30 characters are rejected.

2 The reason of choosing a cryptographic hash like md5 is to allow to have a well balanced distribution of the files per directories. Non-cryptographic hashes (e.g. adler32) were tried, but the distribution was not well balanced for file names that differed from only a few characters.
supervisor operating only on one site). The operation consists in a single query, and does not need to be parallelized between multiple workers.

2 The supervisor sends the list of datasets to the DQ2 lookup workers that query the DQ2 catalog that stores the dataset content. A query is performed for each dataset and therefore takes advantage of working on multiple workers. The result returned is a list of logical files identified by a GUID (Globally Unique Identifier).

3 The list of GUIDs is then used as input by the LFC lookup agents that identify the replicas (URI) associated to the GUID in the target site.

4 The list of replicas extracted by the LFC lookup is used as input by the LFC addreplica agent, which for each replica that follows the old naming convention will register in the LFC a new replica with the new convention. If some replicas already follow the new convention, they are skipped.

5 The list of files with the old and new convention is used as input by the Storage Rename agent, that interacts with the Storage Element. It renames the files with the old convention on the Storage Element to the new convention using the WebDAV [5] protocol. To speed-up the renaming, all the files from a job are renamed within the same HTTPS session.

6 The last part of the rename is to delete the old replicas from the LFC. This is done by the Cleaner agents.

By proceeding this way, the files physically present on the Storage Element are always available in the catalogs. Moreover the workers get the jobs\(^3\) to process from the gearman server, and if some of the jobs fail, the server ensures that they are reprocessed. In case of massive failure (for instance if the storage system of a site goes down) the renaming can be stopped at any step and when it restarts, the infrastructure has been designed to restart where it broke, which means it will skip the already processed files: it ensures the fault tolerance and robustness. The

\(^3\) A job consists of a task like an LFC lookup or a Storage renaming on a list of files of fixed size (e.g. 2000 files for an LFC lookup, 10 files for the storage renaming).
Figure 2. Total renaming rate (Hz) on a site over ~250k files. It represents the frequency of the joint physical (storage renaming) and logical operations (LFC renaming). The number of Storage rename workers used for this site is 20.

renaming process is fully automated: once a supervisor is started it will process all the files from the site. Many supervisors operating each on different sites can be run concurrently.

For the physical renaming, the protocol that is used is WebDAV. It is supported by DPM [6], dCache [7], StoRM [8] and EOS [9] which covers more than 90% of the ATLAS sites. The physical renaming of each file requires at most 7 interactions: 2 to test if the destination and source files are available, 2 to test existence of the hash directories (the scope directory is always created before) that host the destination file and 2 to create them if they do not exist, 1 for the actual renaming. An other potential candidate for physical renaming would have been SRM, but this protocol is being ruled out in the medium term and was not considered.

4. Performance
The performance of the system is monitored via Graphite [10], a tool to store time series and visualise them in real-time. The slowest part of the renaming is the physical renaming, since the renaming of one single file with WebDAV ranges from O(100) ms to few seconds. This mainly depends on the Round Trip Time which can reach up to 250ms for some sites in North America or Asia Pacific zone. But, many different storage renaming processes can be run simultaneously allowing to reach renaming rates between 10 and 40 Hz (joint physical and logical operations) as shown in Figure 2. The physical renaming rate increases linearly with the number of Storage rename workers up to 20-40 parallel processes and for higher values the renaming speed scales sublinearly. The number of Storage rename workers is then chosen to stay in the domain of linearity (between 20 and 40 depending on the sites). For the other parts, the number of workers for each task has been tuned to: 20 workers for the DQ2 lookup, 4 workers for the LFC lookup, 4 workers for the LFC rename and 4 workers for the LFC cleanup.

We first focussed on renaming files in some big sites (Tier1s and big Tier2s). As of beginning September 2013, more than 50% of the files (around 200 million) already follow the new convention, as shown on Figure 3. The total number of files renamed per day can easily reach 1.5 million by only using one machine to run the whole infrastructure, and by design, the system

4 For the sites with technologies that do not support WebDAV, the sites are asked to rename the files using some local protocol.

5 The machine used here is a dedicated node (Intel(R) Xeon(R) CPU L5520 2.27GHz 2 GB RAM).
Figure 3. Percentage of files following the old (red) and the new (blue) convention extracted from the central LFC. The absolute number of files fluctuates with time, at the beginning of September 2013 being 400 million files.

can scale horizontally. Actually, what currently limits the renaming rate is the deployment of WebDAV on the sites (not all sites have already enabled it).

5. Conclusion

The renaming infrastructure developed to prepare the transition to Rucio has demonstrated that it satisfies all the requirements: it is automatised, robust, transparent for the users, fault tolerant, storage technology agnostic and requires no work from the sites except to configure WebDAV. At this time of writing, it has already successfully renamed hundreds millions of files without disrupting the ongoing computing activities, and if all sites provide a WebDAV access, it should be possible to finish the renaming campaign in time for Rucio deployment.

References
[1] V. Garonne et al., “Rucio - The next generation of large scale distributed system for ATLAS Data Management.”, these proceedings.
[2] V. Garonne et al., “The ATLAS Distributed Data Management project: Past and Future”, Journal of Physics: Conference Series, 396 (032045), IOP 2012.
[3] R. Rivest, “The MD5 Message-Digest Algorithm”, RFC 1321, 1992.
[4] Gearman: http://gearman.org/
[5] The IETF Trust, “HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)”, RFC 4918, 2007.
[6] A. Alvarez et al., “DPM: Future Proof Storage”, Journal of Physics: Conference Series 396 (032015), IOP 2012.
[7] M. Ernst et al., “dCache, a distributed data storage caching system”, Proceedings of Computing in High Energy Physics, 2001.
[8] A. Carbone et al., “Performance studies of the StoRM Storage Resource Manager”, Proceedings of Third IEEE International Conference on e-Science and Grid Computing, (10-13 Dec. 2007).
[9] A. J. Peters and L. Janyst, “Exabyte Scale Storage at CERN”, Journal of Physics: Conference Series 331 (052015), IOP 2011.
[10] Graphite: http://graphite.wikidot.com/