ATLAS Distributed Computing in LHC Run2

To cite this article: Simone Campana and on behalf of the ATLAS collaboration 2015 J. Phys.: Conf. Ser. 664 032004

View the article online for updates and enhancements.

You may also like
- ATLAS Distributed Computing Monitoring tools during the LHC Run 1
  J Schovancová, S Campana, A Di Girolamo et al.
- ATLAS Distributed Computing Shift Operation in the first 2 full years of LHC data taking
  Jaroslava Schovancova, Alessandro Di Girolamo, Johannes Elmsheuser et al.
- AGIS: The ATLAS Grid Information System
  A Anisenkov, A Di Girolamo, A Klimentov et al.

Recent citations
- Spanish ATLAS Tier-1 & Tier-2 perspective on computing over the next years
  Santiago González de la Hoz et al
- Role of the ATLAS Grid Information System (AGIS) in Distributed Data Analysis and Simulation
  A. V. Anisenkov
- AGIS: Integration of new technologies used in ATLAS Distributed Computing
  Alexey Anisenkov et al
ATLAS Distributed Computing in LHC Run2

Simone Campana[1], on behalf of the ATLAS collaboration

[1] CERN, Geneva, Switzerland

Email: Simone.Campana@cern.ch

Abstract. The ATLAS Distributed Computing infrastructure has evolved after the first period of LHC data taking in order to cope with the challenges of the upcoming LHC Run-2. An increase in both the data rate and the computing demands of the Monte-Carlo simulation, as well as new approaches to ATLAS analysis, dictated a more dynamic workload management system (Prodsys-2) and data management system (Rucio), overcoming the boundaries imposed by the design of the old computing model. In particular, the commissioning of new central computing system components was the core part of the migration toward a flexible computing model. A flexible computing utilization exploring the use of opportunistic resources such as HPC, cloud, and volunteer computing is embedded in the new computing model; the data access mechanisms have been enhanced with the remote access, and the network topology and performance is deeply integrated into the core of the system. Moreover, a new data management strategy, based on a defined lifetime for each dataset, has been defined to better manage the lifecycle of the data. In this note, an overview of an operational experience of the new system and its evolution is presented.

1. Introduction

The LHC accelerator has been upgraded in 2013 and 2014 to be able to reach its higher energy (6.5 TeV per proton beam) and higher luminosity. This will bring new computing challenges for the ATLAS experiment, as it will operate at more than double the trigger rate (which implies more event to process by a factor two) and the events will be more complex. In addition, major detector improvements (mostly in tracking and calorimeters) will need to be understood also from the software and computing aspect. The computing resources however will not considerably increase: a growth of computing power proportional to Moore’s law, i.e. 20% every year is expected, rather than linear scaling with the number of events to be processed. In this contribution we will describe the distributed computing system and model that ATLAS will adopt in the coming LHC run and how such system will face the challenges described above.

2. New distributed computing systems

The ATLAS distributed computing systems served very well during LHC Run-1. It was evident, however, that in order to face the Run-2 challenges major improvements were needed and for many services this ended up in a complete re-design.

2.1. Distributed Data Management system

ATLAS developed a new distributed data management system, called Rucio [1], replacing the old one, DQ2. Rucio was developed during the LHC shutdown and went into production on Dec 1st 2014. After a ramp-up phase, Rucio has been performing at the same scale or higher that its predecessor and with
equivalent stability. For example, Rucio can manage the transfer of more than 1M files per day (corresponding to 4PB of data per week) and delete more than 10M files per day. Fig. 1 compares the transfer volume of the data management system before and after the migration to Rucio and underlines how the performance of the new system compares with that of the previous one. Additionally, Rucio implements many new functionalities that will be commissioned during Run-2. For example it integrates natively different protocols for data transfer and data access (SRM, gridFTP, HTTP, XrootD), it provides improved handling of file and dataset level metadata, and it supports advanced features in terms of ownership, permissions and quotas which will facilitate the space management.

Complementary to Rucio, ATLAS deployed a Federated Storage infrastructure based on the XrootD protocol (FAX) [2]. FAX will allow a more flexible utilization of computing resources, as jobs will not be forced to run at sites hosting the data but will be able to overflow to sites with free CPUs and access the data remotely through the storage federation. The intent is to throttle the number of overflow jobs to 10% of the total number of running jobs at most, in order to limit the possible impacts of remote reading on site storage and network. Recent measurements show that the job efficiency for overflow jobs (76% success rate) is similar to the one for local area network access (83%), while in terms of CPU/WallTime ratio remote access (43% efficiency) is disfavored with respect to local access (84%). Tuning the right amount of overflow to benefit from available CPUs at an acceptable efficiency cost will be one of the challenges of Run-2.

Figure 1: the daily volume of data transferred by the ATLAS Distributed Data Management system

2.2. Workload Management system

ATLAS has evolved its workload management system to offer a highly automated framework for distributed data processing and distributed analysis. The new production system, Prodsys-2 [3], consists of four major components: a Request Interface allows production managers to define and handle production requests, the DEFT (Database Engine for Task) service translates production requests into an appropriate set of tasks with dependencies, JEDI (Job Execution and Definition Interface) dynamically creates jobs from the task definitions and PanDA executes the jobs in the distributed environment. While PanDA has been heavily utilized during Run-1, the other components have been newly developed during the LHC shutdown. JEDI and PanDA offer also a platform for
distributed analysis, which was deployed in August 2014 with no noticeable impact on user activities. Prodsys-2 went in production in Dec 1st 2014 (together with Rucio) and, after a validation phase, started processing at a larger scale than the previous system (up to 160K concurrent running jobs), providing capabilities for new workflows previously not supported (for example for the Derivation Framework, as described later on). Prodsys-2 also comes with a new monitoring system for requests, tasks and jobs. Fig. 2 shows the number of concurrent production and analysis jobs as a function of time. Some drop is visible at the time of the Prodsys-2 migration while no impact has been observed when migrating to JEDI for distributed analysis.

3. Improvements in resource utilization
ATLAS identified several areas in software and computing where major improvements would result in a large reduction of resource utilization. In some cases, the changes were not limited to the distributed computing area, but extended to the processing and analysis model.

3.1. Simulation
A considerable amount of CPU resources (up to 40%) are utilized for detector simulation. ATLAS developed an Integrated Simulation Framework (ISF) capable of accommodating both full Geant4 simulation and fast simulation (based on parameterization). The ISF can also run a mixture of full and fast simulation within the same event, so that the physically interesting regions of the event can be simulated with more precision. The ISF will be tuned in 2015 and its utilization increased during LHC Run-2.

3.2. Reconstruction
Event reconstruction is a very memory-intensive process and consumes non-negligible CPU time. During the LHC shutdown, many improvements were carried out on the reconstruction code (mostly in particle tracking), yielding a reduction of the CPU time per event by almost a factor four over the last two years (from 64 to 18 seconds per event) [4]. The sequence of steps of such improvements are represented in Fig. 3 on the left hand side. In order to keep memory usage below the 2 GB per core
(the standard in WLCG resources) ATLAS enabled multi-processing in the reconstruction framework [5]. This allows a reduction in memory consumption from 19 to 11 GB on an 8 core job slot for digitization and reconstruction. A mixture of single core and multicore jobs will run on the Grid during Run-2 with a relative ratio of approximately 50% each. The right hand side of Fig. 3 compares the memory utilization (RSS) in digitization and reconstruction between 8 single core Athena processes and a single multi-process payload (AthenaMP) running on 8 cores.

3.3. The Analysis Model

ATLAS analysis for Run-2 will be based on the xAOD format, which is readable both from the offline framework (Athena) and ROOT. The predecessor during Run-1, the AOD format, was not readable by ROOT, therefore most analysis groups spent an extra processing step just to convert AODs in flat ROOT ntuples. The xAODs therefore will reduce the processing capacity needs and reduce the disk space utilization.

Coupled with the xAODs, ATLAS implemented a Derivation Framework [6] as the basis for Run-2 analysis. The petabyte scale xAODs are centrally processed through Prodsys-2 to produce many terabyte scale derived xAOD output, each serving a particular analysis channel. Those outputs will be reduced to 1% of the original size and therefore will be more portable and easy to access for fast turnaround.

![Figure 3: on the left, the improvement of the reconstruction code (in time per event) for various releases. On the right, the memory utilisation for single and multi core jobs](image)

4. Opportunistic resources utilization

ATLAS relies heavily on the utilization of beyond pledge resources especially for Monte Carlo simulation. During Run-1 and the shutdown period, 30% of the resources used were of opportunistic nature. It has been therefore very important to evolve the computing systems in order to make use of the largest heterogeneous set of resources available. The resources above the yellow line of Fig. 2 are non-pledged.

4.1. Cloud Resources

ATLAS has set up a dedicated effort to enable access to cloud based resources and to utilize them for data processing [7]. The resource provisioning is fully integrated in the PanDA system and is dynamically throttled based on the resource need. As concrete example, ATLAS instrumented the high level trigger (HLT) farm at the experiment pit (P1) with an Openstack interface, so that the approximately 20,000 cores could be made available to the offline production activities in the periods when the online activities were stopped (like for large part of the last two years shutdown and in future LHC technical stops and winter shutdowns) [8]. The main use case is detector simulation, but thanks
to the elasticity of the cloud provided resources, it was possible to utilize them also for reconstruction
on 8 core job slots, during the summer of 2014, for the Data Challenge Monte Carlo reconstruction
under Run-2 conditions.

4.2. High Performance Computing
High Performance Computers (HPCs) also known as Supercomputers have been designed for
massively parallel applications and therefore the embarrassingly parallel use case of High Energy
Physics is not the best to fit their architecture. At the same time at most HPC centers the utilization
never reaches 100% and therefore empty cycles can be used in backfill mode. In addition, ATLAS has
succeeded in getting granted proper allocations at many centers in the US and Europe.

Integrating HPC resources with the production system has not been an easy task: the centers are
very heterogeneous in terms of processor architecture (x86 compatible or not), access policies
(outbound/inbound connectivity, rules on the utilization of a shared file system) and provisioning
schemas (limited number of jobs granted). Therefore, different solutions have been implemented and
will be used in the future.

At some HPC centers all workflows (event generation, simulation, reconstruction), apart from
analysis, which is too I/O demanding, have been commissioned. This is the case for the LRZ, MPPMU
and CSCS centers, which deliver today approximately 4000 cores to ATLAS [9,10]. In MIRA at
Argonne, only physics generators have been enabled as the CPU architecture (PowerPC) has not been
enabled yet for simulation and reconstruction. MIRA is massively utilized therefore for event
generation thanks to the allocation of 50M hours for 2014 and 2015 and the capability to reach up to
100,000 cores simultaneously. In Titan at Oak Ridge ATLAS could utilize 90,000 cores (equivalent to
70% of the experiment total capacity on the Grid) in opportunistic backfill mode for many hours [11].

During Run-2, ATLAS will commission more workflows on HPCs, integrate new centers and
further improve their utilization.

4.3. Volunteer Computing
The target of the ATLAS@home project [12] is to enable users’ desktops and laptops running
simulation in the idle periods. The system is based on the Boinc technology (widely adopted in the
internet community), which has been integrated with the production system through the ARC
Computing Element and the ARC Control Tower, offering a secure mechanism for dispatching the
payloads and handling the input and output data. ATLAS@home today delivers 4,000 cores to
simulation, running at roughly 50% efficiency. The number of users and hosts increases linearly with
time and counts today more than 16,000 distinct hosts. The project started as an outreach activity to
become an important source of opportunistic resources for ATLAS.

4.4. The Event Service
Opportunistic resources in many cases can be very volatile: the Amazon “spot market” price is very
competitive but guarantees resources for at most 15 minutes after which these can be reclaimed at any
time; many HPC centers can be backfilled with short payloads. In order to effectively provision and
utilize opportunistic resources it is therefore important to have very short payloads and in ATLAS
today the shortest possible is processing one single event. For this reason, the Event Service [13] was
implemented. Such a system is capable of streaming the input, processing and storing the outputs at
the event-by-event level. The Event Service is based on existing building blocks such as JEDI, FAX
and AthenaMP. In addition, for the stage out of the data it utilizes Object Store technologies provided
recently by many centers such as CERN, BNL and RAL. The Event Service will be commissioned
during 2015 and used in production for Monte Carlo simulation, on opportunistic and non-
opportunistic resources, including conventional Grid sites.
5. The Data Lifecycle management model

Because of the very demanding physics program and the effectiveness in leveraging pledged and opportunistic resources, ATLAS constantly faces a lot of pressure in managing disk and tape utilization. During Run-1, a dynamic model was put in place for managing disk space: a popularity service was implemented to account for all possible access to data via PanDA and DQ2. The datasets were replicated at Grid sites based on a two-fold policy: a minimal number of copies was created and pinned on disk; a certain number of extra cached copies was created based on access patterns as measured by the popularity system. The number of cached copies could be reduced if the data were not accessed for some time. The system allowed for operation of the data placement for the length of Run-1 in a very automated way, avoiding manual interventions and human decisions. However, ATLAS ended up with a lot of pinned replicas and little room for cached data (at T1 sites, the cached space was as little as 5% of the total) largely reducing the effectiveness of the model. Examination of the pinned data and access patterns showed that a significant fraction had never been accessed even if created more than one year earlier.

The Data Lifetime model was put in place based on the experience of Run-1 and to solve the issues mentioned above. The model foresees each dataset to have a lifetime defined at creation time. The lifetime can vary based on the type of data, the data taking period and the processing conditions; it can also be infinite (this is the case for RAW data for example). The lifetime can be extended in case the data is accessed. Datasets with an expired lifetime can disappear at any time from disk and tape. Within the boundaries of the data lifetime, computing systems can flexibly manage the space: the number of replicas can be increased or decreased based on popularity information, data can be distributed flexibly between sites and the most used data can be kept on disk, while those that are accessed infrequently can be stored on tape and the disk to tape migration can take place at any time based on access patterns.

This simple model has fundamental consequences. The tape usage will increase as tape will be utilized more and more often for less popular data that in the past were pinned on disk. For the first time in ATLAS, tapes will be not only written but also deleted, which will challenge the capability of the Grid sites to recuperate space from tape media. In the steady flow, ATLAS will delete as much data as it will write, as the increase of storage resources is rather negligible compared to the read/write rate. The system relies heavily on the correct measurement of data accesses and it is known today that this is not perfect (direct access through some storage backdoor is not accounted for), so it will be a priority of 2015 to improve this situation.

After a partial implementation of the Data Lifecycle model, carried out in January 2015, the amount of never accessed data dropped from 8PB to 1.2PB. Approximately 10PB of data has been deleted from tape while 5PB of pinned space could be migrated to accommodate cached copies.

6. Conclusions

The ATLAS distributed computing project carried out an ambitious upgrade program in preparation for LHC Run-2. Major components such as Rucio and Prodsys-2 have been developed during the LHC shutdown, commissioned at the end of 2014 and are now ready for production. Major improvements in the simulation and reconstruction software will be leveraged to optimize resource utilization. The new analysis framework together with the new xAOD data format have been established to rationalize the end user analysis process and benefit from the central production and analysis tools such as Prodsys-2 and JEDI. Efficient usage of opportunistic resources remains vital and considerable development effort has been spent enabling utilization of beyond-pledge CPU capacity. Clear examples are HPC resources, Cloud resources and volunteer computing. The dynamic usage of storage resources has been further evolved with respect to LHC Run-1 by introducing a data lifetime model. A first partial implementation of the model already allowed improvement of the disk utilization and the recovery of space on tape storage. Overall, ATLAS computing is ready for LHC Run-2 and will continue evolving to leverage more advanced functionalities, developed during the shutdown but not exploited yet.
7. References

[1] Garonne V et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[2] Vukotic I et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[3] Klimentov A et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[4] Mitrevski J et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[5] Tsulaia V et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[6] Catmore J et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[7] Taylor R et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[8] Brasolin F et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[9] Hostettler M et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[10] Mazzaferrro L et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[11] Panitkin S et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[12] Cameron D et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*
[13] Wenaus T et al. 2015 Computing for High Energy Physics *J. Phys.: Conf. Series*