Evolution of the ATLAS Distributed Computing system
during the LHC Long shutdown

S Campana[1] on behalf of the ATLAS collaboration

[1] CERN, Geneva, Switzerland

E-mail: Simone.Campana@cern.ch

Abstract. The ATLAS Distributed Computing project (ADC) was established in 2007 to
develop and operate a framework, following the ATLAS computing model, to enable data
storage, processing and bookkeeping on top of the WLCG distributed infrastructure. ADC
development has always been driven by operations and this contributed to its success. The
system has fulfilled the demanding requirements of ATLAS, daily consolidating worldwide up
to 1PB of data and running more than 1.5 million payloads distributed globally, supporting
almost one thousand concurrent distributed analysis users. Comprehensive automation and
monitoring minimized the operational manpower required. The flexibility of the system to
adjust to operational needs has been important to the success of the ATLAS physics program.
The LHC shutdown in 2013-2015 affords an opportunity to improve the system in light of
operational experience and scale it to cope with the demanding requirements of 2015 and
beyond, most notably a much higher trigger rate and event pileup. We will describe the
evolution of the ADC software foreseen during this period. This includes consolidating the
existing Production and Distributed Analysis framework (PanDA) and ATLAS Grid
Information System (AGIS), together with the development and commissioning of next
generation systems for distributed data management (DDM/Rucio) and production
(PRODSYS2). We will explain how new technologies such as Cloud Computing and NoSQL
databases, which ATLAS investigated as R&D projects in past years, will be integrated in
production. Finally, we will describe more fundamental developments such as breaking job-to-
data locality by exploiting storage federations and caches, and event level (rather than file or
dataset level) workload engines.

Introduction

The ATLAS Distributed Computing project was established in 2007 with the goal to deliver an
infrastructure for the needs of the ATLAS experiments in terms of data handling and data processing.
ADC consists of two main working areas: the software development and computing operations. The
two activities show a very large overlap, since ADC from the beginning pragmatically decided to
adopt an operations-driven approach. This has been recognized as a key component of the success of
ATLAS computing during the LHC Run-1 and will be preserved in the future. The ATLAS distributed
computing system relies on resources spread over approximately 100 computer centres around the
world and embraced the Grid paradigm to enable data movement and data processing across resources.
The ADC software stack therefore builds on top of WLCG baseline services, implementing the
specific aspects of the ATLAS computing model. The success of the ATLAS computing framework during LHC Run-1 can be easily understood looking at the key figures: the workload management system managed to execute up to 1.5M jobs/day over more than 100 sites, filling in peak periods all available CPU resources and even resources beyond pledge (reaching 200K CPU cores simultaneously occupied). It demonstrated additionally the capability to keep this pace at steady state for many consecutive days. The ATLAS data management system was capable to deliver an aggregated traffic exceeding 10GB/s, again sustained over a period of many days, without the loss of even a single RAW event. The system has been built for reliability following modern techniques and best practices for high availability services (e.g. resiliency/load balancing, monitoring and alarming) so that the core ADC services have shown availabilities larger than 99.9% over the last several years, including the time for upgrades. Finally, the system has been improved to deliver a very high level of automation, so that it can be run by a very reduced team of experts for central operations, allowing also remote shift teams from ATLAS institutes to contribute to the operational effort.

LHC Run-2 will present new challenges for distributed computing: the ATLAS collaboration foresees to collect detector data at much higher trigger rate (1KHz, to be compared to the average rate of 450Hz during Run-1) and higher luminosity (which will therefore yield a higher event pile-up) and this implies the need to produce a higher amount of simulated data as well. This will change the scale at which the ATLAS distributed computing system will have to operate: more data will have to be managed, moved across the network and accessed through the storage interfaces; more jobs will have to be executed in the distributed environment for both data simulation and data analysis; the footprints of the payloads will become more demanding (higher memory usage, need for longer jobs). Unfortunately, this increase in scale and complexity will not be accompanied by a commensurate increase in resources (both CPUs and storage). We can only expect a simple resource increase based on Moore’s law, which will not be sufficient to accommodate the scenario described above, if we operate the system similarly to Run-1. Therefore, ATLAS computing went through a major revision of the core components and the established workflows and planned a system evolution to be able to cope with the challenges of the new years of data taking. This paper will present an overview of such evolution, referring to more detailed contributions for the specific description of various subjects.

The Workload Management System

The ATLAS Workload Management System builds on top of the core service for Production and Distributed Analysis (PanDA). PanDA consists of a workflow engine for job brokering and dispatching to the distributed computing resources, based on the concept of late binding of jobs to CPUs. The distributed analysis system consists of a set of client tools injecting user workflows into the PanDA server for remote execution. The Production System (Prodsys) instead is based on a task manager translating physics requests into collections of jobs to be injected in PanDA. Prodsys is probably the oldest component in ATLAS computing and in recent years has shown its limitations operating at Peta-scale. Particularly, the lack of flexibility in defining complex workflows and the static definition of jobs does not allow optimizing the utilization of CPUs in a heterogeneous environment such as WLCG. Therefore, ATLAS is working on the next generation Prodsys-2. The new system will consist of two services stacked on top of PanDA which will remain the core component: DeFT will offer an interface for the definition of meta-tasks (e.g. the request of 1M simulated events with some software tag for simulation and reconstruction) and will convert the meta tasks into real chained tasks. JEDI will define the jobs from the various tasks and inject them in PanDA. The capability of JEDI to create jobs “just-in-time”, based on information of previous jobs will allow to optimize the job requirements to fit the available resources and will allow to optimize task chaining such as merging of output data into larger files, more suitable for the data transfer and bookkeeping systems. Furthermore, DeFT and JEDI will offer a new platform for distributed analysis,
where most of the current functionalities will be moved from client to server side, offering a real state
game for analysis tasks. A detailed description of Prodsys-2 is provided in [1] and [2].

The Data Management System

The ATLAS Distributed Data Management System (DDM) consists of a set of resilient services for
data location, data movement and data deletion. It builds on top of WLCG common baseline services
such as the File Transfer Service and the LCG File Catalogue and complements their functionalities by
implementing the ATLAS specific concepts and policies.

1.1. The next generation of ATLAS Distributed Data Management: Rucio

While DDM fulfilled with large contingency the use cases of Run-1, several improvements need to be
foreseen. Today for example the management of disk space is not optimal: Grid storages are
fragmented in hard partitions, and for accounting reasons, multiple copies of the same data exist at the
same location under different namespaces. Additionally, new technologies in terms of file access and
transfer protocols are very difficult to integrate in the system, while they would offer considerable
improvements in the performance and the network utilization. Finally, the current system based on
datasets as the unit of data location and replication proved to be too coarse grained for some use cases
such as dynamic replication and deletion based on popularity (number of file accesses from Grid user
jobs). ATLAS agreed therefore to go through a major revision of the data management model (and
consequently the data management framework) in order to address the shortcomings above. This
project called Rucio is described in detail in [3].

1.2. Federating ATLAS storage system using Xrootd: FAX

Complementary to Rucio, which will serve as the engine for data management and planned data
transfer, ATLAS has been working on building a federated storage infrastructure. The FAX initiative
[4], which started two years ago as an R&D project, proposed and is currently commissioning an
overlay system on top of the Grid storage at the various computer centres, based on the xrootd
protocol, offering catalogue-less file access over the wide area network. FAX will initially be utilized
by the ATLAS Grid jobs to fall back to remote storage in case of failures in data access to the storage
local to the job. This should improve the production and analysis reliability while offering the
possibility to commission the system at scale. At a later stage, ATLAS foresees to relax the current
analysis model based on data locality (CPUs and data co-located at the same site) into a more flexible
situation where jobs would prefer data locality but be ready to accept some data access through the
WAN, if this improves the overall efficiency of resources utilization. Obviously the mechanism would
work more efficiently if some local caching could be provided by the federated infrastructure and if
the workload management system could be made cache aware. While this has not yet been
implemented nor demonstrated, it will be the next step in optimizing the CPUs vs storage vs network
multidimensional matrix.

Enabling the usage of opportunistic computing resources

The previous sections described the main evolutions in order to optimize the usage of WLCG
resources offered to ATLAS through Grid interfaces. More opportunistic resources however might
become available and ATLAS needs to be ready to utilize them at least for a meaningful subset of use
cases. In the last two years, ATLAS launched an R&D initiative aimed to explore how to better utilize
those resources. We identified at least two areas of work, which will be briefly described here.
1.3. Cloud infrastructures and Cloud interfaces

A Cloud infrastructure allows to “elastically” demand resources through an established interface. It will provide (if possible) a virtual machine and instruments to manage and customize it. It is then up to the user (in this case the ATLAS computing services) to utilize it. This represents a rather radical change with respect to the standard Grid Computing Elements, offering a gateway to inject payloads to some batch system: the user becomes “administrator” of his cluster. Very quickly over the last years, more and more resources have become available through cloud interfaces. For example, ATLAS decided to provide access to its High Level Trigger farm (more than 15K cores) through the Openstack interface during the periods of shutdown, and several academic facilities are offering free of charge access to their infrastructure through a cloud interface. In other cases, commercial clouds like Amazon and Google are becoming viable under very special circumstances such as grants of resources for research or collaborative purposes, or favourable pricing models such as Amazon spot pricing. ATLAS therefore invested quite some effort in integrating its production system with cloud infrastructures and demonstrated its capability to utilize cloud resources (both the HLT and commercial clouds) with efficiencies comparable to the “standard” Grid submission. The setup is based on the HTC ondr scheduler [5] to aggregate the cloud Virtual Machines into a pool of resources to be utilized by PanDA as if they were Grid resources. The next steps will consist in automating the process of managing the VMs since still today it requires quite some efforts in managing their creation and lifetime through the cloud APIs. This work is describe in detail in [6].

1.4. High Performance Computing

High Performance Computing (a.k.a. supercomputers) offers for High Energy Physics an important opportunity worthy of exploration. While the HEP use case (processing of independent events) is not the one for which HPCs have been designed and optimized, still there is the possibility to parasitically utilize empty cycles at many centres. Unfortunately the task is complicated by the very wide spectrum of architectures and, most importantly, site policies. At most sites, no external connectivity from the CPU core to the outside world is allowed and the payload can rely on a fairly small amount of local disk storage capacity (with respect to what we are used to for Grid jobs in WLCG). Additionally, in most cases it is not possible to ask for pre-installation of Grid clients as it is done in WLCG and the distribution of the ATLAS software itself (now days relying on the CERNVM file system) becomes also problematic. Some of the architectures are not even x86_64 compatible and there is no build of the ATLAS software which today can run on such resources. Therefore, it is unlikely that one solution will fit all possible scenarios. HPC exploitation, started as an R&D project, is today a core activity coordinated by ATLAS distributed computing. An architecture compatible with most restrictions indicated above has been defined and is being commissioned: the arcControlTower, a component successfully already in use since many years to enable ATLAS Grid jobs on Nordugrid resources (accessible through the ARC middleware) will function as edge-service between HPC sites and ATLAS/WLCG central services. The ARC Computing Element functions both as job execution service and data manager for the HPC job slots. With regard to software compatibility with the HPC architecture, many core packages have been running successfully on non x86_64 HPC resources (ROOT, Geant and several physics generators), which means some use cases such as event generation (today utilizing more than 10% of ATLAS Grid resources) could be executed on those architectures.

1.5. The ATLAS Event Service

In many aspects of ATLAS distributed computing, being able to deal with sudden termination of the job execution at any point in time while preserving the produced data would be a great advantage. For example, cloud resources at spot price are very cost effective, but the cloud provider can reclaim the resource back at any moment and with no prior warning. Usage of HPC resources for HEP activities
needs to parasitically utilize empty cycles which could be available for a very short period of time.
Even at WLCG Grid sites, the computer centre administrators would benefit from short job lifecycles in order to better schedule and perform non-transparent system upgrades, reducing the time lost to queues draining. At the same time, the most CPU intensive ATLAS workflow, Geant4 full simulation, while representing the best use case for HPC and Cloud utilization thanks to the very limited I/O requirements, is also the most time consuming per single event processed (up to 12h to execute a payload producing 50 events). The ATLAS Software and Computing teams have therefore decided to initiate a project to evaluate and possibly build a system capable of executing payloads at the granularity of the single event. Such Event Service would deliver event definitions (or tokens) to be picked up and executed on computing resources by the ATLAS pilot framework. The event tokens would sequentially be passed to Athena running in multi process mode and contain all relevant information to allow retrieving the input data from remote storage. The output data would be stored on a per event basis to a central aggregator, producing the files to be persistently stored. The new JEDI component would deal with the per event bookkeeping, as well as providing the workflow engine. A prototype of the Event Service based on existing ATLAS components is being implemented and should be tested at least at low scale and for a subset of use cases starting from 2014.

Relational and NoSQL databases

ATLAS Distributed Computing has been relying on Oracle relational database services for basically all core services. Oracle services at CERN today provide the engine for the state machine of PanDA and Prodsys-I, the back end for the DDM dataset and file catalogues and the back end for all relevant monitoring applications (dashboard and PanDA Monitor). Oracle services at CERN and some other Tier-1s offer the state engine for WLCG services that ATLAS relies upon (e.g. the FTS back end), the back end for ATLAS condition data access through Frontier and the back end for the ATLAS TAG data (a condensed version of the reconstructed data). Some other relational database technologies are in use for more limited and lower scale use cases, for example MySQL is utilized as local cache for DDM transfer agents and as the Hammercloud [7] database. The Oracle technology and in particular the CERN Oracle service has been working very well for ATLAS during Run-I at its scale. A very big improvement in terms of performance and stability has been achieved after the migration from Oracle11g to Oracle12g in January 2012. This was accompanied by a hardware upgrade in the RAC Tier and the back end for the ATLAS TAG database, and as the Hammercloud [7] database. The Oracle technology and in particular the CERN Oracle service has been working very well for ATLAS during Run-I at its scale. A very big improvement in terms of performance and stability has been achieved after the migration from Oracle11g to Oracle12g in January 2012. This was accompanied by a hardware upgrade in the RAC Tier and the back end for the ATLAS TAG database, and as the Hammercloud [7] database. The Oracle technology and in particular the CERN Oracle service has been working very well for ATLAS during Run-I at its scale. A very big improvement in terms of performance and stability has been achieved after the migration from Oracle11g to Oracle12g in January 2012. This was accompanied by a hardware upgrade in the RAC Tier and the back end for the ATLAS TAG database, and as the Hammercloud [7] database. The Oracle technology and in particular

At the same time, it was recognized that some ATLAS use cases would better fit with a non-relational solution. The respective WLCG Technical Evolution Group, with strong contributions from and based on the experience of the ATLAS NoSQL R&D initiative, indicated Hadoop as the baseline technology for NoSQL databases in WLCG. ATLAS has adopted Hadoop as back end storage for the DDM Accounting service as well as for the DDM log files data mining system and gained production experience with it. A major project under development in ATLAS Distributed Computing, the Event Index [8], is considering Hadoop as the most suitable candidate for the service back end. The Event Index consists of a complete catalogue of all ATLAS events in any format. Its aim is to deliver a lightweight system offering a subset of the functionalities of the current ATLAS TAG database, focusing on a restricted number of use cases, such as single event lookup (event picking), skimming of reconstructed data files, completeness and consistency checks and providing the inputs for the Event Service. The system is under development and should reach maturity by the beginning of Run-2.

The ATLAS Grid Information System
The described services providing the infrastructure for ATLAS Distributed Computing need a system to provide centralized and consistent configuration across them. ATLAS developed the AGIS [9] service for this purpose. AGIS fetches information from Grid providers such as GOCDB/OIM and the WLCG BDII and complements the ATLAS specific definitions (e.g. PanDA queues and PanDA resources, DDM endpoints). The result is stored in a centralised Oracle database and exposed through a web interface as well as through a REST API. AGIS has been designed for high concurrence access and supports a caching system. Authentication and authorization based on single user DNs as well as VOMS groups and roles is built in. AGIS is used in production for services and clients configuration since 2012 and improved drastically the rationalization of information across ATLAS computing services and clients.

Conclusions

The ATLAS Distributed Computing project delivered an infrastructure and provided the operational effort for computing activities during LHC Run-1 data taking, coping successfully with ATLAS requirements and with contingency as well. In many cases, expectations were significantly exceeded. Run-2 will present new opportunities and challenges, and for some years ADC has been engaged in a major, non-disruptive evolution of core components. Several activities begun as R&D projects have matured to production quality and today provide critical services for ATLAS. Other products went through design studies and are in the development process, to be ready and commissioned by the start of Run-2.

Bibliography

[1] The ATLAS Collaboration, "Evolution of the ATLAS PanDA Workload Management System for Exascale Computational Science", CHEP 2013, Amsterdam, NL
[2] The ATLAS Collaboration, “Task Management in the New ATLAS Production System”, CHEP 2013, Amsterdam, NL
[3] The ATLAS Collaboration, "The next generation of large scale distributed system for ATLAS Data Management", CHEP 2013, Amsterdam, NL
[4] The ATLAS Collaboration, “Data Federation Strategies for ATLAS Using XRootD”, CHEP 2013, Amsterdam, NL
[5] Douglas Thain, Todd Tannenbaum, and Miron Livny, "Distributed Computing in Practice: The Condor Experience" Concurrency and Computation: Practice and Experience, Vol. 17, No. 2-4, pages 323-356, February-April, 2005.
[6] The ATLAS Collaboration, “ATLAS Cloud Computing R&D”, CHEP 2013, Amsterdam, NL
[7] The ATLAS Collaboration, “Grid Site Testing for ATLAS with Hammercloud“, CHEP 2013, Amsterdam, NL
[8] The ATLAS Collaboration, “The ATLAS EventIndex: an event catalogue for experiments collecting large amounts of data”, CHEP 2013, Amsterdam, NL
[9] The ATLAS Collaboration, "AGIS: The ATLAS Grid Information System", CHEP 2013, Amsterdam, NL