The ATLAS Tier-3 in Geneva and the Trigger Development Facility

S Gadomski[1], Y Meunier[1], P Pasche[2], J-P Baud[3] for the ATLAS collaboration

[1] Université de Genève, Département de physique nucléaire et corpusculaire, 24 Quai Ernest-Ansermet, CH-1211 Genève, Switzerland

[2] Université de Genève, Division informatique, 24 rue du Général-Dufour, CH-1211 Genève, Switzerland

[3] CERN, IT Department, CH-1211 Genève, Switzerland

szymon.gadomski@unige.ch

Abstract. The ATLAS Tier-3 farm at the University of Geneva provides storage and processing power for analysis of ATLAS data. In addition the facility is used for development, validation and commissioning of the High Level Trigger of ATLAS [1]. The latter purpose leads to additional requirements on the availability of latest software and data, which will be presented. The farm is also a part of the WLCG [2], and is available to all members of the ATLAS Virtual Organization. The farm currently provides 268 CPU cores and 177 TB of storage space. A grid Storage Element, implemented with the Disk Pool Manager software [3], is available and integrated with the ATLAS Distributed Data Management system [4]. The batch system can be used directly by local users, or with a grid interface provided by NorduGrid ARC middleware [5]. In this article we will present the use cases that we support, as well as the experience with the software and the hardware we are using. Results of I/O benchmarking tests, which were done for our DPM Storage Element and for the NFS servers we are using, will also be presented.

1. Introduction

The ATLAS group at the University of Geneva has built a large Tier-3 facility. In addition to serving as a local computing facility for the final stages of data analysis done by the group, our farm has become a Trigger Development Facility. Anyone working on the ATLAS Trigger system [1] is entitled to have an account at our Tier-3. Commissioning and development of the Trigger software are possible thanks to:

- availability of the latest software releases, including nightly builds, via AFS from CERN,
- access to Trigger n-tuples, which are produced during data reconstruction at the Tier-0 and copied automatically to our cluster,
- a possibility to transfer small samples of recent data directly from the Tier-0.

We will describe our setup in section 2. The usage of the system will be presented in section 3 and the performance results of our systems are described in section 4. We conclude by briefly describing the outlook in section 5.

2. The setup

2.1. Hardware and the operating systems

The photograph of our hardware is shown in figure 1. We currently have 63 machines intended for running of physics data processing applications, which have in total 268 CPU cores. 48 of the machines, with 232 cores, are batch worker nodes, running the CERN Scientific Linux 5 (SLC5) operating system. The remaining 15 machines serve for user login and interactive analysis. Most login machines run the SLC5. A few of them still run the SLC4, which is necessary for compilation of user code against old releases of the ATLAS offline software. The login and batch machines represent four different generations of hardware, all from Sun Microsystems (models V20z, X2200, blade 6220 and blade 6240). All the machines have two CPU; the number of cores per PC varies between 2 and 8. The oldest hardware is now five years old.

The storage consists of nine file servers, also all by Sun. Eight of the file servers, containing over 95% of our disk space, are Sun X4500 or X4540 servers. They run the Solaris operating system and we rely on the zfs file system, which comes with the OS, to provide the redundancy against disk failures. The total usable disk space is 180 TB and the reliability record has been excellent so far. In five years of experience we have never lost data through a hardware failure.

We also have five machines that run central services such as the batch system, the grid interfaces, web server, the monitoring and installation server (used to install the Linux machines). In total the hardware in production has 77 machines.

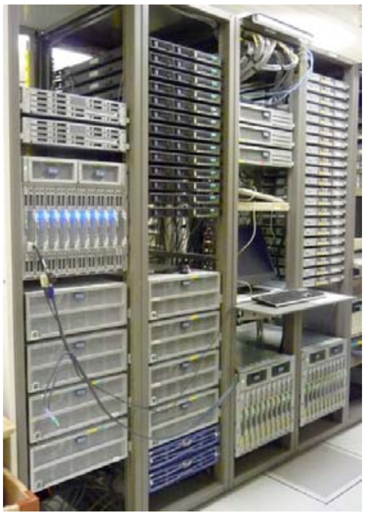

Figure 1. Hardware of the ATLAS Tier-3 at the University of Geneva.

2.2. Batch system and the NFS

The batch system we are using is Torque with the scheduler Maui [6]. Maui is a necessary addition for our setup because it provides scheduling based on fair share, where job priority is lower for users who have used the system heavily during the few preceding hours. This feature is essential in a setup where mass production has to be combined with a quick turnaround time for single jobs. In addition the scheduling gives preference to short jobs.

Most of our storage, 110 TB at the time of writing, is accessible via the standard Network File System protocol, in our case NFS 3, exported by five file servers. Our experience with the NFS 3 is in general very positive. We realize that, because a file system always corresponds to a single machine, we have reached the limits of performance of the NFS 3 for data-intensive tasks. The performance of our storage systems is described in section 4.1.

2.3. Grid middleware: NorduGrid ARC and the DPM Storage Element

Since the beginning, in autumn of 2005, the Tier-3 in Geneva runs the NorduGrid [5] middleware. It was originally estimated that the middleware is the only possibility for a relatively small setup, operated by a University group, as opposed to a computer center. Five years later we still think the NorduGrid middleware is an easier choice for a Tier-3 center. The entire configuration is contained in one file. As we have found out ourselves, it is possible to get started working alone and reading the documentation. Once the setup is running, the maintenance load is minimal. We continue to see the NorduGrid as a simple, low-maintenance way to make our spare CPU cycles available to the rest of the ATLAS collaboration. Details of the usage are provided in the next section.

Since Summer 2009 we also operate the Disk Pool Manager Storage Element, developed at CERN [3]. The DPM SE provides the following functionality, which we see as absolutely vital:

1. The SRM interface, necessary for integration with the Distributed Data Management system of ATLAS.
2. Distribution of logical directories, which correspond to ATLAS datasets, over multiple physical file servers.
3. Sufficient I/O performance for our size of the batch system, if one assumes that the application is the ATLAS offline software Athena.

The installation of the DPM was done in close collaboration with the CERN IT. Our setup was the first instance of the DPM SE in which the file servers run the Solaris OS. We currently have four file servers, with the total usable space of 70 TB, in the DPM system. Upgrades of storage foreseen later this year and next year will contribute more space to the SE.

3. Observed usage of the system

Thanks to the NorduGrid, the T3 in Geneva has been contributing to the ATLAS grid since September 2005. Since Spring 2007 we have observed a gradually increasing usage of the batch system by the local users, the Geneva ATLAS group. Recently our colleagues working on the neutrino experiments have started using the system, as well as contributing to its further development.

The data shown in figure 2 is based on our batch system statistics. The figure on the left shows the total of CPU days per month, counting wall time hours. It should be noted here that the system available in 2005 had below 9% of the present CPU core count. Over six months preceding the CHEP 2010 conference (April to September) the batch machines were on average 45% used. It would be possible to increase the average usage by allowing more grid jobs. The maximum number of grid jobs is currently limited to 120, for 232 slots. This choice represents a compromise between maximizing the usage, and providing quick turn-around time for local users. We think that providing a quick turn-around time, i.e. having CPU accessible on short notice, is vital for a Tier-3, which is meant to be mainly a local facility for final data processing steps. The pie chart on the right hand side of figure 2 shows the breakdown of the CPU usage, based on batch system statistics, for the six-month period of April to September 2010. The grid usage, in spite of its lower priority, still accounts for nearly 50% because of the nearly constant demand of the ATLAS production system.

Running local batch jobs over the data in our Storage Element is an efficient way to work. The DDM data transfers are reliable; we estimate that their failure rates are below 10^{-3}. Local batch jobs can be made very reliable, with failure rates of the order of 10^{-4} or 10^{-5}. Once the data are in the SE, the turnaround time is short.

The interactive usage is not quantitatively known. We observe that the login machines are often heavily used for final data analysis steps done in interactive sessions, code development and testing on small data samples. Trigger n-tuples, which are automatically copied to Geneva, are regularly analyzed to verify the performance of the real time data selection in ATLAS.

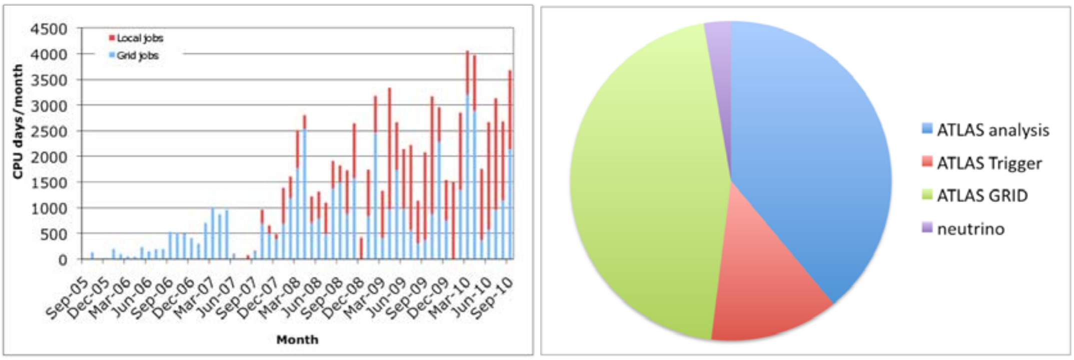

Figure 2. Batch system statistics since 2005 (left) and breakdown of the batch system usage during six months, April to September 2010.

4. Performance of the system components

The ATLAS computing is data-intensive. We therefore pay close attention to the data movement performance of our systems. We have tested the performance of our storage (section 4.1) and we have also calculated the rates at which the data are copied to us from other grid sites (section 4.2).

4.1. Performance of our storage systems

The tests were done using the batch system. One hundred jobs were launched, simultaneously writing or reading the data files. Each job was writing or reading a single file of 5 GB size.

Writing to NFS was done by using the Linux command dd, giving /dev/zero as the data source. Writing to the Storage Element was done by first creating a 5 GB file in a local scratch directory on a disk of the batch worker node (with dd as above) followed by rfcp of the file to the Storage Element. Reading was done by doing cp (from NFS) or rfcp (from the Storage Element) of the 5 GB files to local scratch directories on local disks of the batch worker nodes. The measured performance numbers are shown in table 1.

Table 1. Measured reading and writing speed for the two storage systems of the cluster. The measurements were done using 100 batch jobs reading or writing a 5 GB data file.

Storage system	Direction	Rate [MB/s]
NFS 3, 1 server	Read	300
	Write	200
DPM SE, 4 servers	Read	800
	Write	210

When reading data, we can appreciate the performance of the DPM, which distributes data over multiple physical servers. In our test the writing did not scale so well. This was not investigated further because we are not concerned by the writing performance. The data in the Storage Element is mainly written by grid data transfers, which run at much lower rates (as discussed below). A typical usage pattern for our batch jobs is to read data from the Storage Element, writing a much smaller output data volume to the NFS, for further analysis in ROOT [7].

In figure 3 an example of our network monitoring plots is shown. The plot shows the total data movement at the cluster, with all machines added together. The data rate in bytes per second is shown as a function of time for a period of one week. The green line shows data getting in; the blue line shows the data getting out. Data movement internal to the cluster produces green and blue traces that follow each other. One can see two large peaks, which correspond to batch runs, with batch jobs reading the data off the Storage Element. Rates up to 1 GB/s have been observed, a performance that is slightly better than the test results described above.

The green line running above the blue line indicates a net influx of data to the cluster, running for several days at a rate of ~20 MB/s. (Performance of the data transfers to Geneva is described in the next section.) A short blue spike of around 370 MB/s, observed on Saturday, corresponds to a data copy from Geneva to CERN.

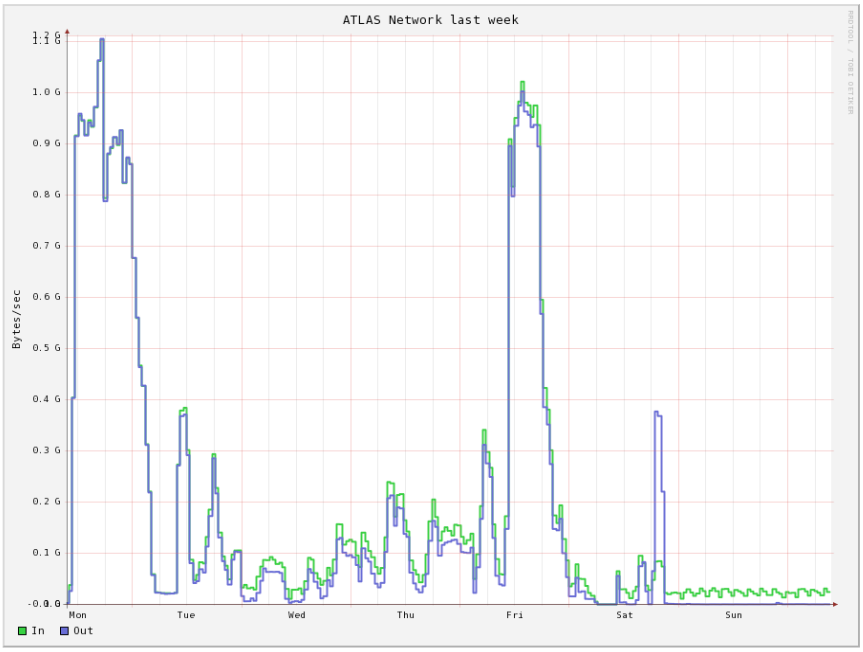

Figure 3. Example of network monitoring showing movement of data during one week.

The performance of our storage would have been sufficient, if the requirements were not changing with time. It has been estimated that an Athena process reading AOD, which we used to consider our most I/O demanding application, needs to read data at around 5 MB/s in order to be CPU-bound on a typical core. It was therefore deemed sufficient to be able to read the data at 800 MB/s in order to feed the batch jobs running in our batch system, where a typical number of running jobs varies between 120 and 150.

A new usage pattern has appeared in Spring 2010. The D3PD datasets, which are ROOT n-tuples, have reached sizes of a few TB. The users started to process such large n-tuples in batch mode, using selection code compiled with ROOT, producing smaller n-tuples for further analysis. In terms of data rate this sort of application can run much faster than an Athena job. Rates of 20 MB/s can easily be reached. In fact such processes are, in our experience, always limited by I/O and never by the CPU. They therefore pose requirements on storage that are in a completely new range. We do not think we can meet such requirements with the technologies we are currently using. The only practical solution found so far was to ask the users to limit the number of simultaneously running hyper I/O intensive jobs, in particular when reading from the NFS. A special batch queue was introduced for that purpose.

4.2. Data transfers to the Tier-3

The rates of data transfers to the Geneva Tier-3 from other grid sites, including but not limited to CERN, were calculated by analyzing time stamps and sizes of the files. Only datasets larger than 10 GB were considered. The exercise was done separately for datasets found on the NFS, which were downloaded using a DQ2 client, and for the datasets in the DPM Storage Element, which are copied via a replica subscription mechanism of the ATLAS DDM system. The results are shown in Figure 4.

One can note a rather large spread of the calculated data rates. We see rates over 60 MB/s with dq2-get and over 100 MB/s with an FTS transfer to the SE, but usually the data movement is slower. The average rates are 6.8 MB/s for the dq2-get transfers and 15.7 MB/s for an FTS transfer.

The hardware and networking of the Geneva cluster would allow for faster data transfers. Our network connection to sites other than CERN is 1 Gb/s. Assuming 50% bandwidth utilization, one could transfer data at rates of ~50 MB/s. Direct transfers from CERN could be done faster because of the 10 Gb/s connection between the CERN/IT and the machine room in Geneva. In case of the transfers from CERN the hardware limit is probably given by the file systems of the machines receiving data. We estimate that this limit is beyond 6 Gb/s. As the hardware limits are between one and two orders of magnitude away, improvements by large factors could be possible without any hardware investments. This would be an attractive prospect for our Tier-3.

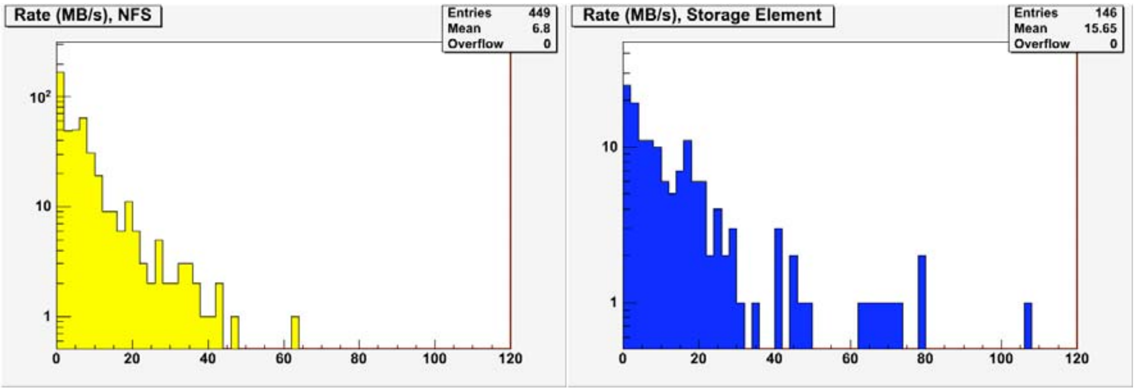

Figure 4. Data transfer rates to the cluster calculated from sizes and creation times of the files found on NFS (left) and in the Storage Element (right). One entry is one dataset.

5. An outlook

Building on the experience gained so far, which was mostly positive, we are planning to develop the cluster further. In the foreseeable future we are planning to continue with the technologies, including grid middleware, described above. Two more file servers, providing additional 34 TB of usable disk space, will be added to the Storage Element before the end of the year. Next year we hope to be able to add another three file servers, as well as 10 more batch worker nodes, which will have 16 CPU cores each. The cluster will be developed as a shared resource for the particle physics groups, not limited to ATLAS. It will also continue to serve all of the ATLAS Trigger community.

References

[1] The ATLAS Collaboration, "Performance of the ATLAS Trigger with Proton Collisions at the LHC", these proceedings (PS32-2-189).

[2] I.Bird, "WLCG - Progress and Challenges", these proceedings (PL-01).

[3] The Disk Pool Manager Storage Element: https://svnweb.cern.ch/trac/lcgdm/wiki/Dpm/.

[4] V.Garonne et al., "Status, News and Update of the ATLAS Distributed Data Management Software Project: DQ2", these proceedings (PS41-5-296).

[5] The NorduGrid ARC Middleware, http://www.nordugrid.org/.

[6] Torque batch system and Maui scheduler: http://www.clusterresources.com/.

[7] The ROOT system, http://root.cern.ch/.