CERN / IT / FIO / FS

Work Report

Report about the ATLAS TDAQ Data Challenge Support by IT-FIO/FS

Author Names **Veronique Lefebure**

Issue date 02/08/2005

Abstract

From 6th of June 2005 till 22nd of July 2005, ATLAS has been running its TDAQ Large Scale Tests on part of the IT LXBATCH cluster at CERN. The tests were successfully terminated after 6 weeks of running on a stepwise increasing farm size from 50 to up to 700 dual CPU nodes. Functionality and verification tests of the ATLAS DAQ and HLT parts of the system were carried out and selected performance measurements were taken. Individual sub-system tests were performed for the online system, the Event Building system, and for the Level2 Trigger system and Event Filter system. On the IT side, the cluster was configured and maintained by the FIO/FS section according to the ATLAS specific requirements, with the help of the ELFms tools. Requirements were on many aspects: hardware specifications, network layout, operating system and specific ATLAS software deployment and upgrades, machine monitoring and problem solving. This document describes how the ATLAS requirements were satisfied by IT.

VERSION	DATE	AUTHOR	CHANGES
1.0	21/07/2005	Veronique Lefebure	Created
1.1	02/08/2005		Integrated German's feedback

Contents

1. Introduction

1.1 Purpose of this document

The purpose of this document is to describe how the current ELFms tools helped IT in satisfying special requirements from an experiment. The particular example of the ATLAS TDAQ Data Challenge concerns worker nodes (CPU's) configuration, monitoring and software deployment.

1.2 Introduction

ATLAS has requested to be provided a cluster of some 700 CPU nodes, to be gradually available according to a predefined schedule (See Table 1), a few hundred nodes being delivered at a time each week, with the proper configuration. Part of the machines was provided by IT-FIO/FS, taken from the batch resources, and part of the machines was provided by the IT-ADC group (87 2.8 GHz machines and 89 2.4 GHz machines).

Date	Number of nodes	capacity reduction for ATLAS	capacity reduction for all experiments
06 - 12June	50 nodes	0 %	0 %
13 - 21 June	100-200	0 %	0 %
22 - 28 June	300	30 %	0 %
29 - 05 July	400	50 %	0 %
06 - 12 July	500	80 %	10 %
13 - 19 July	700 +[1]	100 %	37 %

Table 1 Schedule for resource delivery to ATLAS, and corresponding effect on the LXBATCH capacity.

2. Network Infrastructure

2.1 ATLAS Requirements

1. Importance was given to homogeneity in the network layout. Interference with other users had to be avoided. The layout should not include the SSR backbone as far as possible.

2. UDP multicast transparency over the network was required.

3. It was desirable to have the planned network layout available for the preparation of the configuration, before the start of the tests.

2.2 IT Service

Machines provided to ATLAS were chosen so that the network requirements were satisfied by default.

ATLAS had to ask Bernd personally in order to get the network layout information.

[1] The "+ "sign relates to a new delivery of CPU nodes expected during that time, with no guarantee for its availability.

3. Hardware

3.1 ATLAS Requirements

1. All nodes are dual processors at least 1GHz and 500MB memory

4. Operational Settings

4.1 ATLAS Requirements

1. SLC3 Operating System to be running on the whole cluster
2. ssh access [2]
3. Access to afs
4. All nodes to be time synchronized via ntp
5. Max number of file descriptors per OS: (cat /proc/sys/fs/file-max) should stay equal to 52345 as is the default on LXBATCH nodes.
6. IP port range (cat /proc/sys/net/ipv4/ip_local_port_range) to keep the values [32768 61000] as is the default on LXBATCH nodes.
7. Max number of file descriptors per process to be set to 8196.
8. login is enabled for the following afs accounts: atlonl, effuser, dcuser, doris, werner, sobreira
    - More logins were later enabled as new requests came
9. /pool/online, /pool/hlt, /pool/df, /pool/tdaq needed with read and execute rights to all accounts
    - This requirement was later extended with: /pool/users writable by all enabled users, so every enabled user can create his/her /pool/users/myname directory there
10. no cleanup of /tmp
    - Related to /tmp: it was later requested by ATLAS to remove the quota on /tmp.
11. The system should be cleaned from servers, daemons etc. installed by previous users
12. No automatic update must be performed. Necessary updates should be scheduled for a fixed time/interval to be agreed upon.

[2] There was an issue regarding the use of ssh, which was observed and then fixed by ATLAS: the problem was that, for their tdaq system, process manager agents were started on all involved hosts with ssh by a build-in procedure. Therefore they could not use the command line option. As there were many nodes, they had to avoid having to log in to each node manually to say “yes” to accept the key. Here is the recipe which finally worked: on the HOME directory, remove the directory .ssh, create it, make sure it is not writable by someone else
> ssh-keygen -f identity -t rsa1
(return)
(return)
> cp identity.pub authorized_keys
in .ssh/config file set :
StrictHostKeysChecking no

13. install bittorent.rpm on all nodes

14. Install of a “tdaq” RPM

15. Install of MySQL, PHP, APACHE on a few (3-4) machines, preferably on faster (xeon) dual machines.

4.2 IT Service

Requirements number 1 to 6 did not imply any change with respect to the standard LXBATCH setup. For satisfying the other requirements, special templates have been created in CDB, for defining the software, monitoring and system configuration of the ATLAS cluster, taking as a basis a clone of the LXBATCH (SLC3) templates:

- pro_software_atlas_tdaq (and pro_software_atlas_tdaq_addon)
- pro_system_atlas_tdaq
- pro_type_atlas_tdaq
- pro_type_atlas_tdaq_slc3_monitoring

Hence a new cluster was created, called “atlas_tdaq”. It had to be added into the password system before any machine being installed under that cluster. This was done by hand. The same was needed on the Remedy side.

- File descriptors: the parameters requested by ATLAS (request number 7) were set using the available “interactivelimits” Quattor component. It was defined in the CDB template at the cluster level.

- Interactive and root access were set using the “access_control” Quattor component. They were defined in the CDB template at the cluster level. For each new access to be granted, the cluster template “pro_system_atlas_tdaq” has to be modified (manually) and compiled, and the component needs to be run on all nodes of the cluster.

- For the setup of the /pool subdirectories, an RPM called “CERN-CC-atlas_tdaq” was created containing the requested structure and privileges. That RPM was deployed on the cluster using SWrep and SPMA, as for all other RPM’s included in the “pro_software_atlas_tdaq” template.

- On “standard” FIO/FS clusters, the automatic cleanup of /tmp is done via a Sensor Alarm actuator or by a daily cronjob. To prevent the automatic cleanup of /tmp, it was necessary to switch off the actuator in CDB (in the “pro_type_atlas_tdaq_slc3_monitoring” template). In addition, the “CERN-CC-tmpwatch” RPM had to be removed from the CDB configuration, as it contains the cron job. This implied removing a list of RPM’s depending on that one (with the agreement of Atlas): xmltex, tetex-afm, tetex-fonts, passivetex, tetex-latex, latex-xft-fonts, tetex-dvips, tetex, jadetex, tetex-xdvi, docbook-utils-pdf, lyx-qt, xmlto, a2ps, docbook-utils, linuxdoc-tools, gtk-doc.
  Regarding the removal of the quota on /tmp: from the quota FIO-expert, it was advised to run “/sbin/quotaoff –a” on all nodes, because removing the quota configuration at the CDB configuration level would require a reinstallation of the machines, which was not possible. It turned out that this was not enough as there was also a quota cron job running daily. Hence it was necessary to remove the “CERN-CC-quota” RPM from the software configuration (in the “pro_software_atlas_tdaq” template).

- LSF RPM’s and monitoring configuration were removed from the CDB templates.

- There was no software update needed from the IT side (no security upgrade). Upgrade of the ATLAS software was made by IT, on demand. There were 6 upgrades of "tdaq-v0102" and 2 upgrades of "tdaq-v0103", each RPM being of about 140MB. These RPM packages were the biggest of all packages stored on the Swrep software repository. ATLAS did not have the time to try to modularise their code in order to have smaller RPM to distribute. This did not cause any problem in this case, although the download times were rather large (of the order of 20 minutes, with the RPM's being transferred in parallel to all nodes via the headnodes).

- The installation of "bittorrent" was done after it was agreed so during a special discussion meeting, because of the security issue related to the usage of that tool (see http://security.web.cern.ch/security/software-restrictions/). See the summary of that meeting in Appendix A below.

- Installation or upgrade of the ATLAS software was made by IT, on demand, and after a checking of its content by German. There were 6 upgrades of "tdaq-v0102" and 2 upgrades of "tdaq-v0103", each RPM being of about 140MB. Each time, the following steps had to be accomplished:

  1. upload the new RPM onto the software repository (with the "swrep-client" application)

  2. Wait for the repository template to be updated by the cron job (maximum 30 minutes)

  3. Edit the pro_software_atlas_tdaq template so that the new version of the RPM is included, compile and commit the template in CDB.

  4. Run SPMA on all nodes of the cluster (with "notify" or "wassh")

  5. Check for success of the SPMA run, correct problems if any (there were some problems of space, at the beginning of the data challenge period, but afterwards, ATLAS users took care of cleaning up the space from log files and coredumps)

  Because both "v0102" and 'v0103" RPM's had the same name "tdaq" and had to be concurrently installed, there was a problem when installing both at the same time, because of a feature of RPM that decides to install only the one with the most recent version.

- The "pro_software_atlas_tdaq-addon" template was created for the installation of MySQL, PHP and APACHE on 3 machines of the cluster. The daemons were started. Permissions were given "by hand" on /var/www/html.

5. Monitoring of the Cluster

5.1 ATLAS Requirements

1. ATLAS requested to be provided a file containing the "online" list of hostnames of all machines belonging to their cluster, with a strict syntax (1 hostname per line, with the domain name).

2. ATLAS requested to be provided a file containing the "online" list of hostnames of all machines belonging to their cluster and being usable, with the same syntax as described previously.

3. ATLAS requested to be notified of any change in any of the two lists mentioned above, and wanted to know what the response time was.

4. The lists would preferably be available locally on any node.

5. The disk capacity and the memory size had to be communicated to ATLAS for each node.

6. On two of the three machines where a MySQL database had been installed, it was requested to increase the sampling frequency of some quantities measured by LEMON.

5.2 IT Service

1. The “online” list of machines present in the ATLAS TDAQ cluster was provided via a script which would access the data stored in CDBSQL (list of machines in cluster “atlas_tdaq” with state “production”.

2. For the list of “usable” machines, the LEMON heartbeat information was used in addition (using the Lemon::FmonMRs_perl module).

3. The CDB field “/system/cluster/itcmccemail” was used so that a carbon copy of any Remedy ticket open for a machine of the ATLAS ticket was instantaneously sent to that mail recipient. This turned out to be quite useful, in particular in the case of some “SWAP_FULL” alarms received by the Operators.

4. The script providing the dynamic list of nodes was available via afs on all nodes.

5. The same script was enhanced with “—disk” and “—ram” options when respectively the disk capacity or the memory size information of all nodes available in the cluster at a given moment, was requested.

6. The sampling period was decreased from 300 seconds to 15 seconds for the “CPUutil”, “Mem_in_use”, “networkIO”, “Interrupts”, and “DiskIOSummary” metrics, using the CDB configuration parameters “/system/monitoring/metric/_<metric ID>/period" =15;” and reconfiguring the monitoring agent on these two nodes with the corresponding NCM Quattor component.

Because of the special requirement of ATLAS regarding the /tmp file partition (see previous section), it was necessary that the monitoring configuration be modified in such a way that, in case of /tmp increasing above a threshold (90%), an email was sent by LEMON to the ATLAS users, instead of triggering an alarm on SURE (delete "/system/monitoring/metric/_20012/reference" in the template to prevent alarms to be sent to the operators).

6. IT Operational Aspects

6.1 Machine Installation

The installation of all machines was done using the “OSUpgrade.pl” tool[3]. In addition, in order to keep at a maximum of LXBATCH capacity and at the same time be able to deliver the resources on time to ATLAS, the “/etc/shutdowntime” file trick was used to prevent job to start on the machines, which would not be completed by the time where the machines had to be given away.
The same tool was used at the end of the challenge for the migration of the machines back to the LXBATCH cluster.

[3] OSUpgrade.pl is a script wrapping all tools used in FIO/FS for managing machines. I t has been implemented to automate the soft move of large number of machines (for kernel upgrade, operating system upgrade, or move of machines from one cluster type to another). It makes use of CDBSQL, CDB toolsuite, SMS, Remedy, LEMON, SURE, LSF, PrepareInstall. See more info on the LEAF web page.

At the beginning, the machines were simply rebooted. But it turned out that there were residual files which where not properly cleaned from the LCG software previously installed[4], that needed a manual cleaning. Consequently it was decided to re-install the machines from then on, before giving them to ATLAS.

The final capacity allocated to ATLAS was distributed as shown in the Table 1 below, while the SLC3 LXBATCH capacity was down to the numbers given in Table 2.

Hardware model	nodes (Ksi2k)
Elonex 2.8GHz	87 (180.44)
SEIL 2.4GHz	551 (938.90)
SEIL 1GHz	85 (78.37)
Sum	723 (1197.71)

Table 2 Resources allocated to the ATLAS TDAQ during the period of maximum allocation.

Hardware model	nodes (Ksi2k)
Transtec 2.8GHz 64	64 (132.74)
Elonex 2.8GHz 64	121 (250.95)
Elonex 2.8GHz	240 (497.76)
SEIL 2.4GHz	126 (214.70)
SEIL 1GHz	6 (5.53)
TechAS 800MHz	16 (10.88)
Elonex 800MHz	61 (41.48)
TechAS 600MHz	52 (28.08)
Sum	686 (1182.13)

Table 3 Minimum LXBATCH capacity reached during the ATLAS TDAQ challenge.

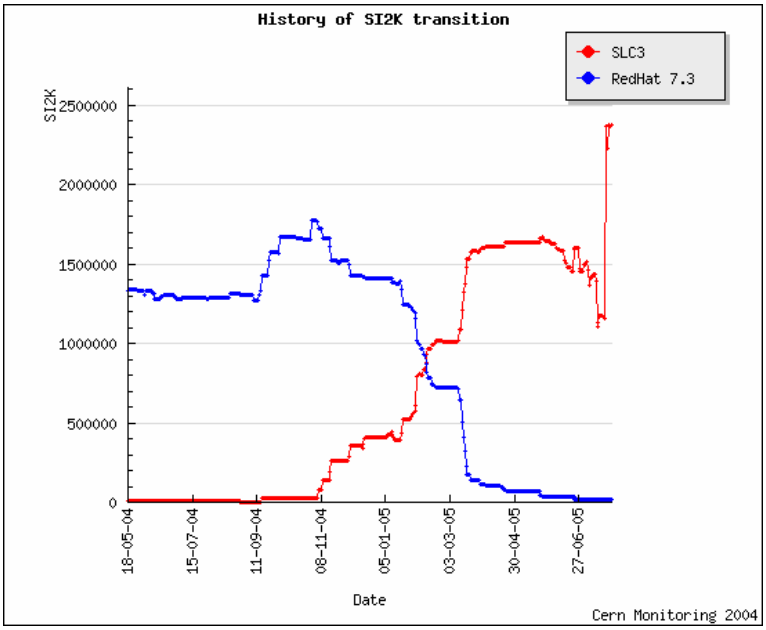

Figure 1 The decrease of LXBATCH capacity during the ATLAS TDAQ challenge period can be seen, starting on the 6th of June 2005, and ending on the 22nd of July when all resources where moved back to LXBATCH.

6.2 RPM feature

In the future users should be warned not to use the same RPM name if more than one version of it needs to be concurrently installed, as RPM installs only the one it considers as being the most recent.

[4] The problem was related to the definition of the $LD_LIBRARY_PATH environment variable, because of left-over files /etc/profile.d/lcgenv.csh and /etc/profile.d/lcgenv.sh.

6.3 Remedy Tickets

All together, 792 Remedy tickets where created for the “atlas_tdaq” cluster, among which 582 were Log Only tickets related to the installation. That makes 210 tickets for other various reasons:

- 71 for RFIOD problems due to the deployment of the “castor2” software (which was in fact not needed on that cluster, but was performed by an other Service Manager)
- 26 “SPMA_OR_ASIS_ERROR” because there was no space left for installing new version of the ATLAS software RPM’s
- 22 “ROOT_FS_FULL” alarms
- 20 installation failures
- 10 “SBATCHD_WRONG” alarms before the LSF configuration be properly removed
- 6 “SWAP_FULL” alarms
- 5 “CPU_WRONG” alarms
- 5 disks problems, spotted by SMARTD, and that required a disk change after stress-tests have confirmed the disk failures
- 2 “HIGH_LOAD” alarms
- 1 “TMP_FULL” alarm, before that alarm was disabled
- 1 “DMA_DISABLED” alarm

7. Possible Enhancements of ELFms Tools

1. Network information:
   The network layout and parameters is currently not directly available to the Service Managers.
2. Interface for privileged users to upload and deploy software (implies ACL’s on the relevant CDB templates)
3. On-demand update of the repository CDB templates (in order not to have to wait for the cron job that runs every 30 minutes).
4. Interface for privileged users to give interactive or root access to a list of machines software (implies ACL’s on the relevant CDB templates as well)
5. Documentation about which components exists and what they can do (searchable by keyword, for example “File descriptors”)
6. Prevent intervention on a cluster (editing of a cluster CDB template) by external service managers (See below, castor2 software deployment)
7. Automatic registration of new clusters in the password system.
8. Automatic registration of new clusters in Remedy.
9. We have many critical tools depending on ORACLE (PrepareInstall, SMS, wassh, …). When ORACLE is down, any machine management activity is blocked. There needs to be a fail-over solution coming from the ORACLE side.
10. It was found that, for the move of the full ATLAS cluster back to LXBATCH, SMS could not cope with 700 nodes at the same time. This was reported and has been fixed by now.
11. There were few hosts running “zombie” LSF jobs: jobs were running but were unknown by LSF itself. These jobs caused a delay in the migration of the machines from the LXBATCH cluster to the ATLAS cluster as the processes had to be checked, killed and possibly the users had to be informed. (This was a consequence of an LSF problem occurring several days before). It would consequently probably be good to have a sanity check of the LSF job tracking system.
12. A new file called “/etc/quattor_install_info” has been put in place, created locally on the host by the kickstart process. The file contains the time of installation and the name of the “pro_type_*” CDB template which was used for the installation. The file, together with associated LEMON metrics, will help in the future to cross-check that the installation of a host has been successful. As a work around, we had to wait for a “safe” amount of time before putting a machine back into production after the reinstallation.

8. Questionnaire for Future Requests

1. Cluster Name (-> password, Remedy)
2. Hardware requirements
3. Network requirements

4. Software requirements (Operating System)
5. File partition description
6. Quota
7. Monitoring of partition space? If not, what to do in case of problems?
8. Private Software: RPM size, names, content
9. Interactive/root access
10. Monitoring sampling
11. Contact persons, responsible persons (CC of Remedy tickets)
12. Software automatic upgrade (security upgrades)
13. Support expectations (24/7?)

9. Acknowledgments

I want to thank Harry Renshall and Thorsten Kleinwort for having put me fully in charge of the ATLAS TDAQ setup and support. It was a good opportunity to cover aspects of machine configuration different from the "standard" cases encountered in the FIO/FS section. It was also very nice to be in very close contact with the users.

10. Appendix A (bittorrent installation request)

Fri 6/10/2005 9:10 PM
Hello Mr. Kleinwort,

Many thanks for your excellent summary, and the constructive spirit of all the participants. I just want to add a few minor clarifications:

Thorsten Kleinwort wrote:

This is a summary of the meeting between IT-FIO and ATLAS online on the issue of the software distribution tool, 10.06.2005 16:00. Please let me know if I have forgotten something important, or if you disagree with something you want to have changed.

Attendees:
IT/FIO: German Cancio, Veronique Lefebure, Vlado Bahyl, Thorsten Kleinwort
ATLAS: Doris Burckhard, Haimo Zobernig, Hegoi Garitaonandia Elejabarrieta

ATLAS wants to investigate the software tool 'Bittorrent' as a software distribution tool for their software, and they want to investigate it on the test cluster they have for the Atlas Online DC. The software they want to distribute is around 5 GB uncompressed and around 1.5 GB compressed. They have done already some tests of this tool on a smaller scale.

Currently, the size of the compressed container file is 1.8 GB.

IT has raised concerns about this software, because it is a peer-to-peer file sharing application, and therefore it is banned for personal use at CERN (http://security.web.cern.ch/security/file%2Dsharing/). In addition, with the ELFms and Quattor framework, IT/FIO have a very powerful tool for distributing software in RPM format, and they are proposing and recommending to use this tool for distributing the ATLAS specific software as well. ATLAS was agreeing to have a look and try to test this software distribution as well, but for the moment, there is no mechanism to create RPM's out of their software repository (CVS), and they have just started to try to create one big RPM out of the software. A better way of splitting up the big amount of software into logical units is not easy to do, and needs some investigation beforehand. This will be not ready for this DC.
IT agreed that it would be not a good idea to have one big RPM containing the whole 5 GB of data, but having some reasonable amount of RPM's with reasonable size, where the software is split into logical units. In general they are used to handle this amount of data for such a number of machines, and the infrastructure is built up for this, like taking into account the current network topology in the CC. It is unclear how a peer-to-peer tool would behave, especially where the network on with the current DC will run is not exclusively reserved for the DC, but shared with the standard batch activity.

ATLAS pointed out that the amount of network traffic can be configured in the Bittorrent configuration and they will watch the traffic carefully. They asked for means of monitoring network traffic. Lemon can give only a view from the host network traffic. ATLAS will contact IT/CS for more information.

As a Bittorrent peer receives pieces of a file, it also becomes a server for the pieces it already has. The Bittorrent peers can be configured to respect a maximum upload rate (to other peers) and the maximum number of simultaneous uploads, among many other parameters. With additional information on the network topology connecting the nodes of the atlas_tdaq cluster it should be possible to avoid loading the network with too much traffic.
We will be glad to get advice from IT in this respect.

ATLAS also pointed out that they want to start their investigations as soon as possible, which means this weekend, beside the network intervention that will take place on Sunday. The possibility of using AFS as a shared file system was also shortly discussed. This could be an option as well, because AFS reliability has improved, and it is very configurable and therefore tunable. Though AFS will probably not be available in the ATLAS online cluster, it could be a general test for a network file system (NFS). This has to be looked at as well, but again not for this DC. In the end the agreement was as follows: ATLAS will use the Bittorrent tool on their ATLAS_TDAQ cluster to investigate software distribution within this cluster for their ATLAS software. They will start these investigations as soon as possible. They agreed to use this tool exclusively within this cluster and to not use it to connect to other peers, especially not offside. They will step by step increase the distributed amount of software and watch the network traffic carefully. The tool does not have to be installed locally, because it will be used out of AFS (for now). In addition, ATLAS will also investigate other ways of distributing the software, especially the means that are available by Quattor, which is software distribution in RPM's. IT can offer some help here, e.g. to make the Quattor framework available for ATLAS members, or how to package software in RPM. IT will inform Computer Security.

Thorsten Kleinwort

Our main goal is measuring the time it takes Bittorrent to distribute the container file as a function of the cluster size. This would happen about once or twice a week during the current tests and is expected to take around 1-2 hrs for the largest cluster size (admittedly a rough guess).
We would also take advantage of the resulting file to then run tests of the Atlas High Level Trigger software from it, but this has no bearing on the networking and security issues discussed today.

Best Regards, Haimo Zobernig