Configuration and Control of the ATLAS Trigger and Data Acquisition

Giovanna Lehmann Miotto[*,a], Igor Aleksandrov[b], Antonio Amorim[c], Giuseppe Avolio[d], Elisabeta Badescu[e], Mihai Caprini[e], Alina Corso-Radu[d], Georgiana L. Darlea[a], Andre dos Anjos[f], Ivan Fedorko[a], Andrei Kazarov[a,g], Serguei Kolos[d], Vladislav Kotov[b], Andrew J. Lankford[d], Marius Leahu[a], Livio Mapelli[a], Raul Murillo Garcia[d], Yuri Ryabov[g], John E. Sloper[a], Igor Soloviev[a], Lourenco Vaz Gil Lopes[h]

[a] *European Laboratory for Particle Physics (CERN), Geneva, Switzerland*
[b] *Joint Institute for Nuclear Research, Dubna, Russia*
[c] *Laboratorio de Instrumentaao e Fisica Experimental, Lisboa, Portugal*
[d] *Department of Physics and Astronomy, University of California, Irvine, California*
[e] *National Institute of Physics and Nuclear Engineering, Bucharest, Romania*
[f] *Department of Physics, University of Wisconsin, Madison, Wisconsin*
[g] *Petersburg Nuclear Physics Institute (PNPI), St. Petersburg, Russia*
[h] *Laboratrio de Instrumentaao e Fisica Experimental de Particulas, Coimbra*

Abstract

ATLAS is the biggest of the experiments aimed at studying high-energy particle interactions at the Large Hadron Collider (LHC). This paper describes the evolution of the Controls and Configuration system of the ATLAS Trigger and Data Acquisition (TDAQ) from the Technical Design Report (TDR) in 2003 to the first events taken at CERN with circulating beams in autumn 2008. The present functionality and performance and the lessons learned during the development are outlined. At the end we will also highlight some of the challenges which still have to be met by 2010, when the full scale of the trigger farm will be deployed.

Key words:
LHC, ATLAS, Data acquisition, Online Software, Run Control

1. Introduction

In this paper we describe the evolution of the Controls and Configuration system of the ATLAS [1] Trigger and Data Acquisition (TDAQ) from the Technical Design Report [2] (TDR) in 2003 to the first events taken at CERN with circulating beams in autumn 2008. We will first present the status of the system in 2003 and explain the reasons for launching a new development project; then we will highlight the areas in which major upgrades were performed and we will assess the status of the system which has been used to acquire the first events recorded by ATLAS with circulating beams in the LHC. We will finally conclude by listing the issues which still need to be addressed before considering the project successfully completed.

2. The system after the ATLAS TDAQ TDR

The ATLAS TDAQ is a large computing system running on 3000 interconnected computers, distributed over two main counting rooms, one underground close to the detector and the other one at the surface. Its aim is to readout, assemble, select and store interesting collision data generated within the ATLAS detector. The control of the system occurs over a dedicated Gigabit Ethernet network and operators act on it via a series of graphical interfaces, in the control rooms.

When the TDR was released in 2003 a complete design and a basic implementation of all aspects of control and configuration of the ATLAS TDAQ existed already. Validation was done on small scale systems with real detector readout [3] or on large computing farms with simulated data [4]. All in all the functionality and performance assessment was positive, except for the fault tolerance of the system, which was deemed insufficient.

Nevertheless, starting in 2004 a set of missing features and areas of concern were identified, most of which had not been formulated as requirements to the original design. The complexity of some of these issues was such that they could not be accommodated within the original system without a new substantial design and implementation effort. Therefore, in autumn 2004, a project was launched, with the goal of reviewing all configuration and control aspects of the ATLAS TDAQ and of upgrading their design and implementation in time for the experiment's start-up.

*Giovanna Lehmann Miotto. Tel.: +41-22-7672369.
Email address: Giovanna.Lehmann@cern.ch (Giovanna Lehmann Miotto)

Since the lifetime of this project was overlapping with the overall commissioning of the TDAQ and of the ATLAS detectors, a major constraint was put on it from the beginning: at any TDAQ software release (every 4-6 months) the project had to deliver working software. In addition, public API changes needed to be minimised whenever possible. Another peculiarity of this project was that in many areas the requirements were very vague and in continuous evolution: it was considered a task of the project coordination not only to clarify requirements but also to anticipate them by gathering users' feedback and by participating to the deployment of the system on the field.

3. The Controls and Configuration System

The system encompasses all the software required to configure and control the data taking of the ATLAS experiment. It is designed following a layered component model. At the very bottom are common libraries to deal with threads, time stamping and command line parsing, an in house developed database [5] and the libraries for the CORBA based inter process communication[6]. Higher up are a set of services, like the configuration service or the process management. Above these layers is the so-called application layer, with the run control, the diagnostics and verification system and the expert system. Finally, on top of everything there is a set of graphical user interfaces allowing the operator to act on the system.

In the following sections we will show how the introduction of new requirements has shaped the original design. For clarity, we have subdivided them into five distinct areas: auditing, security, scalability, fault tolerance and operability.

4. Auditing

Under auditing we mean the capability of analysing a posteriori what has happened during a data taking session and of reproducing results.

4.1. Reproducing a Run

It shall be possible to repeat a run with exactly the same configuration used in a past data taking session. For each run it shall be possible to know which configuration was used.

These requirements were addressed by introducing two new components into the system: the run number service and the OKS archiver. The run number service allows each run to be assigned a unique number throughout the lifetime of the experiment. The OKS archiver stores the configuration used for each run into a relational database [7].

4.2. Error Analysis

It shall be possible to browse and analyse the errors which occurred during a run. This requirement had a deep impact not only for the controls and configuration but also for the rest of the TDAQ software. In order to provide a searchable archive of log messages and to be able to analyse them in an automated way, a new error reporting package was introduced at the bottom of the software dependency tree and all other code needed to be changed to adopt it. In addition, a log service [8] component was introduced, with the task of subscribing to all messages produced by other applications during a data taking session and to store them into a relational database. A Java application was developed to browse through the messages and search them according to a set of criteria and tools were put in place for automated error cataloguing.

4.3. Management of Log Files

It had always been assumed that log files could not be used in such a large distributed system. Nevertheless, application log files exist and are deemed to be necessary by some developers, for debugging purposes. Two requirements were formulated in this area:

- It shall be possible to see application logs without the need to log onto TDAQ machines
- Log files shall be regularly removed from the computers and archived for a short period of time.

These two requirements were addressed by an extension of the graphical user interface of the verification and diagnosis component [9] for remote log file browsing and by the development of a new component, the Farm Tools, in charge of archiving log files on a large disk and removing them from all computers which participated to a data taking session.

5. Security

Computing security was never really considered an issue in previous high energy physics experiments. Nevertheless, the scale of the ATLAS collaboration (over 2000 people) and its use of conventional networking and computer operating systems, exposes the experiment to a high risk of dead time due to malicious intrusions or unintentional mistakes. Within the frame of the controls and configuration project we developed an access management [10] system based on the role based access control paradigm, which then was adopted for the complete experiment, well beyond data acquisition. The main idea of this access management model is that authorisation for a specific action can be granted to a user based both on his profile and expertise and on the experiment's status.

In order to effectively use access management a prerequisite is that users can be identified, not only when they interactively operate on the system but also when they ask the process management to launch or terminate processes on their behalf. This requirement made it necessary to completely re-design and re-implement the process management component. The main idea was to split the server part running on each machine into a single daemon and a number of launchers (one per process to control) which allow the change of process ownership from the account under which the daemon is run to the requester's account. More details can be found here [11].

6. Scalability

The requirements on scalability of the system continued to evolve until 2008. At the time of the TDR the system was expected to be able to control and configure in the order of 5000 software processes distributed on 3000 computers; but the introduction of multi-core technologies in the High Level Trigger farms, progressively scaled up the number of processes by one order of magnitude. This evolution did not require any major re-design of components but a lot of optimisations in the interprocess communication, at the level of central services, such as the message reporting service [12] and the resource manager, and at the level of the configuration database schema. At present only one third of the trigger farms have been purchased in ATLAS. Thus the largest real configuration which has been used is of the order of 10 thousand applications distributed on 1500 computers. Dedicated tests have proven that no scalability issues appear for 30 thousand processes. Larger configurations could not be measured because of lack of hardware resources.

7. Fault Tolerance and Error Recovery

The fault tolerance of the TDAQ had been identified as fairly weak in the TDR. Besides the capability of ignoring some non-essential processes which failed or restarting them, there was no real means of recovering from complex errors which affected many components. In order to allow for a more flexible evolution of the error recovery aspects, the run control and expert system components, which originally were very tight together, were re-designed and re-implemented with a clear separation of duties. While the run control has the task of starting processes at the right moment and keeping the TDAQ system in a coherent state (by means of a finite state machine), the expert system is put in charge of reacting only to anomalous situations. Both systems are organised in a hierarchical tree which can, but does not need to, overlap [13]. This new structure has allowed to progressively introduce more and more sophisticated error recovery scenarios. Of course the expert system will still evolve in future years when the experiment will reach a mature state and the number of actions to be taken automatically without the need for a decision by the operator will increase.

8. Operability

Until the end of 2006 the operation of the TDAQ system was mostly performed by experts. The commissioning runs at the experiment's site showed the importance of simplifying the working environment for the operator, having easy to use graphical interfaces and providing an online help to the tools. The effort in this area started in 2006 and led to the re-implementation of most of the graphical user interfaces as well as to the development of the so-called Control Room Desktop, a working environment, based on Kiosk [14], which exposes via simple menus and icons all the tools and web sites that the operator needs to access. In 2009 the training of ATLAS operators without any knowledge of the TDAQ has become important: a complete training package has been prepared and will be used for monthly training sessions as of May.

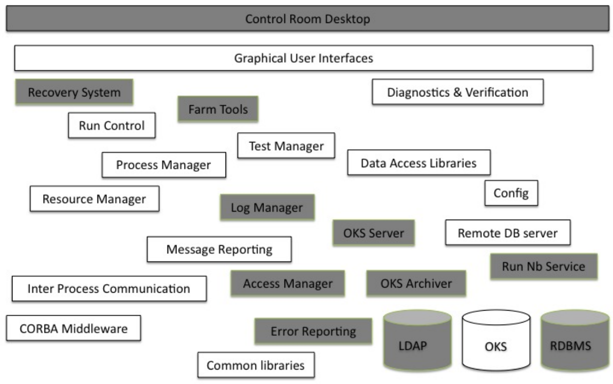

Figure 1: Main components of the controls and configuration software. The figure also indicates dependencies: low level libraries at the bottom and are used by the higher software layers. The elements depicted in grey are components which did not exist or had not been implemented in the original design.

9. Results and Outlook

The control and configuration project delivered a complete system in time for the first circulation of protons at the LHC in September 2008, thus achieving its main goal. Figure 1 shows the layered component structure of the system and its evolution since 2003.

Despite the fact that most of the components foreseen in the original design were upgraded and that several new components were introduced, the overall structure of the software and its main design guidelines could be preserved. In particular, the substantial change of performance requirements in terms of scalability, which could have caused the invalidation of the complete communication model (i.e. of one of the main design choices), could fortunately be addressed by localised software optimisations and by an increase of computing and networking resources, thanks to the excellent performance of the CORBA based interprocess communication.

While some of the new features were introduced smoothly, by adding standalone components, some other aspects proved to be quite problematic and required the reworking of many software packages. In particular, adding security concepts on top of an existing system has been very cumbersome and better solutions could have been envisaged if this requirement had been taken into account already from the initial design. Similarly, the introduction of a common error reporting package at an advanced stage of the TDAQ and detector software development was quite painful: the result is that not all developers followed the guidelines completely, thus making automated error analysis very complicated and only partially useful.

The project is basically completed from a software development point of view, but it will be closed only when the complete trigger farms will be purchased and installed at the experiment's site in 2009-2010. It is only at that moment that we will be able to prove without any more doubts that the controls and configuration system is capable of handling 50 thousand processes distributed over 3000 computers in an effective and reliable way.

References

[1] ATLAS Collaboration, The ATLAS Experiment at the Large Hadron Collider, 437 pp., JINST 3:S08003, 2008

[2] ATLAS Collaboration, ATLAS High-Level Trigger, Data Acquisition and Controls Technical Design Report, CERN/LHCC/2003-022, June 2003, see http://cdsweb.cern.ch/record/616089/.

[3] S. Gadomski et al, Deployment and use of the ATLAS DAQ in the combined test beam, IEEE Trans.Nucl.Sci.53:2156-2161, 2006

[4] D. Burckhart-Chromek et al, Testing on a large scale: Running the ATLAS data acquisition and high level trogger software on 770 PC nodes, ATL-DAQ-2006-002, Feb 2006. Prepared for International Conference on Computing in High Energy Physics (CHEP2006), Mumbai, Maharashtra, India, 13-17 Feb 2006

[5] R. Jones, The OKS persistent in-memory object manager, Nuclear Science, IEEE Transactions on Volume 45, Issue 4, Aug. 1998 Page(s):1958 - 1964, 1998

[6] R. Jones et al, Applications of CORBA in the ATLAS prototype DAQ, Nuclear Science, IEEE Transactions on Volume 47, Issue 2, Apr. 2000 Page(s):331 - 336, 2000

[7] J. Almeida et al, The ATLAS DAQ System Online Configurations Database Service Challenge, J.Phys.Conf.Ser.119:022004, 2008

[8] R. Murillo Garcia, G. Lehmann Miotto, A log service package for the ATLAS TDAQ/DCS group, Nuclear Science, IEEE Transactions on Volume 54, Issue 1, Feb. 2007 Page(s):202 - 207, 2007

[9] A. Kazarov et al, A Rule-Based Verification and Control Framework in Atlas Trigger-DAQ, Nuclear Science, IEEE Transactions on Volume 54, Issue 3, Jun. 2007 Page(s):604 - 608, 2007

[10] M.C. Leahu et al, Access Control Design and Implementation in the ATLAS Experiment, Nuclear Science, IEEE Transactions on Volume 55, Issue 1, Feb. 2008 Page(s):386 - 391, 2008

[11] G. Avolio et al, The process manager in the ATLAS DAQ system, Nuclear Science, IEEE Transactions on Volume 55, Issue 1, Feb. 2008 Page(s):399 - 404, 2008

[12] I. Fedorko et al, The Message Reporting System of the ATLAS DAQ System, ATL-DAQ-CONF-2007-029, presented at 10th ICATPP Conference on Astroparticle, Particle, Space Physics, Detectors and Medical Physics Applications, Como, Italy, 8 - 12 Oct 2007

[13] J.E. Sloper et al, Dynamic Error Recovery in the ATLAS TDAQ System, Nuclear Science, IEEE Transactions on Volume 55, Issue 1, Feb. 2008 Page(s):405 - 410, 2008

[14] Kiosk, http://en.wikipedia.org/wiki/Kiosk_(KDE)