

The Control and Configuration Software of the ATLAS Data Acquisition System: Upgrades for LHC Run 2



G LEHMANN MIOTTO^a, I ALEKSANDROV^b, G ANDERS^a, G AVOLIO^a, M CAPRINI^c, A CORSO RADU^d, M D'ASCANIO^a, J DE CeSTRO VARGAS FERNANDES^e, A KAZAROV^f, B KOLOBARA^{a,g}, A LANKFORD^d, F LAURENT^{a,h}, L MAGNONI^a, L PAPAEVGENIOU^a, Y RYABOV^f, A SANTOSⁱ, J SEIXAS^e, I SOLOVIEV^d, G UNEL^d, Y YASU^j

a: CERN (CH), b: Joint Inst. for Nuclear Research – JINR (RU), c: IFIN-HH Bucharest (RO), d: University of California Irvine (US), e: Univ. Federal do Rio de Janeiro (BR), f: B.P. Konstantinov Petersburg Nuclear Physics Institute – PNPI (RU), g: Ministere des Affaires Etrangeres et Europeennes (FR), h: École Polytechnique Fédérale de Lausanne (CH), i: Universidad Nacional de La Plata (AR), j: High Energy Accelerator Research Organization (JP)

Requirements Extension

In a large, heterogeneous system such as the ATLAS TDAQ, it is essential to be able to verify the correct functioning of hardware and software components. Therefore, a TDAQ functional testing framework existed and was used already throughout Run 1. Additional requirements were identified with the experience gained during data taking:

- "Experts shall be able to define the order in which tests should be executed for a component; the sequence may dynamically change based on the result of completed tests"
- "Experts shall be able to define the order with which inter-related components shall be tested; the test sequence may change depending on the result obtained for the components."
- "Experts shall be able to define what should be done upon failure of a test or a component to further diagnose the issue or recover."

Functional testing is used by:

- The Run Control (RC) system that periodically verifies the functioning of the components it is in charge of;
- The Central Hint and Information



Software Refactoring

The **Run Control** (RC) system steers the data acquisition by starting and stopping processes and by carrying all data-taking elements through welldefined states in a coherent way. Given the size and complexity of the TDAQ system (2000+ PCs, 30000+ applications, 9000+ network ports,...), errors and failures are bound to happen and must be dealt with: in ATLAS this task is carried out by an **expert** system.

The RC and expert system components were tightly coupled in the first implementation of the software, with the consequence that their separation of duties became increasingly unclear with the additions of features during operations in Run 1.



Therefore these two components were completely redesigned, using a distributed run control tree and a separate central expert system application, the **Central Hint and Information Processor** (CHIP).



Applications in the ATLAS TDAQ system are organized in a tree-like hierarchical structure (the run control tree), where each application is managed by a parent **Controller**. The topmost node of the tree is the **Root Controller**. Controller applications are responsible to keep the system in a coherent state by starting and stopping their child applications and by sending them the proper commands needed to reach a state suitable for data-taking. Controller applications interact with CHIP by informing it about any changes (their own or their children). CHIP acts as an **intelligent system** having a global view on the TDAQ. It is capable of recovering from error conditions and guiding the TDAQ system through automated procedures in order to take data efficiently. The RC has been re-implemented with state of the art C++ technologies such as boost and Intel Threading Building Blocks. CHIP has been designed and implemented based on a third party open source java based Complex Event Processing (CEP) engine, ESPER.

ATLAS in Run 1

The ATLAS experiment at the Large Hadron Collider (LHC) operated very successfully in the years 2008 to 2013, identified as Run 1. It achieved an overall data taking efficiency of 94%, largely constrained by the irreducible dead-time introduced to accommodate the limitations of the detector read-out electronics. Out of the 6% dead-time only about 15% could be attributed to the central trigger and DAQ system, and out of these, a negligible fraction was due to the control and configuration sub-system. Despite these achievements, the first long LHC shutdown (2013-2014) was used to carry out a complete revision of the control and configuration software. The goals were three-fold: properly accommodate additional requirements that could not be seamlessly included during steady operation of the system; re-factor software that had been repeatedly modified to include new features, thus becoming less maintainable; seize the opportunity of **modernizing software** written at the beginning of the years 2000, thus profiting from the rapid evolution in IT technologies. This upgrade was carried out retaining the important constraint of minimally impacting the mode of operation of the system and public APIs, in order to maximize the acceptance of the changes by the large user community.

This poster illustrates, using a few selected examples, how the work was approached and which new technologies were introduced into the ATLAS DAQ system. Despite these being specific to this system, many solutions can be considered and adapted to different distributed DAQ systems.

Code Modernization

 Message Transport System (MTS) MTS underwent a review of the requirements that led to a complete redesign and new implementation to match its actual role (fast and reliable transport layer for TDAQ Error Reporting

System messages). The redesigned system is reliable, scalable and its performance has been improved. The plot shows the rates of messages reported in MTS in technical runs in April 2014. In the condition of the plot (60000 applications running) MTS reached a maximum rate of 18kHz of delivered messages.



• Information System Archiver, P-BEAST (a Persistent Back-End for the ATLAS TDAQ)

P-BEAST is a **new component** designed and implemented to archive operational monitoring information for analysis by experts. It provides CORBA and REST interfaces for data access. Its implementation is based on Google protobuf (data persistence), CORBA (internal protocol and user programming interface) and libmicrohttpd (Web server).



Resource Manager

After an initial review and simplification of the requirements, the system underwent partial changes with the introduction of Boost multi-index containers. As a result the code base has been reduced by 40% against the previous implementation thus leading to a more maintainable system. The plot shows that the resource manager introduces a negligible overhead to the initialization of the RC tree.



Web Based Tools

• P-BEAST Dashboard

This web application offers an interface to visualize any operational monitoring data published by the TDAQ system through configurable and customizable dashboards. The data is provided by P-BEAST and the application is based on the Grafana project, adapted to support a custom data source within the AngularJS framework.

• ELisA

The ATLAS electronic logbook (ELisA) is a web application used to record and share messages about ATLAS data taking activities by system operators, experts and automated services. The information is stored in an ORACLE database. The adoption of an MVC-driven architecture has allowed one to focus code development on specific features of the project, while profiting from the reliability of established third-party technologies such as the Spring framework. The tool also provides an HTTP-based REST APL such that other



()	https://atlaso	p. cem.ch /elisa/display?log	gbook=ATLAS		, (🖻 🔩 🔻 🚼 🕶 Google	艶 ◆ 合 ☆ 自 🛛 **▼
/ou are	logged in as	s <u>alina</u> Logout					ChangeLog v8.0.12
Flat V	iew Thre	eaded View New Entry	Advanced Search Displ	lay Thread			Contact us @ 👔 🚺
-	•	Search:				Showin	g 1 to 15 of 499 entries Show 15 🔽 entries 🗹 🗲 🕨
e Sh	ow / Hide col	umns Date&Time	- Author	\$ Subject	≎ Message Type		≎ Text
0	246225	2014-10-30 09:27	Tiesheng Dai	Tuirn on MDT	Default Message Type	DCS, MDT	Turn on MDT for checking EO chambers. Followi
٢	246224	2014-10-30 09:23	DCS_IS	INF_Smoke_SR1_TechnicalFault	Default Message Type	DSS, Tech. Infra	DSS Alarm: INF_Smoke_SR1_TechnicalFault
٥	246223	2014-10-30 08:47	Stephanie Ulrike Zimm	Turbine rack interlock script stopped	Default Message Type	DCS, MDT	I stopped the turbine rack interlokc process/script
٥	246222	2014-10-30 07:09	Iskander Ibragimov	Shift Summary for Pixel desk	Shift Summary	ID Gen. (IC), Pixel	Shift Summary for Pixel desk Shifter(s): Iskander I
0	246221	2014-10-29 22:52	Laura Franconi	Shift Summary for ID DCS Watcher s	Default Message Type	ID Gen. (IC), Pixel	Shift Summary for Pixel desk Shifter(s): Laura Fra
٢	246220	2014-10-29 19:14	Atlas Slimos	Shift Summary for SLIMOS desk	Shift Summary	DSS	Start of Shift: ====================================
٢	246219	2014-10-29 17:27	Stephanie Ulrike Zimm	Cooling restored for MDT EO racks,	Default Message Type	DCS, MDT	We got the confirmation that the beampipe bakeo
	Reply Edit	Subject: Cooling resto We got the confirmation the ResetNet removed. EO chambers have been	red for MDT EO racks, HV/LV h that the beampipe bakeout has i n reincluded in the JTAG asnd PS	ave been put back in operational mode finished and rack cooling for the HO (EO) rac S FSM trees in DCS. Please note that a few c	ks restored. All explicit interlo nambers seem interlocked st	icks on EO HV and LV have be ill due to gas conditions or gas	en cleared, and the "Latch Off" of CAEN communication via interlock flags asserted by Philipp.
٢	246218	2014-10-29 16:35	Oleg	Tile Laser Runs (Runs 243805-2438	Tile	Tile	Run 243805: Tile Laser LG Scan Run 243807: Til
0	246217	2014-10-29 16:06	Oleg	Tile CIS and Pedestal Runs (Runs 2	Tile	Tile	Run 243802: Tile CIS Mono Run 243803: Tile CIS
0	246216	2014-10-29 15:09	Marius Cornelis Van W	Laser calibration runs	Tile	Tile	Laser calibration runs were taken. run number, n

rams		its features
gianis	Lan alless	115 IEdiules.

246213 2014-10-29 14:30 Wainer Vandelli New RobinNP SW version Default Message Type DAQ A new version of the RobinNP-related libraries ha	246213 2014-10-29 14:30 Wainer Vandelli New RobinNP SW version Default Message Type DAQ A new version of the RobinNP-related libraries ha	0	246215	2014-10-29 14:52	Siinn Che	Shift Summany for ID DCS Watcher s	Default Massage Type	ID Can (IC) Pival	October 20, 2014, Divel DCS Morning Shift Sijan	
O 246214 2014-10-29 14:30 Wainer Vandelli New RobinNP SW version Default Message Type DAQ A new version of the RobinNP-related libraries ha Image: Contact us: Bugs, feedback, improvements (access from outside P1 only). Currently supported browsers: Firefox, Safari, IE.	246214 2014-10-29 14:30 Wainer Vandelli New RobinNP SW version Default Message Type DAQ A new version of the RobinNP-related libraries ha	•	240215	2014-10-29 14.55	Simili Che	Shint Summary for 10 DCS Watcher S	Delaun message Type	id Gen. (iC), Fixer	October 29, 2014, Pixer DC3 Morning Shint Smith	
Thu Oct 30 10:39:59 CET 2014 Contact us: Bugs, feedback, improvements (access from outside P1 only), Currently supported browsers: Firefox, Safari, IE.	I Oct 30 10:39:59 CET 2014 Contact us: Bugs, feedback, improvements (access from outside P1 only). Currently supported browsers: Firefox, Safari, IE.	0	246214	2014-10-29 14:30	Wainer Vandelli	New RobinNP SW version	Default Message Type	DAQ	A new version of the RobinNP-related libraries ha	j.
hu Oct 30 10:39:59 CET 2014 Contact us: Bugs, feedback, improvements (access from outside P1 only), Currently supported browsers: Firefox, Safari, IE.	a Oct 30 10:39:59 CET 2014 Contact us: Bugs, feedback, improvements (access from outside P1 only). Currently supported browsers: Firefox, Safari, IE.	4								•
		hu Oct 30	0 10:39:59 CE	T 2014		Contact us: Bugs, feedback, improvements (access	from outside P1 only). Currently	supported browsers: Firefox, Safa	ri, IE.	

Conclusions & Outlook

The control and configuration software has contributed to the physics results obtained by the ATLAS experiment during Run 1 by ensuring smooth and efficient data taking. It was completely revised during 2013-2014 in order to accommodate additional requirements, improve maintainability and profit from advances in IT technologies: all this was done applying minimal changes to APIs, such that the large amount of client code would not need significant adaptations. The control and configuration software has already proved to be stable and well performing since the start of LHC Run 2 (2015 - 2018) and is prepared to face the new challenges that will arise during Run 2 operations. This experience has also demonstrated that the overall architecture of the control and configuration system is flexible and supports partial upgrades, as well as step-wise modernization of its components: this is fundamental for a system that is foreseen to run for the next 20 to 30 years and that will undergo several more upgrade iterations.