The ALICE Configuration Tool

M Boccioli¹, F Carena¹, S Chapeland¹, V Chibante Barroso¹, M Lechman¹, A Jusko² and O Pinazza¹,³ for the ALICE Collaboration

¹ CERN, Geneva, Switzerland
² School of Physics and Astronomy, University of Birmingham, Birmingham, UK
³ Sezione INFN, Bologna, Italy

Email: Vasco.Chibante.Barroso@cern.ch

Abstract. ALICE (A Large Ion Collider Experiment) is the heavy-ion detector designed to study the physics of strongly interacting matter and the quark-gluon plasma at the CERN Large Hadron Collider (LHC). It includes 18 different sub-detectors and 5 online systems, each one made of many different components and developed by different teams inside the collaboration. The operation of a large experiment over several years to collect billions of events acquired in well defined conditions requires predictability and repeatability of the experiment configuration. The logistics of the operation is also a major issue and it is mandatory to reduce the size of the shift crew needed to operate the experiment. Appropriate software tools are therefore needed to automate daily operations. This ensures minimizing human errors and maximizing the data taking time. The ALICE Configuration Tool (ACT) is ALICE first step to achieve a high level of automation, implementing automatic configuration and calibration of the sub-detectors and online systems. This presentation describes the goals and architecture of the ACT, the web-based Human Interface and the commissioning performed before the start of the collisions. It also reports on the first experiences with real use in daily operations, and finally it presents the road-map for future developments.

1. Introduction

1.1. The ALICE experiment

ALICE (A Large Ion Collider Experiment) [1] is the detector designed to study the physics of strongly interacting matter at extreme energy densities at the CERN Large Hadron Collider (LHC), where the formation of a new phase of matter, the Quark Gluon Plasma (QGP), is expected. Consisting of 18 sub-detectors, the ALICE experiment will primarily target heavy-ion collisions (in particular Pb-Pb) to study the QGP, but it also has an ambitious physics program for proton-proton and proton-ion collisions. After 15 years of design and installation, a series of dedicated sessions were organized during 2008 and 2009 not only to commission the sub-detectors, the online systems and the online-offline interfaces but also to exercise operational and support activities. At the time of the LHC restart in March 2010, the ALICE experiment was ready to detect and record the first collisions and has been collecting millions of events ever since.

1.2. The ALICE Configuration Tool
The operation of the ALICE experiment over several years to collect billions of events acquired in well defined conditions requires repeatability of the experiment’s configuration. Appropriate software tools are therefore needed to automate daily operations, minimizing human errors and maximizing the data-taking time. The ALICE Configuration Tool (ACT) fulfills these requirements, allowing the automatic configuration of the different systems and sub-detectors.

2. Architecture
The core of the ACT architecture is a configuration repository from which any ALICE system can extract its currently selected configuration. A publish/subscribe mechanism based on the Distributed Information Management (DIM) [2] system is used to notify changes. Although it is possible to directly interact with the repository with a generic mechanism, some intermediate clients have been added to actively drive some of the configurations to the subsystems through specific interfaces.

As shown in figure 1, the ACT is operated by the Run Coordination via a Web-based Graphical User Interface (GUI). Whenever a configuration request is submitted, the selected configurations are broadcasted via DIM to two dedicated modules (implemented in C), running as daemon processes:

- **ECS Dedicated Daemon (EDD):** handles the configuration requests for ECS [3], DAQ [3], CTP [3] and HLT [3].
- **DCS Dedicated Daemon (DDD):** handles the configuration requests for the ALICE sub-detectors.

In the case of the sub-detectors, the requested configuration is sent by DDD - via DIM - to another module called Run Control Tool (RCT), which then makes it available to each detector’s individual DCS [3] Finite States Machine (FSM) where the sub-detectors configuration is executed.

An Application Programming Interface (API) implemented in C provides numerous functionalities to the different components.
2.1. Taxonomy
In order to define the different systems and sub-detectors components to configure, the ACT introduces the following concepts:

- **System**: an ACT system represents a physical or logical element of the ALICE experiment (e.g. CTP). Each system normally has several configurable components.
- **Item**: an ACT item corresponds to a configurable component of a specific ACT system (e.g. /CTP/VALID.BCMASKS). Each item normally has several possible configurations defined.
- **Instance**: an ACT instance defines a possible predefined configuration for a specific ACT item (e.g. 50ns_1236b+1small_1180_37_1152_144bpi). At any given time, each item can only have one active instance.

2.2. Items Locking
A locking mechanism prevents the configuration of items that are either being configured or used by an online system (e.g. a sub-detector being used for data-taking). If being configured, the items are locked by the corresponding daemon (EDD or DDD). If being used by one or more online systems, the items are locked by those systems (an item can therefore have more than one lock). An item is considered as locked if at least one entry with its ID exists in the ACTlockedItems DB table.

2.3. Items Activation Status
At any given time, each item is in a specific state, represented by its activation status. As shown if figure 2, there are four possible statuses:

- ‘update requested’: a configuration has been requested for the item.
- ‘applying’: a configuration is being executed for the item.
- ‘active’: the item is configured as requested.
- ‘update failed’: an error occurred while configuring the item.

![Figure 2. Items activation status state diagram.](image)

3. Databases

3.1. ACT DB
This DB, running on a MySQL server, is used to store the definition of the different elements of the ACT. InnoDB is used as a storage engine for its support of both transactions and foreign keys constraints. Daily backups are performed to a RAID 6 disk array and the CERN Advanced STORage manager (CASTOR) [4]. The main tables are:

- **ACTsystems**: stores the information related with the systems.
- **ACTitems**: stores the information related with the items.
- **ACTinstances**: stores the information related with the instances. The instance name is normally a human-friendly label and the instance value is the actual configuration to be applied by the system.
- **ACTlockedItems**: stores the list of locked items.

3.2. Detector Control System DB
This DB is hosted on the CERN Central Oracle Services. The service is provided by CERN’s IT department. The sub-detectors store their configuration data in 2 types of table schemas:
- **Config DB**: developed within the JCOP framework project [5], it allows the definition of generic sets of parameters and supports basic versioning of their values.
- **FERO (Front-end and Readout Electronics) DB**: specific designed table schema for a particular sub-detector.

### 4. Interfaces

Some specific interface modules have been developed to minimize changes in the existing ALICE online systems in production, and cope with their heterogeneous nature. These modules transform the generic notification and information pull mechanism offered by the configuration repository to a set of subsystem-specific push mechanisms.

#### 4.1. ACT-ECS/DAQ/HLT/CTP interface

The interface between ACT and ECS, DAQ, HLT and CTP is implemented by the EDD daemon process. When a configuration request is received by EDD (via DIM), it checks which items have been marked for update and propagates the selected configuration to the corresponding system. It also handles the locking and activation status changes of the affected items.

#### 4.2. ACT-sub-detectors interface

The interface between ACT and the numerous sub-detectors is more complex not only because they exist in different platforms (ACT in Linux, sub-detectors in Windows/PVSS) but also due to the fact that several sub-detectors already had their specific configuration tools. Therefore, the following modules were developed:

- **Run Control Tool (RCT)**: implemented in PVSS, this module receives from DDD (via DIM) the list of items to be configured and the values of the corresponding active instances and stores them on sub-detector specific PVSS datapoints. At the same time, if the sub-detectors and the data taking status allow it, it issues the configuration command to the affected sub-detectors.
- **FERO Central Configuration (FERO CC)**: this tool was designed to create and manage the mappings between parameters defined in ACT DB and identifiers used by specific version control systems for accessing the data in FERO DB.
- **Devices FSM Configuration (FSMconf)**: implemented in PVSS and based on JCOP Framework components, this module resides in the sub-detectors control system and serves as an abstraction layer connected to the hardware (e.g. power supplies). It supports versioning and provides different groups of settings (recipes) that can be loaded from the Config DB according to the command received (e.g. `CONFIGURE/PHYSICS`) and the device state (e.g. `READY`). It applies the recipes directly to the hardware (e.g. setting the desired voltage level, operating a power supply).

As shown in figure 3, when a configuration is requested for one or more sub-detectors the following sequence of actions is executed:

1) DDD sends via DIM the list of items to be configured – together with the values of the selected instances – to RCT.
2) RCT writes the received pairs on sub-detector specific PVSS datapoints.
3) RCT sends an FSM `CONFIGURE` command to each affected sub-detector.
4) The sub-detector(s) access the desired configuration from their specific PVSS datapoint and propagate the `CONFIGURE` command down to the FERO CC and FSMconf of each device.
5) The FERO CC translates the desired configuration sent by DDD into the corresponding set of identifiers from FERO DB. On this basis the sub-detector algorithm loads the requested...
configuration from the FERO DB, applies it to the devices and sends feedback of the result. The FSMconf does the same for the Power Supplies.

6) The sub-detector(s) write the configuration feedback in their specific PVSS datapoint.
7) RCT sends the result of the configuration back to DDD.

![Figure 3. Sub-detectors configuration using the ACT.](image)

Similar to EDD, DDD also handles the locking and activation status changes of the affected items.

5. Graphical User Interfaces

5.1. ACT Web Interface

The ACT’s Web-based GUI was developed using modern Web technologies, including PHP5, Javascript and Cascading Style Sheets (CSS). It uses the PHP Zend Framework to implement the Model-View-Controller (MVC) architecture. It is hosted on an Apache web server and can be accessed from the experimental area (inside the experiment's technical network), the CERN General Purpose Network (GPN) and the internet.

5.1.1. Authentication and Authorization. Authentication is implemented via the CERN Authentication central service, providing Single Sign On (SSO) and removing the effort of authenticating the users from the ACT software. Authorization is based on CERN’s egroups. In order to access the GUI, users must be members of the ALICE-ACT egroup.

5.1.2. Expert Mode. The Expert Mode section is used to populate the ACT DB, allowing system experts and ACT administrators to create and modify systems, items and instances. It also allows the execution of different actions and the display of different status tables.

5.1.3. Run Coordination Mode. The Run Coordination Mode section is used to configure ALICE during data taking periods, allowing the Run Coordination and the Shift Leaders to globally configure the different ALICE sub-detectors (see figure 4) and systems.

5.2. RCT PVSS Panel

The RCT PVSS Panel allows the DCS shifters and experts to monitor the status of both the RCT and the sub-detector’s configuration. As shown in figure 5, for each sub-detector the users can access the timestamp and the status of the last configuration.

5.3. FERO CC Panels

As shown in figure 6, the FERO CC panels enable the DCS experts to link specific configurations of sub-detectors with possible requests coming from DDD. The tool also allows tracking the history of changes in these mappings.

6. Current Status
After a successful commissioning campaign at the beginning of 2010, the ACT has been used by the Run Coordination to configure ALICE since the first day of LHC operations at 7 TeV in March 2010. With constant addition of new configurable items and instances, the ACT has been steadily extending its scope to become the global ALICE configuration tool. Currently there are 30 systems, 132 items and 862 instances available, covering all installed sub-detectors and the main online systems.

7. Future plans
The future plans include the expansion of the ACT to new systems and items and improvements to the ACT Web Interface to facilitate the shift crew operations. The possibility to add a run scheduler feature was discussed but abandoned in favor of a dedicated tool.

8. Conclusions
In large scientific experiments such as ALICE, it is essential to have tools that allow shifters to easily reuse predefined configurations without expert intervention. Additionally, to keep operational costs at a reasonable level over a lifespan of several years, the reduction of the size of the shift crew needed to operate the experiment is of paramount importance. In production since February 2010, the ACT strongly contributed not only to the success of the ALICE operations in 2010 but also to the reduction of the size of the ALICE shift crew - from more than 20 in 2009 to the current 6.

References
[1] The ALICE Collaboration 2008 The ALICE experiment at the CERN LHC 2008 JINST 3 S08002

[2] Gaspar C, Donszelmann M and Charpentier P 2000 DIM, a portable, light weight package for information publishing, data transfer and inter-process communication Int. Conf. on Computing in High Energy and Nuclear Physics Padova, Italy

[3] The ALICE Collaboration 2004 The Technical Design Report of the Trigger, Data-Acquisition, High Level Trigger, and Control System CERN-LHCC-2003-062

[4] Cancio G, Duellmann D and Pace A 2009 Data Management Evolution and Strategy at CERN Proc. CHEP 2009 (Prague, CZ: JPCS)

[5] JCOP Architecture Working Group: Framework Design Proposal, CERN, Geneva, 2001, CERN-JCOP-2000-008