

CHALLENGES OF THE ALICE DETECTOR CONTROL SYSTEM FOR THE LHC RUN3

P. Chochula, A. Augustinus, P. M. Bond, CERN, Geneva, Switzerland
 O. Pinazza, CERN, Geneva, Switzerland, also at INFN Sezione di Bologna, Bologna, Italy
 A. N. Kurepin, CERN, Geneva, Switzerland also at INR RAS – Institute for Nuclear Research of
 the Russian Academy of Sciences, Moscow, Russia
 J.L.Lang, University of Helsinki, Finland
 M.A.Lechman, CERN, Geneva, Switzerland also at Slovak Academy of Sciences, Bratislava

Abstract

The ALICE Detector Control System (DCS) has provided its services to the experiment since 10 years. During this period it ensured uninterrupted operation of the experiment and guaranteed stable conditions for the data taking.

The DCS has been designed to cope with the detector requirements compatible with the LHC operation during its RUN1 and RUN2 phases. The decision to extend the lifetime of the experiment beyond this horizon requires the re-design of the DCS data flow and represents a major challenge. The major challenges of the system upgrade are presented in this paper.

THE ALICE DETECTOR CONTROL SYSTEM (DCS)

The ALICE Detector Control System is based on commercial SCADA system WINCC OA. Wherever possible it uses industrial components and tools to provide its functionality.

power supplies, PLCs and detector front-end electronic modules, and devices interfaced via industrial fieldbuses.

The hardware abstraction layer provides unification of the communication between a large variety of devices and the control system software. It consists of industrial OPC servers and of ALICE FED servers [1] providing OPC-like functionality on nonstandard devices. The communication interface of the FED servers is based on the on lightweight DIM protocol [2] developed at CERN.

The data acquired from the devices is processed in the DCS Controls layer. A farm of about 120 servers runs a distributed WINCC OA SCADA system configured to behave as one large system. Each received value is compared with the operational limits and if needed a corrective action is taken or the issue is brought to the attention of the ALICE shift crew. The control layer configures the control devices and sends commands to them. The configuration data as well as all acquired data is stored in the ORACLE database. The control layer is in charge of about 1 million parameters.

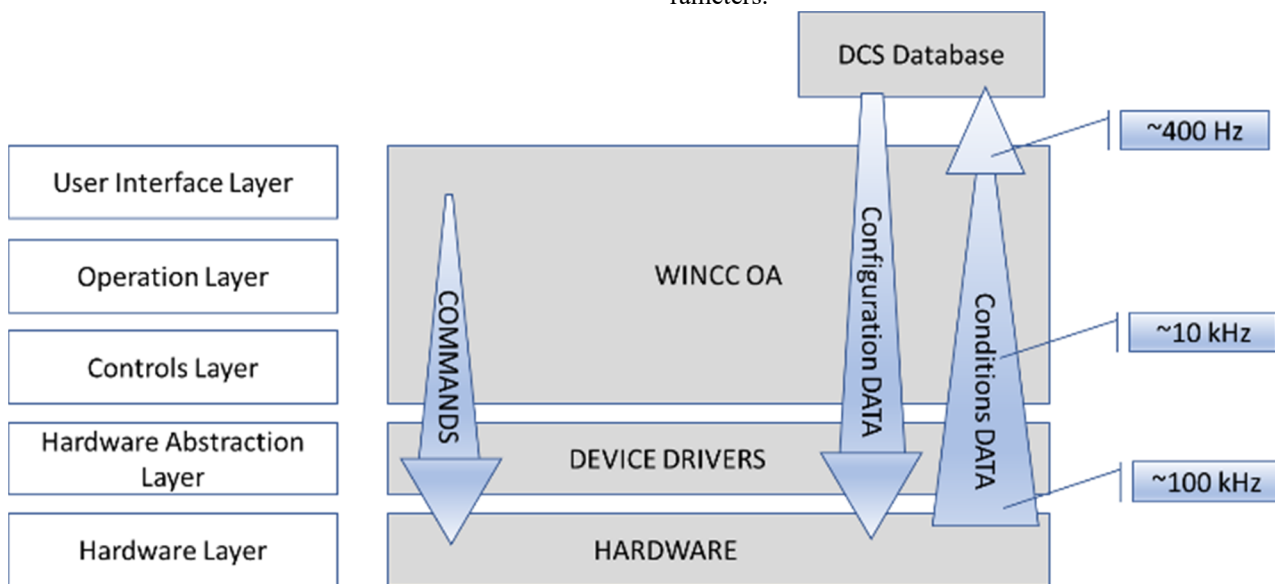


Figure 1: The mapping of the DCS layers to the hardware and software system components and the general data flow.

The DCS is using a five tier architecture (see Fig. 1). At the bottom, the hardware layer provides all devices and sensors required for the operation of the experiment. It consists of about 1200 network attached devices – mostly

The operation of the experiment must follow a carefully tuned sequence of steps. For example a detector module cannot be turned on, unless sufficient cooling is provided, certain electronic modules may be powered only after the high voltage has been applied to the sensors, etc. This logic is implemented in the operations layer. Based on the

SMI++ package [3], this layer represents all devices and their channels as a finite state machine integrated into a global hierarchy. This approach allows the operator to send a global command, for example to prepare the experiment for data taking and the controls layer will synchronize all actions between the detectors and their components.

Finally, the User Interface layer visualizes all DCS components and their states in a user friendly form, allowing for the operation of the experiment from one workplace by a single operator.

THE DCS DATA FLOW IN THE ALICE EXPERIMENT

The ALICE experiment operation follows the operational cycle of the LHC. All devices controlled by the DCS are configured according to their operational requirements related to the beam mode and the type of physics being studied by ALICE. For example, the operation with protons typically requires settings different from the ion mode. During the periods without the beam presence the detectors usually take calibration data and need to be reconfigured again.

The configuration data required for each operation mode is read from the ORACLE database and distributed to all DCS devices by the WINCC OA systems (see Fig. 1). The raw configuration data stored in the database is typically inflated along its path to the devices. The storage-optimized configuration records are compiled and adjusted to a format recognized by the devices. In the front-end electronics domain, the data is largely repetitive and the database stores one configuration record common to group of modules instead of a set of identical individual records. Before sent to the destination, the data volumes can be increased by two orders of magnitude in the data abstraction layer (the FED servers).

The detector conditions are monitored, regulated and acquired data is stored in the ORACLE database. The conditions data flow direction is opposite to the configuration data flow and its size is being reduced.

The detector conditions are recorded continuously, independent on the experiment operation mode. During the periods without the beams, the conditions are recorded mainly for reference or for calibration purposes. Once the beams are accelerated and brought to the collisions, the role of the DCS is to assure a stable data taking. The stability of the conditions is crucial for the physics analysis. The signals of the detectors triggered by the detected particles are processed by the Data Acquisition System (DAQ). The data stream from DAQ is stored using a separate infrastructure independent from the DCS. For the final analysis, the DAQ data must be merged with the DCS information stored in the ORACLE database.

A period during which the LHC provides particle collisions is called a fill. It can last several hours and the role of the experiment is to records as much physics data as possible. During a fill, the data taking might be interrupted due to a physical condition change like a voltage channel trip due to overcurrent caused by high particle flux, or various

software or hardware reasons. A period of stable data taking is called a run, each fill can be split into several runs.

Data recorded during a run is sent to further levels of analysis, using the computer GRID infrastructure. The signals of the detectors are assigned to particle tracks and the particle properties (type, momentum, and trajectory) are calculated. This reconstruction is largely dependent on the DCS parameters. A dedicated mechanism, called ALICE Shuttle, requests from the DCS a set of recorded condition parameters after each run. The DCS retrieves the requested data from the ORACLE database and provides it to the Shuttle using a custom server-client mechanism called AMANDA [4].

THE DCS ALICE UPGRADE FOR THE LHC RUN3

The interaction rates at LHC in ALICE during the RUN3 period, planned to start in 2021, will increase by a factor of 100. A new Combined Online Offline computing facility (O2) [5] has been developed to cope with the new data processing requirements.

The detector readout will be upgraded and it will provide 3.4 TBytes/s of data, carried by 9 000 optical links to a First Level Processing (FLP) farm consisting of 270 servers. The data taken during the same period of time by individual detectors is merged and sent 1 500 Event Processing Nodes (EPN), deploying ~100 000 CPU cores. After initial processing, a compressed volume of 100GByte/s will be transferred to the computing GRID facilities.

Due to high interaction rates, it is not possible to store the detector data on disk and perform the processing once the run ends. Even the local disk storage of 50PB, which will be installed in ALICE, is not sufficient for the whole data processing. The detector data will be therefore processed immediately on the EPN farm. The event reconstruction will allow for significant reduction of the data volumes before it is passed to the GRID for further analysis. To allow for such analysis, the DCS conditions data needs to be provided together with the detector data.

The new readout electronics will be accessed through the CERN GBT link [6]. The traffic on this link will be shared between the DCS and the data acquisition system, which requires a new concept for the front-end control to be implemented.

The conditions data handling and front-end electronics access represent the two main fields of new DCS developments for the LHC RUN3.

Conditions Data

While the current DCS assembles the conditions data block using the ORACLE archive after each run, the upgraded system must ensure a steady streaming of all conditions. Each data frame, covering 50 ms of data taking must be complemented with a block of about 100 000 DCS parameters to allow for reconstruction in the O2 facility. This requirement results in the increase of the DCS data publishing rate by a factor of 5000 and it cannot be covered by the present system.

The typical monitoring frequency for the DCS devices is ~ 1 Hz. It largely depends on each device and is usually defined by the hardware vendors. Furthermore, the DCS data readout is usually not triggered externally and it is driven by the conditions change. As a result, the data arrival time to the DCS is not regular and cannot be predicted. The DCS devices equipped with commercial OPC servers usually do not allow for a parameter readout on demand. The firmware of most of the devices performs internal poll of all devices, before it makes the data available to the OPC server. The polling time usually exceeds the available 50ms window and the DCS is not able to assemble a fresh conditions data block for each data frame.

The newly developed approach is based on a process image stored in the computer memory. This large formatted data block contains all information about the monitored channels required for the O2 processing. This information consists of the channel name, timestamped values and various flags describing the data type and quality information. The DCS creates this block at the startup time and populates it with the latest known values. As the new data arrives to the DCS, the process image is updated. At each period of time the process image contains the actual status of all condition parameters. This block is then streamed to the O2 at the requested time intervals.

New Conditions Data Flow Implementation

The conditions data flow from the DCS to O2 is managed by the Alice DataPoint Service (ADAPOS). Its task is to collect the DCS data and assemble a memory block with conditions data required for the further stages of reconstruction.

Data to be published to the O2 is scattered across ~ 100 WINCC OA systems. At the startup time, the ADAPOS server first needs to locate the data and subscribe to each

The current experience with the control system shows, that the distributed system registers more than 100k incoming values each second. The fluctuations of the individual parameters are usually not significant for the operation as these are caused by the noise. A filtering mechanism implemented at the driver level reduces this amount to about 10000 values per second. The role of this first level data reduction is to suppress fluctuations caused by noise. In WINCC OA a further volume suppression, based on the processing dead band defined for each monitored parameter, is performed. Although each value arriving to the system is processed, only those exceeding the dead band settings are archived in the ORACLE database. Using this method, the DCS archives in total only ~ 400 parameters each second, without losing any information significant for the reconstruction. The smoothing efficiency is based on several years of detector operation and is constantly improving. The same principle will be applied to the new data publishing mechanism and will significantly reduce the traffic between the DCS system and the ADAPOS.

The current simulations show, that the ADAPOS process image shall receive in total ~ 1 k updates each second, after applying the noise rejection filtering. Using a simulated environment, ADAPOS can safely cope with this amount of input data with a contingency of about 2 orders of magnitude. It is expected, that in the first phases of the operation with new detectors, the smoothing efficiency will not be optimal and the DCS infrastructure will need to cope with bigger amounts of data. The present implementation of ADAPOS can cope with the full DCS data load even without the smoothing.

The process image is streamed to the O2 infrastructure. Each of the 1500 EPN nodes will receive the same conditions data for each 50ms data frame. In the test setup, stable streaming of the process image containing 100k parameters

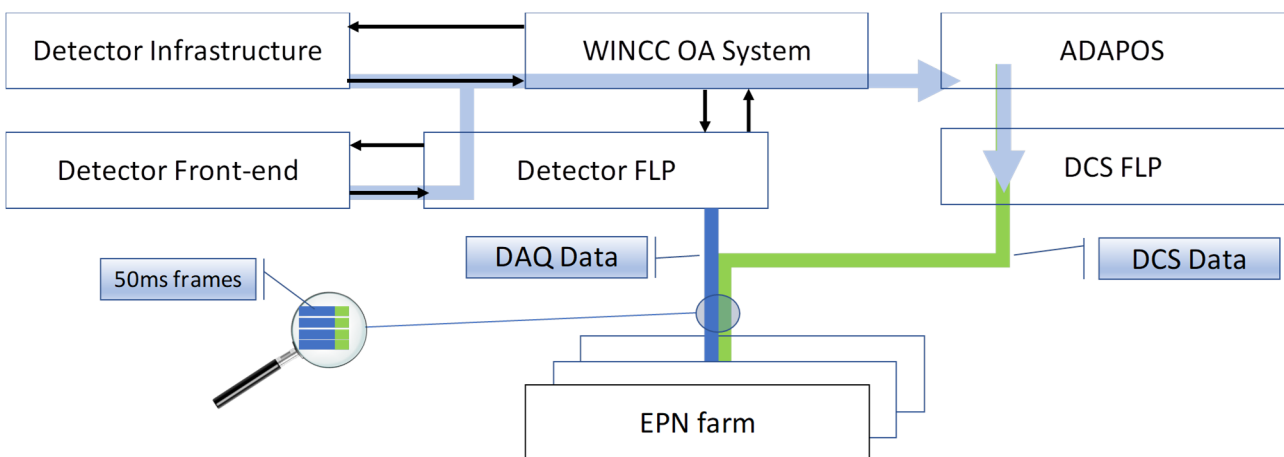


Figure 2: The DCS dataflow in the O2 architecture.

value. The present implementation is based on DIM protocol, supported on the WINCC OA platform at CERN. For each value a corresponding DIM service is created and the published value is updated on each change. Using the DIM DNS service ADAPOS finds the values and establishes a peer-to-peer connection to each publishing server.

has been achieved using the 0MQ protocol. The data flow from the DCS devices to the O2 processing farm is shown in Fig.2.

The immediate processing of the physics data in the O2 facility does not tolerate any downtime of the ADAPOS

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2017). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

service. A system redundancy is one of the critical requirements as missing conditions would block the data reconstruction algorithms. The developed architecture allows for parallel operation of multiple ADAPOS servers at the same time. Each of these servers subscribes to the conditions data and maintains the same process image. In case of server failure, the O2 client will automatically reconnect to the next available ADAPOS server.

The assurance of data availability is one of the challenging tasks for the control system. Due to the missing triggering mechanism for the conditions data readout, the system cannot predict the arrival time of the next update. A faulty channel that will stop updating the values might escape the system attention for a relatively long time. A proactive check of the parameters requires a deployment of relatively complex algorithms customized for each individual system in the distributed environment. ADAPOS provides an elegant way for addressing this problem. A dedicated software can access the process images and perform regular checks of the values. Noisy channels, or channels not publishing the values can be detected by a common quality assurance system and corrective actions can be taken.

Modified Frontend Access

New electronics, prepared for the LHC RUN3 requires also modifications of the whole front-end DCS subsystem. The front-end modules, installed near the detectors in the ALICE experiment cavern use the CERN GBT link to transfer the detector signals at 4.4 Gbit/s speed. The link is supervised by the Common Readout Unit (CRU) installed in the FLP computers. In total, 270 FLPs will host 540 CRUs and control 10 000 GBT links. The same path is used for accessing the DCS information.

The physics information acquired from the detectors flows directly from the FLPs to the EPN farm. Information from the individual detectors is stored in a data structure containing all ALICE information recorded during the 50ms interval. This data is later merged with the DCS information provided by the ADAPOS and the common DCS FLP (see Fig. 2).

Physics data frames are interleaved with frames carrying the DCS information. This part of the data is stripped off the main data stream and redirected to the DCS.

The DCS data packets are created by the front-end electronics and are inserted into the data stream. Alternatively, the readout of the DCS parameters can be triggered by the DCS, using dedicated commands sent to the electronics using the same optical link.

A typical detector front-end module implements the DCS functionality using the CERN developed Slow Control Adapter (SCA) chip interfaced to the GBT link [7]. It provides a wide range of possibilities to access the controlled parameters using multiple busses like JTAG, I²C, parallel I/O bus etc. Although the front-end architectures and DCS requirements of the detectors are largely different, the common SCA-GBT architecture allows for standardization at the software level.

The GBT links are controlled by the Common Readout Units (CRU) installed in the FLPs. Here the DCS data is extracted from the data stream and sent to ALF (Alice Low Level Front-end) interface, which publishes the data to the upper layers of the software.

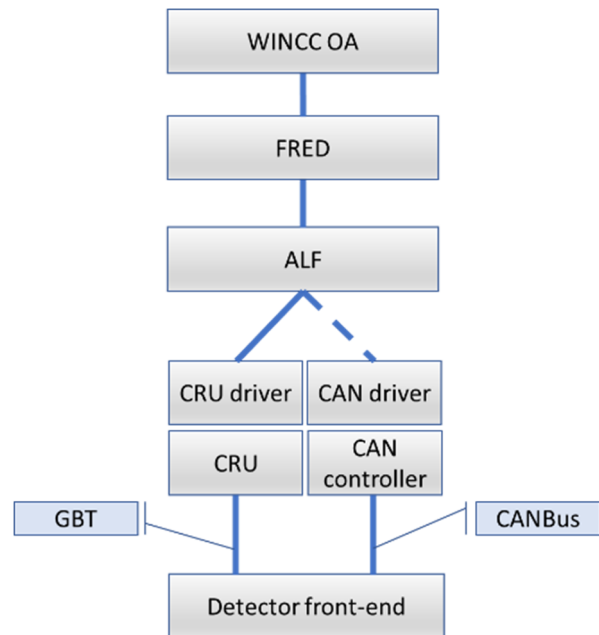


Figure 3: The ALF-FRED architecture of the DCS.

ALF can also receive commands and converts them to data words to be sent to the front-end electronics. To keep the ALF detector neutral, its functionality is restricted to the basic I/O operations. In the current implementation, the ALF can read/write registers implemented on the front-end modules and publish the data using a DIM service. The data published by ALF could be single values, or blocks of data prepared by the electronics modules.

Communication between the WINCC OA systems and the ALFs is managed by the Front-End Device (FRED) module. This layer provides the necessary translation of high level WINCC OA commands to simple ALF transaction and unpacking the ALF data before it is published to the WINCC OA.

The ALF-FRED architecture decouples the front-end details from the high level SCADA system. Separating this task into 3 layers of software – the drivers, the ALF and the FRED (see Fig. 3), brings clear advantages – the ALF remains detector independent and can be deployed to all detectors. The FRED layer provides highly customizable detector-specific module which implements all resource intensive calculations, such as data unpacking and first level filtering.

The WINCC OA system implements the standard controls functionality covering the control, monitoring, alert handling and data visualization and archival. From the WINCC OA perspective, the ALF-FRED behaves as any other standard device.

The present ALF-FRED implementation is based on DIM protocol. It allows for easy integration of complex detector granularity into a coherent system. The ALF modules are installed for each FLP belonging to the same detector and are serviced by a dedicated FRED. The SCADA system recognizes the FRED as a device that recognizes high level commands (such as configure, turn on/off) and publishes its data as single services. It is the task of FRED to translate the high-level commands into a sequence of atomic actions to be carried out by ALF.

The separation of the commands and management of their complexity through the different ALF-FRED layer brings an additional advantage. At the lowest layer, the ALF is interfaced to the CRU through a driver developed in ALICE. It takes care of CRU transactions over the GBT link. Replacing this driver allows for use of a different field bus (such as CANbus) without the need of extensive software modifications. The ALF with modified driver will provide the same functionality to FRED and the modification of the front-end access will remain transparent. This approach is used in ALICE for improving the redundancy. Some detectors implement CANbus in their front-end. During the downtime of the GBT link (for example during the FLP maintenance), the CANbus will take over the controls functionality and will eliminate the need for shutting down the operations. The communication speed will be strongly reduced, however it will remain sufficient for assuring the detector safety and will even allow for less demanding applications (such as detector calibration or debugging of the operational procedures).

CONCLUSIONS

The ALICE upgrade is a challenging task for the whole collaboration. Upgraded detector hardware and change of the data processing strategy required the redesign of the DCS data flow. The data will be provided to the O2 processing facilities in streamed mode for which the DCS data stream has to be diverted from its standard path.

Access to new front-end modules will be shared between the DCS and the Data acquisition. A new device access strategy covers the complexity of the new hardware and provides a flexible mechanism for transferring the information between the DCS and the hardware. Separation of the functionalities using a three layer architecture allows for splitting of the common and detector specific tasks. This approach improves the maintenance and allows for shared developments.

REFERENCES

- [1] P.Chochula *et al.*, "Control and monitoring of the front-end electronics in ALICE", *9th Workshop on Electronics for LHC Experiments*, Amsterdam 2003.
- [2] C. Gaspar, M. Donszelmann, "DIM – A distributed information management system for the DELPHI experiment at

CERN", *Proceedings of the 8th Conference on Real-Time Computer applications in Nuclear, Particle and Plasma Physics*, Vancouver, Canada, June 1993.

- [3] B. Franek, C. Gaspar, "SMI++ - An object oriented framework for designing distributed control systems", *IEEE Trans. Nucl. Sci.*, Vol 45, Num 4, p. 1946-1950.
- [4] P. Chochula *et al.*, "Handling large amounts of data in ALICE", in *ICALEPCS07*, Knoxville, USA 2007.
- [5] P. Moreira, A. Marchioro and K. Kloukinas, The GBT: a proposed architecture for multi-Gb/s data transmission in high energy physics, in *Topical Workshop on Electronics for Particle Physics*, Prague Czech Republic September 3–7 2007, pg. 332 [CERN-2007-007.332].
- [6] M.Richter, A design study for the upgraded ALICE O2 computing facility, *J. Phys.: Conf. Ser.* 664 082046.
- [7] A.Caratelli *et al.*, The GBT-SCA, a radiation tolerant ASIC for detector control and monitoring applications in HEP experiments, in *Topical Workshop on Electronics for Particle Physics 2014*, Aix En Provence, France, 22 - 26 Sep 2014, pp.C03034.