

REUSING THE KNOWLEDGE FROM THE LHC EXPERIMENTS TO IMPLEMENT THE NA62 RUN CONTROL

M. Gonzalez-Berges, N. Lurkin, F. Varela[#], CERN, Geneva, Switzerland

Abstract

NA62 [1] is an experiment designed to measure very rare kaon decays at the CERN SPS planned to start physics data taking in 2014. Until this date, several intermediate run periods have been scheduled to exercise and commission the different parts and subsystems of the detector. The Run Control system monitors and controls all processes and equipment involved in data-taking. This system is developed as a collaboration between the NA62 Experiment and the Industrial Controls and Engineering (EN-ICE) Group of the Engineering Department at CERN. In this paper, the contribution of EN-ICE to the NA62 Run Control project is summarized. EN-ICE has promoted the utilization of standardized control technologies and frameworks at CERN, which were originally developed for the controls of the LHC experiments. This approach has enabled to deliver a working system for the 2013 Technical Run that exceeded the initial requirements, in a very short time and with limited manpower.

INTRODUCTION

The Detector Control Systems (DCS) of the four major LHC Experiments are a good example of the utilization of common components and finding synergies across collaborations. However, this situation is unfortunately rather exceptional in the HEP domain. For instance, a different Run Control system was implemented for each of the Trigger and Data Acquisition (TDAQ) system of the LHC Experiments. This is justified, to some extent, due to the fact that TDAQ systems make use of a large variety of custom and dedicated hardware, which imposed very stringent requirements, in terms of flexibility and openness, on the software used for the supervisory layer. In addition, in three out of the four collaborations, the various controls systems within one experiment were implemented by different teams and using different tools. This has lead to the proliferation of small control systems and the need of dedicated interfaces amongst them to operate the experiments coherently. The exception to this is LHCb [2] where the decision was taken to implement Run Control utilizing the same technologies, concepts and design principles as for the DCS. This approach has lead to an integrated overall control system, which allows operating all different services and sub-detectors of the experiment in a coherent and homogeneous manner. A similar approach to LHCb is currently being followed to implement the Run Control System and the DCS [3] of the NA62 Experiment.

[#]fernando.varela.rodriguez@cern.ch

INDUSTRIAL CONTROLS AT CERN

Although not the first ones, the DCS of the four major LHC experiments constitute, without any doubt, an inflection point in the usage of commercial industrial technologies and components-off-the-shelf at large scale for controls amongst the HEP community.

The DCS of the LHC Experiments were developed in the frame of the Joint COntrols Project (JCOP) [4]. JCOP was set up in 1998 as a collaboration between the four major LHC experiments and the central controls group at CERN. The main goal of JCOP was and continues to be, to increase efficiency by avoiding duplication of efforts, and by promoting the usage of standard components and interfaces in the control systems.

Similar motivations were at the origin of EN-ICE in 2009 as result of merging different controls groups at CERN working with similar technologies although in different domains. For these reasons, the promotion of industrial technologies, as well as the standardization of components used in control systems at CERN became the flagship of EN-ICE. Dedicated and ad-hoc solutions are restricted to areas where no industrial technology is applicable or cost is a limiting factor.

EN-ICE is involved in the development and operation of more than 130 applications that control vital services of the accelerators, the experiments, as well as of the technical infrastructure. All these applications build on top of three main components:

- Standardized hardware like Siemens or Schneider PLCs, commercial power supplies, as well as industrial standards like CANbus or Profibus.
- The commercial SCADA package WinCC Open Architecture [5] from ETM (a Siemens company).
- The JCOP [6] and UNICOS [7] Frameworks.

These technologies have proven to be especially well suited for the implementation of the experiments' DCS. Experience from LHCb now also confirms that these technologies can be efficiently used in the TDAQ domain.

NA62

The main goal of the NA62 experiment is to detect $O(100) K^+ \rightarrow \pi^+ vv^-$ events over two years of data taking with less than 10% background. To collect the 10^{-13} required kaon decays, the 75 GeV beam intensity is very high. The beam will contain 7% of kaons. For this reason, the detector will contain a KTAG differential Cerenkov counter to positively identify them. The GigaTracker spectrometer is a fast silicon pixel detector that will have to sustain the 800 MHz particle rate and measure the kaon

momentum with a time resolution of 200 ps and a 0.1% momentum resolution. The charged pion will be identified by the RICH ring-imaging Cerenkov with respect to muons and its momentum will be measured by the Straw spectrometer. All the other detectors are vetoes that will ensure the hermeticity to photons (LKr, IRC, SAC and LAVs), to muons (MUVs) and finally to veto any charged particle from accidental interaction with upstream detectors (CHANTI and CHOD). All these components are tied together to form a coherent detector with a high-performance TDAQ system. The TDAQ system is formed of number of acquisition and control boards that are developed specifically for the experiment.

NA62 RUN CONTROL

The NA62 Run Control system is developed as a collaboration between of NA62 and EN-ICE. This collaboration started back in 2008 for the DCS and in 2012 its scope was extended to cover the implementation of the Run Control. The work on the NA62 Run Control started only a few months prior to the first technical run scheduled in Q3 2012. The very short time available to develop the system, as well as the limited manpower (1-2 developers), determined many decisions taken on the design of the Run Control system and the technologies used for its implementation.

Requirements

Run Control is responsible for the configuration, monitoring and operation of all equipment and processes that intervene in the trigger and data acquisition of NA62. Run Control provides the glue between the DAQ, Trigger and the DCS. Figure 1 shows the context of the NA62 Run Control.

The main requirements are:

- Hierarchical and homogeneous control and configuration of all different parts of the TDAQ and NA62 sub-detectors according to a well defined set of states and transitions amongst them.
- Support of the various data taking modes providing an easy way to reconfigure the readout electronics for these different types of runs.
- Partitioning, allowing for concurrent and stand-alone operation of the different NA62 sub-detectors.
- A powerful error handling strategy and an intuitive way to localize possible problems in the large set of hardware and software elements of the TDAQ system.

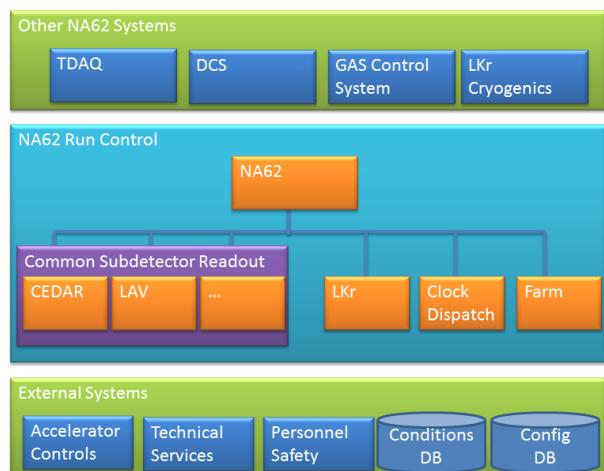


Figure 1: NA62 run control context.

Philosophy of Implementation

Two key principles drove the design and development of the NA62 Run Control:

- Selection of well-proven industrial technologies and components currently used in many controls applications at CERN.
- Re-utilization of the know-how acquired during the implementation of the controls of the LHC experiments.

The NA62 Run Control is largely inspired on the LHCb run control [2]. It uses the same concepts and the same tools to control and monitor the TDAQ as the DCS uses to control the detector hardware. This approach allows native interfaces between the DCS and Run Control and, although still to be decided, it enables the development of an integrated overall control system like in LHCb.

However, unlike for the LHC experiments where a highly customizable framework was developed centrally and then provided to the sub-detector developers, due to the very limited resources and the smaller size of the experiment, the NA62 Run Control was developed centrally with punctual contributions from various hardware experts. In all cases, dedicated developments were always based on common skeletons provided centrally to ensure the coherency and homogeneity of the implementations and to ensure their long-term easy maintenance.

Building Blocks

The NA62 Run Control builds on top of WinCC OA package and the JCOP and UNICOS Frameworks. WinCC OA provides the basic features of the control system like drivers to communicate with the hardware, archiving, trending and alarm handling. Both the JCOP and the UNICOS Frameworks build on top of WinCC OA to provide a set of generic tools for developers to build control systems. The main difference between the JCOP and the UNICOS Frameworks is that, whereas the latter defines the top-down architecture and look and feel of the

application, the former is more like a toolbox for engineers to build control systems leaving to the developers the presentation layer.

Three components of the JCOP Framework are of special relevance for the NA62 Run Control:

- The Distributed Information Management (DIM) [8]: a multi-platform middleware, which provides a network transparent inter-process communication layer. In NA62, specific actions on the detector electronics are performed by dedicated programs written in C/C++ by the hardware experts. These programs make use of DIM to report status and summary data to the SCADA application, and to receive commands.
- A Finite State Machine toolkit (FSM) [9] that allows modelling the different components of the control system as logical decision units that are characterized by a set of well-defined states and possible transitions amongst them. The FSM toolkit permits arranging these different units in a tree-like structure where the state of a particular node is calculated according to logical rules from the state of its children. Moreover, upon the reception of a command from its parent, each node in the tree propagates the command down to its children until the commands reach the hardware. The FSM allows for sequencing of actions and automatic error recovery. Furthermore, the partitioning capabilities of the JCOP FSM allow for the concurrent operation of the different parts of the tree as required for the stand-alone operation of the NA62 sub-detectors.
- The Configuration Database Tool [10], which allows storing and versioning of the configuration parameters in a relational database outside of WinCC OA. The different operational modes of experiment impose the need to quickly reconfigure the readout electronics. All these parameters are stored in an Oracle database to ease the access from third-party applications.

Architecture

The decision was taken by the NA62 collaboration to develop and use standardized readout electronics. The resulting TEL62 board [11] is derived from the TELL1 used in LHCb. The LTU (Local Trigger Unit) are in number the second type of controlled devices (20 in total). Each subsystem contains one LTU that will be used to relay the trigger signal from the central trigger processor (L0TP) or emulate triggers in case of independent run. The TEL62 boards (80 in total) are now used by practically all sub-detectors and are, hence, the device that defines the way the control software is developed. All the other types of boards used in the

TDAQ system implement the same communication interface.

Following a similar approach to how the JCOP Framework was developed, the NA62 Run Control software is also divided into different components to allow for modularity and to ease the integration of new DAQ hardware. There are four kinds of components:

- Hardware interfaces: These programs are DIM servers written in C/C++ by the hardware experts. These programs are based on a common skeleton and enable the communication between the hardware and the WinCC OA application of Run Control. This intermediate layer encapsulates the knowledge of the hardware and implements a generic interface with the core component.
- Core: Contains all common libraries and visual elements, as well as a collection of scripts to gather information from various sources, e.g. the end of burst signal from the SPS, and make them available to other services of the experiment via DIM.
- Devices: Each of these components contains libraries and visual elements to communicate and represent each device type.
- FSM: This component implements the FSM behaviour for each device and each logical node in the hierarchy.

Figure 2 shows the FSM tree of the NA62 Run Control. In this tree-like structure, leaves model the devices whereas the nodes above are logical groups representing the hierarchical structure of the TDAQ system.

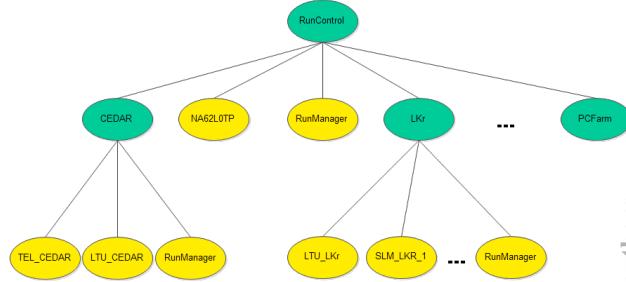


Figure 2: NA62 simplified FSM tree.

The Run Manager entities define the type of trigger used during a run, i.e. global vs. local. Sub-detector Run Manager nodes enable the standalone operation of a sub-detector using local triggers whereas for global data-taking, the overall RunManager instance is used.

Figure 3 shows the FSM state diagram followed by the NA62 Run Control, i.e. the top-most node in Figure 2. IDLE represents the state of the experiment after a power up or a reset. The INITIALIZE command forces the configuration of the readout electronics according to the desired type of data-taking to be performed. The successful execution of this command brings Run Control into the INITIALIZED state. The START_RUN command triggers the readout electronics to acquire data,

thus bringing Run Control into the READY state. It is important to note that although data-acquisition has been started, at this stage, the data are not sent to the PC farm. The successful execution of the ENABLE_TRIGGER command causes the data to be sent to the farm for processing and recording. In event of a failure, the NA62 Run Control moves to the ERROR state and only a complete reset or a re-initialization of the electronics can recover the system from this situation.

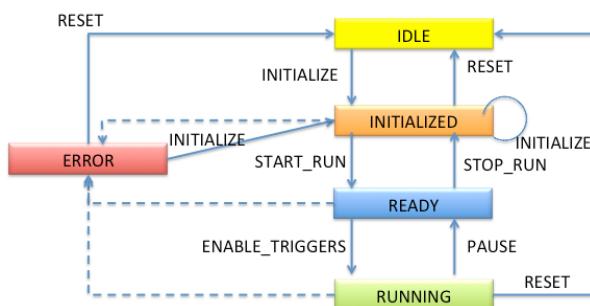


Figure 3: NA62 run control states and transitions.

Figure 4 shows the main user interface of Run Control, where the state of each sub-detector is colour coded. This user interface provides operators with a powerful way to operate the data-taking with a few mouse-clicks, as well as with an intuitive method to localise problems.



Figure 4: Main user interface of the run control.

The core of Run Control is physically located on a data server on the CERN technical network, whereas all other components of Run control are on the experiment network. A single communication point accessible from both networks acts as a bridge for the communication with the hardware. The user interface is installed on a computer in the control room and remotely connected to the central system.

CONCLUSIONS

The reutilization of tools originally developed for the DCS of the LHC experiments and the accelerator services, as well as the know-how acquired during their implementation allowed for a rapid development of the NA62 Run Control with very limited resources. A first version of the system was available for the technical run only a few months after the start of the implementation.

The functionality of the system exceeded the original expectations. It is important to highlight that despite the reduced investment done so far, the experience gathered with the current system proved that it builds on top of solid principles and foundations, thus proving the correctness of the decisions taken. For these reasons, the final NA62 Run Control will be based on the current implementation. However, there is still a long way ahead to extend the current functionality and to improve the robustness of the system and the required effort must not be underestimated. Based on the current experience, we believe that the NA62 Run Control is on the right track and we look forward to the start of data-taking in 2014.

ACKNOWLEDGMENT

We would like to express our most sincere gratitude to the EN-ICE and NA62 colleagues working on the NA62 DCS for the sharing of the computing infrastructure with Run Control, as well as to R. Fantechi, M. Sozzi and G. Lamanna for defining the requirements for the system. Thanks also to C. Gaspar for the useful input on the LHCb Run Control and her continuous support on DIM.

REFERENCES

- [1] F. Hahn (ed.) "NA62 Technical Design Document".
- [2] C. Gaspar et al., "The LHCb Experiment Control System: On the Path to Full Automation", ICALEPCS'2011, Grenoble, France (2011).
- [3] J. Arroyo-Garcia et al., "Integrating Controls Frameworks: Control Systems for NA62 LAV Detector Test Beams", ICALEPCS'2011, Grenoble , France (2011).
- [4] D. R. Myers et al, "The LHC experiments Joint COntrols Project, JCOP", ICALEPCS'1999 Trieste, Italy, (1999).
- [5] ETM professional control GmbH – WinCC Open Architecture, <http://www.etm.at/>
- [6] M. Gonzalez-Berges et al., "The Joint COntrols Project Framework", Computing in High Energy and Nuclear Physics, La Jolla, California, 2003.
- [7] H. Milcent et al., "UNICOS: An open framework", ICALEPCS'2009, Kobe, Japan (2009).
- [8] C. Gaspar et al., "DIM, a portable, light weight package for information publishing, data transfer and inter-process communication", Computer Physics Communications 140 1+2 102-9, 2001.
- [9] B. Franek and C. Gaspar, "SMI++ - an object oriented Framework for designing distributed control systems", IEEE Trans. Nucl. Sci. 45 4 1946-50, 1998.
- [10] L. Del Cano, "Extending the capabilities of SCADA – Device Modelling for the LHC Experiments", ICALEPCS'2003, Gyeongju, Korea (2003).
- [11] E. Pedreschi et al., "From TEL1 to TEL62", NA62 Note NA62-10-06, 2010.