Status of the CMS Detector Control System

Gerry Bauer [6], Ulf Behrens [1], Matthew Bowen [2], James Branson [4], Sebastian Bukowiec [2], Sergio Cittolin [4], Jose Antonio Coarasa [2], Christian Deldicque [2], Marc Dobson [2], Aymeric Dupont [2], Samim Erhan [3], Alexander Flossdorf [1], Dominique Gigi [2], Frank Glege [2], <u>Robert Gomez-Reino</u> [2], Christian Hartl [2], Jeroen Hegeman [2a], Andre Holzner [4], Yi Ling Hwong [2], Lorenzo Masetti [2], Frans Meijers [2], Emilio Meschi [2], Remigius K. Mommsen [5], Vivian O'Dell [5], Luciano Orsini [2], Christoph Paus [6], Andrea Petrucci [2], Marco Pieri [4], Giovanni Polese [2], Attila Racz [2], Olivier Raginel [6], Hannes Sakulin [2], Matteo Sani [4], Christoph Schwick [2], Dennis Shpakov [5], Michal Simon [2], Andrei Cristian Spataru [2], Konstanty Sumorok [6]

[1] DESY, Hamburg, Germany
[2] CERN, Geneva, Switzerland
[3] University of California, Los Angeles, Los Angeles, California, USA
[4] University of California, San Diego, San Diego, California, USA
[5] FNAL, Chicago, Illinois, USA
[6] Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

[a] Now at Princeton University

E-mail: robert.gomez-reino@cern.ch

Abstract. The Compact Muon Solenoid (CMS) is a CERN multi-purpose experiment that exploits the physics of the Large Hadron Collider (LHC). The Detector Control System (DCS) is responsible for ensuring the safe, correct and efficient operation of the experiment, and has contributed to the recording of high quality physics data. The DCS is programmed to automatically react to the LHC operational mode. CMS sub-detectors' bias voltages are set depending on the machine mode and particle beam conditions. An operator provided with a small set of screens supervises the system status summarized from the approximately 6M monitored parameters. Using the experience of nearly two years of operation with beam the DCS automation software has been enhanced to increase the system efficiency by minimizing the time required by sub-detectors to prepare for physics data taking. From the infrastructure point of view the DCS will be subject to extensive modifications in 2012. The current rack mounted control PCs will be replaced by a redundant pair of DELL Blade systems. These blade servers are a high-density modular solution that incorporates servers and networking into a single chassis that provides shared power, cooling and management. This infrastructure modification associated with the migration to blade servers will challenge the DCS software and hardware factorization capabilities. The on-going studies for this migration together with the latest modifications are discussed in the paper.

1. Introduction

The CMS detector is a general purpose detector designed to access a broad range of physics topics over the full range of luminosities provided at the LHC.

The CMS layout corresponds to a layered cylindrical detector structure. Each detector system is usually composed of a cylindrical barrel with the axis along the particle beam direction and two end caps enclosing those barrels. In order to measure precisely the momentum of particles, a large solenoidal magnetic field is used. To create such a field, a superconducting solenoid sits in the middle of CMS. This solenoid is designed to produce a 4 T field [1]. Due to the large return magnetic field, a long 1.5 meter iron return yoke is used. Four muon stations are integrated between the layers of the return yoke resulting is a compact muon system [2] with full geometric coverage. Inside the magnetic solenoid there is an inner tracker [3] and a calorimeter. Ten layers of silicon microstrip detectors provide CMS with the required granularity and precision. Furthermore, another three layers of silicon pixel detectors placed close to the interaction point improve the measurement of the impact parameter of charged-particle tracks, as well as the position of secondary vertices. An electromagnetic calorimeter (ECAL) [4] surrounds the tracker system. Between the magnetic solenoid and the electromagnetic calorimeter a hadronic calorimeter (HCAL) [5] is placed.

The **Detector Control System** (DCS) is responsible of ensuring the safe and optimal operation of the experiment so that high quality physics data can be recorded by the data acquisition system [7]. The DCS input data rate is in the range of 10^4 MB/s. This data is not only stored to a database but it also needs to be processed at this rate in order to take automatic decisions and sequence commands that are sent to different parts of the system.

2. The DCS Overview

The distributed control system of the CMS experiment has to meet some challenging requirements derived from the scale and characteristics of the experiment. Some of its hardware needs to operate in an environment with radiation doses up to 10^4 Gy/year and a high magnetic field. In addition, many locations inside the detector are inaccessible without moving sections of the detector weighting several tons.

The design of the control system, like in other international projects of the size of CMS, involved many groups working locally at CERN in collaboration with others groups working remotely. The organization of the project and its infrastructure needs to accommodate this working style with local and remote man power.

Other requirements are imposed from operational aspects. The DCS system has to run without interruption, eventually only stopping for long shutdown periods. The system should provide an interface intuitive and powerful enough to allow a single operator, with a few hours of training, to control and monitor the whole experiment. Finally, the system should work in synchronization with the LHC machine state so that the experiment is ready for data taking whenever LHC is delivering stable colliding beams and, at the same time, it makes sure that the experiment is in a safe mode for other not stable LHC conditions.

2.1. The control system infrastructure and technologies

CMS experiment facilities are located in Cessy (France). From the DCS point of view the building complex (Fig. 1) can be divided in three main areas: the surface buildings, the underground detector hall and the underground computer hall. In the surface counting room (SCX) sit the experiment online database together with the web servers, sharing room with a 3000 server farm used for the physics event building and filtering. Also on the surface, the experiment control room (also in SCX) contains the DCS human-machine interface. The operator controls the experiments from this room in collaboration with a small shift crew in charge of other systems. The detector is installed in the detector hall (UXC55) where there are a few hundreds of electronics racks. The DCS equipment in these racks needs to be radiation tolerant. Low voltage power supplies regulating the power to each channel are located in this area, as close as possible to the detector, since low voltage lines cannot

travel through long distances without significant attenuation. Also, some high voltage power supplies are installed in the detector hall mainly due to the complicated cabling structure needed to bring them to another location. The computer hall (USC55) also hosts the power supply mainframes cannot withstand the detector hall environment. Together with the industrial power supplies CMS uses in-house power supply solutions where the industrial equipment didn't satisfy sub-detector needs.
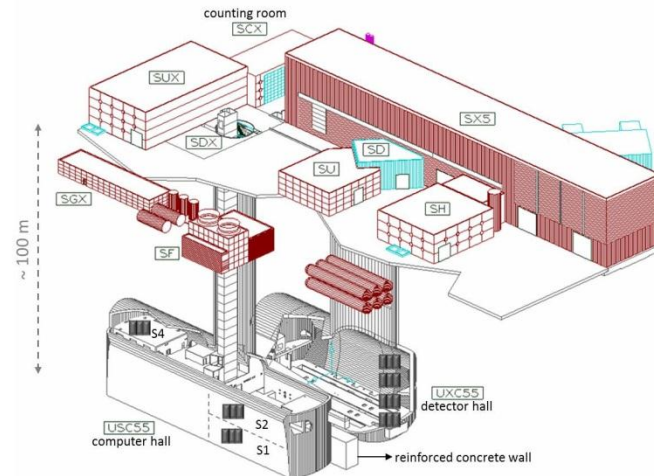
Figure 1. The CMS experimental facilities

In addition to the equipment supplying power there are also CERN-made solutions used in the DCS. The Embedded Local Monitoring Board (ELMB) [6] is a radiation tolerant data acquisition board developed by the ATLAS experiment. It provides analog and digital input/outputs. The card is used all over the experiment facilities (including inside the detector cavern) for monitoring all kind of sensors (like temperature, humidity and pressure) and also to control other hardware by means of its digital and analog outputs. The Detector Control Unit (DCU) and the Readout Boxes (RBX) are CMS-made solutions installed on the detector itself. Both solutions have a controller module that is used to control and monitor different bias voltages used by the electronics of the particle detector sensors. Finally, Programmable Logic Controllers (PLCs) are used to control critical processes involving safety aspects. **Table 1** summarizes the list of the most broadly used hardware in the experiment DCS. The table includes the estimated number of parameters directly associated with the readout and control of the equipment. The communication drivers used are also listed in the table. The commercial power supplies used are addressed using OPC [7] (Object Linking and Embedding (OLE) [8] for Process Control) either over Ethernet or using Controller Area Network (CAN) [9] field buses. The OPC standard specifies the communication of real-time infrastructure data between control devices from different manufacturers. The Distributed Information Management system (DIM) [10] created at CERN is used by the custom hardware to communicate via Ethernet with the distributed control system PCs. A DIM server can be configured to publish DIM services containing information relative to the monitored hardware. These services can also be used to send information, like commands or settings, to the hardware. Any number of DIM clients can connect to a DIM service. PLCs are connected to the DCS system using industrial communication drivers (SIEMENS S7 [11] and Modbus [12]) over Ethernet.

Table 1. Example of some representative DCS control and monitoring hardware used by CMS

Device type	Usage	Brand	Driver	Parameters
Power supply	Front end electronics and detector biasing	CAEN W-IE-NE-R CMS-made	OPC/Ethernet OPC/CAN DIM	~2.5 M
Embedded Local Monitoring Board (ELMB)	Temperature, humidity and pressure monitoring Water leak detection Laser monitoring	CERN-made	OPC/CAN	~24 K
DCU, RBX	Detector monitoring	CMS-made	DIM	~0.5 M
PLC	Safety, Cooling Rack electrical distribution	Siemens Schneider	S7 Modbus/tpc	~12 K

2.2. The Control System Software
The detector control system rack mounted PCs (about 100) are also located in USC55 in S1, S2 and S4 distributed in 14 racks. The next sub-sections gives the details on the technology choices for the operating system, the computer management tools and the software chosen to build the distributed control system.

2.2.1. *The Operating System*. The choice of the operating system for the control PCs was driven by the use of OPC communication. At the time of the choice the OPC protocol was only available for Windows operating systems. Windows XP was chosen by CMS as operating system for all control PCs.

2.2.2. *The Controls Software*. PVSS commercial Supervisory Control and Data Acquisition (SCADA) software was selected [13] as the development package for the detector controls at CMS and the other LHC experiments. The engineering controls CERN group together with the LHC experiments founded the Joint Controls Project (JCOP) in charge of identifying common needs across the experiments and developing common solutions for them. The result of this project, the JCOP framework, consist on a set of tools and templates built in PVSS to simplify the task of connecting and creating the PVSS infrastructure needed to control and monitor the hardware commonly used by the experiments.

2.2.3. *The Computer Management*. CERN has built its own framework to manage Windows PCs across whole site. The Computer Management Framework (CMF) allows for creating sets of computers than can be customized by the experiments to configure the software and the policies that are installed on each of them. All the Windows machines used at CMS for controls are configured using CMF. The central DCS team takes care of the maintenance of the whole DCS computing infrastructure so that from the sub-detector point of view the PCs are no more than a computing service. This strategy ensures that the exact states of the machines can be automatically recovered in a new PC when replacing a faulty production one.

2.3. The System Architecture
Figure 2 shows an overall view of the CMS DCS architecture. The whole detector is controlled from a single operator station. A user interface (1) remotely connects to the central DCS PVSS system providing the operator with a global control of the whole detector (3). This control is provided by

means of a logical node tree which has a parent node for each of the experiment sub-detectors. This way, the central tree takes control of each of the sub-detector trees (8). The central DCS operator can partition out parts of the control tree and delegate their operation to the sub-detector control stations (9). In addition to the sub-detectors, central DCS also connects to other services (5) and models those services in the control tree. In particular, the power distribution of the ~500 experiment racks is modeled in a sub-tree as well as well as the experiment cooling system. The communication to the LHC machine (4) is modeled in another tree that interprets different defined handshake protocols. On the other hand, the experiment Run Control system (2) is not modeled in the control tree but it does spy on it to be aware of the status of each of the detectors participating in a physics run. The sub detectors are modeled (10) following a set of conventions and rules dictating the commands and states they should implement on the higher layers so that central DCS can communicate homogeneously with the sub-detectors most top nodes. The particularities of each sub-detector are hidden down on their control tree. At the bottom of the trees there are the proxy type nodes. These nodes, called device units (DU), model the hardware devices. A direct line connects CMS protection mechanism to the hardware (11). This direct connection goes in parallel to the control tree so it ignores any possible created partition, setting directly the hardware devices to a desired safe state when required. Central DCS automates most of the detector control and uses the central control tree to dispatch commands to the sub-detectors (6).
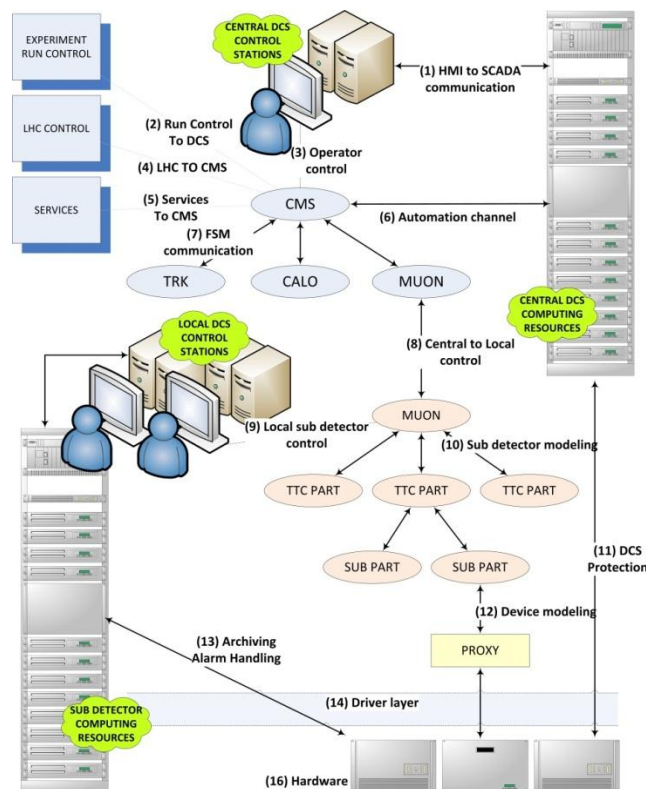

Figure 2. The DCS architecture

2.3.1. *The Control Tree*. The modeling of the CMS detector is implemented using the Finite State Machine (FSM) toolkit. The FSM toolkit is only partly developed in PVSS. Most of its modelling functionality is based in the State Modelling Interface (SMI++) [14]. SMI++ was originally developed

by DELPHI experiment during the LEP era. It uses its own language, the State Manager Language (SML), to model objects. The PVSS interface to SMI++ translates the FSM rules into SML language. PVSS connects at run time with SMI++ executable instances where the FSM node state calculations are done.

The complete CMS FSM tree is made of approximately 32000 nodes. To ensure a homogeneous and maintainable result from the development of each of the sub-detector trees the central DCS team created a set of naming and programming rules governing the design of FSM unit types and how they are put together to create the trees [15]. Even with the convention and rules provided, the JCOP FSM offers a good degree of flexibility and the complexity of the behavior of an FSM tree grows very fast when adding a very few levels and tree nodes types. For this reason, in addition to design guidelines, the central DCS team collaborates since 2009 with a group from the Technische Universiteit Eindhoven (TU/e) creating a set of tools that can automatically analyze the FSM tree looking for and finding potential problems before they happen in the production system [16]. The SMI++ SML language constructs used in the FSM tree are translated to mCLR2 [17] mathematical language. The aim is not to model the CMS detector but to model how the FSM itself behaves so that it can be investigated what might happen for particular trees (in this case CMS tree). The model can then be questioned for property verification. A first useful verification test was to inquire from the model whether with CMS's FSM tree it is possible that a combination of states generates a situation where an infinite loop of state changes can happen. Verifying properties of an FSM tree in mCRL2 is time consuming and computationally very expensive. The FSM tree modeled in mCRL2 has been found to have a state space of at least 10^{30000} states. It is of course computationally impossible to run checks on such a state space. Using Bounded Model Checking (BMC) [18] many tree properties can be studied locally in single FSM trees, reducing the computation time and state space ($\sim 10^{1190}$). The current objective is to create tools that can detect most of the problems already at design time. CMS is sharing these tools and other experiments are also using them to analyze their FSM trees.
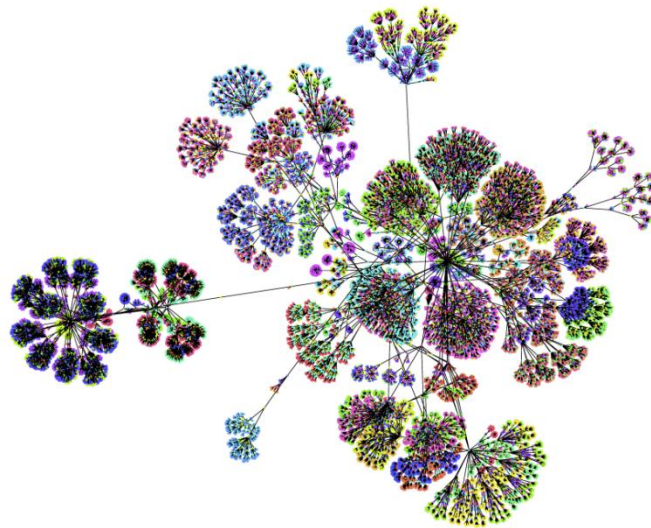

Figure 3. The CMS FSM graph

Figure 3 shows a representation of the CMS FSM tree created with one of the visualization tools of the TU/e group. The figure provides visual evidence of the size and complexity of this control tree.

2.3.2. *Automation and Protection*. The main reason to use automation in CMS is to contribute to the increase of efficiency in the recording of physics data. By automating the detector high voltage power ramping-up between LHC particle fills and stable colliding beams declaration, the DCS reduces to the

minimum the preparation time for physics data taking. There also automatic behaviours programmed to make sure that the sub-detectors are operated under safe conditions and turning off power when sub-optimal conditions are detected. When talking about the DCS Automation [19] in CMS we are referring to the central DCS automation. It is based on a three-dimensional action matrix where the sub-detector partitions are in one dimension, the LHC beam mode (like *injecting, ramping, squeezing…*) on another one and the LHC machine mode (*setup, proton_physics, ion_physics…*) on the third one. For each point on this three dimensional distinct space there is a DCS automatic action.

There are two types of automatic actions, the standard actions and the protection ones. Standard actions are sent using the FSM control tree. Since they use the control tree these actions are subject to the FSM partitioning rules. If any of the node that is supposed to be receiving a command is not in central but in local operation mode, then it will not receive the command. The second type of automatic action is the protection one. Protection actions do not use the FSM to propagate commands to the hardware and they are therefore not subject to FSM partitioning rules. The central DCS team designed this protection mechanism to ensure the maximum software protection that was possible to achieve within the JCOP framework.

2.3.3. *Plug-in infrastructure mechanism*. To achieve a flexible architecture and simplify the maintenance CMS, in collaboration with CERN engineering group, designed a plug-in mechanism [19] allowing for installing detector controls functionality with packages called components. These components follow the JCOP framework component implementation standards and are installed by means of a JCOP installation tool (to which CMS made many contributions). The functional components can be moved across the different distributed control nodes providing load balancing capabilities.

2.4. Remote monitoring and control tools

The central DCS team has developed a set of web applications allowing for remote monitoring of the control system. These applications are used on a daily basis by hundreds of CMS users. For this an Oracle Application server is used and the applications are developed as J2EE Java Portlets. The main web applications developed and some of their features are:

- Infrastructure monitoring: provides the information of the production PVSS project processes allowing also for commands to start/stop them. It also displays the configuration in terms of components installed in each of the projects and the connectivity within the distributed system.
- Component installation: it interfaces the installation tool used in CMS to deploy functional detector components into the production PVSS projects. The application users can download directly their developed functional components from their framework repository and target them into production project.
- DCS Operation: shows the summary information of the control system in terms of readiness for physics data taking. Includes information about the current LHC machine and beam modes and the automatic actions programmed for them. A log window showing the latest events and operator actions is also displayed.
- FSM monitoring: allow for browsing the entire FSM tree states and display color-coded tree diagrams.
- Alert screen: provides and online version of the DCS alert screen, including the possibility to visualize the alert history.
- Plotting tool: can access the database where all the conditions of the detectors are stored and make time plots.
- Rack and Crate monitoring: shows the information of the environment (mainly temperature, humidity and ventilation turbine currents) and the electrical distribution status of all the experiment racks. It also provides means to control remotely some experiment VME crates.

3. The DCS History & Current Status

Figure 4 summarizes the evolution of the detector control system since 1999 when PVSS was selected. Before 2003 the detector control system was briefly described in the experiment technical design report and most of the work was dedicated to choose the technologies. The JCOP framework started to be developed during those years. CAEN and WIENER were chosen as industrial providers for power supplies. The period 2003-2004 was mostly devoted to individual implementation of sub-system controls performed independently by the sub-detectors. The central DCS team started defining the global DCS design concepts in 2003 [20]. The work on the definition of the main development concepts resulted in the creation of the first version of a guidelines report [15] to be used by anyone developing controls for a CMS sub-system. In parallel to this work, the sub-detectors and the central DCS worked in a first proof of concept version of the control system for the Magnet Test Cosmic Challenge [21]. This test included a small fraction of each of the sub-detector and a simplified supervisor system. The system was operated by an expert for each of the systems participating.
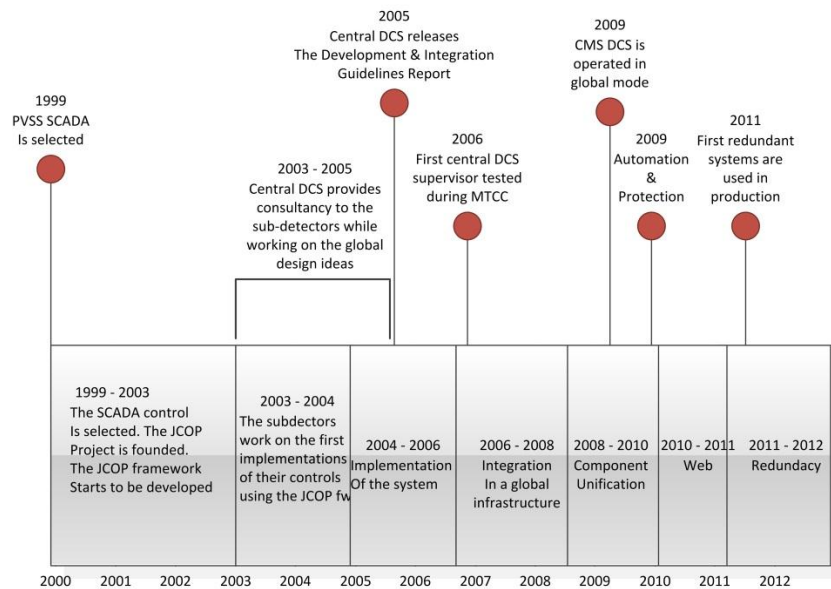
Figure 4. The DCS history

With the Magnet Test Cosmic Challenge (MTCC) operational experience the central DCS team started a long process of systems integration. Between 2006 and 2008, working closely with central DCS, each sub-detector and service system went through a process where the systems were adapted until they fully complied with the defined development guidelines. Due to this integration work the central DCS team could create a common system infrastructure to manage the controls PCs and their configurations simplifying the maintenance of the systems. Furthermore, the design guidelines governing the control tree implementation allowed for a coherent operation in central mode before LHC started its operation. With the systems integrated a new effort on the direction of identifying similar behaviors, patterns and needs in the sub-detectors started. The *unification* period, between late 2008 and 2010, was used by central DCS to replace controls algorithms in the sub-detectors that were doing similar things by common functional components solutions. The protection systems represent one of the main results of this period. With the machine operating stably during 2010-2011, the effort of the central DCS team was invested in gathering operational experience, improving the computing infrastructure and developing the extensive set of web applications. By the end of 2011 the central DCS team planned how to confront two upcoming events. First the end of life of the operating system in use (Windows XP) and second the aging of the whole computer infrastructure and the need for a replacement.

3.1. The Infrastructure Upgrade

CMS chose a challenging approach for the DCS computing infrastructure upgrade that can be divided in three parts: an OS upgrade, a hardware upgrade and an upgrade in the sub-detector components to make full use of PVSS redundancy feature.

3.1.1. *The OS upgrade.* CMS will upgrade the OS from Windows XP to Windows 7. CMS is the first experiment bringing production DCS systems to this OS. For this reason, a lot of investigation is been carried on within CMS and the CERN engineering controls group (EN-ICE) on the performance of the currently used drivers and the PVSS software on this OS. Additionally, many CMF packages are been adapted to work as well in the new OS. CMS has already successfully moved a few central DCS production system to Windows 7.

3.1.2. *The computing hardware upgrade.* The current computing infrastructure consists in about 100 rack mounted PCs scattered across the experiment (see Fig. 1). Rather than simply replacing the current PC by newer models, the central DCS group has chosen for a high-density modular solution (DELL Blade [22]) incorporating PCs and networking into a single chassis that provides shared power, cooling and management. The current objective is then to bring the whole distributed system to one rack hosting two DELL Blade-PC chassis, each one with 16 Blade-PCs running Windows 7 in the underground service cavern and a redundant copy in the surface counting room. The Blade-PCs used have 48 GB of RAM memory and 4 processors with 6 cores each. The new computing power should allow the reduction from the ~100 current servers (~3 GB RAM and dual core processors) to 32 Blade-PCs.

3.1.3. *The redundancy upgrade.* The Central DCS team has gathered experience on running redundant PVSS systems. The use of this functionality built in the SCADA allows the configuration of a PVSS system to run in two different PCs where one of them (the passive one) waits in a standby mode to take over whenever the active one fails. Within CERN, CMS is so far the only experiment using this PVSS feature. The second rack with two Blade chassis installed on the surface will run passive partners for each of the nodes running actively in the underground chassis blades. A paper in this same conference, titled "*High availability through full redundancy of the CMS detector controls system*", gives details on the current experience running PVSS redundant systems, the problems found and the plans for completing this upgrade.

4. Summary

The implementation of the CMS DCS fulfils the operational, functional and environmental requirements. It provides an efficient automatic detector operation and ensures its safe operation. It succeeds in coping with a few million parameters that are monitored and processed raising alerts when necessary and storing a set of these parameters into a database for later analysis.

To reduce developing efforts, the JCOP framework was used for the implementation of the control system. Additionally, CMS created a development guidelines report to be followed when using the JCOP framework. The use of these guidelines made possible the integration of all the sub-detector control systems in a common infrastructure facilitating the software maintenance.

The last objective/task of the central DCS's effort was to create a compact and redundant system infrastructure. The first production systems were already moved to the new infrastructure and the rest will be moved before the end of 2012 providing CMS with a high availability infrastructure.

References

[1] The CMS Collaboration, *The CMS magnet project : Technical Design Report*, CERN LHCC 97-10 (1997)

[2] The CMS Collaboration, *The CMS muon project : Technical Design Report*, CERN LHCC 97-32

(1997)

[3] The CMS Collaboration, *CMS Tracker Technical Design Report*" CERN LHCC 98-6 (1998)

[4] The CMS Collaboration, *The CMS ECAL Technical Design Report*, CERN LHCC 97-37 (1997)

[5] The CMS Collaboration, *The Hadron Calorimeter Project Technical Design Report*, CERN LHCC 97-31 (1997)

[6] B. Hallgren et al, *The Embedded Local Monitor Board (ELMB) in the LHC Front-end I/O Control System*, 7th Workshop on Electronics for LHC Experiments, Stockholm (2001)

[7] Wikipedia, http://en.wikipedia.org/wiki/OLE_for_process_control

[8] Wikipedia, http://en.wikipedia.org/wiki/Object_Linking_and_Embedding

[9] Wikipedia, http://en.wikipedia.org/wiki/CAN_bus

[10] C. Gaspar et al, *DIM, a Portable, Light Weight Package for Information Publishing, Data Transfer and Inter-process Communication*, International Conference on Computing in High Energy and Nuclear Physics, Padova (2000)

[11 SIEMENS, http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/Pages/Default.aspx.

[12] MODBUS ORG, http://www.modbus.org/specs.php.

[13] A. Daneels, *Selection and Evaluation of Commercial SCADA Systems for the Controls of the CERN LHC Experiments*, ICALEPCS, Trieste (1999)

[14] C Gaspar, *SMI++ - Object Oriented Framework for Designing Control Systems for HEP Experiments*, International Conference on Computing in High Energy and Nuclear Physics, Berlin, (1997)

[15] R. Gomez-Reino, *CMS DCS Integration Guidelines*, https://twiki.cern.ch/twiki/bin/view/CMS/DCSIntegrationGuidelines (2007)

[16] Y. L. Hwong et al, *An Analysis of the Control Hierarchy Modelling of the CMS Detector Control System*, International Conference on Computing in High Energy and Nuclear Physics, Taipei, (2010)

[17] J. Groote, *Analysis of distributed systems with mCRL2*, Process Algebra for Parallel and Distributed Processing (2009)

[18] A. Biere, *Bounded Model Checking*, Advanced Computing (2003)

[19] R. Gomez-Reino. et al, *The LHC Compact Muon Solenoid experiment Detector Control System*, Computing in High Energy and Nuclear Physics, Taipei (2010)

[20] R Gomez-Reino et al, *CMS DCS Design Concepts*, The International Conference on Accelerator and Large Experimental Physics Control Systems, Geneva, (2005)

[21] D. Lazic, *The CMS Magnet Test and Cosmic Challenge (MTCC) Operational Experience and Lessons Learnt*, 10th Topical Seminar on Innovative Particle and Radiation Detectors, (2007)

[22] DELL, http://www.dell.com/us/business/p/poweredge-blade-servers