The Run Control and Monitoring System of the CMS Muon Drift Tubes

S. Badoer[1], L. Berti[1], M. Biasotto[1], E. Frizziero[1], M. Gulmini[1,3], F. Lelli[1], G. Maron[1], P. Molini[1], A. Petrucci[1], S. Squizzato[1], N. Toniolo[1], S. Traldi[1], S. Ventura[2]

1 INFN Laboratori Nazionali di Legnaro, 2 Dipartimento di Fisica and INFN Padova, 3 CERN

INTRODUCTION

The CMS detector is under construction for imminent operation at the LHC machine at CERN. About 80 of a total of 250 muon drift tubes chambers were built in a large hall of the Legnaro National Laboratories and shipped to CERN. Chambers, front-end electronics, and data acquisition software require a flexible and reliable control and monitoring system for operating and testing the functionalities of the hardware and software components. Functionality tests are continuously performed both in Legnaro and at CERN. During summer and autumn 2006 the CMS collaboration took the crucial opportunity to operate pieces of all the sub-detectors together with cosmic rays and magnetic field. The Magnet Test and Cosmic Challenge (MTCC) included a 60° sector of the Muon System comprising drift tube chambers.

This report briefly describes the Run Control and Monitoring System of the Drift Tube used both during the MTCC for global CMS operation and in local DAQ systems in Legnaro.

CMS SOFTWARE OVERVIEW

Two general software frameworks for data acquisition, control and monitoring purposes have been developed for the CMS experiment. The XDAQ[1] framework provides the basic behavior of the single DAQ components, the communication protocols to configure the applications, to interface the hardware and to synchronize the readout and the event building.

The RCMS[2] framework, based on the GRIDCC (Grid Enabled Remote Instrumentation with Distributed Control and Computation)[3][4] middleware, provides a Java Web Service software infrastructure for Run Control and Monitoring functionalities, including protocols to interface to the XDAQ software, a default JSP graphical user interface and the means for developing custom control software and Graphical user interfaces. The RCMS framework has been developed by a collaboration between the CERN CMS DAQ group and the authors of this report.

The component of the RCMS responsible for performing the actual communication with the controlled set of resources is the Function Manager (FM). The RCMS FM allows for resource aggregation and hierarchic organization, thus achieving the goal of breaking down the complexity of the system and organizing the resources in groups. At the same time it provides the necessary building blocks that allow the customization of the control system. The FM receives control commands from the users that are operating the system and from other FMs. It can also receive inputs from the controlled resources, like states or errors, so it can react in an autonomous way to unexpected behaviors by allowing automatic recovery procedures to be started.

FM main components are the Input Manager, the Event Processor and the Finite State Machine Engine. The Input Manager waits for external inputs from users, other FMs or controlled resources and presents the corresponding events to the Event Processor. The Event Processor elaborates the events by triggering the proper control actions in the proper resource proxies. The Finite State Machine Engine is specifically designed in order to simplify the event processor algorithms. It provides a callback mechanism where writing the action that must be performed according to a particular triggered state transition.

THE DT DAQ AND DCS SOFTWARE

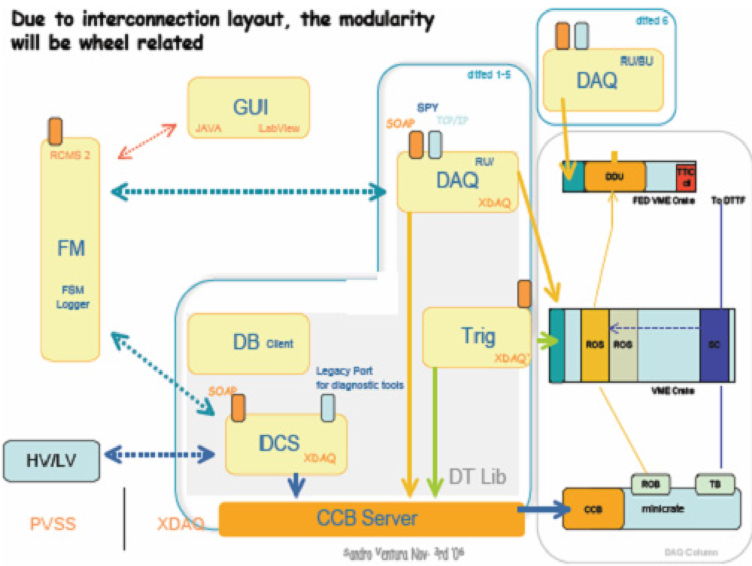

FIG. 1: *The DT DAQ and DCS Software*

The design and development of the data acquisition, control and monitoring software for the muon drift tubes, shown in Fig.1, follows the main guidelines of the CMS

DAQ team and it is then based on both the frameworks: the XDAQ framework for the basic DAQ functionalities (events read out from the front-end electronics and event building); the RCMS framework for all the tasks related to the experiment control, including the Detector Control System (DCS), the XDAQ nodes and the interface to the users. A RCMS FM controls the DAQ applications and the DCS functionalities using a SOAP protocol. In the DT case, DCS has the task to setup, control and monitor the front-end electronics (minicrates) and to provide the high and low voltage of all the DT chambers. A custom JSP GUI provides the main user interface for local data taking. A RCMS logging collector is also used for monitoring the behavior of the system.

THE DT FUNCTION MANAGER

The CMS global DAQ operation mode foresees a two-level hierarchy of Function Managers (fig. 2).

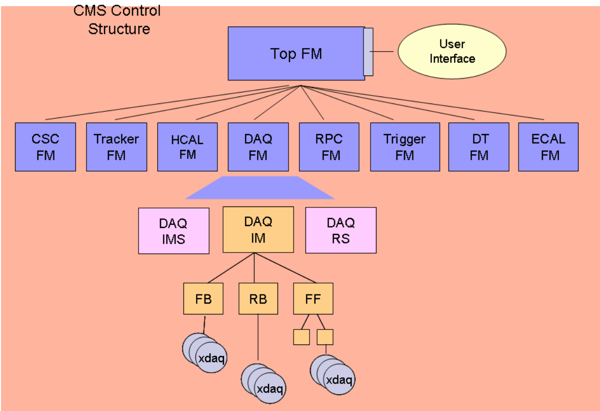

FIG. 2: *The CMS Run Control Structure*

The user can operate the system through a JSP or Labview GUI connected to the top FM, by choosing the sub-detectors participating to the global data taking and the related FEDs to read out. It can then issue commands to configure, start and stop the data acquisition. The top FM propagates the user commands to the sub-detectors FMs. It also propagates the Run Number, provided by a central "Run Information" database, and the configuration key to be used by the sub-detectors to configure their system according to the type of run. The top FM monitors the state of the controlled sub-detectors, the completion status of the commands being executed, and possible error conditions.

The sub-detector FMs are responsible for all the control operations specific to their DAQ and DCS software. They are also responsible for reporting their status, errors, and information messages to the top FM.

A sub-detector template FM is provided by the RCMS team to ease the integration and operation of all the sub-detectors. It defines a common sub-detector state machine model, the communication protocols for synchronous and asynchronous messaging, and the default parameters to exchange, including the run number.

The proposed finite state machine model allows to perform the usual DAQ operations, like initialize the system, configure it, start, stop, pause and resume the data acquisition, reset the system to a well-known state.

The DT FM has been developed according to the template and the recommendations to operate the CMS sub-detectors together. It is responsible of creating and destroying the DAQ and DCS applications, to issue the proper commands to the DAQ applications and monitor the state of the DT DCS. It also monitors the DAQ applications verifying the correct behavior of the system and reporting problems asynchronously to the top FM.

The same FM with the same state machine model is also used for local data taking, at CERN and in Legnaro, when only the DT system is involved in the data taking. The approach, that allows for an easier maintenance and evolution of the software, has been followed also for the development of the DAQ and DCS applications. The switch between local and global mode is a configuration parameter of the FM, forwarded to the DAQ applications. In local mode the FM is directly connected to a JSP GUI specifically developed for the DT system (fig. 3). The run number is managed using a local "Run Information" database, while configuration parameters are directly issued by the GUI. Status and errors are also directly reported to the GUI, with an AJAX asynchronous mechanism.

FIG. 3: *The DT Run Control GUI*

CONCLUSIONS

The RCMS DT Function Manager and its local GUI have been successfully used in local DAQs and during the MTCC. In preparation of the first run of CMS at the end of 2007, works are in progress to satisfy new CMS requirements, like a new state machine model that includes a test mode for the TTS system, and DT specific requirements.

---

[1] The XDAQ framework, http://xdaqwiki.cern.ch
[2] The RCMS framework, http://cmsdoc.cern.ch/cms/TRIDAS/RCMS
[3] The GRIDCC project, http://www.gridcc.org
[4] E.Frizziero et al., Instrument Element: a new Grid component that enables the control of remote instrumentation, CCGRIDW'06, Singapore 16-19 May 2006