# DISTRIBUTED CONTROL AND MONITORING OF HIGH-LEVEL TRIGGER PROCESSES ON THE LHCb ON-LINE FARM.

P.Vannerem, C.Gaspar, B.Jost, N.Neufeld, CERN, Geneva, Switzerland

*Abstract:*

The on-line data taking of the LHCb experiment at the future LHC collider will be controlled by a fully integrated and distributed Experiment Control System (ECS). The ECS will supervise both the detector operation (DCS) and the trigger and data acquisition (DAQ) activities of the experiment. These tasks require a large distributed information management system. The aim of this paper is to show how the control and monitoring of software processes such as trigger algorithms are integrated in the ECS of LHCb.

## ECS ARCHITECTURE

LHCb is an experiment dedicated to the study of CP asymmetries and rare decays in the b-hadron system at the future Large Hadron Collider (LHC) at CERN. The Experiment Control System will control and monitor both hardware devices and software processes in the DCS and DAQ in a homogeneous and uniform way [1]. In addition, the supervision of the experimental infrastructure and the communication with other control systems, such as the control system of the LHC accelerator, will be integrated.

The central concept of the ECS architecture is a strict hierarchical logical tree of nodes, where each node can be operated with a user interface (Fig. 1). At a given level in the tree each Control Unit is able to configure, monitor and control the nodes and the subtree below. This includes handling and propagation of alarms to supervision control levels and local decision capabilities. At the lowest level the nodes are so called Device Units, which drive and operate concrete devices, which can be hardware or software. An important design feature of the ECS architecture is the concept of partitioning, which allows stand-alone concurrent control and operation of subsystems.

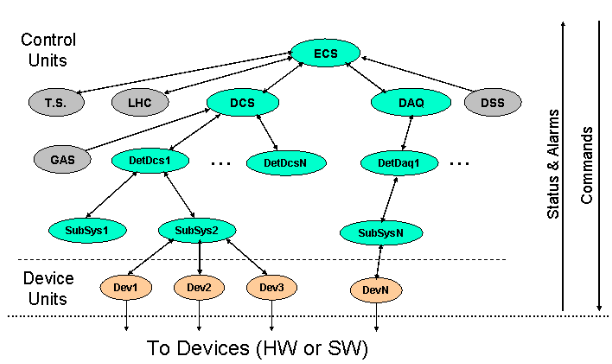


Fig. 1 The architecture of the Experiment Control System of LHCb.

Hence multiple operation modes of the nodes are foreseen. The ECS is being implemented using the commercial software package PVSS II [2], which complies with the industrial control standard SCADA. The ECS is based on the control framework that has been developed for the four LHC experiments [3], into which the Finite State Machine toolkit SMI++ [4] has been integrated to implement the desired hierarchical control tree structure.

## HIGH-LEVEL TRIGGER PROCESSES

### *The Event Filter Farm*

By applying increasingly complex criteria to increasingly more data from a particular event, several consecutive trigger levels decide whether the event contains enough interesting physics phenomena to be stored for off-line analysis. The LHCb specific requirement of on-line tagging of the quark-flavour of the event makes triggering a big technological challenge. The DAQ and trigger system consists of the following trigger stages: Level-0, Level-1 and the High-Level Trigger (HLT). The L0 Trigger is executed by custom electronic boards at the LHC beam-crossing rate of 40 MHz. The L1 trigger reduces the L0 accepted event rate of about 1 MHz to about 40 kHz by looking for predefined physics entities in a subset of the data coming from a few subdetectors. The HLT algorithms further reduce the event rate to about 200Hz and results in a recordable data stream of about 40 MB/s to permanent storage for offline analysis. While the Level-1 algorithm will act on a subset of the data the HLT algorithms have access to the complete data of an event.

Both Level-1 trigger and the HLT processes will run concurrently on one large dedicated farm of about 1800 general-purpose CPUs. The farm is partitioned in subfarms, each of which is controlled by a Sub-Farm Controller (SFC). The SFCs are connected to the front-end DAQ modules through a large switching network with a total bandwidth of about 7.1 GB/s. The event data flow and the load balancing between the CPUs within a subfarm are controlled by the SFCs, which continously supply CPUs with L1 trigger and HLT data. The SFCs and the farm CPUs will be supervised by ECS PCs that are connected to all farm equipment with a separate Ethernet Local Area Network connection that operates independently from the connection for event data flow. A schematic view of the event filter farm is given in Fig. 2.

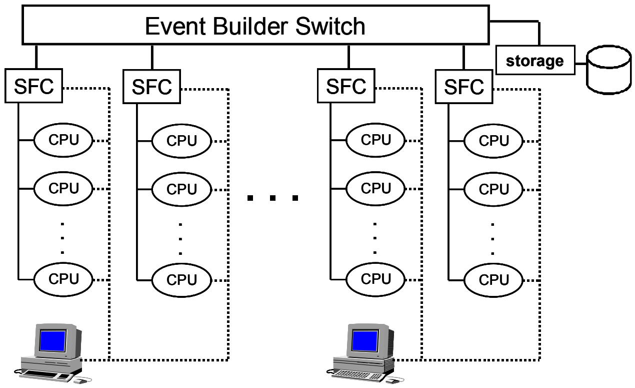

Fig. 2. Schematic view of the control of the Event Filter Farm.

### *Control of GAUDI trigger processes*

The trigger algorithms are implemented with GAUDI [5], an experiment-independent software framework for event data processing applications. GAUDI is a high-level object-oriented programming environment where algorithm objects process event data objects using services and transient data stores provided by the framework. A central idea for the event data analysis is that the same software framework is used both for physics algorithms running as triggers on the event filter farm and for offline physics analysis.

From the ECS point of view, the trigger processes written in GAUDI are devices, similar to electronic equipment, which have to be controlled and monitored on-line by the SCADA system. The processes have to be started and controlled by commands from the ECS, through its user interface or programmatically triggered by a node higher in the controls hierarchy. Starting of a process also implies that each CPU has to download the appropriate trigger software from its ECS PC.

In a GAUDI application, the processing sequence of the algorithms and the steering of services required by the algorithm objects is coordinated by an Application Manager object. Configuration of a trigger process therefore involves communication of specific values for the properties of the Application Manager and the algorithm objects to the trigger processes. These configuration parameters will be retrieved from an external configuration database interfaced to the ECS.

For each CPU device in the system there will be an ECS Device Unit that drives the GAUDI processes on the node. Implementing the hierarchical philosophy of Section 1, nodes are then supervised by a few subfarm Control Units, of which the number does not necessarily needs to be the same as the SFCs. The subfarm Control Units on their turn are controlled by a farm Control Unit, which allows the interaction with the complete farm for control purposes through a single user interface.

### *Monitoring of GAUDI trigger processes*

Status and error messages from the trigger processes have to be communicated to the ECS. In addition, while data-taking, quality checks of the trigger processes consist of verifying statistical data collected by the trigger processes, either by an automated process or by a human operator on the supervisory level. Hence, monitoring data such as counters, rates, histograms, status and error messages have to be communicated from the GAUDI trigger processes to the ECS. On the ECS side, statistical data collected by individual CPUs are combined into summary data. Counters and rates are to be presented in time graphs. The histograms are to be presented to the operator using either the user interface of the SCADA package or another viewer program that has been interfaced to the ECS. In order to trace problems, the contributions of processes to a summary histogram can be investigated in a top-down manner on demand.

## THE GAUCHO PACKAGE

The interfacing package GAUCHO (GAUdi Component Helping Online) implements control and monitoring functionality for GAUDI trigger processes running on the event filter farm. It uses the inter-process communication package DIM [6], which facilitates asynchronous, interrupt-driven data transfer across different platforms. DIM registers data services from processes wishing to publish data under a unique name, after which a client process can retrieve the service directly, and either poll for the data at fixed time-intervals or request to be updated whenever the data changes.

The scheme for the different constituting components in the GAUCHO package is illustrated in Fig. 3. A DimController is added to the GAUDI application, which steers the process by interacting with the Application Manager. In order to have well-defined states that can accept a predefined number of control commands (such as start, configure, pause, stop) a state machine is built in. The finite state machine of the processing algorithm is mirrored in the finite state machine of the ECS Device Unit for the CPU in the SCADA software. Configuration, i.e. the setting of the properties of the Application Manager or algorithm objects, is done by sending commands to a DimPropertyServer, which also acts as a server to read the current properties. For the monitoring part, a DimEngine provides point-to-point connections for counters, rates, messages and histograms from the trigger process to the SCADA system. On the ECS side a DIM API for the SCADA system is used, which allows the system to act as a server or as a client for specific data services. A control manager takes care of collecting the data from the GAUDI HLT processes and combines it into summary data, which is displayed in a monitoring user interface panel for the operator. An interactive control panel allows sending commands to the HLT process.

GAUCHO contains a generalized concept of on-line server histograms, which can provide several client processes with their parameters and bin contents. The package uses the AIDA interface classes [7] for histograms.

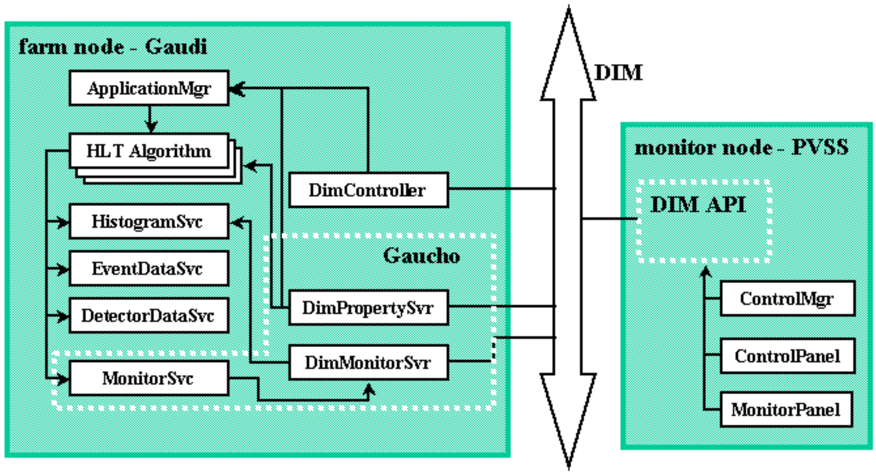


Fig 3. The constituents of GAUCHO, a component to control and monitor GAUDI applications with a SCADA system.

Therefore the online histogram concept is independent from GAUDI. It is easy to let other processes take part in the communication of histograms. This has been tested by implementing a histogram viewer using the ROOT framework [8] which displays histograms served either by a GAUDI trigger algorithm or by the SCADA system in real-time.

## SUMMARY

In this paper we presented the control and monitoring of trigger processes in the homogeneous and integrated experiment control system of LHCb. The trigger processes run on a commodity event filter farm and can be controlled and monitored using the SCADA system that is used to control the hardware devices. A package has been developed which interfaces GAUDI, the software framework for event data processing, to the SCADA system by using a layer of inter-process communication software. The package facilitates the steering of GAUDI trigger processes by commands and the extraction of monitoring data for supervision and data quality checking purposes in an online environment.

## REFERENCES

[1] P.R. Barbarosa Marinho et al.,"LHCb Online System, Technical Design Report", CERN/LHCC 2001-040.
[2] PVSS II, http://www.pvss.com.
[3] S. Schmeling et al., "Controls Framework for LHC Experiments", presented at the 13th IEEE-NPSS Real Time Conference, Montreal, Canada, May 18-23, 2003
[4] B. Franek and C. Gaspar, "SMI++ - An Object Oriented Framework for Designing Distributed Control Systems," in IEEE Trans. Nucl. Sci., Vol 45, Num 4, p.1946-1950, August 1998.
[5] G. Barrand et al., "GAUDI - A software architecture and framework for building LHCb data processing applications," Proceedings of CHEP 2000, Padova.
[6] C. Gaspar et al., "DIM, a Portable, Light Weight Package for Information Publishing, Data Transfer and Inter-process Communication", proceedings of CHEP 2000, Padova.
[7] A. Pfeiffer, "AIDA – Abstract Interfaces for Data Analysis," presented at the HepVis 2001 conference, Boston.
[8] R. Brun and F. Rademakers, "ROOT - An Object Oriented Data Analysis Framework," Proceedings of AIHENP'96 Workshop, Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86.