Contribution ID: **20** Type: **not specified**

Managing the CMS Online Software integrity through development and production cycles

Tuesday, 29 September 2015 09:45 (30 minutes)

The Data Acquisition system of the Compact Muon Solenoid experiment at CERN is a distributed system made of several different network technologies and computers to collect data from more than 600 custom detector Front-End Drivers. It assembles events at a rate of 100 kHz, transporting event data at an aggregate throughput of 100 GByte/s.
The architecture takes advantage of the latest developments in the computing industry. For data concentration, 10/40 Gbit Ethernet technologies are used while a 56Gbps Infiniband FDR CLOS network has been chosen for the event builder with a throughput of ˜4 Tbps.

The CMS Online Software (CMSOS) infrastructure is a complex product created specifically for the development of large distributed data acquisition systems as well as all application components to achieve the CMS data acquisition task. It is designed to benefit from different networking technologies, parallelism available on a processing platform such as multi-core or multi-processor systems. It provides platform independent framework that builds upon industrial standards, open protocols and libraries, tools for local and remote interprocess communication, configuration and control. It is designed according to the object-oriented model, and is implemented using the C++ programming language.
The processing infrastructure is made scalable by the ability to partition applications into smaller functional units that can be distributed over multiple processing units. The CMSOS is highly modular. It is organized in several interdependent packages that can be individually deployed for operations.

The CMSOS product development and the maintenance such as repairing problems and deficiencies, enhancement and new requirements occur in a well-established environment.
Achieving a high level of the CMSOS product quality during its lifetime is an iterative process that consists of collective activities like technical and administrative direction and surveillance, whose purpose is to maintain the integrity of the software products as these products evolve from specification through design, development and production cycles (e.g. releases, milestones, baselines, versions, etc.)
A cost-effective implementation of an application development process and technology environment is described to address the product evolution with the required quality over time.
The provided environment has shown benefits at accounting the complexity issues associated with the software itself, accommodating the problem of transient developers and overcoming different development cultural values.
The approach based on Software Configuration Management practices attempts to achieve the right balance between the support to a visible and traceable development and the overhead due to shuffling paperwork.

This work is based on the experience of the authors at CMS DAQ group and the experiences of others in different organizations who have been faced with similar concerns.

Availability

Tuesday

Will you need the training center (Workshops)?

No

Primary authors: SIMELEVICIUS, Dainius (Vilnius University (LT)); ORSINI, Luciano (CERN)

Co-author: ROBERTS, Penelope Amelia (Nottingham Trent University (GB))

Presenter: ORSINI, Luciano (CERN)