Available on CMS information server

CMS CR 2007/057

CMS Conference Report

4 October 2007

The CMS Monte Carlo Production System: Development and Design

D. Evans[1)a)], A. Fanfani[2)], C. Kavka[3)], F. van Lingen[4)], G. Eulisse[5)], W. Bacchi[2)], G. Codispoti[2)], D. Mason[1)], N. De Filippis[6)], J. M. Hernández[7)], P. Elmer[8)]

Abstract

The CMS production system has undergone a major architectural upgrade from its p redecessor, with the goal of reducing the operational manpower needed and prepar ing for the large scale production required by the CMS physics plan. The new pro duction system is a tiered architecture that facilitates robust and distributed production request processing and takes advantage of the multiple Grid and farm resources available to the CMS experiment.

Presented at *Hadron Collider Physics Symposium HCP 2007*, La Biodola, Isola d'Elba, Italy, 20 – 26 May 2007.

1) Fermi National Accelerator Laboratory, Batavia, IL, USA

2) Università degli Studi di Bologna and INFN Sezione di Bologna, Bologna, Italy

3) INFN Sezione di Trieste, Trieste, Italy

4) California Institute of Technology, Pasadena, CA, USA

5) Northeastern University, Boston, MA, USA

6) INFN Sezione di Bari, Bari, Italy

7) Centro de Investigaciones Energeticas, Medioambientales y Tecnológicas, Madrid, Spain

8) Princeton University, Princeton, NJ, USA

The CMS Monte Carlo Production System: Development and Design

D. Evans [a] *, A. Fanfani [b], C. Kavka [c], F. van Lingen [d], G. Eulisse [e], W. Bacchi [b], G. Codispoti [b], D. Mason [a], N. De Filippis [f], J. M. Hernández [g] and P. Elmer [h]

[a]Fermi National Accelerator Laboratory, Batavia, IL, USA

[b]Università degli Studi di Bologna and INFN Sezione di Bologna, Bologna, Italy

[c]INFN Sezione di Trieste, Trieste, Italy

[d]California Institute of Technology, Pasadena, CA, USA

[e]Northeastern University, Boston, MA, USA

[f]INFN Sezione di Bari, Bari, Italy

[g]Centro de Investigaciones Energeticas, Medioambientales y Tecnológicas, Madrid, España

[h]Princeton University, Princeton, NJ, USA

The CMS production system has undergone a major architectural upgrade from its predecessor, with the goal of reducing the operational manpower needed and preparing for the large scale production required by the CMS physics plan. The new production system is a tiered architecture that facilitates robust and distributed production request processing and takes advantage of the multiple Grid and farm resources available to the CMS experiment.

1. Introduction

The CMS experiment at CERN relies on a distributed computing model for the Monte Carlo production. This paper describes the main architectural components of the production system.

2. The MC production system architecture

The architecture of the Monte Carlo (MC) production system [1] consists of three components (figure 1): the Request system (*ProdRequest*), which acts as a frontend application for (user) production request submissions into the production system; the Production Manager (*ProdManager*), that manages these user requests, performing accounting and allocating work to a collection of Production Agents (*ProdAgents*).

The agents ask for work when resources they manage are available and manage submissions, posible errors and resubmissions while performing the required local cataloging operations.

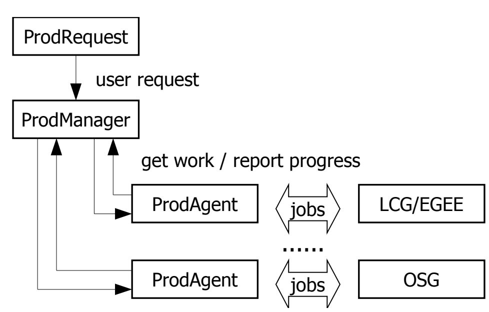

Figure 1. The structure of the MC production system.

The ProdAgents themselves are defined internally in terms of autonomous components that communicate via an asynchronous and persistent

*Corresponding author. Email address: evansde@fnal.gov

message system. Components subscribe to messages of interest, get messages addressed to them waiting only if there are no messages available, and publish their messages without waiting for data to be transferred. Delayed and queued message functionality enables the ProdAgents to adequately deal with third party component interaction, like for example the CMS cataloguing system and the file transfer systems [1].

At the core of the system there is a database that registers the persistent status information like messages, job states and local dedicated data management services (DBS).

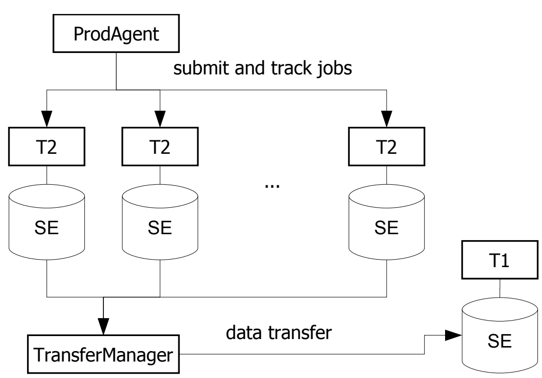

Figure 2. The job workflow model.

The Monte Carlo production job workflow is shown in figure 2. The ProdAgents create processing jobs, which are submitted to the Tier2 centers [1]. These jobs produce output files that are stored in the local storage elements (SE) and registered in the local DBS catalogs of the Tier2 where the job was running. When enough production data is available at a site, merge jobs are automatically submitted by the ProdAgent. These jobs merge previously produced data and remove unmerged files. The ProdAgents take care of error handling and resubmission of both production and merge jobs, automatically triggering data transfers to the target storage system at the Tier1 centers, and performing registration in global catalogs.

Various complementary monitoring tools provide end-to-end monitoring of the system to track down potential problems. A local monitoring tool provides information about the status of all components (access to log files, etc), while a global monitoring tool provides summaries of the status of the productions performed by all ProdAgents that are running in the production system.

As the framework is based on a set of independent components and structured around an asynchronous persistent messaging and triggering infrastructure, other CMS related software projects like CRABServer [2] (supporting user analysis) and Tier0Agent (real time handling of detector generated data) have based their applications on this framework, leading to more convergence in the CMS software code base and better maintainable code.

3. Conclusions

The MC production system for CMS aims at providing a high level of automation with support for multiple Grids. The system is designed to scale to multiple request, manager and agent components thereby minimizing single points of failure. Its autonomous component based design allows many developers independently contribute to the development. The new production system has been successfully used to generate many hundreds of millions of events [3], making heavy use of LCG and OSG resources.

REFERENCES

1. *CMS Computing TDR*, CERN-LHCC-2005-023, 20 June 2005.
2. D. Spiga et al. *CRAB, CMS Remote Analysis Builder*. Proceedings of the Hadron Collider Physics Symposium HCP 2007, Elsevier, 2007.
3. A. Pompili et al. *CMS Monte Carlo Production Operations in a Distributed Computing Environment*. Proceedings of the Hadron Collider Physics Symposium HCP 2007, Elsevier, 2007.