Journal of Physics: Conference Series **331** (2011) 062032 doi:10.1088/1742-6596/331/6/062032

CMS Distributed Computing Integration in the LHC sustained operations era

C.Grandi[1]; B. Bockelman[2]; D. Bonacorsi[1]; I. Fisk[3]; I. González Caballero[4]; F. Farina[5]; J.M. Hernández[6]; S. Padhi[7]; S. Sarkar[8, 9]; A. Sciabà[9]; I.Sfiligoi[7]; F. Spiga[5]; M. Úbeda García[9]; D.C. Van Der Ster[9]; M. Zvada[10]

[1]INFN-Bologna & U.Bologna; [2]U.Nebraska, [3]FNAL; [4]U.Oviedo; [5]INFN-Milano Bicocca; [6]CIEMAT; [7]UCSD; [8]INFN-Pisa; [9]CERN; [10]KIT

Claudio.Grandi@cern.ch

Abstract. After many years of preparation the CMS computing system has reached a situation where stability in operations limits the possibility to introduce innovative features. Nevertheless it is the same need of stability and smooth operations that requires the introduction of features that were considered not strategic in the previous phases. Examples are: adequate authorization to control and prioritize the access to storage and computing resources; improved monitoring to investigate problems and identify bottlenecks on the infrastructure; increased automation to reduce the manpower needed for operations; effective process to deploy in production new releases of the software tools. We present the work of the CMS Distributed Computing Integration Activity that is responsible for providing a liaison between the CMS distributed computing infrastructure and the software providers, both internal and external to CMS. In particular we describe the introduction of new middleware features during the last 18 months as well as the requirements to Grid and Cloud software developers for the future.

1. Introduction

The role of the CMS *Integration of Distributed Facilities and Services* task is to act as the liaison between the CMS Computing program and software providers for what concerns the distributed infrastructure. Software providers are both internal to CMS (the CMS Offline program) and external (mainly middleware developers from the WLCG, EGEE/EMI, OSG and VDT projects). After many years of preparation the CMS computing system has reached a situation where stability in operations limits the possibility to introduce innovative features. Nevertheless it is the same need of stability and smooth operations that requires the introduction of features that were considered not strategic in the previous phases.

We describe here some recent activities in the evolution of CMS Computing that we think may have impact on current and future activities of middleware providers: in section 2 we describe the modifications foreseen in the CMS storage model; in section 3 we describe the recent activities in the field of job management; in section 4 we describe the authorization mechanisms used by CMS on the distributed infrastructure and the issues encountered; in section 5 we describe recent developments in the field of monitoring; in section 6 we present our conclusions.

2. Storage Model

CMS, as other LHC experiments, is still using a hierarchical model [1][2] inherited from MONARC [3]. The cost and performance of the network changed significantly since the MONARC studies were done (see Figure 1): sites are indeed "closer" than initially foreseen. On the other hand the tape technology and costs did not evolve as hoped and transparent access to data on tape sometimes causes inefficiencies.

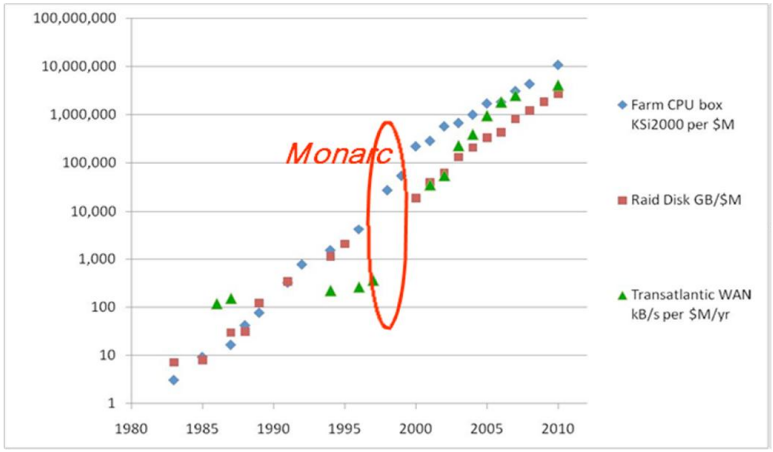

Figure 1: Cost of CPU, disk and network in the period 1980-2010 (arbitrary units) [4]

Data placement is largely managed by humans and choices (location, number of replicas, etc…) are based on previsions on data popularity that are very difficult to make. For example in 2010 over 3500 CMS datasets were subscribed out to Tier-2's central space by Analysis Operations but only a small number are accessed frequently.

There is more to learn about what data needs to be heavily replicated and what will be used once and never again. For these reasons CMS, as well as other LHC experiments, started a review of the storage model to address these issues.

2.1. Partition Tier-1 tape and disk resources

CMS is currently using a unique T1D0 storage element at Tier-1s with a huge buffer and access to the tape is done transparently. CMS is thinking to split tape and disk storage at Tier-1s. The T1D0 system with have only a small buffer while the T0D1 system that will serve data to jobs running on the computing farm will have most of the Tier-1 disk capacity. Data will have to be explicitly replicated to the T0D1 system to process them. This would require modifications in the data distribution and data access systems.

2.2. Dynamic data placement

CMS is interested in investigating technologies that could take decisions on replication and cancellation of data based on actual use as reported by the workload management system (via the CMS Dashboard [5]). Caching mechanisms already known to IT could be reused for this purpose. This could reduce the number of copies that need to be manually pre-placed.

2.3. Remote access to data

CMS is currently sending jobs to sites hosting the data with no exceptions. Relaxing this requirement could reduce the number of copies that need to be manually pre-placed. It could also help in case access to a specific portion of RAW data is needed when processing RECO or AOD data.

For this reason CMS proposed to the WLCG community a demonstrator to investigate the possibility to use xrootd [6] to access data locally and remotely (with the usual X509/GSI/VOMS authorization).

When a process tries to open a file on xrootd a redirector is first contacted. The redirector has no data, but locates a file on a disk server and the client is (seamlessly) redirected to the disk server. The process (shown in figure 2) is transparent and remote access may happen in case a file is not found locally.

It should be noted that in the CMS test the site storage is not changed. A proxy is installed at the site that exports data in the xrootd protocol and federates itself with the other xrootd instances (one global redirector has been used in the test). The namespace is the logical CMS one. And the translation to the site namespace is done at the site itself via the Trivial File Catalogue.

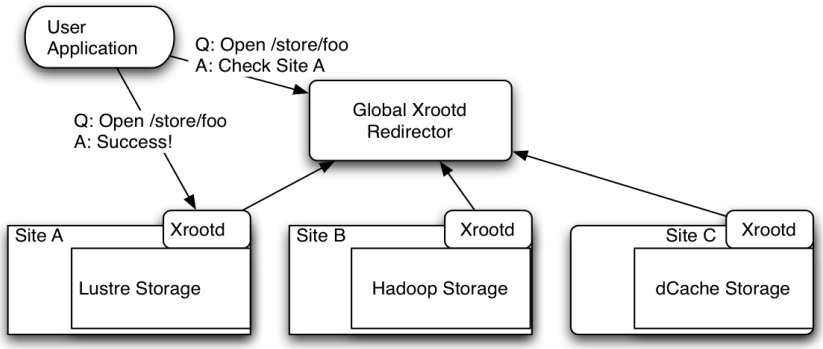

Figure 2: Global redirection mechanism in data access via Xrootd

Beside Nebraska (coordinating the test) the other participating CMS Tier-2 sites are Bari, Caltech, Florida, FNAL, PSI, Purdue, UCR, UCSD and Wisconsin.

Another important use case for remote access is the analysis of data on Tier-3 sites without the need to produce official replicas via PhEDEx [7].

Since a similar behavior may be achieved using other protocols, CMS is also interested in evaluating other solutions based on e.g. NFS 4.1 (POSIX interface) or WebDAV (RFC 4918 proposed standard).

2.4. Software installation

CMS will investigate CernVM File System (CernVM-FS) [8] to distribute CMS software to sites. CMS will start testing on Tier-3s.

3. Job Management

Jobs are currently submitted through the gLite-WMS [9] and the glidein-WMS [10] (using the pilot jobs approach). Job submission frameworks (Production Agent and CRAB) have an abstraction layer that allows using different plug-ins for distributed and local submission (BossLite). The gLite-WMS and glidein-WMS are both heavily used for analysis and production activities. Furthermore CMS is trying to validate the use of CREAM [11] in view of the end of support to the LCG-CE.

3.1. gLite-WMS

BossLite has been using the python API that provided the needed details in the commands output for many years. Unfortunately the use of the API caused incompatibilities between the python versions used by the CMS and grid software. Recently BossLite switched to the new gLite-WMS CLI with json output that offers the same level of detail provided by the API. This allowed isolating the gLite environment with respect to the CMS environment. CMS now needs a similar CLI with json output for the glite-wms-job-logging-info command.

The main problem reported by gLite-WMS users is that there is a long tail in the distribution of the job termination time. This is largely due to the fact that the match-making process based on the information published by the sites in the Information System is often non-optimal. A mechanism that cures this problem would be welcome.

3.2. glidein-WMS

The main problem related to the use of the glidein-WMS is the still limited diffusion of glexec on the Worker Nodes that is needed when submitting analysis jobs. Because of that CMS is currently not changing the identity on the WN in analysis jobs.

3.3. CREAM-CE

The CREAM CE is supposed to fix a number of drawbacks of the LCG-CE, including the fact that it is not being supported on recent versions of the operating systems. With the recent fixes in CREAM and in ICE CMS can use CREAM-CEs via the gLite-WMS. Instead the use via the glidein-WMS is not enabled yet because of issues in the configuration of the authorization layer with pilot jobs.

4. Authorization

CMS authorization is based on VOMS [12] attributes i.e. on VOMS groups and roles. Attributes are assigned by VO Manager on request by group coordinators. The main attributes in use by CMS are:

/cms/Role=lcgadmin	SAM tests and software installation
/cms/Role=production	MC production
/cms/Role=t1production	Processing at Tier-1s
/cms/Role=t1access	Analysis at Tier-1s
/cms/Role=priorityuser	Privileged analysis users
/cms/Role=NULL	All standard analysis activities

4.1. Access to CPU resources

At the Tier-1s there are predefined shares for *production*, *t1production* and *t1access* roles. No access for other analysis users. Membership to those groups is controlled by the CMS VO Manager on indication by the Data Operation team leaders and Tier-1 coordinators (for the *t1access* role)

At the Tier-2s there are predefined shares for *production*, *priorityuser* and normal CMS users. Membership to the *priorityuser* role is controlled by the CMS VO Manager on indication by the physics group managers.

Figure 3 shows the CPU allocations at Tier-1 and Tier-2 centres.

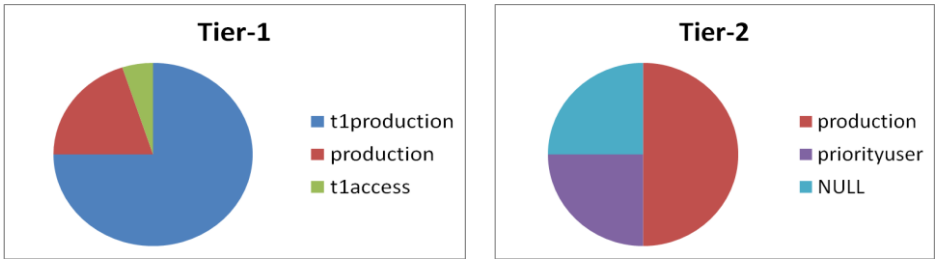

Figure 3: Shares at the Tier-1 and Tier-2 centres for the different roles

4.2. Access to storage resources

Read access is always allowed to all CMS users. Write access to the Tier-1 storage is allowed to production roles only. Write access on the Tier-2 storage depends on the namespace. Official data areas are writeable only by production roles. Group areas are in general writeable by users with the *priorityuser* role, regardless their group. User areas in principle should be writeable only by the users directly supported by the site. This would require maintaining explicit lists of DNs supported at each site. Actually most sites opened the user area to all CMS users.

Group and user quotas are controlled by the site managers that may contact users and group managers in case of problems.

At each site a selected number of users may have write privileges over portions (or all) of the storage and the data (typically the user running the local PhEDEx agents which need write access to the official data areas to place data transferred to the site).

Recently CMS tested a solution where the */cms/muon/Role=production* role can also write on the group area */store/group/muon*. This would provide more fine grained control over the authorization and would delegate to the group managers part of the load.

4.3. Authorization in transfers

Authorization in transfers is controlled by the PhEDEx authorization layer where an explicit request approval by site data manager is foreseen.

4.4. Technical problems

The current authorization mechanism relies on the mapping of the primary attribute to a local unix uid/gid; the unix authorization is then used for local scheduling and for controlling the access to the storage. This is a limitation since different attributes may be needed for the two scopes. For instance the group *production* role would be needed in order to write to the group but at the same time the *priorityuser* role is needed for the prioritization of jobs on the batch systems. A possibility is to get rid of the mapping to uid/gid for scheduling and new middleware (e.g. ARGUS [13]) may help in that.

Similarly the coexistence of pilot and traditional job submission may give problems because pilot jobs require special privileges (glexec execution) that are granted only to */cms/Role=pilot*. This means that this role must be used as primary attribute and there is not the possibility to use the *priorityuser* on the compute element.

Since remote file access will probably be possible in future CMS may need special authorization to protect the network and the remote servers.

5. Monitoring

Optimizing the work and data flows in a complex experiment like CMS requires to correctly and closely monitor the various aspects involved. To accomplish that task there is a need to continuously develop and integrate new tools that correlate all the possible systems and sources of information. The following are examples of new monitoring tools being integrated in CMS.

5.1. CMS Tier-1 Local Job Monitoring

New tools for monitoring the local batch queues at Tier-1s have been developed and are being deployed. This has been done in order to early spot problems on local Tier-1 batch systems not visible through grid monitoring tools and provide a central location for this information. This will allow Facilities, Data and Analysis Operation teams to take fast and clever decisions and it will offer monitoring information through independent channels complementing other existing monitoring tools.

CMS developed dedicated information providers that query the batch system and publish information to an XML file with a well defined format that describes all the quantities CMS needs to follow. A DB backend and a presentation layer have been developed as well. The information downloaded from the Tier-1 sites is collected at a central place and the plots are added to the CMS Dashboard [5] and integrated in the monitoring shift patterns, possibly via the HappyFace [14] interface.

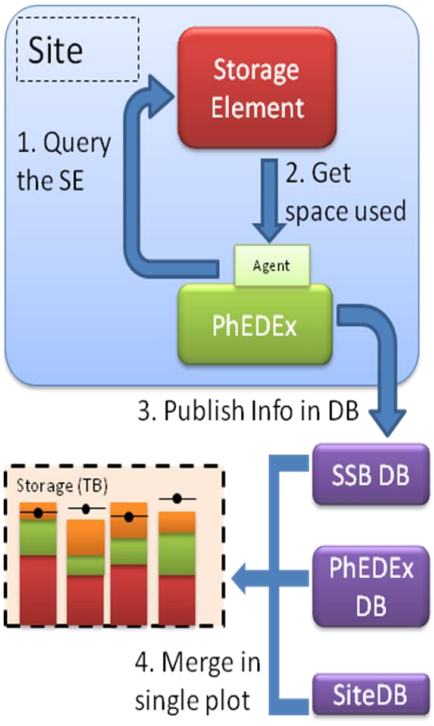

Figure 4: Group and storage space monitoring

5.2. Monitoring user and group storage space

In absence of common tools for setting quotas and provide accounting on storage systems, CMS developed tools to monitor the space used on Storage Elements by the data directly produced by users, physics and detector groups. The tools allow comparing the total space with the pledges; identify cases of overuse and of saturated sites and eventually balance

the CMS storage resources.

This is being achieved, as shown in figure 4, by adding a new PhEDEx agent using the plug-in architecture that will be implemented in the next release that finds out the amount of disk used in the user and group namespaces. The information will be published to a central Database, merged with other information coming from other sources and compared with the pledges for each site. The plots are finally added to the CMS Dashboard and integrated in the monitoring shift patterns.

5.3. HammerCloud

HammerCloud (HC) [15] is a distributed analysis testing system built around Ganga. It was developed initially by ATLAS and is now being extended to support CMS and LHCb.

HC may be used for frequent functional tests to validate the services and to perform on-demand stress tests to commission new sites or give benchmarks for site comparisons.

In CMS, HC will replace the Job Robot for site functional tests. Furthermore it will run data I/O efficiency tests on Grid sites allowing finding the best site configuration and data access mechanism for a given site and will stress test beta versions of CMSSW on a distributed environment.

6. Conclusions

In this paper we described recent activities concerning the evolution of CMS Distributed Computing that potentially will make the infrastructure more scalable and operable. Some of these activities may have impact on current and future activities of middleware providers.

References

[1] C.Grandi, D.Stickland, L.Taylor et al. “The CMS Computing Model” CERN-LHCC-2004-035/G-083 (2004)
[2] The CMS Collaboration, “CMS Computing Technical Design Report”, CERN-LHCC-2005-023 (2005)
[3] M.Aderholz et al., “Models of Networked Analysis at Regional Centres for LHC xperiments (MONARC) - Phase 2 Report,” CERN/LCB 2000-001 (2000)
[4] R.Mount, Jamboree on Evolution of WLCG Data & Storage Management, Amsterdam (2010)
[5] J.Andreeva et al. "Dashboard for the LHC experiments", Conference on Computing in High Energy and Nuclear Physics (CHEP07), Victoria, Canada (2007)
[6] Xrootd: http://xrootd.slac.stanford.edu/
[7] R.Egeland, T.Wildish, S.Metson, "Data transfer infrastructure for CMS data taking", Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2008), Erice, Italy, PoS ACAT08:033 (2008)
[8] J.Blomer, T.Fuhrmann "A Fully Decentralized File System Cache for the CernVM-FS", 19th IEEE International Conference on Computer Communications and Networks (ICCCN) 2010; doi: 10.1109/ICCCN.2010.5560054
[9] M.Cecchi et al. “The gLite Workload Management System”, Conference on Advances in Grid and Pervasive Computing,, ISBN 978-3-642-01670-7 (2009)
[10] I.Sfiligoi et al. “The Pilot Way to Grid Resources Using glideinWMS", WRI World Congress on Computer Science and Information Engineering, ISBN: 978-0-7695-3507-4 (2009)
[11] C.Aiftimiei et al., "Design and Implementation of the gLite CREAM Job Management Service", Future Generation Computer Systems, Volume 26, Issue 4 (2010)
[12] A.Ceccanti et al., “Virtual Organization Management Across Middleware Boundaries”, Conference on e-Science and Grid Computing (e-Science 2007), Bangalore, India (2007)
[13] ARGUS: https://twiki.cern.ch/twiki/bin/view/EGEE/AuthorizationFramework
[14] HappyFace: https://ekptrac.physik.uni-karlsruhe.de/trac/HappyFace/
[15] D.Van Der Ster et al. "HammerCloud: A Stress Testing System for Distributed Analysis", Presented at this conference