Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Managing the CMS Data and Monte Carlo Processing during LHC Run 2

To cite this article: C Wissing and CMS Collaboration 2017 *J. Phys.: Conf. Ser.* **898** 052012

IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052012 doi:10.1088/1742-6596/898/5/052012

Managing the CMS Data and Monte Carlo Processing during LHC Run 2

C Wissing[1] for the CMS Collaboration
[10]DESY, Notkestraße 85, 22607 Hamburg, Germany
E-mail: christoph.wissing@desy.de

Abstract. In order to cope with the challenges expected during the LHC Run 2 CMS put in a number of enhancements into the main software packages and the tools used for centrally managed processing. In the presentation we will highlight these improvements that allow CMS to deal with the increased trigger output rate, the increased pileup and the evolution in computing technology. The overall system aims at high flexibility, improved operational flexibility and largely automated procedures. The tight coupling of workflow classes to types of sites has been drastically relaxed. Reliable and high-performing networking between most of the computing sites and the successful deployment of a data-federation allow the execution of workflows using remote data access. That required the development of a largely automatized system to assign workflows and to handle necessary pre-staging of data. Another step towards flexibility has been the introduction of one large global HTCondor Pool for all types of processing workflows and analysis jobs. Besides classical Grid resources also some opportunistic resources as well as Cloud resources have been integrated into that Pool, which gives reach to more than 200k CPU cores.

IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052012 doi:10.1088/1742-6596/898/5/052012

1. Introduction

The CMS experiment is one of the two multi-purpose experiments located at the LHC storage ring at CERN. CMS observes proton-proton collision provided by the LHC. For the running period that started in 2015, typically referred to with "Run 2", the center of mass energy of the provided collision could be increased from 8 GeV to 13 GeV. Improvements of the LHC accelerator complex allowed to increase the beam currents and final focusing at the collision points. That leads to an increase in luminosity and the number of pile-up events, which is the number of proton-proton scatterings during one bunch crossing. Increased pile-up causes more complex events that are more demanding regarding the CPU requirements for event reconstruction.

1.1. CMS Computing Infrastructure

The computing infrastructure utilized by the CMS experiment is organized in a distributed approach and builds on top of the World-wide LHC Computing Grid (WLCG) [1] structure. The tiered structure of sites has one Tier-0 at CERN, where the experiment is located. The Tier-0 resources are primarily used to process the data that come from the experiment and to archive the raw detector data on tape. The CMS experiment is supported by 7 Tier-1 centers, one located in the US and six European centers located in Germany, France, Italy, Russia, Spain and the United Kingdom. Another roughly 40 Tier-2 centers around the world also support CMS. All sites provide CPU and disk capacity. Tier-1 sites and the Tier-0 also provide long term archival of data on tapes. Beyond these "classical" Grid resources experiments have started to use other resources including the usage of the high level trigger farms (HLT) for offline processing, usage of academic and commercial clouds and utilization of time based allocations on high-performance computing systems (HPC).

1.2. LHC and CMS Performance

While the first year of Run 2 in 2015 saw just a moderate integrated data volume, the performance in 2016 exceeded the planning. The LHC quickly reached the expected instantaneous luminosity. The re-filling of the LHC storage ring happened even more efficiently than planned and the originally forecast beam delivery time was already reached a few weeks before the scheduled end of the 2016 running. Also the CMS detector was in excellent shape and could very efficiently record the provided proton-proton collisions. Since neither the trigger nor the data acquisition system was limiting the data collection, CMS could record and promptly reconstruct events with an average rate of 1.2 kHz [2] exceeding the originally planned target of 1.0 kHz. The longer than planned beam delivery time, the higher data logging rate and the complexity of the recorded events challenged the computing infrastructure, both regarding CPU capacity and storage capacity.

2. Increasing Flexibility of Resource Usage

CMS workflows can be coarsely grouped into four classes. Production of Monte Carlo events is characterized by little I/O demands and high CPU needs. The reconstruction of MC events is very I/O intense particularly when events with a number of pile-up interactions need to be simulated. For each event to be simulated N pile-up events need to be read in. Data (re-)reconstruction has moderate I/O demands. Analysis jobs from individual users can have a broad spectrum of characteristics.

In Run 1 CMS operated the distributed computing infrastructure with a rather close coupling of workflow classes to resource types. This is illustrated in Figure 1. Since not all workflow classes are submitted with a constant fraction the resource utilization was often asymmetric.

In order to enhance the flexibility a number of developments were made in CMS before Run 2:

IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052012 doi:10.1088/1742-6596/898/5/052012

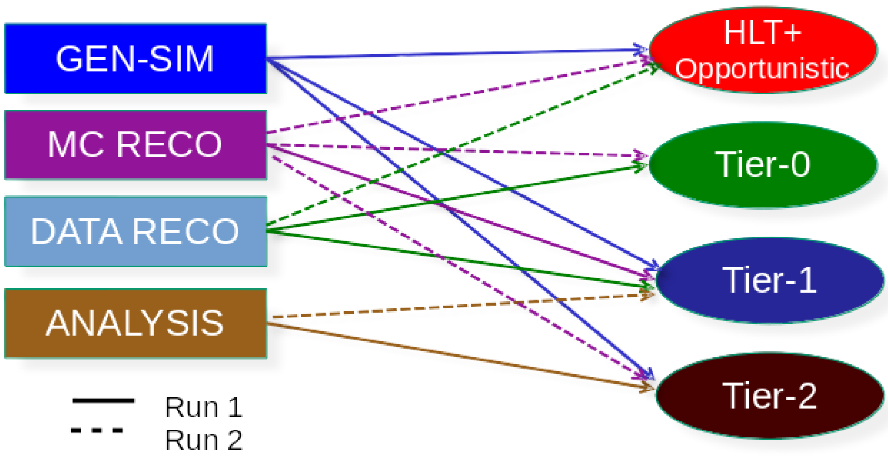

Figure 1. Types of workflows and resources where they could be assigned to in Run 1 (solid line) and new options in Run 2 (dashed line).

- Pooling of CPU resources: All CPU resources are reachable through a single Global Pool and execution priorities are controlled in one central place.
- Pooling of storage resources: The majority of the disk space is joined into one big space that is managed dynamically by a management system.
- Remote data access: In order to allow job assignment in the most flexible way, jobs need the capability to access data remotely without waiting for any data placement at the execution site.
- Access to new types of resources: Any additional resource that can be integrated into the computing infrastructure gives more flexibility to the research program. It should be possible to profit also from resources that are available only for limited amount of time.

In the following sections some examples are described that show how these developments are used in recent larger processing campaigns of CMS.

3. Multi-thread enabled Global Pool

3.1. Multi-threaded CMS Applications

The clock speed of CPUs could not be further increased over the last couple of years. Nevertheless processors have become more and more powerful by increasing the number of CPUs cores. In order to exploit the potential of modern CPU architectures the software needs to be adopted to make use of the higher number of cores. Therefore the development of thread-safe code has become an important aspect in software development. Another advantage of multi-threaded applications is their lower memory footprint compared to multiple executions of single-threaded applications.

The CMS software framework was transitioned into a multi-threaded application[3]. A lot of effort has been spent into providing thread-safe modules, which is crucial to achieve efficient program execution with a high number of threads. Since 2015 already the CMS Prompt Reconstruction is executed in 4-thread mode[4]. Re-reconstructions are also run in 4-thread mode. Figure 2 shows the CPU efficiency of re-reconstruction campaign, which runs typically with an efficiency of 80%.

CMS is in progress of transitioning more and more workflow types to multi-threaded applications. In 2016 the DIGI-RECO workflow got commissioned for multi-threading. This workflow requires thread-safe modules for the digitization in addition to the reconstruction.

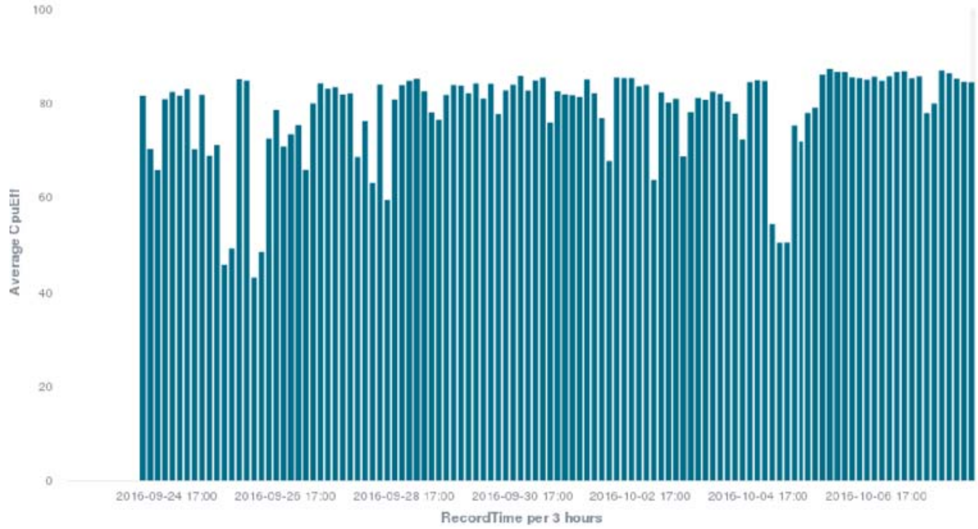

Figure 2. CPU efficiency for a recent RE-RECO campaign using a 4-threaded reconstruction program.

The effort to achieve this has been considerable, because the implementations are specific for most sub-detectors. There is an effort ongoing to enable also the Monte Carlo production in a multi-threaded approach.

3.2. Transition to multi-core Pilots

In order to be able to execute multi-threaded applications CPU resources need to be allocated in a suitable way. Like all other LHC experiments CMS employs a pilot-job approach. The CMS solution builds on HTCondor technology with GlideinWMS [5]. Pilot-jobs are sent to the sites and once they have started on the compute nodes the resources get jointed in one large Global HTCondor Pool. Having all resources combined in one large single pool enables CMS to control the priorities of all payload-jobs that are submitted through the CMS submission infrastructure.

For multi-threaded applications the pilots need to allocate slots that expand over several CPU cores. To ease the local scheduling of multi-core jobs it has been agreed in WLCG that the default multi-core pilot allocates 8 CPU-cores. In collaboration with the Grid sites that support CMS basically all Tier-1 and Tier-2 resources got enabled to receive multi-core pilots. Figure 3 shows an example of utilized cores. The Global Pool could be operated stably at a utilization of 130,000 cores. Over 90% of the resources are allocated via multi-core pilots.

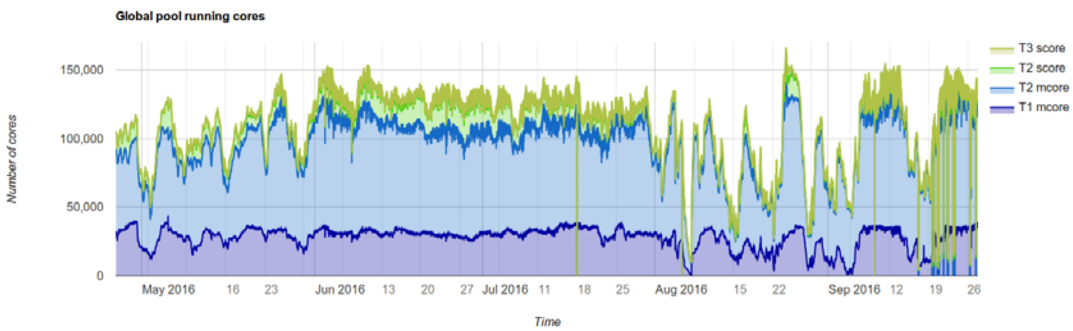

Figure 3. Utilization of the CMS Global Pool.

If a compute resource supports multi-core pilots CMS sends only those to the site. The full capacity of cores gets in-cooperated into the Global Pool and payloads are executed following a "dynamic partitioning" approach. That means that a multi-core pilot can run a mixture of payloads inside, e.g. a 8-core pilot can run one 4-threaded job and 4 single-threaded jobs [6].

4. Tools for Workflow Handling

The central CMS production system employs a request manager and WMAgents [7] to store requests and drive the corresponding jobs. The request manager got recently completely renewed and the new version has been put into production in 2016. The development activities for the WMAgent were primarily targeted at scalability and improved reliability.

During Run 1 the assignment of workflows was largely depending on operator effort. The much more flexible and complex assignment of workflows in Run 2 required the development of a tool that steers the workflow assignments. The Unified tool [8] has been developed with the aim to largely automatize the handling of all types of workflows. Unified takes care of the data placement prior to the start of the workflow. Depending on the workflow various configurations are passed to the WMAgent, e.g. the enabling of remote data access. Selection of custodial tape subscriptions as well as announcing the finished workflows is a functionality of Unified. The system handles several thousand requests per week.

5. Improvements in Data Handling and Access

A very large fraction of all disk resources that are provided to CMS have been joined to one large pool. The Dynamic Data Management (DDM) system handles replication and removal of data based on the popularity and a set of rules. The storage elements at the various sites are members of a large data federation. Clients can ask for unique logical file names (LFN) and get re-directed to a storage instance, that is hosting the requested data, where the file gets opened for remote read. The dynamic management and the possibility to access all files that are on disk somewhere largely enhances the flexibility where to store data and where to run jobs.

5.1. Dynamic Data Management

All Tier-1 and Tier-2 sites and the Tier-0 site contribute a large fraction of their disk capacity to the DDM system [9]. In 2016 the size of the disk space pool that was managed by DDM has been about 54 PB. The majority of that space is used to host data and MC files for analysis by CMS physicists. Another fraction hosts files that are input to centrally managed production and processing campaigns.

The DDM takes care of creating additional replicas and their removal. The system mainly considers the following metrics.

- Data popularity: DDM has access to information what datasets are requested as input of analysis jobs. For frequently accessed datasets additional replicas are created. Replicas of less popular datasets are reduced.
- Space constrains: DDM fills sites up to a threshold of the available space. Once this is reached removal of datasets starts.
- Policies: DDM can be configured by an arbitrarily complex set of rules that allow the implementation of almost any condition. Rules can be spelled out for certain data tiers, they can contain name pattern or consider a lifetime.
- Explicit locks: This method is used for the handling of input data that is required by central production. Needed datasets get registered in a central lock list and DDM will not touch those until the lock is released again.

IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052012 doi:10.1088/1742-6596/898/5/052012

5.2. Remote Data Access

In the original computing models of the LHC experiments jobs where always supposed to be executed on a CPU close to the data and any I/O operations was expected to go via the LAN only. The experiences showed however that the network bandwidth and the reliability of the networks developed much better than originally assumed. That development let the experiments explore the usage of workflows that read in their data over the network from remote. CMS put quite some effort into the optimization of the I/O layer of their applications such that they can cope efficiently with latency caused by remote reads.

In order to easy the access to any data on disk CMS has setup the "Any data Anywhere Anytime" (AAA) storage federation[10] based on the XRootD technology[11]. All storage elements subscribe their inventory to a regional re-director. At the moment there are two regions, one US region and a EU region that also includes Asia. The regional re-directors subscribe to a global re-director hosted at CERN.

The CMS software applications are usually configured to attempt a local file open first and fall back to a remote open via the AAA federation, if the local attempt has failed. In the recent past an increasing number of workflows has been sent intentionally to sites, where input files are not available and the successful completion of the jobs would entirely depend on file reads from remote locations. The first class of workflows that run in that mode was Monte Carlo generation, which has only little data to read in to keep the CPUs fully busy. In 2016 a bigger re-reconstruction campaign was configured to allow remote data read for a fraction of the jobs. The success rate of those jobs is shown in Figure 4. Successful jobs with input data being local (red) and successful jobs with remote input (orange) are displayed together with failed jobs reading locally (dark blue) and failed jobs reading from remote (light blue).

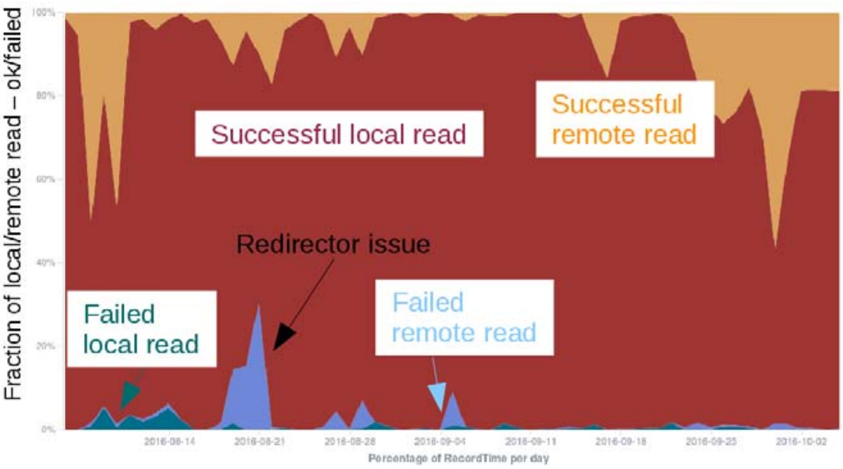

Figure 4. Fraction of successful jobs comparing jobs with local data access to those with remote data read.

The overall fraction of jobs reading form remote was sustained over a longer period at about 20% and reached some peaks around 50%. The failure rate was comparable for both types of jobs with one exception, where a technical issue with the AAA federation caused a number bad remote read attempts. The resilience in the re-director infrastructure and the robustness of the application is subject of ongoing developments.

6. Resources beyond classical Grid Sites

The good LHC performance leads to an increased demand for computing resources. Additional capacity enhances the flexibility of the physics program that can be addressed by the experiment.

IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052012 doi:10.1088/1742-6596/898/5/052012

Therefore all LHC experiments have ongoing efforts to explore the usage of resources beyond the established Grid sites. The high level trigger (HLT) farms of the experiments are used for triggering purposes only during data taking. Utilizing the HLT for offline processing is natural approach to increase resources. The provisioning of resources via Cloud technologies has become an establish process for various businesses and is being exploited in research as well. Yet another class of resources are time based allocations on high-performance computing systems [12]. The experiments get encouraged by providers and funding agencies to use such resources in order to enlarge the utilization of these expansive machines.

6.1. Improved Utilization of the High Level Trigger Farm

The CMS HLT group successfully managed to provide the HLT CPUs via an OpenStack installation for offline processing during longer breaks in beam operations[13]. In a next step the capability to the HLT during inter-fill periods has been addressed. With typical refilling times a few hours it has been attempted to execute jobs with a target run time of two hours. Due to the very good LHC performance in 2016 also the inter-fill time periods turned out to be too short to accommodate these short jobs. In a new approach the system has been setup to allow for suspension of the virtual machines, while the HLT is used for triggering purposes [14]. This new approach also allows to use parts of the HLT farm even during beam operation. The full capacity is only needed at the beginning of a fill. When beam currents decrease also trigger rates go down and more and more suspended virtual machines can be restarted to continue offline processing.

6.2. Cloud Resources as a Site Extension

After several smaller scale case studies CMS demonstrated in 2016 a large scale usage of cloud resources. In this example the existing Tier-1 at FermiLab was dynamically extended with resources bought from Amazon Web Services (AWS). An advantage of this approach is that no additional resources need to be included into the CMS infrastructure, all provisioning happens at the level of the site. Figure 5 illustrates the number of utilized CPU cores. The additional cloud CPUs could increase the overall number of cores by almost 50%.

The large scale demonstrator contributed in the order of 5% to the overall Monte Carlo production of CMS in 2016. A number of valuable experiences were made during the exercise [15]. Workflows needed some special adaption for running in the cloud, because a number of operations comes at significant charges. Particularly data handling has to be evaluated carefully. Since cloud providers typically charge for data exports, workflows have been configured to run the full chain from Monte Carlo input to reconstructed output. No intermediate data products were saved. The costs of running resources in the cloud compared to in-house computing have been evaluated and have been found to be higher, but not an order of magnitude. It can therefore be expected that sites will provide parts of their resources via cloud purchases to cope with peak load.

7. Summary

The second year of LHC Run 2 was very successful due to the excellent performance of the accelerator and the detectors. The resulting challenges to process and analyze the data could be met thanks to various improvements that were into the CMS computing system. Developments were targeted to achieve more automation and higher flexibility regarding the resource selection for a particular workflow type. Most important ingredients are the setup of a single Global HTCondor pool for all kinds of resources and the possibility to execute a higher fraction of workflows accessing data remotely via a data federation. There is an ongoing effort to exploit and include resources beyond classical Grid sites such as Clouds and HPC resources.

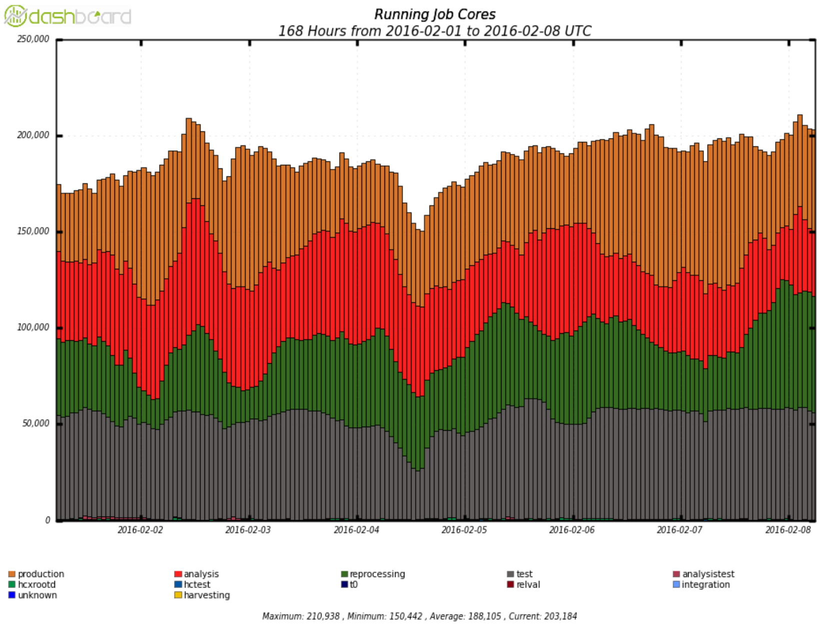

Figure 5. Number of utilized CPUs per generic workflow class. The HEPcloud contribution is labeled "test" (gray).

References

[1] Knobloch J *et al.* 2005 LHC Computing Grid Technical Design Report CERN-LHCC-2005-024
[2] Van Mulders P (for the CMS Collaboration) 2016 127th LHCC Meeting Open Session `https://indico.cern.ch/event/563488/`
[3] C D Jones and E Sexton-Kennedy 2014 Stitched Together: Transitioning CMS to a Hierarchical Threaded Framework *J. Phys.: Conf. Series* **513** 022034
[4] C D Jones *el al.* 2015 Using the CMS Threaded Framework In A Production Environment *J. Phys.: Conf. Series* **664** 072026
[5] J Letts it el al. 2015 Using the glideinWMS System as a Common Resource Provisioning Layer in CMS *J. Phys.: Conf. Series* **664** 062031
[6] J E Ramirez, A Perez-Calero Yzquierdo, J M Hernadez 2016 Exploiting multicore compute resources in the CMS experiment *J. Phys.: Conf. Series* **762** 012018
[7] S Wakefield *et al.* 2012 The CMS workload management system *J. Phys.: Conf. Series* **396** 032113
[8] J R Vlimant 2016 Software and Experience with Managing Workflows for the Computing Operation of the CMS Experiment This Conference
[9] Y Iiyama *et al.* 2016 Dynamo - The dynamic data management system for the distributed CMS computing system This Conference
[10] Bloom K (for the CMS Collaboration) 2014 CMS Use of a Data Federation *J. Phys.: Conf. Series* **513** 042005
[11] `http://xrootd.slac.stanford.edu`
[12] D Hufnagel 2015 Enabling opportunistic resources for CMS Computing Operations *J. Phys.: Conf. Series* **664** 022025
[13] D Colling *et al.* 2015 The Diverse use of Clouds by CMS *J. Phys.: Conf. Series* **664** 022012
[14] M Dobson *et al.* 2016 Dynamic resource provisioning of the CMS online cluster using a cloud overlay This Conference
[15] G Garzoglio *et al.* 2016 The HEP Cloud Facility: elastic computing for High Energy Physics This Conference