The CMS TierO goes Cloud and Grid for LHC Run 2

2015 J. Phys.: Conf. Ser. 664 032014

(http://iopscience.iop.org/1742-6596/664/3/032014)

The CMS Tier0 goes Cloud and Grid for LHC Run 2

Dirk Hufnagel (on behalf of the CMS Collaboration)
Fermilab, P.O. Box 500, Batavia, IL 60510-5011, USA

E-mail: Dirk.Hufnagel@cern.ch

Abstract. In 2015, CMS will embark on a new era of collecting LHC collisions at unprecedented rates and complexity. This will put a tremendous stress on our computing systems. Prompt Processing of the raw data by the Tier-0 infrastructure will no longer be constrained to CERN alone due to the significantly increased resource requirements. In LHC Run 2, we will need to operate it as a distributed system utilizing both the CERN Cloud-based Agile Infrastructure and a significant fraction of the CMS Tier-1 Grid resources. In another big change for LHC Run 2, we will process all data using the multi-threaded framework to deal with the increased event complexity and to ensure efficient use of the resources. This contribution will cover the evolution of the Tier-0 infrastructure and present scale testing results and experiences from the first data taking in 2015.

1. Introduction

The CMS (Compact Muon Solenoid) experiment is a general purpose particle physics detector at the LHC (Large Hadron Collider) at CERN Ref. [1]. It is located in an underground cavern at LHC P5 (Point 5) in Cessy, France. When the experiment is running and collecting data, that data is first written to a large disk buffer at P5. In a secondary step, the data is then transferred to CERN, where it is further processed. When we talk about the CMS Tier0, what we mean is all the immediate and automatic data handling and processing at CERN.

2. Changes for the CMS Tier0 between LHC Run1 and Run2

Figure 1 shows the dataflow in the CMS Tier0 for the upcoming LHC Run2. Its starts from P5, shows the major Tier0 tasks and PromptCalibration loop, which feeds back conditions to the Tier0 processing. In addition to the Tier0, which includes the automatic data handling and processing, there is also the CAF (Calibration and Alignment Facility), which includes manual data handling and processing tasks. The Prompt Calibration workflows that are run manually will be run on the CAF. If you compare this to the same diagram for LHC Run1 Figure 2, the differences are mainly the split of the Prompt Reconstruction between Tier0 resources at CERN and the Tier1.

For reference, figure 3 shows an overview of all the internal processing steps of the Tier0 in Run2. The same overview for Run1 would essentially look the same.

If the dataflow and processing steps are so similar, where are the big differences then between the CMS Tier0 used for LHC Run1 and the CMS Tier0 used for LHC Run2 ? The answer to this is that the differences are in how these dataflows and processing steps are executed on the computing resources as shown in table 1.

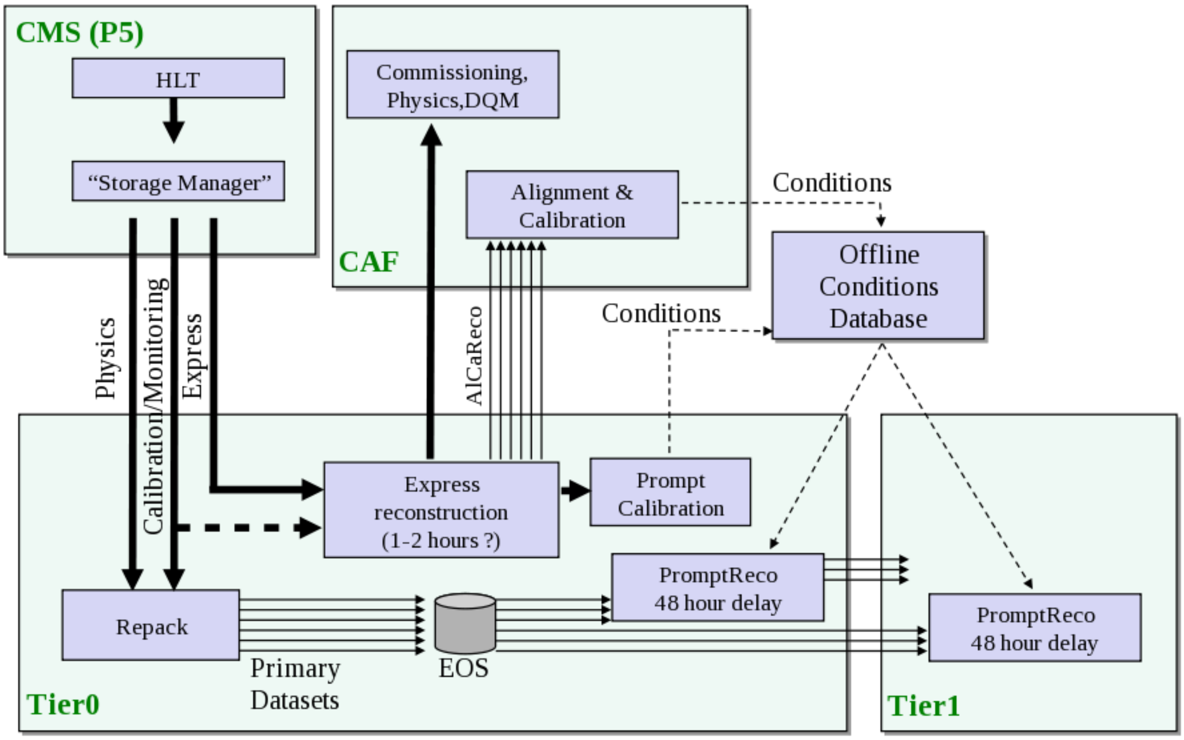

Figure 1. CMS Dataflow Summary for LHC Run2

LHC Run1	LHC Run2
LSF job submission (bsub)	HTCondor/GlideInWMS job submission
Dedicated LSF queue at CERN	CERN Cloud integrated into GlideInWMS
Tier0 only runs at CERN	Tier0 can offload fraction of PromptReco to Tier1

Table 1. Differences between Run1 and Run2

All these changes bring the Tier0 much more in line with how the remainder of CMS Computing operates. We now use the same job submission system, we use sharable resources that can easily run regular CMS Production jobs when the Tier0 is not using all resources and at the same time we also allow the Tier0 to use more general CMS resources at the Tier1.

3. CERN Agile Infrastructure and its integration into GlideInWMS

CERN Agile Infrastructure (or short CERN AI) is an Openstack based CERN internal cloud. It provides resources to various projects. Each project manages their own quota, virtual machine (VM) images etc.

GlideInWMS and HTCondor form the backbone of the CMS job execution system. For a description of the pilot based glideinWMS system see here Ref. [3]. Between LHC Run1 and Run2 CMS migrated from using it in various separate HTCondor pools to using it through one common Global pool. For more details on this migration please refer to Ref. [5].

CERN AI is fully integrated into GlideInWMS via the existing EC2/GlideInWMS interface. Through this interface the system can access a cloud controller the same way as it would access a CE, except instead of submitting jobs that start a pilot it requests to start VM that have a pilot integrated into the image. We use custom VM images without any tight integration into

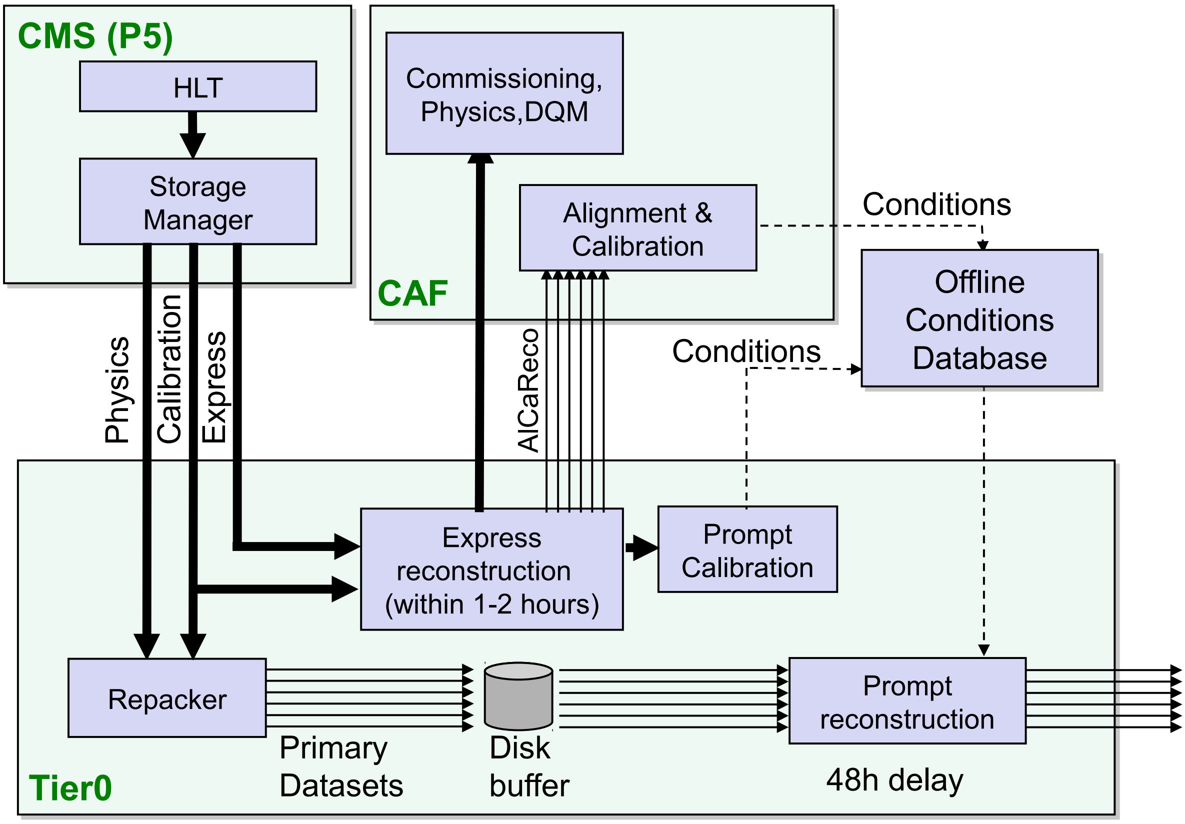

Figure 2. CMS Dataflow Summary for LHC Run1

the CERN environment (no Kerberos, no AFS, no CERN user accounts). In the future we plan on evaluating switching to micro-CERN VM Ref. [4], but we want to keep the standalone nature of our images as much as possible to minimize external dependencies. For more on how CMS is integrating cloud based resources into its computing infrastructure see Ref. [6].

At the moment there is no dynamic resource scheduling on CERN AI, resources are assigned to projects in a very static way. As such, we currently run very long-lived VM, with a lifetime of 30 days. Originally this was also chosen because of very bad performance when contacting the CERN cloud controller. The resulting necessary rate limit in creating virtual machines forced us to configure for very long-lived VM just to keep a high enough number of VM running all the time. These problems have largely been resolved now, so for maintenance reasons (VM update cycles) we might reduce the VM lifetime to a few days to a week in the future.

Still, even if we shorten the VM lifetime, we foresee a steady-state operations model with a short ramp up of number of VM to our quota and then small fluctuations as old VM expire and are quickly replaced.

While in the long run we want to integrate the CERN AI Tier0 resources into the Global pool like all other CMS resources, as a precaution for now they form a separate Tier0 pool. This takes into account the special nature of the Tier0 as a processing resource and its importance in the big picture view for CMS. For now we take the cautious approach and will gather experience with both the Tier0 pool and the Global pool and then decide how to proceed later. Flocking is enabled from the Tier0 pool to the Global pool, so that the Tier0 can submit jobs to the Tier1. Flocking will also soon be enabled from the Global pool to the Tier0 pool so that general CMS production activities can overflow into the Tier0 resources as well.

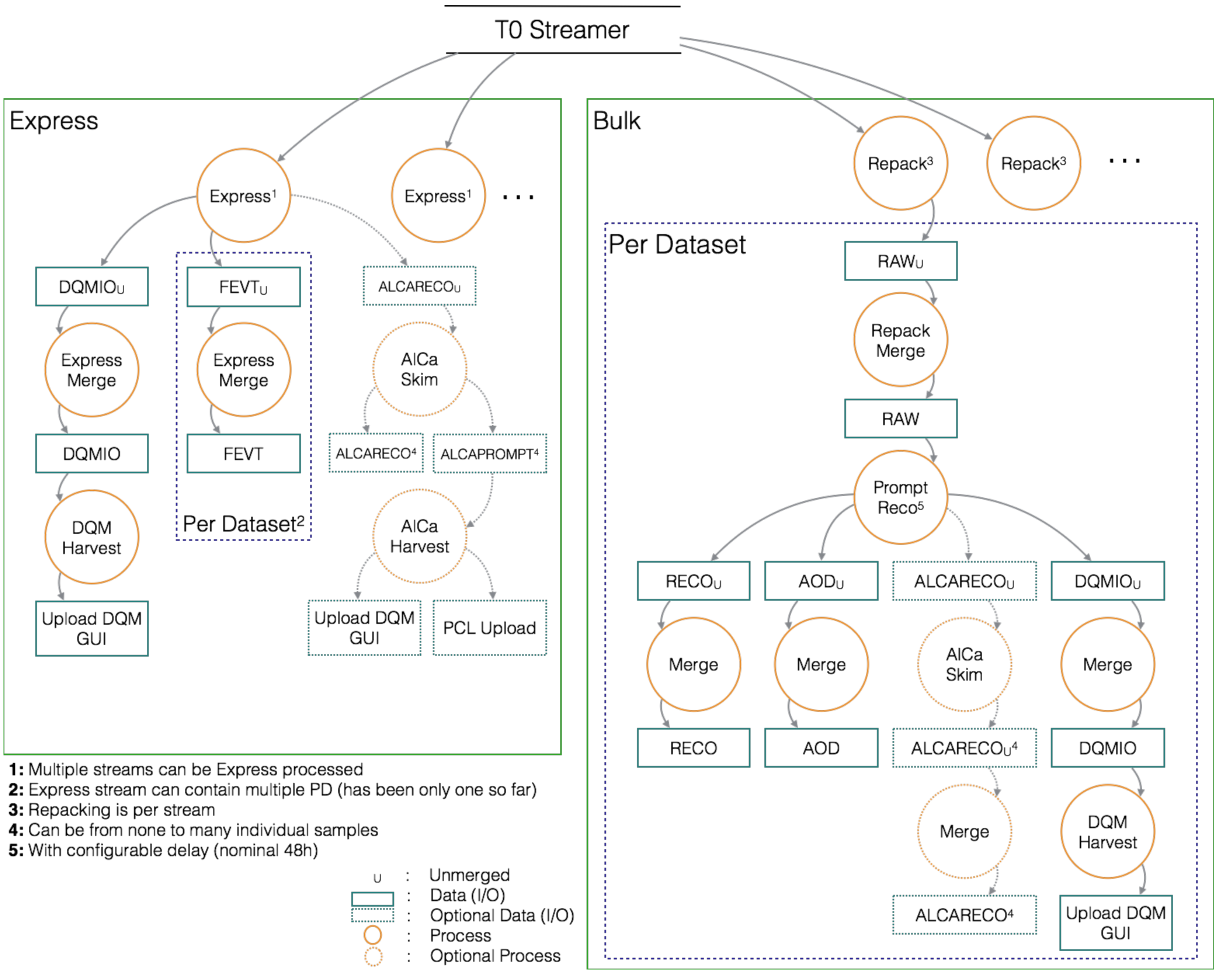

Figure 3. Processing steps within the CMS Tier0

4. Multicore

One major work item for CMS Offline during LS1 (Long Shutdown 1, the downtime between LHC Run1 and Run2 data taking periods) was finishing the changes to the CMS software framework CMSSW to allow it be used in multi-threaded mode in a production environment. For a status report on this see Ref. [7]. This work has been complete and we plan to run all Tier0 reconstruction type workflows (Express and PromptReco) multi-threaded. In fact, we count on the multi-threaded framework to allow us to handle the higher rates and higher pileup in LHC Run2. Multi-threading helps for higher pileup because it allows the reconstruction jobs to stay within a reasonable memory budget. It also indirectly helps with higher rates because it allows us to make use of all available cpu cores, with single-threaded jobs we would not be able to do so (again due to memory restrictions). There is also a secondary effect in that it reduces the number of jobs in the system, which helps reduce load on our workflow management systems. This secondary effect is very important for the high rates we are expecting for LHC Run2, with just single-threaded jobs we would run into limits here and its not clear that our workflow management systems would support this many single-threaded jobs.

As such, the Tier0 will have to be able to submit to multicore enabled resources. For the Cloud AI resources at CERN this isn't a problem. We control these resources and decide which flavor (core count, memory, etc) of VM we are using. For the Tier1 the situation is a bit different though as we have to work through the Tier1 site contacts to provide multicore job slots. All CMS Tier1 have been asked to provide at least 50% of their pledged CMS resources as multicore slots, but this is still work in progress. We have standardized on 8 core job slots here, although we can also accept different core counts (and at some sites might have to because the resources provided are not multiples of 8 cores). As far as the reconstruction jobs are concerned, we currently plan to run 4 core jobs here, but this number will be reviewed once we have some experience with multicore operations.

For more on the status of commissioning multicore operations at the Tier1 (and also in general) you can check Ref. [8].

5. Operational experience with the new system

We have run Tier0 production instances supporting data taking tests and submitting jobs to CERN AI via GlideInWMS since July 2014. Tier0 test instances using old saved 2012/2013 data have been exercised even earlier.

The currently deployed production Tier0 instance is on its final hardware (physical compute node) and runs against a production Oracle instance. It has been supporting continuous cosmics data taking with and without magnet since February 2015. Just recently it processed the very first 2015 collision events in CMS.

6. Scale testing the new system

As current data taking only produced light processing loads, we also did some scale tests using old saved 2012/2013 data. When we ran these tests on the CERN AI Tier0 resources, we reached a total of 6000 cores (the then maximum quota for the project) for a few days as you can see in figure 4. We also ran scale tests for PromptReco at the Tier1, figure 5 shows the results. At peak we almost reached our goal of 50% of all CMS Tier1 resources, but this wasn't sustained very long and also the fraction we managed to use varied wildly between the various T1 sites. At some T1 sites we reached significantly more than 50% of our pledged resources for multicore PromptReco jobs, at other T1 sites we had very few multicore PromptReco jobs running. Within the limits of available resources the system worked ok, but we will have to repeat these tests as more resources become available to test at expected LHC Run2 data taking scales.

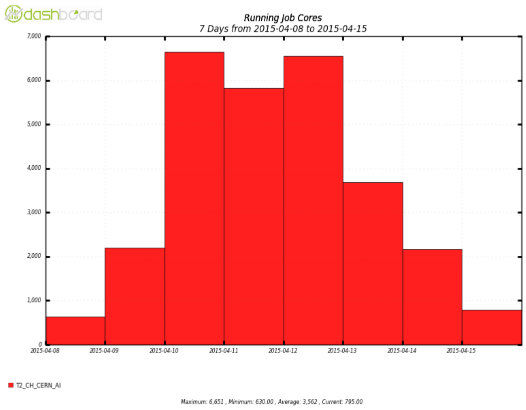

Figure 4. Core utilization in CERN AI

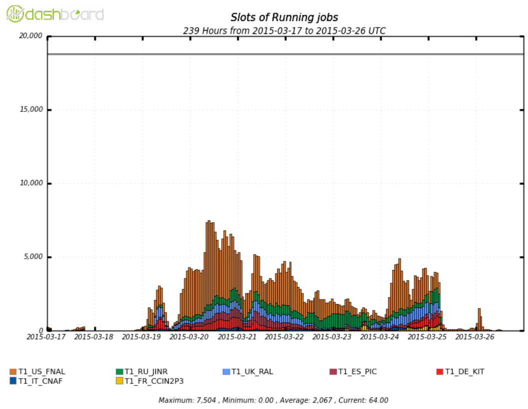

Figure 5. PromptReco at the T1 sites

7. Summary and Outlook

Much has changed for the CMS Tier0 between LHC Run1 and LHC Run2. The new setup is now much more integrated into the general CMS Computing Infrastructure, which should allow a much more flexible use of resources, allowing the Tier0 to use more general CMS computing resources if needed, but also allowing general CMS production work to overflow into Tier0 resources when the Tier0 itself does not process at full capacity.

We are currently supporting ongoing cosmics, beam splash and collision data taking with a fully functional production Tier0 deployment. At the same time we will continue scale tests to make sure we are ready for high luminosity pp collision data taking.

Maybe as the most important point, we make use of the CERN AI based cloud resources via the direct interaction of GlideInWMS with the EC2 interface of the Openstack cloud controller. For CMS, this is the first large scale production deployment of cloud resources for arguably the most important computing resource we have.

References

[1] S. Chatrchyan *et al.* [CMS Collaboration], "The CMS experiment at the CERN LHC," JINST **3**, S08004 (2008).
[2] D. Hufnagel [CMS Offline and Computing Collaboration], "The architecture and operation of the CMS Tier-0," J. Phys. Conf. Ser. **331**, 032017 (2011).
[3] I. Sfiligoi, D. C. Bradley, B. Holzman, P. Mhashilkar, S. Padhi and F. Wurthwrin, "The pilot way to Grid resources using glideinWMS," WRI World Congress **2**, 428 (2009).
[4] J. Blomer, D. Berzano, P. Buncic, I. Charalampidis, G. Ganis, G. Lestaris, R. Meusel and V. Nicolaou, J. Phys. Conf. Ser. **513**, 032009 (2014) [arXiv:1311.2426 [cs.DC]].
[5] James Letts, "Using the glideinWMS System as a Common Resource Provisioning Layer in CMS" proceedings of CHEP2015
[6] David Colling, "The diverse use of clouds by CMS," proceedings of CHEP2015
[7] Antonio Perez-Calero Yzquierdo, "Evolution of CMS workload management towards multicore job support," proceedings of CHEP2015
[8] Christopher Jones, "Using the CMS Threaded Framework In A Production Environment," proceedings of CHEP2015

Acknowledgments

Fermilab is operated by Fermi Research Alliance, LLC under Contract No. De-AC02-07CH11359 with the United States Department of Energy.