CMS Tier-0: Preparing for the future

Dirk Hufnagel, Fermilab (for CMS Computing)
E-mail: Dirk.Hufnagel@cern.ch

Abstract. The Tier-0 processing system is the initial stage of the multi-tiered computing system of CMS. It is responsible for the first processing steps of data from the CMS Experiment at CERN. This presentation covers the complete overhaul (rewrite) of the system for the 2012 run, to bring it into line with the new CMS Workload Management system, improving scalability and maintainability for the next few years.

1. Introduction
The CMS (Compact Muon Solenoid) experiment is a general purpose particle physics detector at the LHC (Large Hadron Collider) at CERN. 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 Tier-0, what we mean is all the immediate and automatic data handling and processing at CERN.

2. Dataflow
Figure 1 shows a summary of the dataflow for CMS data at CERN, starting from P5, showing the major Tier-0 tasks and PromptCalibration loops, feeding back conditions to the Tier-0 processing. In addition to the Tier-0, which includes the automatic data handling and processing, there is also the CAF (Calibration and Alignment Facility / CERN Analysis facility), which includes manual data handling and processing tasks. The Prompt Calibration workflows that are run manually will be run on the CAF.

3. Before data gets to the Tier-0
Before the Tier-0 there is the data acquisition system, the High Level Trigger (HLT) and the StorageManager (system which writes data to the disk buffer) at P5. What is written to disk at P5 is the input to the Tier-0 and its structure determines to some extent how we have to handle it. The HLT output is separated into online streams, for proton-proton collisions they are

- Physics stream: up to 1000Hz (included parked data), of which about 300Hz are prompt reconstructed
- Express stream: about 30Hz, a subset of Physics for fast monitoring/analysis and prompt calibration
- Various calibration/monitoring streams: from a few Hz to about 10kHz
Within an online stream there is no sorting of events by physics/trigger classification. Events are handed from the HLT to the StorageManager and written to disk in a special streaming format (streamer files). There are multiple StorageManager instances to optimize the write rate to disk, which causes data taken within very small time windows to be split across multiple streamer files.

4. Tier-0 requirements
   (i) Repacking

   First, the special data format of the streamer files is not software release independent. This means that there is no guaranteed forward or backward compatibility for streamer files, to first order they need to be read with the software release they were written. This makes them unsuitable for custodial long-term storage. Second, luminosity for CMS data can only be calculated for clearly defined short data segments (called a lumi section). At the moment this is set to about 23s. Due to the way the StorageManager writes out the data at P5, a lumi section is split across multiple streamer files. Having a lumi section split across multiple files leads to complications in the data handling as it introduces dependencies between files. And third, the data in the streamer files is not sorted by physics/trigger classification. This leads to 3 requirements, which all will be addressed in a single workflow, the repacking.

   • Convert streamer files into ROOT-based custodial data format
   • Assemble lumi sections into files that only contain complete lumi sections
   • Split data into sub-samples according to trigger classification of events
(ii) PromptReconstruction
Reconstruct the data after a 2 day wait for updated conditions from the PromptCalibration.
Once started, finish the reconstruction within 24 hours.

(iii) ExpressProcessing
Provide a fully reconstructed subset of the data after 1 hour for fast monitoring/analysis and feedback. The first stage of this is already done at the HLT level, writing a separate Express online stream, which is the input for the ExpressProcessing. The PromptCalibration loops that calculate conditions for PromptReconstruction also run on output of the express processing.

(iv) PromptCalibration
There are two PromptCalibration loops, one fully in the Tier-0, the other via the CAF. For the latter, output from the express processing is made available on the CAF, where Alignment and Calibration workflows can then be run manually or semi-automated. In contrast, the PromptCalib loop within the Tier-0 has to work automatically, without any manual steps.

5. Current Tier-0 general architecture
A detailed description of the current production Tier0 can be found in Ref. [2]. It is slightly dated, but the only major change to the currently deployed Tier0 is the addition of the Prompt Calibration Loop (which was already implied in Ref. [2]).

As mentioned in Ref. [2], the current production Tier-0 is based on the ProdAgent architecture (CMS MC Production System). It consists of many individual components, each component having a well defined task. In particular, for every type of job we need to run (Repack, Express, Merge, etc), there is a separate component that handles creating jobs.

![Diagram of Tier-0 jobs](image)

**Figure 2.**

Figure 2 shows an example for the Repack, Express and Merge jobs. The RepackInjector and ExpressInjector look at all the (streamer) files from P5 and determine based on file status (whether it is already assigned to a job) and file meta-data (run number and online stream) which files needs to be processed and how jobs should be created. If needed, it creates a job and passed it to the ProdAgent execution layer. Similar for the Tier0Merger, which looks at all the unmerged files and depending on their status and meta-data (run number and Physics dataset) creates merge jobs.
6. Changes in CMS workload management
Since Ref. [2] was written, major changes have occurred in the wider world of CMS workload management. During the implementation of the Tier-0, it became clear that the underlying ProdAgent system just did not provide all the needed features to handle Tier-0 data processing. That’s why the Tier-0 implemented its own file and job bookkeeping and only used the ProdAgent execution layer to run jobs.

Partly from the experiences with the Tier-0 implementation in a ProdAgent environment, a process was started to work on a new workload management system to address the deficiencies in ProdAgent. During 2011, the ProdAgent has been retired as the CMS production system and was replaced with a new system, WMAgent.

As of this time, the only major system left still using (parts of) ProdAgent is the Tier-0.

7. WMAgent architecture
For details see Ref. [3]. In general, the WMAgent system relies on requests, which define a set of data and what to do with the data. The smallest building block of a request consists of a fileset (some data), a workflow (a processing configuration) and a subscription tying these two together. These building blocks can be tied together by defining the output of one step as the input to another one. In this way one can build complicated processing chains or processing trees.

Once these request (or composite requests) are defined, they are handled by a set of generic WMAgent software components. There is for instance only one component creating jobs. It will check each subscription, see which processing configuration is set in the workflow, what data it needs to be applied to and create jobs accordingly.

This is fundamentally at odds with how the current production Tier0 works. Therefore switching the Tier-0 to use the WMAgent was quite difficult, it involved not just a simple port, but more a redesign and rewrite of the whole system.

8. Tier0/WMAgent architecture
One consequence of the different design philosophies and big difference between the current Tier0 and the Tier0/WMAgent is that the current Tier0 can deal with changes in processing configurations for different data on the fly. The component dealing with a processing step just adjusts the processing configuration to the data it is dealing with. This simply does not work anymore in the WMAgent. The actual job creation is done by a generic component that has no idea about the differences between run N files and run M files. The input data for a processing step needs to be consistent in the way it is processed.

Figure 3 shows how this is dealt with by partitioning the data. Repack and Express configurations apply to an online stream and can change at run boundaries, therefore the largest unit of data with a guaranteed consistent processing configuration is a run/stream. Similarly for PromptReco, there we configure for Physics datasets and the configuration can change at a run boundary, leading to a unit of data of a run/dataset.

To do this, we have one Tier-0 specific software component, Tier0Feeder, on top of a standard WMAgent. For every new run/stream we receive data for from P5, it creates a run/stream specific fileset and populates it (and keeps populating it with new data). It also, based on the Tier-0 configuration, creates a Repack or Express workflow to process the data for this run/stream and ties the workflow and fileset together with a subscription. After the Repack workflow creation, we also have the information about which Physics datasets is populating which output fileset. When the delayed PromptReco is released 2 days after the end of a run, the Tier0 will create PromptReco workflows (again based on the Tier0 configuration) and tie them with subscriptions to their input filesets (which are still known from the Repack workflow creation).
That’s basically the whole Tier-0 in a WMAgent environment. Very simple, but only because the system complexity is hidden within the workflow definitions. The WMAgent does not care how complex and deep these workflow definitions are, it just processes them one step at a time.

For instance, the PromptReco workflow consists of an initial processing step that usually does write at least three different outputs, all of which are merged. One of these merged samples is then input to the Data Quality Monitoring harvesting and upload of information.

9. Changes in Tier0 Monitoring
The current production Tier0 has standalone separate monitoring. Migration to a WMAgent based Tier0 will give us the opportunity to switch to the same monitoring system used for all other CMS production activities.

10. Status and Outlook
The Tier0/WMAgent is currently still under development. Test versions are feature complete as far as actually data processing is concerned. What is missing are features at the edge of the system, like for instance Data Quality Monitoring harvesting and upload of information and the Prompt Calibration.

The current Tier-0 has been in use for almost 3 years now (in various states of feature completeness). Continuous feedback from operation resulted in many tweaks and protections against anomalous running conditions. It is difficult to replicate this in the new system without having to go through a re-commissioning phase.

References
[1] CMS Collaboration, The Computing Project, Technical Design Report, CERN/LHCC 2005-023
[2] The architecture and operation of the CMS Tier-0, Dirk Hufnagel (for Cms Offline and Computing) 2011 J. Phys.: Conf. Ser. 331 032017
[3] The CMS workload management system (CHEP2012 id 579)