PROCEEDINGS OF SCIENCE

LHCb computing in Run 2 and its evolution towards Run 3

Antonio Falabella - On behalf of the LHCb collaboration*
INFN - CNAF (Bologna)
E-mail: antonio.falabella@cnaf.infn.it

This contribution reports on the experience of the LHCb computing team during LHC Run 2 and its preparation for Run 3. Furthermore a brief introduction on LHCbDIRAC, i.e. the tool to interface to the experiment distributed computing resources for its data processing and data management operations, is given. Run 2, which started in 2015, has already seen several changes in the data processing workflows of the experiment. Most notably the ability to align and calibrate the detector between two different stages of the data processing in the high level trigger farm, eliminating the need for a second pass processing of the data offline. In addition a fraction of the data is immediately reconstructed to its final physics format in the high level trigger and only this format is exported from the experiment site to the physics analysis. This concept have successfully been tested and will continue to be used for the rest of Run 2. Furthermore the distributed data processing has been improved with new concepts and technologies as well as adaptations to the computing model. In Run 3 the experiment will see a further increase of instantaneous luminosity and pileup leading to even higher data rates to be exported. The signal yield will further increase which will have impacts on the data processing model of the experiment and the ways how physicists will analyse data on distributed computing facilities. Also connected to the increased signal yield is the need to produce more Monte Carlo samples. The increase in CPU work cannot be absorbed by an increase in hardware resources. The changes needed in the data processing applications will be discussed in the area of multi-processor aware applications, changes in the scheduling framework of the physics algorithms and the changes in the experiment data event model to facilitate SIMD instructions.

38th International Conference on High Energy Physics
3-10 August 2016
Chicago, USA

*Speaker.

PoS(ICHEP2016)191

1. Introduction

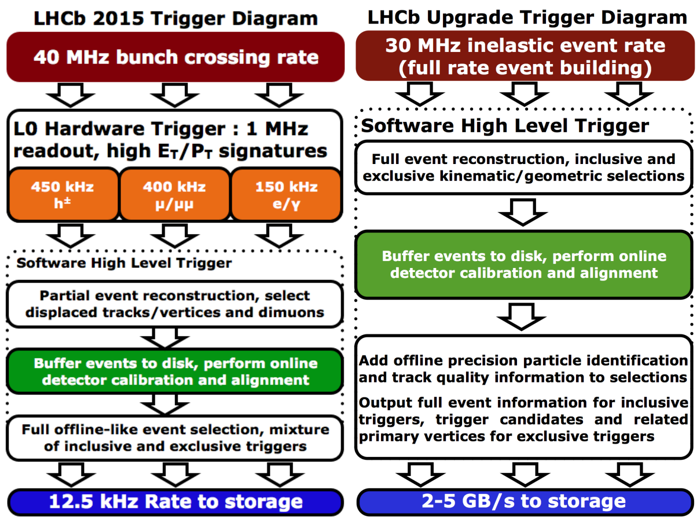
LHCb[1] is one the four main experiments currently operating at the CERN Large Hadron Collider in Switzerland. LHCb aims at the precise measurement of b and c-hadrons decays and CP violation parameters.

2. LHCb Computing Model

The LHCb computing model allowed to successfully collect data during Run 1, on the other hand the experience led to some modifications for Run 2. Currently events passing the trigger selections are sent to offline via a "FULL" stream of RAW events. One copy of the RAW data is kept at CERN while another one is distributed at the other 7 Tier-1s sites. A reconstruction step is then performed on the RAW data. The reconstruction step runs democratically at CERN and Tier1s. A the end of the year a reprocessing step is performed. Each reconstruction is followed by a stripping step that produce an output data format called DST, or μDST depending on the quantities saved. The stripping step consist of event selections defined by physics working groups. These selections are grouped in 12 streams. The stripping output is distributed to the CERN and the other Tier1s. The stripping output are the main input for user analysis. An additional step under investigation is centralization of the user analysis step. User jobs runs at the Tier0 and Tier1s, but shall not access RAW files or unstripped DSTs. An evolution with respect to Run 1 distributed data processing model consists of the role of Tier2 sites. In Run 1 they were essentially used for Monte Carlo data productions while in Run 2 this constraint has been relaxed given that some of them provide also storage space. Tier storage storage is useful because it allow user analysis, and in *Mesh Processing* where they contribute to reconstruction and stripping.
The distributed computing activities are handled with the LHCbDirac[2] software. It is an extension of the Dirac software[3]: a community Grid solution with experiment agnostic job submission and data management capabilities. It consists of several subsystems interfaced with services or lightweight agents (similar to unix cron jobs). LHCbDirac includes the extension to Dirac that are specific of the LHCb experiment, in particular the workflow concept of Dirac is extended to handle real data processing, stripping and simulation. A specific component is also the Bookkeeping system which keep metadata informations of the files and an interface for user analysts.

3. Run 2 Data Processing

In Run 2 the trigger output event rate is 12.5KHz. The trigger configuration is implemented in two stages as shown in Fig.1a. The first stage ($L0$) is an hardware trigger as in Run 1, that reduces the frequency from the LHC bunch crossing frequency of 40MHz to 1MHz. The second stage called HLT(High Level Trigger) is a software trigger further divided in HLT1 and HLT2. Between these two sub-levels a disk buffer is placed in order to have the time ($\sim O(min)$) to compute the calibration and alignment constants. Using HLT2 algorithms as close as possible offline algorithms avoid the need to run reprocessing campaigns as in Run 1. Another novel concept developed and implemented in Run 2 is the "TURBO"[4] stream. This stream consists of selected trigger lines reconstructed online and persisted on μDST format. The "TURBO" stream output is then already



(a) Schema of the LHCb 2015 trigger configuration.

(b) Schema of the LHCb Upgrade trigger configuration.

available for physics analysis. As shown in Fig.2 apart from the "FULL" and "TURBO" streams there is also a "CALIBRATION" stream used to compute data driven efficiency used for online and offline reconstruction.

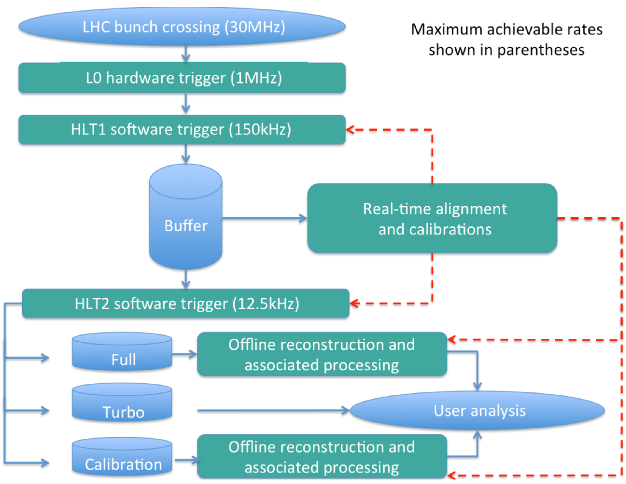


Figure 2: LHCb trigger output streams for Run 2.

4. Towards the LHCb Upgrade

Run 2 is a useful test bed for the upgrade for the Run 3 data taking (start in 2021). The strategies for the computing upgrade must be carefully evaluated mainly because the trigger will output one order of magnitude more data with a constant computing budget. The various parts of the computing model can be broadly grouped into two groups. The ones that will be adapted (*Evolution* concept) and the ones that will be redesigned (*Revolution* concept). The first group consists of the data processing and analysis model, external software like Dirac and the Simulation framework. The parts that has been recognized to need a redesign are the core software framework, the event model, the detector description and the conditions databases. Before describing each part some words must be spent on the foreseen trigger configuration [5]. The schema of the LHCb trigger upgrade can be seen in Fig.1b. The idea is to remove the hardware trigger and use a software trigger only operating at an input rate of 30MHz. This is required because of the saturation of the $L0$ hadronic trigger lines. This strategy means that the concepts of stripping and streaming will be blurred. The TURBO stream will be then the default one. The requirement for offline data processing will be then considerably reduced, on the other hand more signal data means a proportional increase of the simulation events. As of now the MC jobs accounts for the 75% of the total delivered CPU power as shown in Fig.3.

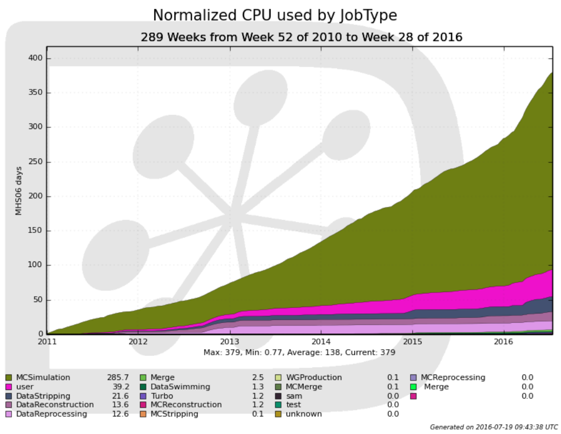


Figure 3: Normalized CPU used in Run 1 and Run 2 sorted by job type.

The LHCb software framework is based on Gaudi[6], and it will also the base for the upgrade. Currently it only support single-threaded execution, the idea is to implement support for multiprocessing at the task level thinking on algorithms as reentrant entities as proposed in the GaudiHive project [7]. Further changes will be also on the area of event model. Currently it is based on an array of structures (*AoS*) memory layout that make difficult to profit of the SIMD features of modern CPUs. Efforts on the parallelism of selected algorithms are already ongoing especially on the forward track finding and track fitting algorithms.
Another topic to be addressed are the so called non-event data. Two data bases: LHCb Detector Description (LHCbDD) and Conditions Data Base (CondDB) are used. Both suffered for the lack of thread-safeness and being based on an XML-based persistency format. A good candidate for the detector description to substitute then is the DD4Hep toolkit [8]. As previously mentioned the CPU resources will become an issue in Run 3 especially for simulation. At a CPU power increase of 20% per year the distributed infrastructure will not cope with foreseen CPU needs because of the luminosity increase. A possible candidate for the Run 3 is *Gaussino* that is part of the Future Circular Collider Software Stack which aims at the support for multi-thread execution and fast simulation.

5. Summary

The experience of Run 1 and Run 2 shows that the current LHCb computing model will not scale for Run 3. In particular the RAW storage and the stripping as well as Monte Carlo simulation will not scale. Several part of the computing model is recognized to require only a evolution will other require a redesign. The Run 2 is ground to test new concept as the TURBO stream. It will probably become the default for Run 3 and will be developed further. Concerning distributed data processing the LHCbDirac framework will not require a deep redesign, but will provide the require flexibility to face the computing model evolution.

References

[1] A.A.Alves Jr. et al.*The LHCb Detector at the LHC*, JINST 3 (2008) S08005. doi:10.1088/1748-0221/3/08/S08005.

[2] F.Stagni et al. *The LHCb DIRAC-based production and data management operations systems*, J.Phys.Conf.Ser. 368 (2012) 012010

[3] Tsaregorodtsev A et al.*DIRAC: a community grid solution*, S G 2008 Journal of Physics: Conference Series 119 062048

[4] A. Puig et al. *The LHCb Turbo stream*, Nucl. Instrum.Meth. Section A, Volume 824, 11 July 2016, Pages 35-37.

[5] A.Dziurda et al. *The LHCb trigger and its upgrade*, Nucl.Instrum.Meth. A824 (2016) 277-279

[6] *The Gaudi Framework*`http://cern.ch/gaudi`

[7] *Concurrent Framework Project*`http://concurrency.web.cern.ch/GaudiHive`

[8] *DD4Hep* `http://aidasoft.web.cern.ch/DD4hep`