PROCEEDINGS OF SCIENCE

PoS(ICHEP2020)730

Flexible selections at 30 MHz in the LHCb Upgrade trigger

Victor Daussy-Renaudin[a,*]

[a] *University of Oxford,*
Keble Rd, Oxford, United Kingdom

E-mail: victor.daussy-renaudin@physics.ox.ac.uk

The first LHCb upgrade will take data at an instantaneous luminosity of $2 \times 10^{33}\text{cm}^{-2}\text{s}^{-1}$ starting in 2022. Due to the high rate of beauty and charm signals LHCb has chosen as its baseline to read out the entire detector into a software trigger running at the LHC collision frequency of 30 MHz. This High Level Trigger (HLT) will enable unprecedented flexibility for trigger selections. To provide a flexible trigger, the key features of the trigger selection framework are discussed here as well as the current performance.

40th International Conference on High Energy physics - ICHEP2020
July 28 - August 6, 2020
Prague, Czech Republic (virtual meeting)

---

[*] Speaker

1. Introduction

During Run 1 and Run 2, the LHCb experiment collected data at an instantaneous luminosity of $4 \times 10^{32}\text{cm}^{-2}\text{s}^{-1}$. This translates to rates of $b\bar{b}$ and $c\bar{c}$ of respectively $\sim$ 45 kHz and $\sim$ 1 MHz based on trigger efficiencies [1] and cross-sections [2, 3]. The LHCb trigger was designed to handle such a large range of rates and provide enough flexibility to reach both high efficiency for charm physics and high purity for rare decays of beauty hadrons. The trigger strategy consisted of a hardware trigger (L0) followed by a two-staged software system (HLT1& HLT2).

Starting in 2022, the LHCb upgrade will record data at an instantaneous luminosity of $2 \times 10^{33}\text{cm}^{-2}\text{s}^{-1}$, five times the Run 1 and Run 2 luminosity. Many beauty and charm hadrons decay rates will fall in the MHz region. To cope with such an increased data stream, the hardware L0 system will be removed and the trigger will be purely software-based. To reach a more flexible system, allowed by the removal of the L0, and retain the current performance, a complete overhaul of the software has to be performed.

The design of the trigger for the LHCb upgrade and comparison with respect to previous runs is detailed in Sec. 2. Sec. 3 describes the selection framework and its key features that will provide the required flexibility and Sec. 4 focuses on the trigger performance.

2. The LHCb trigger upgrade

The LHCb trigger for the upgrade is composed of two levels: HLT1 and HLT2 as shown in Fig. 1. HLT1 handles the triggerless readout of the detector performed at 30 MHz. It also carries out a partial reconstruction of charged tracks and muon identification. HLT1 selections are based on the properties of single tracks and two-track composite objects and permit a reduction of the event rate by around a factor 30, depending on the tuning of the selection cuts [4].

The HLT2 application performs the complete event reconstruction, including particle identification, and all the selections required by physical analyses. The output of the HLT1 is buffered which also allows the execution of a near-real-time alignment and calibration of the detector. Hence the offline computing is reduced to streaming and final selection as offline-quality processing is achieved within the trigger system. In terms of computing resources, the HLT1 track reconstruction and event selection will be performed on $\sim$ 500 GPUs and the HLT2 application will run on $\mathcal{O}(1000)$ CPUs [4, 5].

If the main HLT1 challenge is to perform partial reconstruction and several selections at 30 MHz, then for HLT2 it is reducing the data bandwidth down to 10 GB/s [5] while remaining flexible. Indeed, $\mathcal{O}(1000)$ [6] unique selection algorithms are expected to run which reflects the variety of physics analyses performed within the LHCb collaboration. The next section describes the selection framework and its key features to meet the HLT2 requirements.

Figure 1: Schematic view of the LHCb trigger for the upgrade.

3. The selection framework

3.1 A new trigger configuration

The HLT2 trigger needs to be both flexible and efficient to handle $O(1000)$ unique trigger selections and a fixed output bandwidth of 10 GB/s.

The trigger selection framework is composed of C++ algorithms running withing the Gaudi framework [7]. For the upgrade, each algorithm is developed in a functional way: they are pure functions which returned output only depends on inputs declared explicitly. A functional design allows processing events within HLT2 in a multi-threaded way which is crucial to reach higher computational performance as described in [8]. Algorithms then form a tree based on data flow dependencies, and can be further composed in a control flow, as shown in Fig. 2, to evaluate a trigger decision per event. A control flow is composed of basic nodes, algorithms and the list of their data dependencies, and composite nodes handling the association logic.

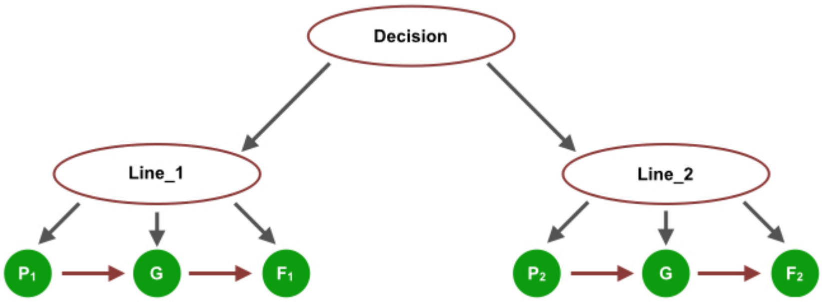

Figure 2: Adapted from [9]. Schematic view of a typical trigger selection control flow with both basic (green) and composite (red) nodes (top). A typical trigger selection consists of an (optional) Prescale (P), a Global Event Cut (G) and a Filter (F).

With the expected number of selections to be run, it leads to a very complex control flow graph. To optimise the execution, a scheduler algorithm creates the list of data dependencies by matching algorithms inputs and outputs [9]. Taking into account these data constraints, the execution order of basic nodes is then reordered, as shown in Fig. 3. A realistic trigger line selecting muon candidates showing both data and control flows is presented in Fig. 4.

3.2 Bandwidth optimisation and selective persistence

The Turbo data processing model [10] is the LHCb reduced event data format where only objects used to create the trigger candidate are stored for offline use. Introduced during Run 2, this reduced event data format was successfully used for several measurements. It was extended to selective persistence to allow for a tunable size of event from turbo data ($\sim$ 15 kB) up to the raw event ($\sim$ 70 kB) with all reconstructed objects kept.

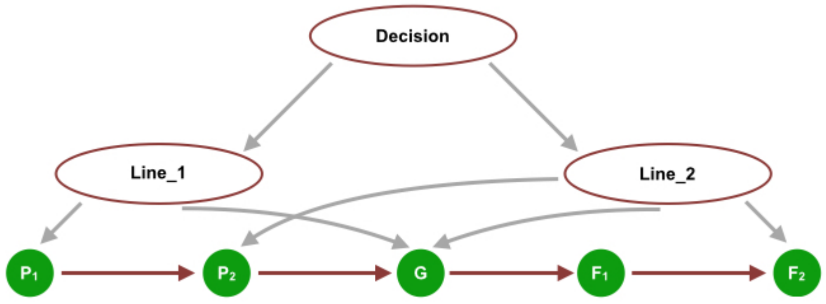

Figure 3: Adapted from [9]. The schematic view of reordering the basic nodes execution to match data constraints.

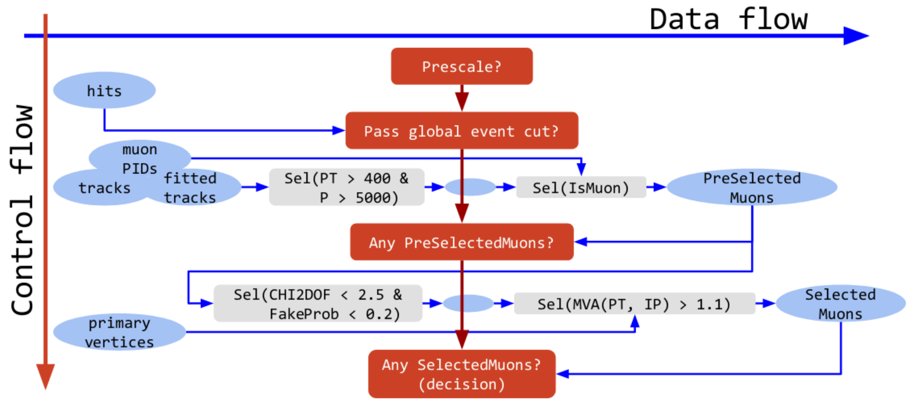

Figure 4: From [8]. Realistic example of a HLT2 trigger selection. Data inputs are shown in blue, control flow nodes in red and selection algorithms in grey.

A flexible output data format is crucial to optimise each trigger selection bandwidth separately. Besides individual tuning of each trigger selection, a genetic algorithm detailed in [11] can be used to achieve an overall optimisation of the HLT2 output bandwidth.

4. Performance of the LHCb trigger

The HLT1 trigger performance is estimated using simulation samples tuned to the expected running conditions for the upgrade. Fig. 5 presents the reconstruction efficiency for tracks originating from the decays of beauty hadrons as well as the performance to correctly identify muons. Overall performance of HLT1 is sufficient to efficiently implement trigger selections representative of the LHCb physics program.

Tab. 1 shows the trigger selection rates implemented in HLT1. The total output rate of HLT1 is also presented and is of the order of 1 MHz which demonstrates the capacity of the HLT1 to reduce

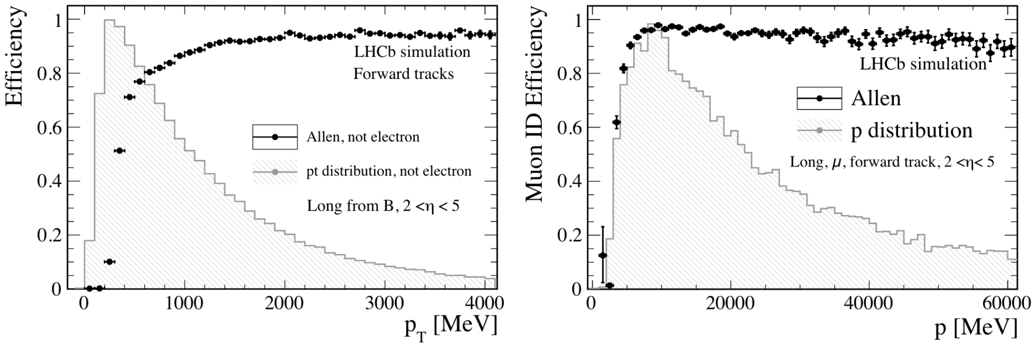

Figure 5: From [12]. HLT1 performance to reconstruct tracks originating from decays of beauty hadrons (left) and muon identification efficiency (right).

the input rate by a factor 30. More details on the HLT1 design and performance can be found in [4] and [12].

The latest HLT2 throughput performance study is shown in Fig. 6. HLT2 is currently able to process 133 Events/s per node [13]. This value is expected to increase substantially as performance improvement techniques learnt during HLT1 development are ported to HLT2. Run 2 trigger selections are also being ported to the new framework, with hundreds of lines already using it. Feedback from the analysts on the new framework is positive, especially concerning its ease of use and debugging capabilities.

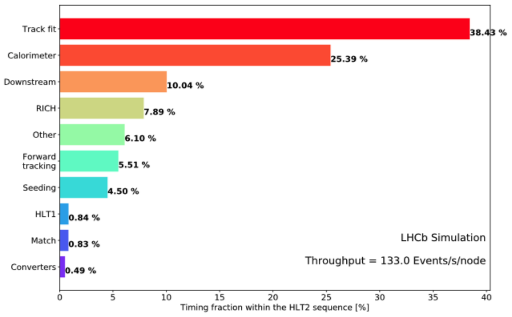

Figure 6: From [13]. Breakdown of the HLT2 reconstruction throughput.

Triger	Rate [kHz]
TrackMVA	409 ± 23
TrackMuonMVA	23 ± 6
singleHighPtMuon	7 ± 3
TwoTrackMVA	503 ± 26
DiMuonHighMass	131 ± 13
DiMuonLowMass	177 ± 15
DiMuonSoft	8 ± 3
D2KPi	93 ± 11
D2PiPi	34 ± 7
D2KK	76 ± 10
Total w/o pass through lines	1157 ± 39

Table 1: From [12]. Trigger selection rates implemented in HLT1 and the total HLT1 output rate.

5. Conclusion

The LHCb trigger is currently undergoing a complete redesign to cope with the upgrade conditions. Millions of beauty and charm decays will occur within the LHCb detector acceptance during every second of data-taking, shifting the trigger goal from rejecting background to categorise a large variety of signal modes. The new software-based trigger consists of the HLT1 reducing the rate from 30 MHz to 1 MHz while performing partial reconstruction and the HLT2 handling the detector calibration and alignment and the $O(1000)$ unique trigger selections. To achieve a flexible

HLT2 trigger, the selection framework was overhauled to allow for multi-threaded execution of algorithms based on their data dependencies and an event data format tunable to optimise each trigger selection bandwidth.

References

[1] R. Aaij et al. Design and performance of the LHCb trigger and full real-time reconstruction in Run 2 of the LHC. *Journal of Instrumentation*, 14(04):P04013–P04013, apr 2019. doi: 10.1088/1748-0221/14/04/p04013. URL https://doi.org/10.1088%2F1748-0221%2F14%2F04%2Fp04013.

[2] Roel Aaij et al. Measurement of the b-quark production cross-section in 7 and 13 TeV pp collisions. *Phys. Rev. Lett.*, 118(5):052002, 2017. doi: 10.1103/PhysRevLett.118.052002. [Erratum: Phys.Rev.Lett. 119, 169901 (2017)].

[3] Roel Aaij et al. Measurements of prompt charm production cross-sections in pp collisions at $\sqrt{s} = 13$ TeV. *JHEP*, 03:159, 2016. doi: 10.1007/JHEP03(2016)159. [Erratum: JHEP 09, 013 (2016), Erratum: JHEP 05, 074 (2017)].

[4] Roel Aaij et al. Allen: A high level trigger on GPUs for LHCb. *Comput. Softw. Big Sci.*, 4(1): 7, 2020. doi: 10.1007/s41781-020-00039-7.

[5] LHCb collaboration. Computing Model of the Upgrade LHCb experiment, 2018.

[6] LHCb collaboration. LHCb Upgrade Software and Computing, 2018.

[7] G. Barrand et al. GAUDI — A software architecture and framework for building HEP data processing applications. *Computer Physics Communications*, 140(1):45 – 55, 2001. ISSN 0010-4655. doi: https://doi.org/10.1016/S0010-4655(01)00254-5. URL http://www.sciencedirect.com/science/article/pii/S0010465501002545. CHEP2000.

[8] Rosen Matev et al. Configuration and scheduling of the LHCb trigger application. *EPJ Web Conf.*, 245:05004, 2020. doi: 10.1051/epjconf/202024505004. URL https://doi.org/10.1051/epjconf/202024505004.

[9] Ekaterina Govorkova et al. A new scheduling algorithm for the LHCb upgrade trigger application. *Journal of Physics: Conference Series*, 1525:012052, apr 2020. doi: 10.1088/1742-6596/1525/1/012052. URL https://doi.org/10.1088%2F1742-6596%2F1525%2F1%2F012052.

[10] R. Aaij et al. A comprehensive real-time analysis model at the LHCb experiment. *Journal of Instrumentation*, 14(04):P04006–P04006, apr 2019. doi: 10.1088/1748-0221/14/04/p04006. URL https://doi.org/10.1088%2F1748-0221%2F14%2F04%2Fp04006.

[11] C. Fitzpatrick et al. Upgrade trigger: Bandwidth strategy proposal, 2018.

[12] LHCb collaboration. Performance of the GPU HLT1 (Allen), 2018.

[13] LHCb collaboration. Throughput and resource usage ofthe LHCb upgrade HLT, 2018.