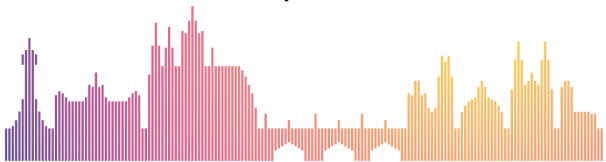
Physics and throughput performance of the real-time reconstruction for the LHCb upgrade

40^{th} International Conference on High Energy Physics (ICHEP)

Renato Quagliani (LPNHE)
on behalf of the LHCb collaboration

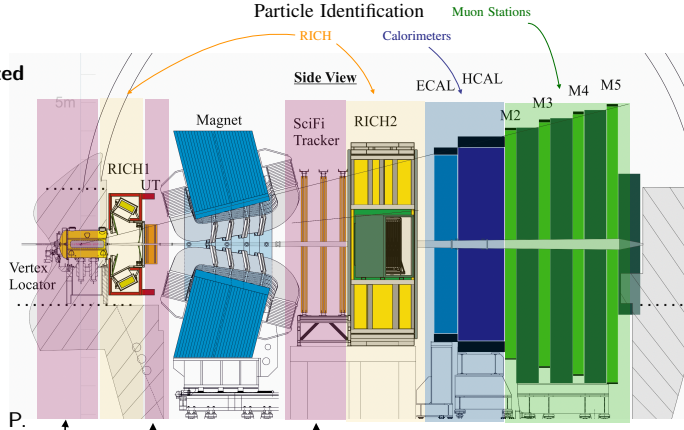
July 29, 2020



The LHCb detector

LHCb is a high precision experiment at LHC optimized for b and c hadrons decays

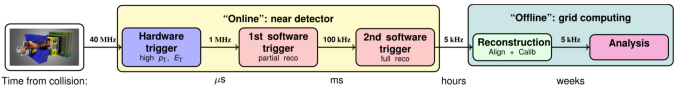
- Forward arm spectrometer in $\eta \in [2, 5]$
- Excellent track and vertex reconstruction
  1. $\sigma_{IP} \sim 20\,\mu\mathrm{m}$ ($p_T > 2\,\mathrm{GeV}/c$)
  2. $\epsilon_{tracking} > 96\%$
  3. $\sigma_p/p \sim 0.5 - 1\%$
  4. $\sigma_\tau \sim 45\,\mathrm{fs}$ for b hadrons.
- Excellent particle identification
  1. $\epsilon_{K-ID} \sim 95\%$
  2. $\epsilon_{\mu-ID} \sim 97\%$
- Benefit of large $b\bar{b}$ and $c\bar{c}$ cross section in pp collision in forward region.
- More on Run 2 performance in *talk* from Martina P.



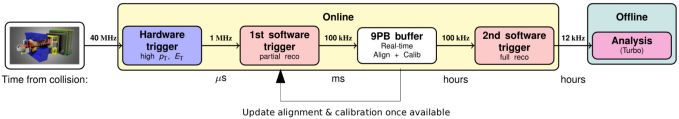
Run2 detector performance
Int. J. Mod. Phys. A30 (2015) 1530022

LHCb DAQ and trigger in Run1-2-3 : a continuos evolution

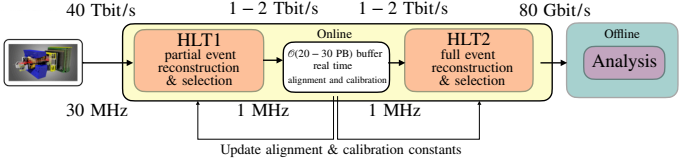
- **Run 1**:



- **Run 2**:



- **Run 3**:

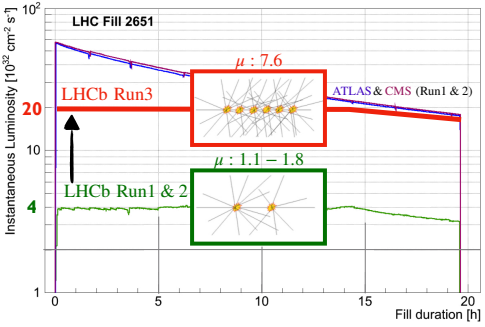


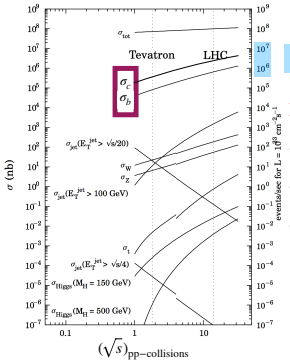
- Hardware trigger: $40 \rightarrow 1$ MHz read-out limit in **Run1,2** based on Muon and Calorimeter signatures
- HLT1(partial) and HLT2(full) event reconstruction split in **Run2**
- **Buffer** data to disk to perform real time alignment and calibration
- Offline quality reconstruction and selection in the online system

- **Run3** : remove Hardware trigger in favour of a fully software based one.
- Event reconstruction at collision rate
- Full detector read-out at 40 MHz

From Run 1,2 to Run3: b, c physics at LHC

- Run 3 data taking period will start in 2021
- LHC pp collisions at $\sqrt{s} = 14\,\mathrm{TeV}$, $25\,\mathrm{ns}$ bunch spacing $\rightarrow$ 40 MHz collision rate.
- LHCb aims at boosting the physics output increasing the instantaneous luminosity and the signal rate.





MHz signal rate

For $\mu \sim 7.6$

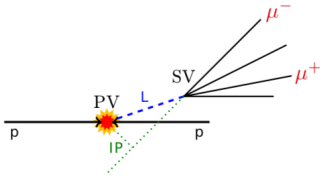
- 0.02 b hadron/event
- 0.2 c hadron/event
- 2 light-long-lived/event

(*)accounting for $\epsilon_{acc}^{LHCb} \times \epsilon_{recolle}^{>=2-daughters}$

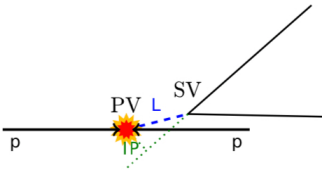
- More PVs, more tracks, more signal
- Almost all events will have a b or c hadron in Run 3

LHCb-PUB-2014-027

Signatures in LHCb from b and c hadrons for triggering



- $m_{head} \sim 5.28\,\mathrm{GeV} \rightarrow p_T^{daughters} \sim \mathcal{O}(\,\mathrm{GeV})$
- $\tau_B \sim 1.16\,\mathrm{ps}$. $\Delta(SV - PV) \sim 1\,\mathrm{cm}$.
- Dispaced tracks carrying high p_T.

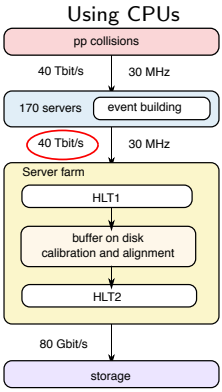


- $m_{head} \sim 1.86\,\mathrm{GeV} \rightarrow p_T^{daughters} \sim \mathcal{O}(\,\mathrm{GeV})$
- $\tau_B \sim 0.4\,\mathrm{ps}$. $\Delta(SV - PV) \sim 0.4\,\mathrm{cm}$.
- Dispaced tracks carrying high p_T.

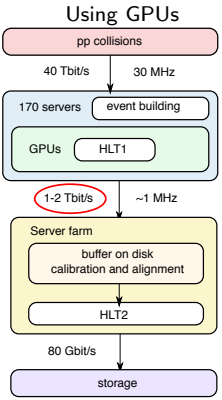
Key ingredients for efficient triggering and signal discrimination

- Primary vertex finding, high p_T tracks reconstruction and optimal μ-Identification
- Inclusive triggers on 1&2 track signatures.
- Challenge in Run3 is not only to have an efficient trigger, but also be able to identify the topology of events as early as possible in the triggering process: more information than single sub-detector read-out needed
- $\rightarrow$ Track reconstruction at collision rate required : **huge computing** challenge

Reconstruction at collision rate for the LHCb upgrade: 2 TDRs



Trigger TDR (2014)



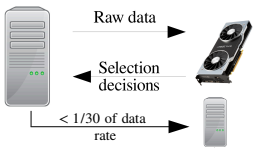
GPU HLT TDR (2020)
Allen project

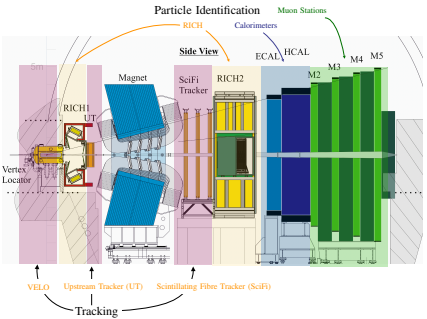
- Both proposals carried out in the last years
- Extensive studies and developments on both architectures
- Brand new algorithms and ideas on pattern recognition developed on both architectures
- **Final decision : use GPUs for HLT1**
- All the work and experience gained for HLT1 reconstruction using CPUs crucial to achieve large speed-up also for the HLT2 reconstruction.
- Benefit of running HLT1 on GPUs :
  1. Reduce network bandwidth between EventBuilder and filter farms
  2. Free up filter farm CPUs for HLT2 only

HLT1 reconstruction: tasks

Highly parallelizable tasks across sizeable set of algorithms

- Full event information copied to GPU (Raw event size 100 kB)
- Process HLT1 at 30 MHz on less than 500 state of the art GPUs.
- Selection reports copied back to CPUs.





Data preparation

- Decode raw data in
  1. VErtex LOcator (VELO)
  2. Upstream Tracker (UT)
  3. Scintillating Fibre Tracker (Sci-Fi)
  4. Muon chambers
- Clustering of VELO pixels into hits

Reconstruction

- Velo tracks reconstruction
- Primary Vertex reconstruction
- Add UT hits to Velo tracks
- Find matching segments in Sci-Fi
- Match tracks to Muon hits
- Make 2-track secondary vertices
- Fit tracks with a (fast) Kalman Filter

Selection

- 1-track selections
- 2-track selections

HLT1 reconstruction on GPUs: parallelization using GPUs

Efficient parallelization can be achieved

- Repeating the same *kernel* or function thousands of times: parallelize intra-event reconstruction.
- Linearize algorithms and algorithm workflows as much as possible
- Organize and redesign data structures in a parallel friendly way for the algorithm purpose
- Pipeline the HLT1 reconstruction in parallel across thousands of events

Raw data decoding in Velo, SciFi, UT, Muon

- Decode binary information from subdetector readout: *parallelize* across readout units and/or sensors.

VELO pixels clustering

- Parallelize across small detector units.

Track reconstruction

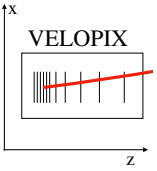
- Pattern Recognition: assign/add hits to a track candidate, *parallelize* across hit combinations
- Track fit: *parallelize* across tracks

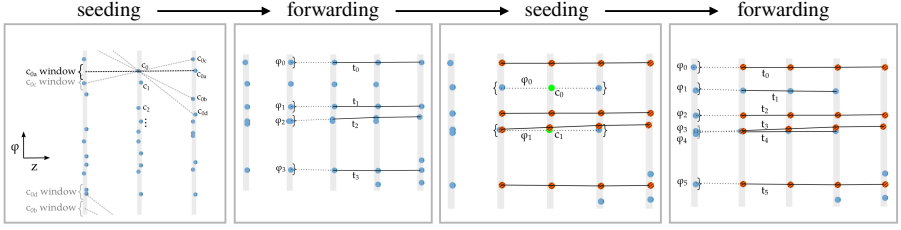
Vertexing

- Combine tracks to form primary and secondary vertices. *parallelize* across tracks and vertex seeds.

HLT1 reconstruction on GPUs: Velo tracking

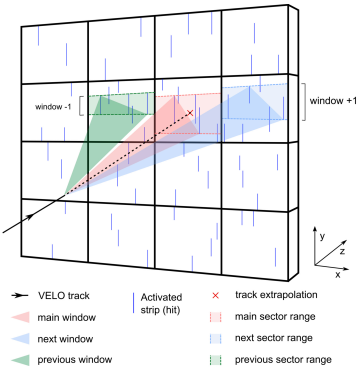
- Velo region has $\overrightarrow{B} = (0, 0, 0)$.
- VELO tracks: straight lines in bending and non-bending plane $\rightarrow \sim$ constant ϕ angle as a function of z
- Search for combinations of hits in parallel
- *Seeding* : Iterate over all possible triplets of VELO modules
- Choice of triplets based on alignment in ϕ
- *Forwarding*: Forward triplet to next layer.
- Algorithm interleaves seeding with forwarding to maximize spatial and temporal locality.

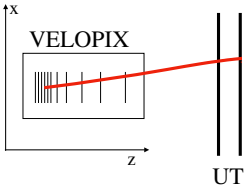




D D. Campora, N. Neufeld, A. Riscos Núñez: "A fast local algorithm for track reconstruction on parallel architectures", IPDPSW 2019

HLT1 reconstruction on GPUs: VELO-UT tracking





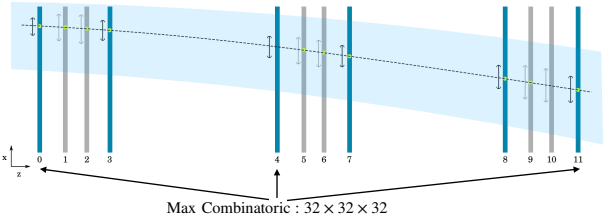
- Find hits in the UT tracker (4 layers) matching the Velo input tracks projections after *small* magnetic field bending.
- Define search regions in each UT plane: hits are stored in sector ranges and optimized for parallel processing.
- Tracklets finding inside windows from the 4 layers building combinatorics in parallel

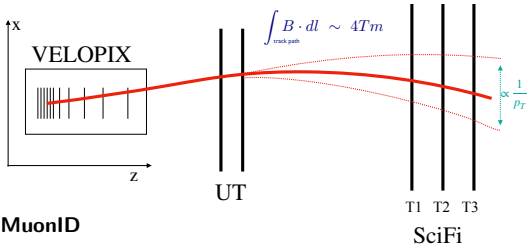
P. Fernandez Declara, D. Campora Perez, J. Garcia-Blas, D. vom Bruch, J. Daniel Garca, N. Neufeld , IEEE Access 7 (2019)

HLT1 reconstruction on GPUs: SciFi tracking and Muon ID

SciFi tracking

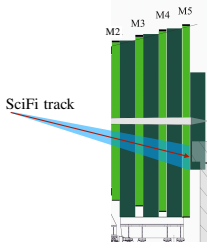
- Extrapolate each Velo-UT track in the 12 layers of the SciFi detector
- Build *triplets* combinations using T1/2/3.
- Best *triplets* selected according to local parameterization of magnetic field[1]
- Forward all triplet(s) to remaining layers with an extra parameterized corrections in the non-bending plane.





MuonID

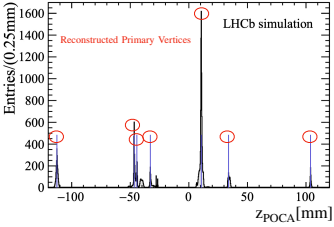
- Project tracks to MWPC muon stations
- Find hits inside the FoI for $\mu - ID$



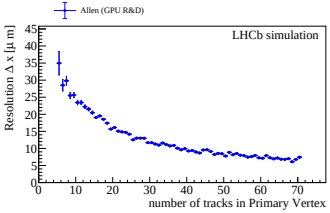
[1] Similarly to arXiv:2007.02591

HLT1 reconstruction on GPUs: PV finding

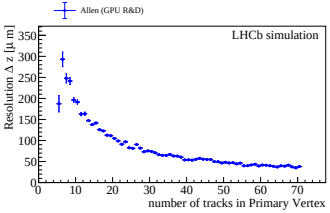
- Algorithm developed for CPU and ported to GPU: based on histogramming along beam-axis.
- Extrapolate in parallel each VELO tracks to the point of closest approach to beamline (z_{POCA})
- Caching covariance matrix at that position: avoid updating it
- Histogram filling for each track in parallel and take uncertainties into account using Gaussian densities
- Peak finding and vertex fitting



PV finding histogram



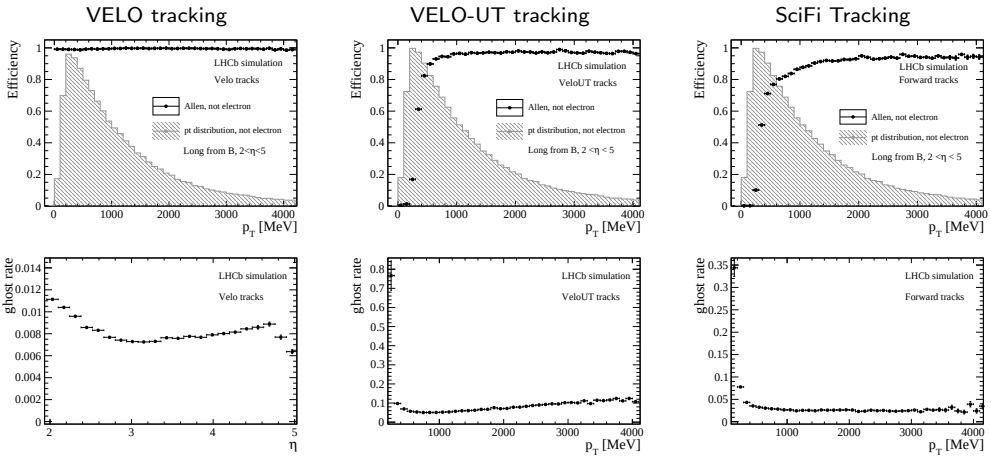
$\sigma_x(PV)$



$\sigma_z(PV)$

LHCB-FIGURE-2020-005

HLT1 physics performance: Track reconstruction efficiencies

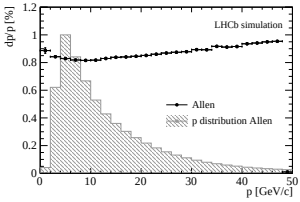


- Tracking down to 0 p_T would cost 20% extra in GPU resources.

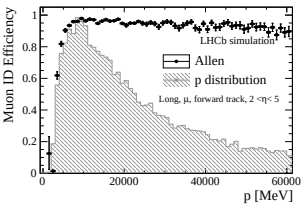
LHCb-FIGURE-2020-014

HLT1 physics performance: Resolution, PV & Muon ID

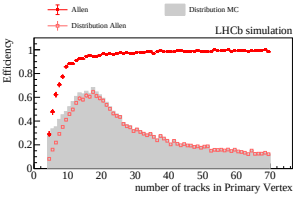
Momentum resolution



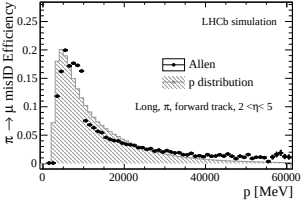
Muon ID efficiency



Primary Vertex reconstruction efficiency



$\pi \to \mu$ mis-ID efficiency



LHCb-FIGURE-2020-014

HLT1 physics performance: Selections

Trigger	Rate [kHz]
ErrorEvent	0 ± 0
PassThrough	30000 ± 0
NoBeams	5 ± 3
BeamOne	18 ± 5
BeamTwo	8 ± 3
BothBeams	4 ± 2
ODINNoBias	0 ± 0
ODINLumi	1 ± 1
GECPassthrough	27822 ± 52
VeloMicroBias	26 ± 6
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

LHCb-FIGURE-2020-014

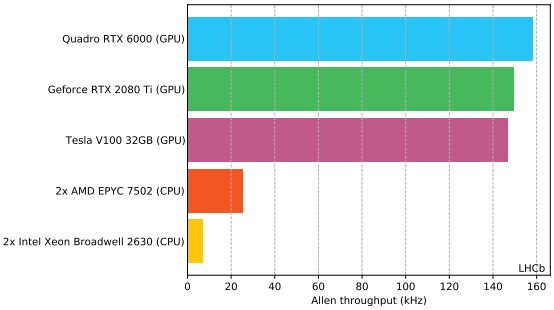
- From 30 $\mathrm{MHz} \rightarrow 1\,\mathrm{MHz}$ event rate reduction
- Can execute $\mathcal{O}(100)$ lines with almost no effect on throughput
- Selection efficiencies fulfill HLT1 requirements for broad range of decays of interest for LHCb

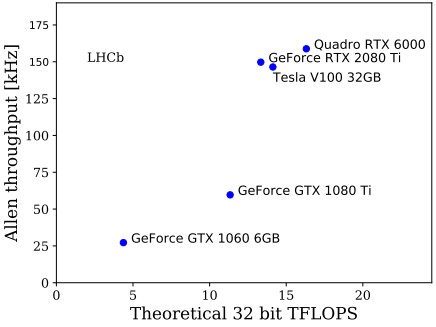
Signal	GEC [%]	TIS-OR-TOS [%]	TOS [%]	GEC × TOS [%]
$B^0 \rightarrow K^{*0}\mu\mu$	89±2	91±2	89±2	79± 3
$B^0 \rightarrow K^{*0}ee$	84±2	69±2	62±2	52± 3
$B_s^0 \rightarrow \phi\phi$	83±3	76±3	69±3	57± 3
$D_s^+ \rightarrow K^-K^+\pi^+$	82±4	59±5	43±5	35±4
$Z \rightarrow \mu\mu$	78±1	99±0	99±0	77±1

GEC : Global Event Cut, TIS: Trigger Independent of Signal, TOS: Trigger On Signal

- Selections for alignment and monitoring implemented as well
- On going: adding more selections

HLT1 computational performance





- Full HLT1 at 30 MHz input rate can be processed using 215 GPU cards. Available slots are 500.
- Computing performance scales well with GPU generations: improvements expected.
- Room already available to include more algorithms to further expand LHCb capabilities, *e.g. PID*, long-lived track reconstruction, *e* optimized track reconstruction....

LHCb-FIGURE-2020-014

Conclusion

Status

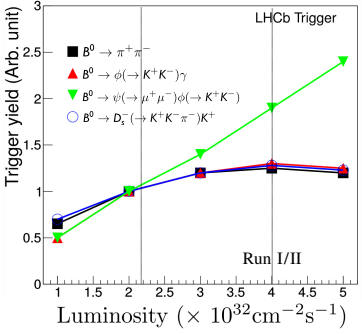
- LHCb is almost ready to face the MHz signal era, changing the trigger paradigm
- From background rejection $\rightarrow$ signal selection and characterization
- Event rate reduction $\rightarrow$ bandwidth reduction (see backup)
- Major detector and DAQ upgrade to perform offline quality event reconstruction in real time
- Partial event reconstruction (HLT1) at 30 MHz input rate using GPUs.
- Full event reconstruction (HLT2) at 1 MHz input rate on CPUs.
- Selective persistency developed for Run II will be used in the upgrade as baseline (see talk from Victor R.)

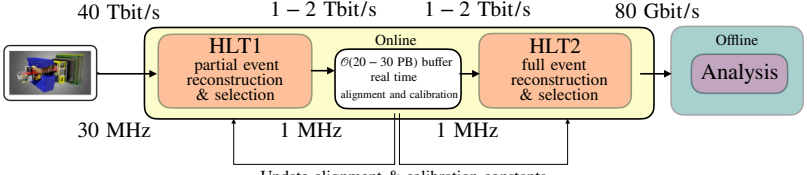
Current developments

- Improve computing performance for HLT2 reconstruction
- Implementation of physics selections for both HLT1 & HLT2
- Get ready for commissioning
- Possibly expand HLT1 reconstruction content using GPUs with great benefit for the LHCb upgrade physics program

LHCb trigger strategy for the upgrade

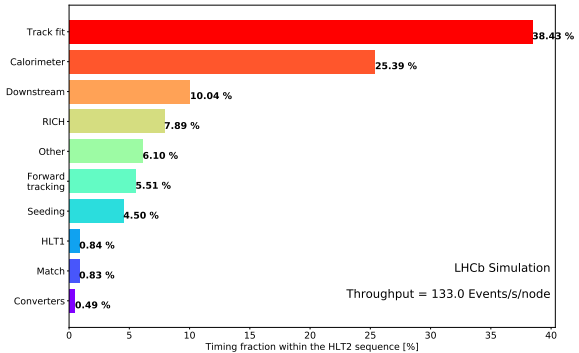
- L0 Hardware trigger output rate of 1 MHz imposed by read-out system fully saturates already in Run 2.
  [Higher rate → higher $p_T^{L0}(\mu)/E_T^{L0}(h^{\pm}/e^{\pm})$ cuts to keep 1 MHz]
- → Full event readout at bunch crossing rate
- → Event reconstruction and triggering in real time
- → Upgrade and replacement of subsystems
  - Cope with higher occupancy
  - Faster/higher precision tracking
  - Full replace of DAQ to support 40 MHz detector read-out
- **LHCb upgrade trigger strategy**: full software based trigger at 30 MHz (non-empty bunch crossing collision rate)





HLT2 reconstruction: tasks

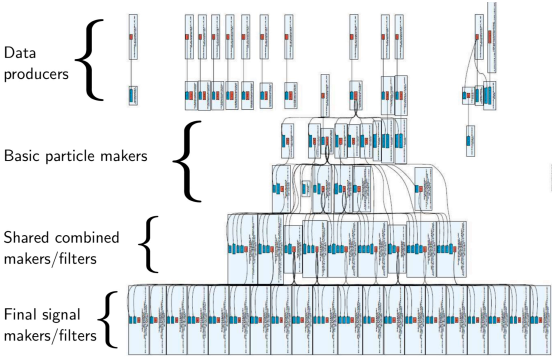
- Using a fully aligned and calibrated detector. More on calibration and alignment in *talk* from Arantza O.
- Offline quality track fit and Particle Identification at 1 MHz input rate
- Knowledge aquired on speeding up CPU solution for HLT1 ported into HLT2



LHCB-FIGURE-2020-007

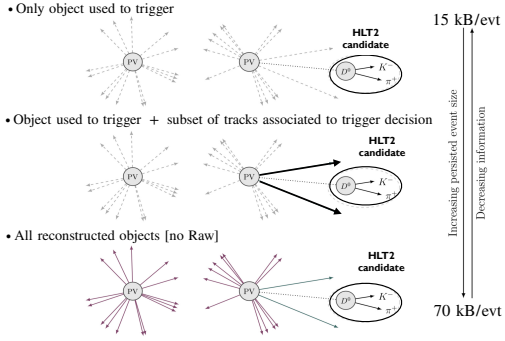
HLT2 selections: the real time analysis paradigm

- Using a fully aligned and calibrated detector. See *talk* from Arantza Oyanguren
- Offline quality track fit and Particle Identification at 1 MHz input rate
- Knowledge aquired on speeding up CPU solution for HLT1 ported into HLT2
- Build offline-like candidates in the online system and perform analysis on direct trigger output.



LHCB-TDR-018, Computing Model TDR

Selective persistency: what is saved to disk?



Extrapolated throughput to tape during the upgrade

STREAM	rate fraction	throughput (GB/s)	bandwidth fraction
FULL	20%	5.9	59%
Turbo	68%	2.5	25%
TurCal	6%	1.6	16 %
Total	100%	10	100%

Bandwidth optimization : Trigger output rate [kHz] $\times$ $\overline{\text{event size}}$ [kB] crucial for final storage [up to 80 Gbit/s].

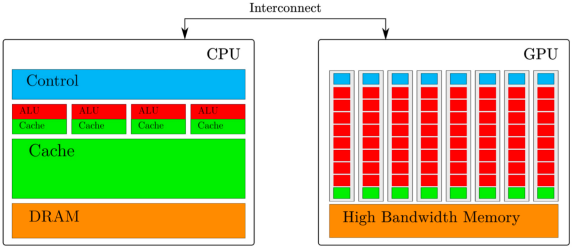
- Offline quality *flexible*-selections available in online system. See *talk* from Victor R.
- Choose what to store to disk to optimize bandwidth.
- Reduced event format and size $\rightarrow$ keep high signal efficiency using the same bandwidth.
- **R**eal **T**ime **A**nalysis concept implemented in Run 2 with `Turbo stream` becomes the baseline in Run 3.

LHCB-TDR-018, Computing Model TDR

GPU architecture design

Interconnect between CPU and GPU

- PCIe 3.0: up to 16 GB/s
- PCIe 4.0: up to 32 GB/s



- Avg bandwidth between CPU and host memory
- Low core count/Powerful ALU
- Complex control unit
- Large caches

Latency

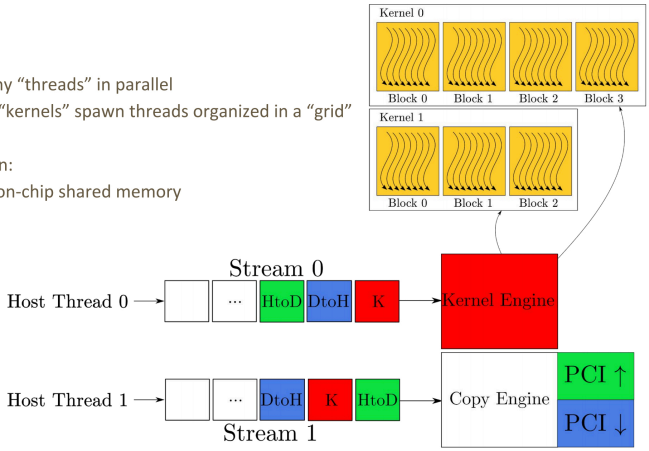
- High bandwidth between GPU cores and GPU memory
- High core count
- No complex control unit
- Small caches

Throughput

Slide taken from here

GPU programming model

- GPU code is executed by many “threads” in parallel
  - Parallel functions, aka “kernels” spawn threads organized in a “grid” of blocks
- Threads in the same block can:
  - Communicate via fast on-chip shared memory
  - Synchronize



Slide taken from here