Triggerless readout:
How to save computing resources by using more of them

Conor Fitzpatrick

ECHEP workshop
University of Edinburgh

UK Research and Innovation

Why triggerless readout?

- This talk is based on (personal) experience with the LHCb trigger
  - The first LHCb upgrade is happening now, and is of a similar scale to that faced by ATLAS/CMS in LS3.
  - To meet the rise in data rates LHCb went with a *Fully software trigger* operating at the LHC bunch crossing frequency (*triggerless readout*)
  - This had significant advantages in terms of maximising online and offline compute efficiency and cost-effectiveness
- I will discuss the reasons for this decision and our experiences doing so
- **Caterina** will discuss one big advantage (reduced data formats) in the next talk
- **Mark** has discussed already how for future upgrades we have to go beyond optimisation.
- LHCb has a very different design and approached things differently as a result- not everything here may be useful for other/future experiments but there is some synergy in computing challenges that may be informative.

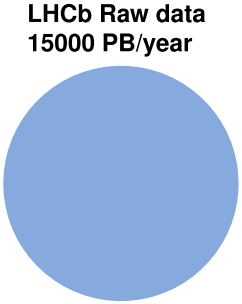
Why Triggerless readout?

- The LHC experiments already operate at high data rates:
  - The LHC collides bunches of protons at 30 MHz
  - At the experiments, each collision is about 100kB (LHCb) - 1MB (ATLAS/CMS)
  - LHC operates for about 5×10^6 seconds/year.

C. Fitzpatrick

Why Triggerless readout?

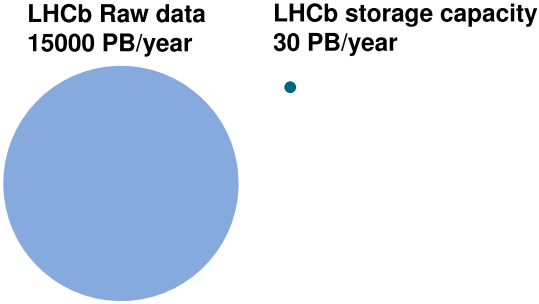
- The LHC experiments already operate at high data rates:
  - The LHC collides bunches of protons at 30 MHz
  - At the experiments, each collision is about 100kB (LHCb) - 1MB (ATLAS/CMS)
  - LHC operates for about 5×10^6 seconds/year.
- Each experiment generates 15-150 exabytes of raw data per year



C. Fitzpatrick

Why Triggerless readout?

- The LHC experiments already operate at high data rates:
  - The LHC collides bunches of protons at 30 MHz
  - At the experiments, each collision is about 100kB (LHCb) - 1MB (ATLAS/CMS)
  - LHC operates for about 5×10^6 seconds/year.
- Each experiment generates 15-150 exabytes of raw data per year

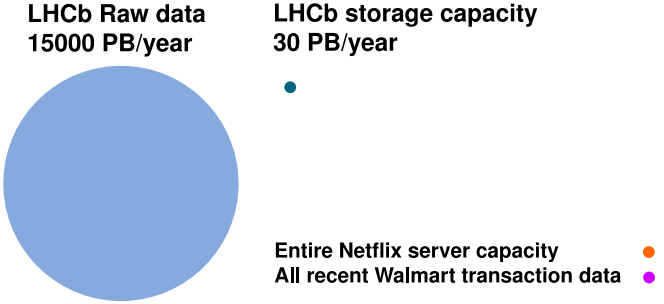


- Storage is limited to tens of PB / year

C. Fitzpatrick

Why Triggerless readout?

- The LHC experiments already operate at high data rates:
  - The LHC collides bunches of protons at 30 MHz
  - At the experiments, each collision is about 100kB (LHCb) - 1MB (ATLAS/CMS)
  - LHC operates for about 5×10^6 seconds/year.
- Each experiment generates 15-150 exabytes of raw data per year

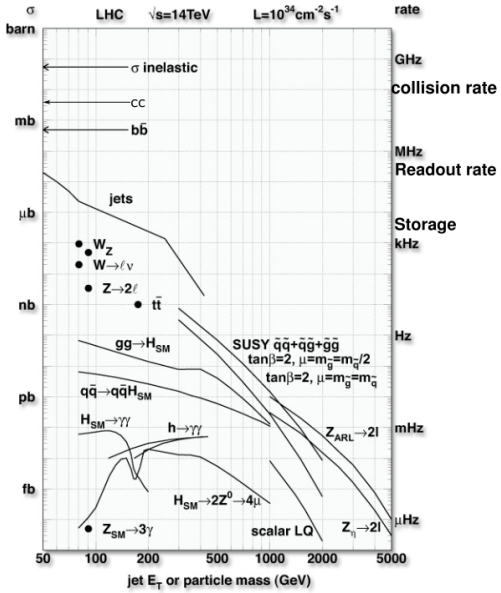


- Storage is limited to tens of PB / year
- LHC experiments have **similar storage requirements to fortune 500 companies**

C. Fitzpatrick

The LHCb trigger

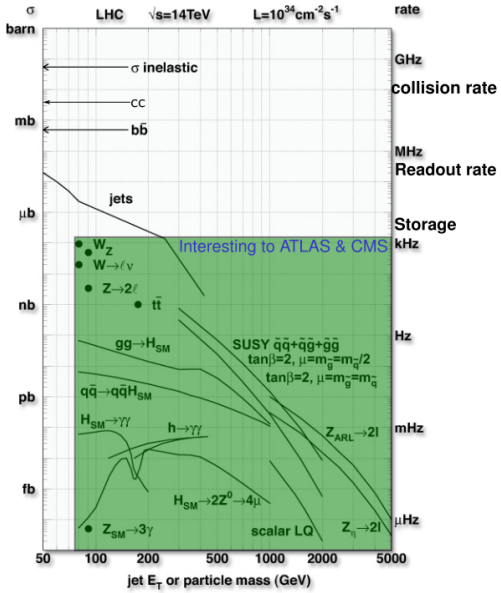
- Traditionally, a trigger is needed to reduce storage and readout costs
- A *good* trigger does so by keeping more signal than background



C. Fitzpatrick

The LHCb trigger

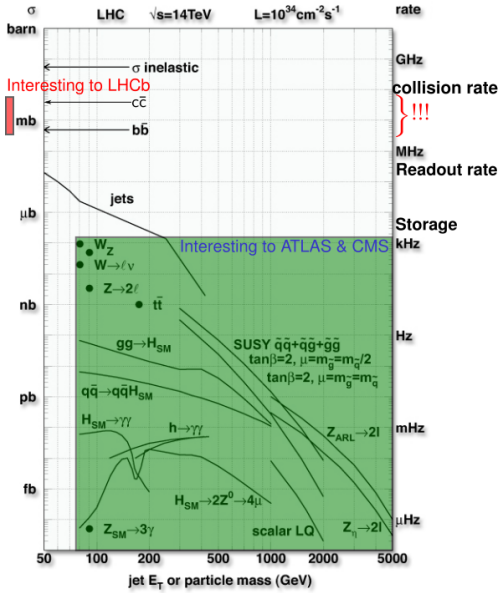
- Traditionally, a trigger is needed to reduce storage and readout costs
- A *good* trigger does so by keeping more signal than background
- ATLAS/CMS are interested in signatures in the kHz region
  - Readout at 100kHz is efficient with reasonably straightforward E_T local requirements



C. Fitzpatrick

The LHCb trigger

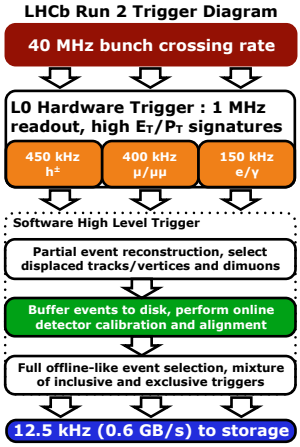
- Traditionally, a trigger is needed to reduce storage and readout costs
- A *good* trigger does so by keeping more signal than background
- ATLAS/CMS are interested in signatures in the kHz region
  - Readout at 100kHz is efficient with reasonably straightforward E_T local requirements
- LHCb faces a unique challenge addressed in Runs 1&2 with:
  - Lower luminosity running
  - 1 MHz readout rate after L0



C. Fitzpatrick

February 18, 2020
UK Research and Innovation

The Run 2 LHCb Trigger



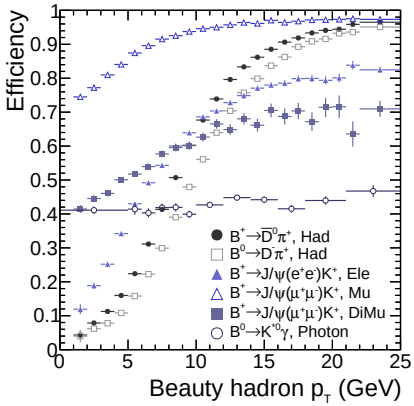
- The LHCb Run 2 trigger (2015-2019)
- Three trigger levels, with a hardware L0 stage:
  - Level-0 trigger buys time to readout the detector with Calo, Muon p_T thresholds: $40 \rightarrow 1$MHz
  - Events built at 1MHz, sent to HLT farm (~ 27000 physical cores)
  - HLT1 has $40\times$ more time, fast tracking followed by inclusive selections $1\text{MHz} \rightarrow 100\text{kHz}$
  - HLT2 has $400\times$ more time than L0: Full event reconstruction, inclusive + exclusive selections using whole detector
- Flexibility comes from software-centric HLT design[1]

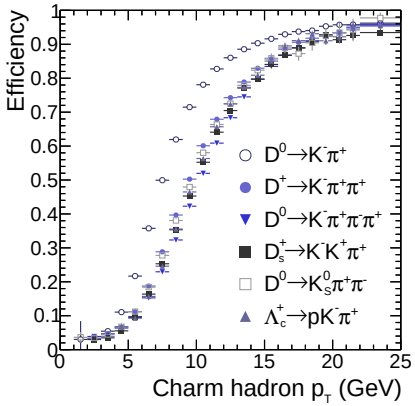
[1] JINST 14 (2019) P04013

C. Fitzpatrick

Level 0

- L0 Uses simple, localised signatures: Transverse energy/momentum thresholds in the muon and calorimeter systems



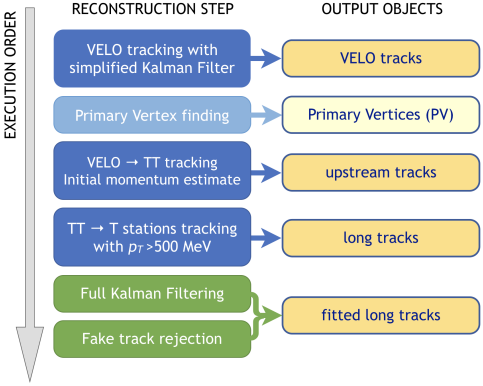


- Genetic algorithm-based bandwidth division balances signal efficiency across entire physics programme within 1 MHz output.
- Typically 40-60% efficient for hadronic beauty 10-30% charm, 90% efficient for muon signatures

C. Fitzpatrick

HLT1

- After readout, events are sent to a 27,000 core CPU farm where the full event is available for processing
- HLT1 performs a fast reconstruction to obtain primary vertices and all tracks above $p_T > 500$ MeV
- These are available for 1- and 2- track MVA selections
- Full muon ID applied to fitted long tracks $p_T > 500$MeV, and an additional fast reconstruction recovers muons with $p_T > 80$MeV.

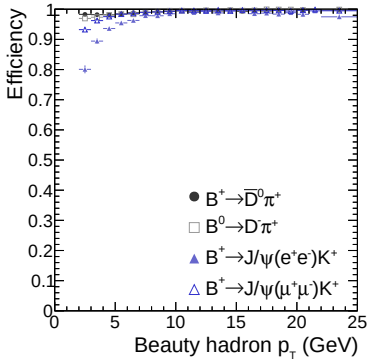


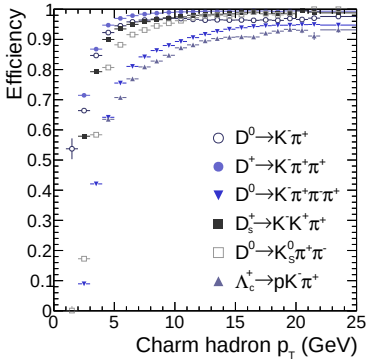
C. Fitzpatrick

February 18, 2020
UK Research and Innovation

HLT1 selections

- Majority of physics at HLT1 selected using 1- and 2- track multivariate algorithms. Rate reduction from 1 MHz $\rightarrow$ 100 kHz:



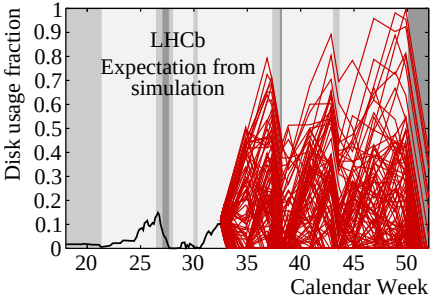


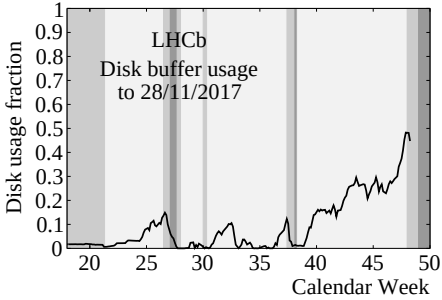
- Extremely efficient ($> 95\%$) for beauty, $70+\%$ efficient for charm

C. Fitzpatrick

Disk Buffer

- HLT Farm: off-the shelf servers, with considerable (11PB) disk capacity
- HLT1 gets written to these disks, allowing HLT2 to run asynchronously. 11PB provides 2 week contingency.





- Effectively doubles trigger CPU capacity, Farm is used twice for HLT, excess used for simulation
- Buffer simulated during data taking, allowing HLT1 output to be tuned
- Asynchronous HLT has another big advantage though...

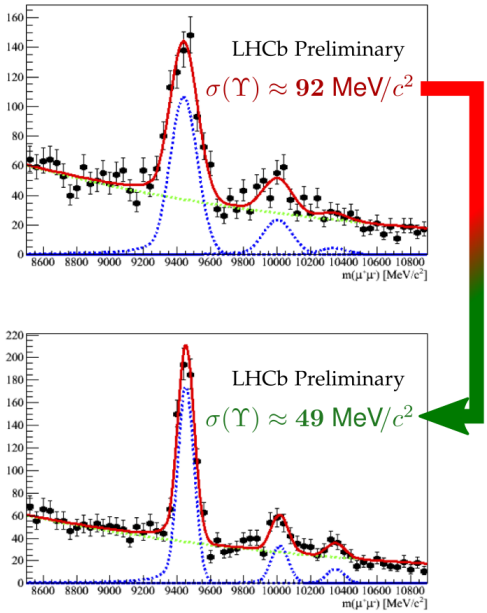
C. Fitzpatrick

February 18, 2020

UK Research and Innovation

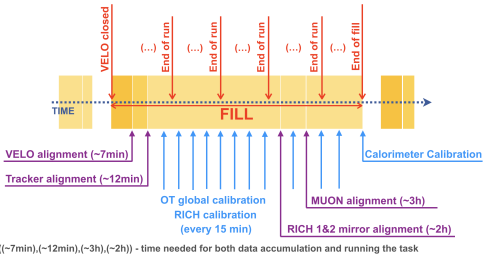
Real-time Alignment + Calibration

- With Run 2 signal rates, efficient & pure output required full reconstruction at HLT2
  - Online selections $\rightarrow$ offline selections
  - Reduces systematic uncertainties and workload for analysts
- Alignment and calibration of full detector in the trigger needed
- While HLT1 is written to disk, alignment & calibration tasks run

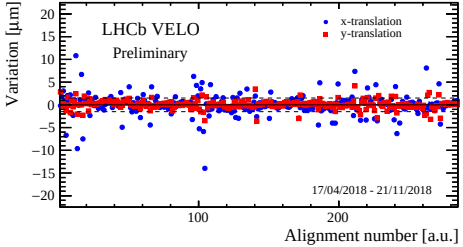


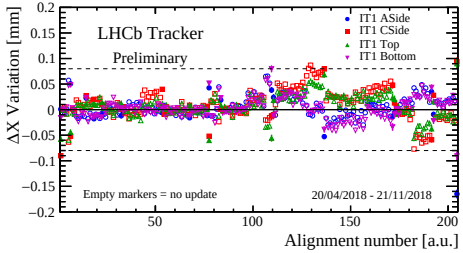
C. Fitzpatrick

A fully aligned detector



- All detectors were aligned & calibrated in-situ using the full HLT1 output rate
- Updates applied automatically if needed prior to HLT2 starting

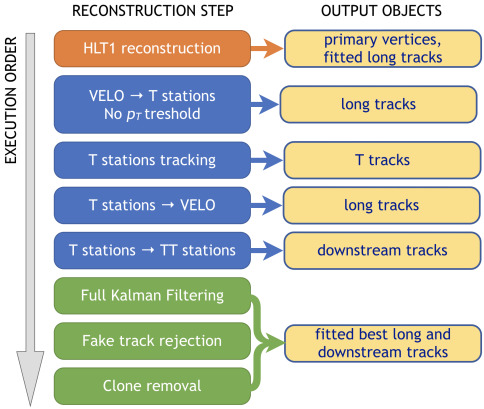




C. Fitzpatrick

HLT 2: Full event reconstruction

- At HLT2 the full reconstruction was performed down to 0 p_T
- Long and downstream tracks are available for physics
- Full Particle ID is available (RICH, MUON, CALO)
- All trigger quantities now 'offline quality' after alignment & calibration
- Several hundred inclusive & exclusive selections, resulting in 6-700MB/s sent offline for analysis

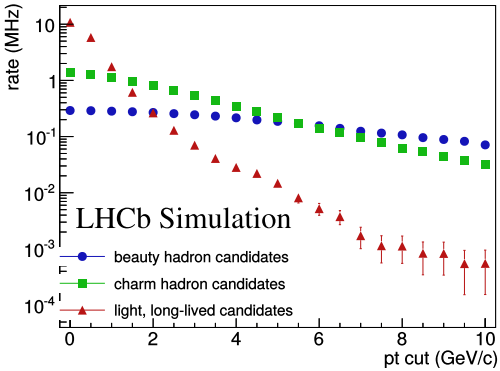


- By definition, HLT2 is $\sim$100% efficient with respect to offline analysis selections because it *is* the offline selection in most cases

C. Fitzpatrick

The MHz signal era

- Starting in 2021, The 'new' LHCb will run at five times the collision rate:



- Even after simple trigger criteria, MHz of signals [2]

[2] LHCb-PUB-2014-027

The MHz signal era

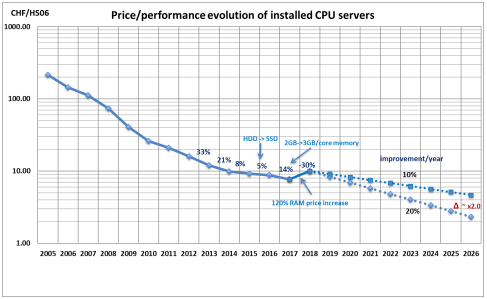
- Starting in 2021, The 'new' LHCb will run at five times the collision rate:

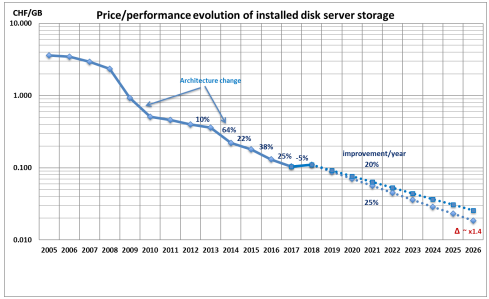
- Even after simple trigger criteria, MHz of signals [2]

[2] LHCb-PUB-2014-027

Can we save ourselves with "Moore's Law"?

- This isn't a problem, right? Just wait a few years and your computing budget will buy twice as much...



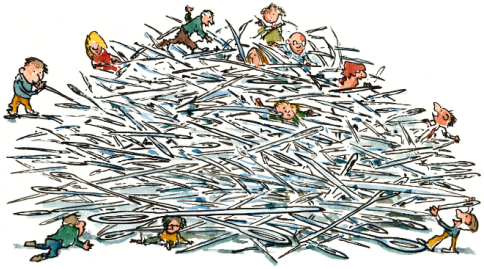


- General trend is one of smaller gains than we have been used to[3]
  - Note: Reasons to be cheerful are the inherently parallel nature of much of our problems. Architectures are moving in this direction.
  - Storage however is also a problem. Rebalancing storage and processing requirements should be considered.

[3] H. Meinhard, RAPID 2018

So what 'stuff' can we throw away?

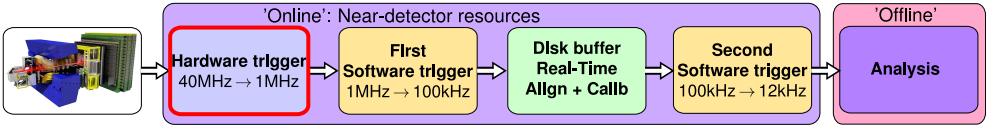
- The problem is no longer one of rejecting (trivial) background
- Fundamentally changes what it means to trigger



- Instead, we need to categorise different 'signals'
  - Requires access to **as much of the event as possible, as early as possible**

C. Fitzpatrick

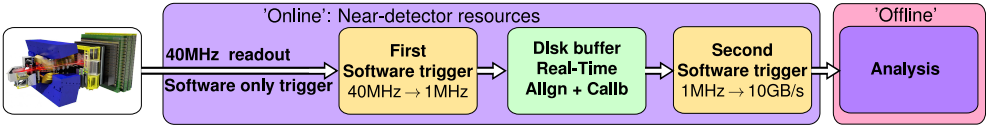
Reading out at 30MHz



- The L0 trigger cannot reduce the rate below the 1 MHz readout limit without being inefficient
- The software triggers are pure: Can use the full event to make the decision

C. Fitzpatrick

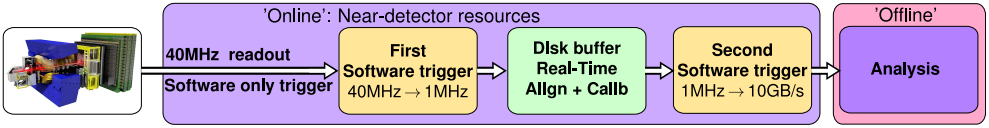
Reading out at 30MHz



- The L0 trigger cannot reduce the rate below the 1 MHz readout limit without being inefficient
- The software triggers are pure: Can use the full event to make the decision
- Solution: Readout and reconstruct 30 MHz of collisions in software

C. Fitzpatrick

Reading out at 30MHz



- The L0 trigger cannot reduce the rate below the 1 MHz readout limit without being inefficient
- The software triggers are pure: Can use the full event to make the decision
- Solution: Readout and reconstruct 30 MHz of collisions in software
  - HLT1 similar to the Run 2 design but now must operate at the 30 MHz visible interaction rate
  - HLT2 input rate increased to 1 MHz and will produce **mostly TLA/Turbo/Scouting output** at 10GB/s

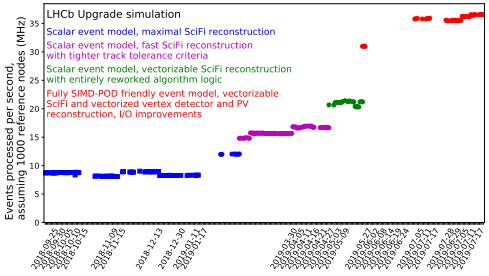
C. Fitzpatrick

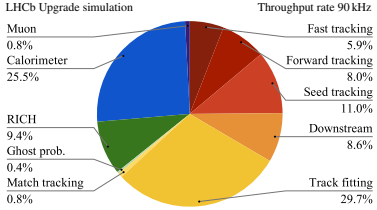
Efficient computing at 30MHz

- There are a couple of distinct advantages to this approach from a computing perspective:
  - **Calibration & Alignment**: MC samples only need made with one set of calibration parameters as the data is always calibrated back to the baseline.
  - **Online reco == Offline reco**: Grid resources originally used for offline reconstruction can be devoted to MC production
  - **Reduced event formats**: No need to keep the raw data for majority of physics analyses, LHCb expects an average reduction of $3\times$ in event size.
- By using more CPU up-front, LHCb can make more efficient use of the entire online+offline computing infrastructure.
- Similarly, 'perfect' is the enemy of 'working'. If you lose permille of tracking resolution and it doubles your throughput, does this kill your physics programme?

Getting to 30MHz with x86

- HLT1 has to process in-fill at 30 MHz
  - This has led to a huge reoptimisation effort in LHCb
  - Gains from moving to modern multithreaded framework and careful refactoring of reconstruction sequence
  - **This taught us a lot**
    - We are not particularly memory hungry
    - While we started with many hotspots, the optimised resonstruction doesn't really have a single 'intractible' problem
    - Code optimised for AVX/GPU can be ported reasonably easily to GPU/AVX

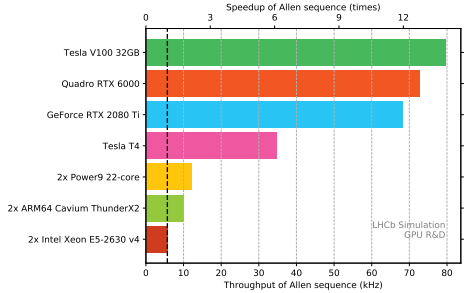


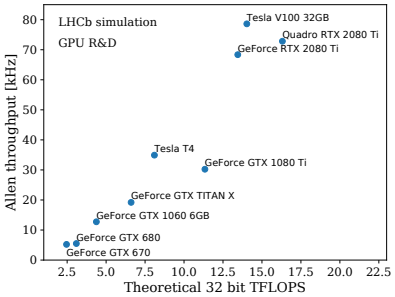


C. Fitzpatrick

Getting to 30MHz with GPUs

- The Allen project is dedicated R&D into a GPU-based HLT1 for LHCb[4]
- Allen works because the **entire HLT1 sequence can be run on cost-effective GPUs**



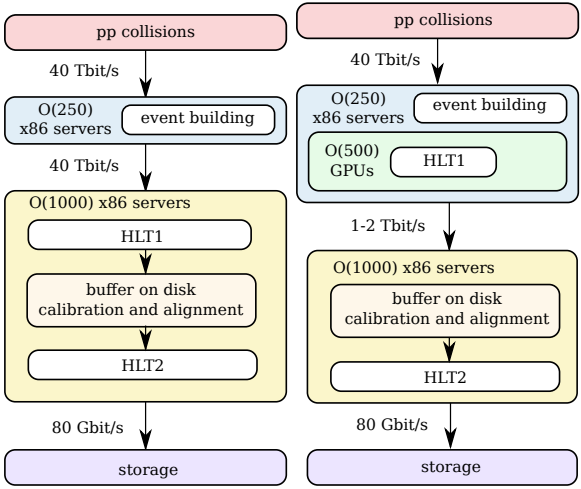


- Personal opinion: GPUs work here because they can run the entire sequence and make a decision/reduction. More specialised/less general purpose hardware may not have the same cost/benefit when including interconnects, network etc.

[4] arXiv:1912.09161

A fully triggerless readout

- The online network for LHCb consists of Event Builder (EB) and Event Filter (EF) nodes.
- GPU and CPU based HLT1 can be integrated as follows:
  - Baseline: HLT1 & HLT2 run asynchronously on CPU event filters
  - HLT1 runs on GPU cards in EB nodes. **Reduced network requirements between EB and EF is cost effective**



C. Fitzpatrick

Conclusions 1

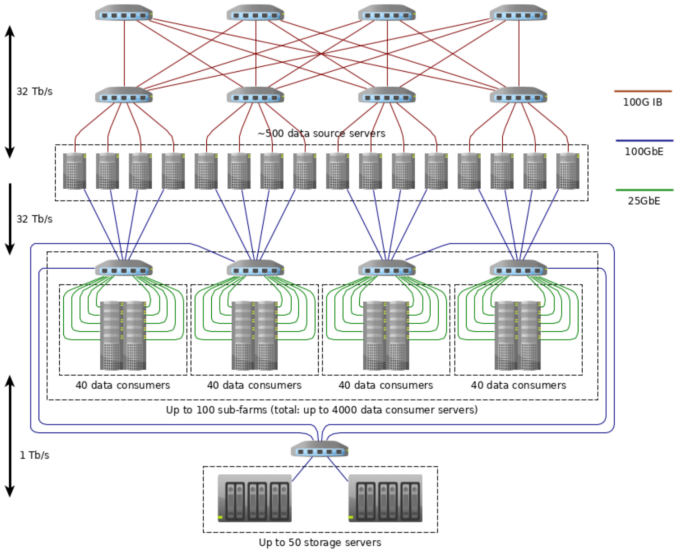
- In order to efficiently categorise MHz signals, LHCb will use a triggerless readout into a software trigger
- Offline quality selections mean only subset of the event has to be saved for analysis
  - Requires fully aligned & calibrated detector in the trigger
- This paradigm allows LHCb to do **More Physics** with **Less (global) resources**

Conclusions 2

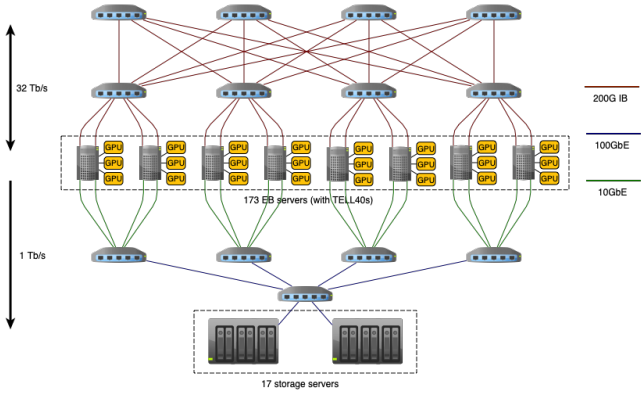
- What did LHCb learn from this process that may help inform HL-LHC upgrades?
  - While going functional/multithreaded led to some performance gains,
  - The big improvements came from dedicated vectorisation of reconstruction algorithms
  - This led to cross-pollination of performance gains in both GPU and CPU implementations.
  - Experience with more dedicated co-processors (FPGA) so far have shown unless they do a *lot* of work the infrastructure surrounding them makes them less cost effective.
- **A dedicated global optimisation of online + offline CPU and storage results in interesting and efficient design choices.**
- LHCb has another upgrade on the horizon and will need to revisit this optimisation again: We hope to learn from the ALTAS/CMS HL-LHC upgrade experiences as they unfold.

Backups

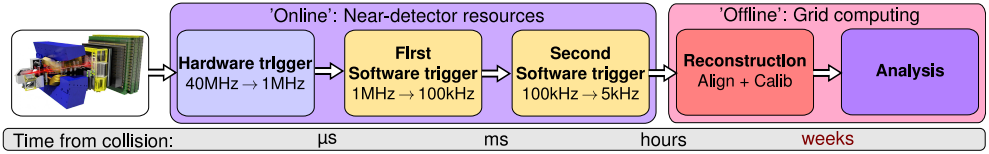
DAQ network, CPU implementation



DAQ network, GPU implementation

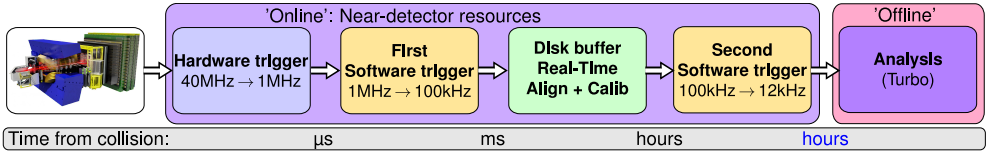


HLT2: Reduced event formats



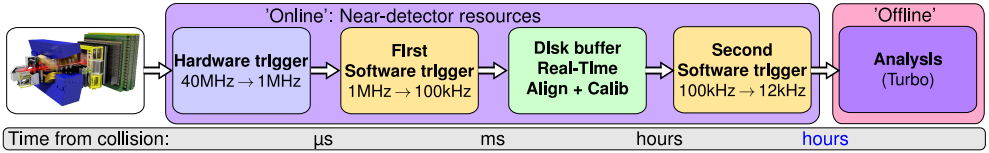
- Trigger *rates* aren't important, output *bandwidth* is
- Offline reprocessing previously needed to recover best quality

HLT2: Reduced event formats



- Trigger *rates* aren't important, output *bandwidth* is
- Offline reprocessing previously needed to recover best quality
- After alignment: online == offline, why reprocess? Do analysis on trigger objects at HLT2, write only the relevant objects offline
- Significant reduction in event size $\rightarrow$ higher rates for the same bandwidth

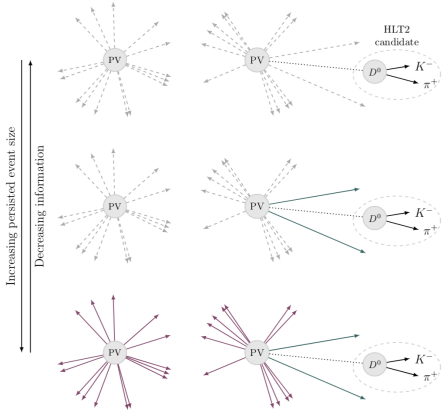
HLT2: Reduced event formats



- Trigger *rates* aren't important, output *bandwidth* is
- Offline reprocessing previously needed to recover best quality
- After alignment: online == offline, why reprocess? Do analysis on trigger objects at HLT2, write only the relevant objects offline
- Significant reduction in event size → higher rates for the same bandwidth
- Added bonus: offline CPU freed up for simulation.

Turbo

- Turbo is LHCb's Real-Time Analysis paradigm for reduced event format data[5]



- High degree of flexibility: Save only as much of the event as is needed for analysis
  - Keep all reconstructed objects, drop the raw event: < 100kB
  - Keep only objects used to trigger: 7kB
  - 'Selective Persistence' objects used to trigger + user-defined selection: $7 \rightarrow 100$kB

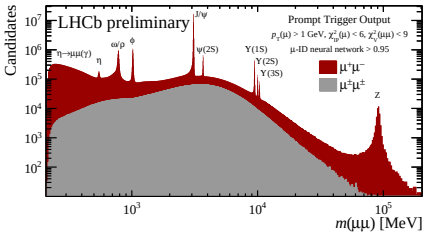
[5] arXiv:1604.05596, JINST 14 (2019) P04006

Turbo usage in Run 2

- 528 trigger lines at HLT2. 50% are Turbo
- 25% of the trigger rate is Turbo but it counts for only 10% of the bandwidth
- Many analyses would not be possible without Turbo[6]

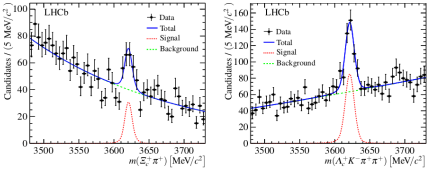
CERN-EP-2017-248
LHCb-PAPER-2017-038
October 5, 2017

Search for dark photons produced in 13 TeV *pp* collisions



CERN-EP-2018-172
LHCb-PAPER-2018-026
October 18, 2018

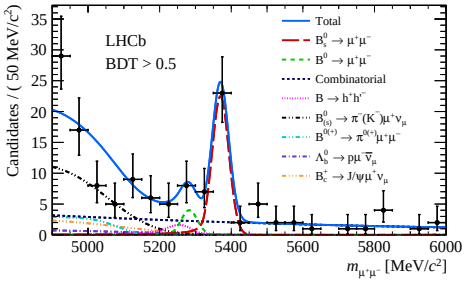
First observation of the doubly charmed baryon decay $\Xi_{cc}^{++} \to \Xi_c^+ \pi^+$

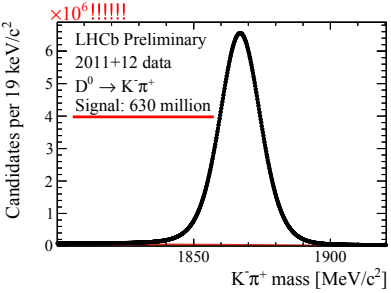


[6] Phys. Rev. Lett. 120, 061801 (2018), Phys. Rev. Lett. 121, 162002 (2018)

The LHCb Run 2 trigger in two plots

- The LHCb trigger had to cover extremes of data taking:



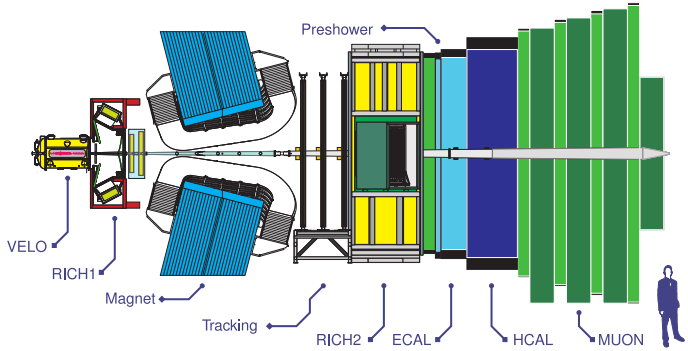


- High efficiency to collect rare decays like $B_s^0 \to \mu\mu$[7]
- High purity for enormous charm signals like $D^0 \to K\pi$[8]
- Requires a high degree of flexibility at high data rates

[7] Phys. Rev. Lett. 118, 191801 (2017)
[8] LHCb-CONF-2016-005

LHCb: The precision flavour experiment

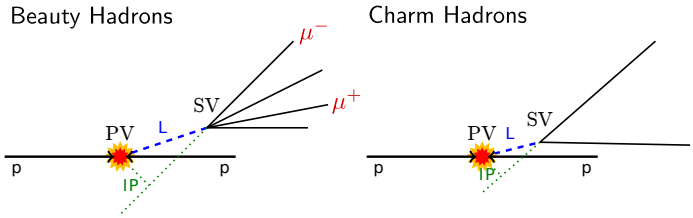
- LHCb was built to exploit the high rates of beauty and charm at the LHC:



- Precise particle identification (RICH + MUON)
- Excellent decay time resolution: $\sim$45fs (VELO)
- High purity + Efficiency with flexible trigger

Signatures

- Typical beauty and charm decay topologies:



- $B^{\pm}$ mass $\sim$5.28 GeV, daughter p_T $\mathcal{O}$(1 GeV)
- $\tau \sim 1.6$ ps, Flight distance $\sim$1 cm
- Important signature: Detached muons from $B \to J/\psi X$, $J/\psi \to \mu\mu$

- D^0 mass $\sim$1.86 GeV, appreciable daughter p_T
- $\tau \sim 0.4$ ps, Flight distance $\sim$4 mm
- Also produced as 'secondary' charm from B decays.

Underlying Trigger strategy:

- Readout based on simple L0 critera, Fast reconstruction at HLT1: Primary Vertices, High p_T tracks, optional Muon ID, Exclusive and inclusive selections at HLT2 with full reconstruction