LHCb Run 3 Computing Model

Marco.Cattaneo@cern.ch
• Single arm forward spectrometer
• Measure properties of known (beauty and charm) particles as precisely as possible
• Search for evidence of new physics by looking for deviations from Standard Model predictions
• Low instantaneous luminosity compared to ATLAS and CMS ($4 \times 10^{32} \text{ cm}^{-2} \text{s}^{-1}$)
## LHC schedule

### What we expect for Run III and Run IV:
- Collect 50 fb\(^{-1}\) at 14 TeV
- Higher luminosity: 0.4 \(10^{33}\) cm\(^{-2}\) s\(^{-1}\) → 2 \(10^{33}\) cm\(^{-2}\) s\(^{-1}\)
- More interactions per beam crossing: \(\mu = 1.1 \rightarrow \mu = 7.6\)
- Detector and trigger have to be adapted to cope with the new conditions

### LHCb expectation: for Run III and Run IV: collect 50 fb-1 at 14 TeV

<table>
<thead>
<tr>
<th>Year</th>
<th>Run III</th>
<th>Run IV</th>
<th>Run V</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019</td>
<td>LS2</td>
<td>LHCb 40 MHz UPGRADE</td>
<td>ATLAS Phase I Upgr</td>
</tr>
<tr>
<td>2020</td>
<td></td>
<td>LHCb Consolidation</td>
<td>ATLAS Phase II UPGRADE</td>
</tr>
<tr>
<td>2021</td>
<td></td>
<td>L = 2 (10^{33})</td>
<td>ATLAS</td>
</tr>
<tr>
<td>2022</td>
<td></td>
<td>300 fb(^{-1})</td>
<td>HL-LHC</td>
</tr>
<tr>
<td>2023</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td>CMS</td>
</tr>
<tr>
<td>2024</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2025</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2026</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2027</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2028</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2029</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2030</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2031</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2032</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
<tr>
<td>2033+</td>
<td></td>
<td>CMS Phase II UPGRADE</td>
<td></td>
</tr>
</tbody>
</table>

- LHCb 40 MHz UPGRADE: \(L = 2 \times 10^{33}\) cm\(^{-2}\) s\(^{-1}\)
- ATLAS Phase I Upgr: \(L = 2 \times 10^{34}\) cm\(^{-2}\) s\(^{-1}\)
- ATLAS Phase II UPGRADE: \(L = 5 \times 10^{34}\) cm\(^{-2}\) s\(^{-1}\)
- ATLAS HL-LHC: \(L = 5 \times 10^{34}\) cm\(^{-2}\) s\(^{-1}\)
- CMS Phase I Upgr: \(300 fb^{-1}\)
- CMS Phase II UPGRADE: \(50 ab^{-1}\)
- CMS HL-LHC: \(3000 fb^{-1}\)

### Belle II:
- 5 ab\(^{-1}\)
- \(L = 8 \times 10^{35}\) cm\(^{-2}\) s\(^{-1}\)
- 50 ab\(^{-1}\)
The upgraded LHCb detector for Run 3

New mirrors and photon detectors
HPDs $\rightarrow$ MAPMTs

New silicon tracker

New vertex locator
silicon strips $\rightarrow$ pixels

New scintillating fibre tracker

New readout electronics for the entire detector

Remove hardware trigger
The upgraded LHCb detector for Run 3

Detector Channels

R/O Electronics

To be kept

To be UPGRADED

New readout electronics for the entire detector

New scintillating fibre tracker

DAQ
24% (2%) of the beam crossings contain a charm (beauty) hadron
In addition to separating signal and background, trigger means also signal categorization
Run 3 will change the definition of trigger: no longer "trivial" background rejection. We will need to effectively separate high statistics signals.

The exploitation of the physics programme of the LHCb upgrade implies
- removing the L0 hardware trigger (output rate limited at 1MHz)
- deploying a full software high level trigger (HLT) with the goal of sustaining triggering at the 30MHz p-p inelastic collision rate
- Performing analysis directly on trigger output.
  - Real-time data analysis requires the best performing reconstruction achieved online
Processing in Run2 and Run3

- The rationale: data processing in Run 3 is based on concepts that were already successfully implemented in Run 2
  - Split HLT with synchronous HLT1 and asynchronous HLT2
  - Real-time alignment and calibrations
  - TURBO stream and selective persistency for real time analysis
- Offline reconstruction of real data in Run 3 will be very limited
- The challenge in Run 3 triggering is also a challenge in event reconstruction
Processing in Run2 and Run3

• The rationale: data processing in Run 3 is based on concepts that were already successfully implemented in Run 2
  • Split HLT with synchronous HLT1 and asynchronous HLT2
  • Real-time alignment and calibrations
  • TURBO stream and selective persistency for real time analysis
• Offline reconstruction of real data in Run 3 will be very limited
• The challenge in Run 3 triggering is also a challenge in event reconstruction

November 5th 2018  M.Cattaneo -- Quantum Computing for HEP Workshop
The road to the Run3 upgrade

- Originally, full event reconstruction up-front
- Strong constraints from available CPU resources and budget

**Partial reconstruction in HLT1**
- Data preparation for tracking and fast track reconstruction
- Rate reduction to 0.5-1 MHz

**Alignment and calibration as in Run2**

**Full event reconstruction in HLT2**
- Best tracking performance
- Particle identification
- Offline-quality selections
Real data flow in LHCb Upgrade

LHC : 30 MHz @ $2 \cdot 10^{33}$

DETECTOR READOUT

HLT1 PARTIAL RECO

Buffer

Real-time alignment and calibrations

HLT2 FULL RECO

X% FULL

Offline reconstruction and associated processing

Y% TURBO & real-time analysis

User analysis

Z% CALIB

Offline reconstruction and associated processing
Alignment and calibration in 2018

• All alignments and calibrations are automated and run in real time
Why do we need online alignment and calibration?

- Better mass resolution

- Better particle identification (PID)

- Store less background → more bandwidth for physics!
The LHCb Turbo stream

- Turbo is the LHCb paradigm for reduced event format data
- High degree of flexibility: Save only as much of the event as is needed
  - Keep all reconstructed objects, drop the raw event: 120kB in Run 3
  - Keep only objects used to trigger: 4-5kB (same as Run 2)
  - ‘Selective Persistence’ anything in between
- Selection done in HLT2, **enabled** by analysis quality calibration & alignment
Challenge:

• Factor 30 increase in HLT1 input rate, with increased event complexity (multiple interactions)

• Traditionally, relied on Moore’s law to increase available (sequential) CPU resources at constant cost

• Improving software performance has become the challenge
Software performance: much to gain!

- Evolution trend of faster single-threaded CPU performance broken 10 years ago.
  - Increase of CPU cores and more execution units.

Software performance: much to gain!

- Evolution trend of faster single-threaded CPU performance broken 10 years ago.
  - Increase of CPU cores and more execution units.

- Gaudi core framework has been in production without major modifications for 17 years

- Its sequential event data processing model leads to
  - Weak scalability in RAM usage
  - Inefficient disk/network I/O

---

Trigger decisions vs. power of trigger farm
Software performance: much to gain!

- Modernize Gaudi and make it fit for current and forthcoming challenges

- Angles of attack:
  - Better utilization of current multi-processor CPU architectures
  - Enable code vectorization
  - Modernize data structures
  - Reduce memory usage
  - Optimize cache performance
  - Remove dead code
  - Replace outdated technologies
  - Enable algorithmic optimization
  - Enforce thread safety to enable multi-threading

Trigger decisions vs. power of trigger farm
Multi-threaded Gaudi

- Multi-threaded framework is ready
  - More than 100 algorithms, including the full HLT1 reconstruction part, have been converted
- Huge gain in memory utilization
Vectorization

• **Multi-threaded framework is ready**
  - More than 100 algorithms, including the full HLT1 reconstruction part, have been converted

• **Main guidelines for optimization:**
  - Code **modernization** (C++98 → C++11 → C++14 → C++17)
  - Code **improvements**
  - **Vectorization** (refactoring of data model required), for example:
    - VeLo tracking
    - RICH rays tracking

---

### Table 2.3: Performance of vectorized Rich's Ray Tracing

<table>
<thead>
<tr>
<th></th>
<th>before SOA</th>
<th>after SOA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function / Call Stack</strong></td>
<td>PrPT_bestHit</td>
<td>PrPT_bestHit</td>
</tr>
<tr>
<td>Clockticks</td>
<td>46958000000</td>
<td>26652000000</td>
</tr>
<tr>
<td>Instructions Retired</td>
<td>25760000000</td>
<td>24932000000</td>
</tr>
<tr>
<td>CPI Rate</td>
<td>1.8229</td>
<td>1.06813</td>
</tr>
</tbody>
</table>

| **MEM_LOAD_UOPS_RETIRED.L3_MISS_PS** | 94002820 | 0 |
| **MEM_LOAD_UOPS_L3_MISS_RETIRED.LOC** | 90002700 | 0 |

---

**Table 2.3**

<table>
<thead>
<tr>
<th></th>
<th><strong>SSE4</strong></th>
<th><strong>AVX2</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>time (s)</td>
<td>Speedup</td>
</tr>
<tr>
<td><strong>double</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>scalar</td>
<td>233.462</td>
<td></td>
</tr>
<tr>
<td>vectorized</td>
<td>122.259</td>
<td>1.90</td>
</tr>
<tr>
<td><strong>float</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>scalar</td>
<td>214.451</td>
<td></td>
</tr>
<tr>
<td>vectorized</td>
<td>55.707</td>
<td>3.85</td>
</tr>
</tbody>
</table>

---

**Figure 2.3**: Top CPU consumers in HLT1 once KeyedContainers are gone

The test ran with 16 threads and a total of 10000 events on a machine with 20 physical cores.

The gains are important and are coming from two effects: less contention on memory allocation and less time spent there, but also a better cache efficiency.

---

2.2.3.3 Suppression of news in PrPixel tracking

As described in 2.2.2.4, high rate of small memory allocation in a highly multi-threaded environment can be a limitation and a contention point. One of the main provider of small allocations and scattered memory in the LHCb HLT1 code was found to be the use of KeyedContainer instead of regular STL containers of objects, with proper reservation of memory.

In order to measure the improvements that a cleanup of our data model on this point can bring, the uses of KeyedContainers in the PrPixel algorithm were replaced by STL containers.

**Figure 2.3**: Top CPU consumers in HLT1 once KeyedContainers are gone

The gains are important and are coming from two effects: less contention on memory allocation and less time spent there, but also a better cache efficiency.

**2.2.4 Overall status, expectations and Task list**

Most of the work done so far has been concentrated on the HLT code and even more specifically on the HLT1 subpart of it. All involved algorithms has now been reviewed extensively and a lot of work has already been achieved to optimize them. This includes:

- reviewing most of the code and cleaning up some useless pieces
- cleaning up the HLT1 sequence to remove unneeded algorithms
- profiling extensively to find where we spent more time than expected

---

November 5th 2018

M.Cattaneo -- Quantum Computing for HEP Workshop
Offline Data Processing & User Analysis

• Classic offline data reconstruction and stripping (streaming / skimming / slimming) reduced to bare minimum

• Main data processing workflow is turbo processing
  • i.e. convert online (LHCb specific) to offline (ROOT) format and streaming
  • In Run 2 this turbo workflow accounts for 0.1 % of the grid work

• User analysis will move from individual to centrally organized data selections
  • Possibility to increase I/O by aggregating multiple selections (train model)
Monte Carlo Simulation

- Order of magnitude increase in recorded event rate requires matching increase in number of simulated events

- MUST speed up the simulation
  - By implementing faster or parameterised simulations
  - By reducing the CPU consumption of the full Geant4-based simulation while maintaining high quality physics monitoring

Legend:

“Sim at 50% of data”:
- FullSim sample is 50% of the data size
- FastSim sample is 50% of the data size

FastSim speed assumed to be 1/10 of FullSim
Fast Simulation options

Broad investigation deploying solutions when mature for physics

- **Simplified detector simulation**
  - Reduced detector: RICH-less or tracker-only. **In production**
  - Calorimeter showers fast simulation. **Under development**
  - Muon lower energy background, used with full muon detector simulation. **In production**

- **Simulation of partial event**
  - Simulate only particles from signal decay. **In production**
  - ReDecay, e.g. use N-times the non-signal decay part of the event. **In production**

- **Fully parametric simulation**
  - Parametrized tracking, calorimeter and particleID objects with a DELPHES-based infrastructure. **Under development**

_D.Mullet et al. - ReDecay: A novel approach to speed up the simulation at LHCb_ [https://arxiv.org/abs/1810.10362]

_M. Whitehead, “A palette of fast simulations in LHCb” @ ICHEP 2018_
Fully parametric fast simulation

- Work in progress on a fully parametric ultra-fast simulation based on the DELPHES package
  - Parametrizes not only the detector response but also the reconstruction
- Crucial to cope with large amount of simulated statistics needed for Run3 and future Upgrade II. **Goal: 100-1000x faster then full simulation.**
- Functional prototype integrated in the current Gauss framework
  - **Tracking** efficiency and resolution
  - **Primary vertices** reconstruction
  - **Photon** calorimetric objects
  - Output LHCb reconstructed high level objects, compatible with the experiment analysis tools

---

**J. De Favereau et al., JHEP 02 (2014) 057**
**B. Siddi, “A fully parametric option in the LHCb simulation framework” @ CHEP 2018**

November 5th 2018
Fast simulation of the Calorimeter system

- Two fast parametrization solutions currently under development
  - *Classic* Frozen Shower Libraries
  - Hits generation based on Generative Adversial Networks (GAN)
  - ... not necessarily mutually exclusive 😊. Could solve the Shower Library problem of a fast search in multi-dimensional phase space by reducing the dimensions with Machine Learning techniques, e.g. autoencoders

  *M. Rama, "Calorimeter fast simulation based on hit libraries in the LHCb Gauss framework" @ CHEP 2018*
  *F. Ratnikov, "Fast calorimeter simulation in LHCb" @ ICHEP 2018*

- Aim to speed up by factor 3 to 10 the simulation of the calorimeters
  - Timing study with dummy filling of calorimeter cells shows overall speed of full LHCb detector reduced by a factor of 2
GAN for LHCb Calorimeters

Starting from latest configuration of CaloGAN, a new Machine-Learning method based on a generator, trained to maximize goodness of produced sample, and a discriminator to classify images (in HEP applicable to jets, clusters)

- Very fast response, but generally long training
- First look with simple mock-up of LHCb ECAL and signal particle gun reproduce the shape reasonably well, need to now tackle variativity. Huge range of energies may be difficult to cover by single generator

V. Chekalina, "Generative Models for Fast Calorimeter Simulation: LHCb Case" @ CHEP 2018
Storage Requirements

• Storage needs are driven by HLT output bandwidth
  • Tape needs incompressible, while mitigations possible for disk
    • E.g. parking scenarios are considered but introduce additional operational costs for the experiment and infrastructure costs for sites
• MC simulation output data format mostly migrated to m(icro)DST format with small contribution to needs introducing a size reduction of factor 20

• LHCb relies on a small amount of sites with disk storage:
  • T0 + 7 T1s + 13 T2s with minimum size requirements especially for T2s
  • Data caching especially on ”small disk sites” is not a major use case
Data Movement

• Introduce multiple streaming layers to keep data set size under control
  • O(10) streams from Online, O(100) streams Offline
  • Expect on average 500 TB per data set / data taking year
  • In case of parking these need to be staged in due time, O(days)

• Throughput to/from tape systems will increase by several factors

• WLCG/DOMA initiative welcome to possibly further reduce costs
  • Especially optimizations on the timescale of Run 3
What next? LHCb Upgrade II

- **Expression Of Interest** for an experimental programme going beyond the current LHCb Upgrade plan, aiming at a full exploitation of the Flavor physics potential of the HL-LHC.
- At L=2 $10^{34}$ almost all bunch crossings contain interesting signal
  - But also vast majority of uninteresting particles from pile-up
- Detector readout and reconstruction will be one of the most challenging issues
What next? LHCb Upgrade II

- Naive scaling (x10) of data rates with respect to LHCb Upgrade I
- Early suppression of pile-up (with timing?)
  - Either at HLT1 or HLT2, with different pros and cons
- Compare with e.g. CMS-TDR-018
  - Event network throughput: 3-6 TB/s
  - Storage throughput: 30-60 GB/s
- LHCb Upgrade II DAQ must process 10x the HL-LHC GPD data rate
- LHCb Upgrade II offline must process same data volume as GPDs

November 5th 2018
M.Cattaneo -- Quantum Computing for HEP Workshop