Real time data analysis at the LHC: present and future

Vladimir V. Gligorov

NIPS 2014, Montreal, Canada
A translation into “physicist-ese”

Real time data analysis

==

Triggering
The heartbeat of the LHC

Collisions at the LHC: summary

Proton - Proton 2804 bunch/beam
Protons/bunch $10^{11}$
Beam energy 7 TeV ($7 \times 10^{12}$ eV)
Luminosity $10^{34}$ cm$^{-2}$ s$^{-1}$

Bunch crossing rate: 15–30 MHz

Between 1–200 proton-proton collisions per crossing (depends on experiment).

New physics rate $\approx 0.00001$ Hz

Event selection: 1 in 10,000,000,000,000
Triggers today
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

This means about 15000 PB of data every year

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

This means about 15000 PB of data every year

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

This means about 15000 PB of data every year

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

This means about 15000 PB of data every year

Data year

- BBC iPlayer: 180 PB
- Facebook: 2500 PB
- Twitter: 3 PB

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

- Input data rate of the LHCb experiment = 1.5 TB/second
- This means about 15000 PB of data every year

Comparison of data sizes (in PB) for various entities:
- AT&T networks
- BBC iPlayer
- Facebook
- Twitter

NB: ATLAS/CMS about a bit more than one order of magnitude above LHCb
Why do we need triggers at the LHC?

Input data rate of the LHCb experiment = 1.5 TB/second

This means about 15000 PB of data every year

Google was at ~7000 PB/year in 2008, so goodness knows where it is today...

AT&T networks

Twitter

Facebook

BBC iPlayer

Data year

3 PB  180 PB  2500 PB  11000 PB

NB : ATLAS/CMS about a bit more than one order of magnitude above LHCb
It’s all about the benjamins

Facebook
180 PB/yr
It’s all about the benjamins

Facebook
180 PB/yr

LHCb
15000 PB/yr
It’s all about the benjamins

Facebook
Computing
$O(500)$ M$/yr$

LHCb
15000 PB/yr

LHCb
Computing
$O(20)$ M$/yr$
It’s all about the benjamins

Facebook Computing
0(500) M$/yr

LHCb Computing
O(20) M$/yr

Storing and distributing data costs more than processing => real time analysis!
Collisions at the LHC: summary

- **Proton - Proton**: 2804 bunch/beam
- **Protons/bunch**: $10^{11}$
- **Beam energy**: 7 TeV ($7 \times 10^{12}$ eV)
- **Luminosity**: $10^{34} \text{cm}^{-2}\text{s}^{-1}$

**Bunch crossing rate**: 15–30 MHz

Between 1–200 proton-proton collisions per crossing (depends on experiment).

**New physics rate**: ≈ .00001 Hz

**Event selection**: 1 in $10,000,000,000,000$
Enter the MHz signal era

In the HL-LHC era triggers will discriminate between different signal classes!
Triggers today

Real-time data analysis tomorrow
The LHCb experiment
$p_T = \text{Transverse momentum}$

$E_T = \text{Transverse energy}$
LHCb

\[ p_T = \text{Transverse momentum} \]
\[ E_T = \text{Transverse energy} \]
**LHCb**

- **ELECTRONS**
- **PHOTONS**
- **HADRONs**

\[ p_T = \text{Transverse momentum} \]
\[ E_T = \text{Transverse energy} \]
LHCb

**p_T** = Transverse momentum

**E_T** = Transverse energy

- **ELECTRONS**
- **PHOTONS**
- **HADRONS**
- **MUONS**

Transverse

Beam
In 2010-2012: 15 MHz of bunch crossings, ~1.5 proton-proton interactions per bunch crossing, ~30 particles produced in the detector acceptance per interaction.

Compare to ATLAS/CMS who aim for 140-200 interactions/crossing after 2025.
LHCb trigger scheme

15 MHz collision rate

L0 Hardware Trigger: 1 MHz readout, high $E_T/P_T$ signatures
- 450 kHz $h^2$
- 400 kHz $\mu/\mu\mu$
- 150 kHz $e/\gamma$

Defer 20% to disk

Software High Level Trigger
- 29000 Logical CPU cores
- Offline reconstruction tuned to trigger time constraints
- Mixture of exclusive and inclusive selection algorithms

5 kHz Rate to storage

See JINST 8(2013) P04022
Triggering b-hadrons at the LHC
B event signatures

“A B is the elephant of the particle zoo: it is very heavy and lives a long time” -- T. Schietinger
Selection cascades

1. Information gathering ("reconstruction") stage
Selection cascades

1. Information gathering ("reconstruction") stage
Selection cascades

1. Information gathering ("reconstruction") stage

2. Event selection stage
Selection cascades

1. Information gathering ("reconstruction") stage

2. Event selection stage

⇒

⇒

⇒

⇒

Selected

Rejected
Selection cascades

1. Information gathering ("reconstruction") stage

2. Event selection stage

3. Next reconstruction stage
1. **Full reconstruction of tracks in vertex locator**

2. **Select displaced tracks**

**Reconstruction of displaced tracks in regions of interest**

Region of interest defined by assumed track $P/P_T$, 3/1.6 GeV in 2012
Single track triggering and pileup

Triggering on a single track makes you basically insensitive to increasing combinatorics, since there is nothing to combine!
50 → 5 kHz: the topological trigger

Figure 7: Lifetime acceptance function for an event of a two-body hadronic decay. The shaded, light blue regions show the bands for accepting a track \( IP \). After \( IP_2 \) is too low in (a) it reaches the accepted range in (b). The actual measured lifetime lies in the accepted region (c), which continues to larger lifetimes (d).

Figure 1: \( B \)-candidate masses from \( B \to K\pi\pi \) decays: (left) HLT2 2-body topological trigger candidates; (right) HLT2 3-body topological trigger candidates. In each plot, both the measured mass of the particle used in the trigger candidate (shaded) and the corrected mass obtained using Eq. 1 (unshaded) are shown. See Section 2 for discussion.

B mesons are long-lived particles; their mean flight distance in the LHC detector is \( O(1 \text{ cm}) \). The HLT2 topological lines exploit this fact by requiring that the trigger candidate’s flight-distance \( \chi^2 \) value be greater than 64. The direction of flight is also required to be downstream, i.e., the secondary vertex must be downstream of the primary vertex. A large flight distance combined with a high parent mass results (on average) in daughters with large impact parameters. The HLT2 topological lines require that the sum of the daughter IP \( \chi^2 \) values be greater than 100, 150 and 200 for the 2-body, 3-body and 4-body lines, respectively.

One of the larger background contributions to the HLT2 topological lines comes from prompt \( D \) mesons. To reduce this background, the HLT2 topological lines require that all \( (n-1) \)-body objects used by an \( n \)-body line either have a mass greater than 2.5 GeV (the object is too heavy to be a \( D \)) or that they have an IP \( \chi^2 > 16 \) (the object does not point at the primary vertex). An exhaustive list of the cuts used in all three of the HLT2 topological lines is given in Table 1.

Table 2 gives the efficiency of the HLT2 topological lines on events that pass the L0 and HLT1 one-track triggers for various offline-selected \( B \)-decay Monte Carlo samples.
50 → 5 kHz: the topological trigger

\[ m_{\text{corrected}} = \sqrt{m^2 + |p_{T,\text{missing}}'|^2 + |p_{T,\text{missing}}|^2} \]
Consider a two-variable boosted decision tree: this is like a binned selection where the BDT algorithm picks the optimal bin sizes and boundaries.
A bonsai boosted decision tree

Consider a two-variable boosted decision tree: this is like a binned selection where the BDT algorithm picks the optimal bin sizes and boundaries.

THEREFORE: discretize the variables yourself!

=> Makes sure that the trigger is insensitive to resolution fluctuations.

=> Transforms the trigger into a 1D lookup table making it essentially infinitely fast.
Topological performance

Figure 10: Response from the BBDT for minimum bias LHC 2010 data (shaded grey), $pp \to c\bar{c}X$ Monte Carlo (blue), $pp \to b\bar{b}X$ Monte Carlo (red) and all minimum bias Monte Carlo (black). The Monte Carlo is not normalized to the data (see text for details).

N.b., no muon or electron requirements were used when making this plot.

Measured output almost 100% consistent with $b\bar{b}$ events!

See also LHCb-PUB-2011-002, 003, 016
http://arxiv.org/abs/1211.3055

Gligorov&Williams http://arxiv.org/abs/1210.6861
The LHCb upgrade and trigger challenges
Why upgrade LHCb?

15 MHz @ 1.5 int/crossing => 30 MHz @ 6 int/crossing!

Once most data is signal, no quick preselection possible!
Upgrade trigger layout

30 MHz inelastic event rate and full event rate building

LLT: 15-30 MHz output rate, select high $E_T/P_T$ ($h^\pm/\mu/e/\gamma$)

Software High Level Trigger

Full event reconstruction, inclusive and exclusive kinematic/geometric selections

Run-by-run detector calibration

Add offline precision particle identification and track quality information to selections

2-10 GB/s rate to storage

Full burden of rate reduction is now on the software trigger.

Rate now given in GB/s to reflect the fact that we want to maximize event rates by storing only partial information about some events => real time analysis.
An example of online signals

Trigger level signal purities and resolutions close to offline, will become even closer in RunII and upgrade once the trigger can use RICH information to separate hadrons.
The run-by-run detector calibration is something which we plan to deploy already in 2015.

For the selections, can we be smarter than having a long list of exclusive triggers?
The run-by-run detector calibration is something which we plan to deploy already in 2015.

For the selections, can we be smarter than having a long list of exclusive triggers?

E.g. by deploying multivariate multi-class algorithms to assign each event a likelihood of being a certain kind of signal?
The run-by-run detector calibration is something which we plan to deploy already in 2015.

For the selections, can we be smarter than having a long list of exclusive triggers?

E.g. by deploying multivariate multi-class algorithms to assign each event a likelihood of being a certain kind of signal?

E.g. by having budgeted classification algorithms which optimize the time-cost of the information used in the classifier?
Challenges at other LHC experiments
The basic approach of all four collaborations can be summarized as follows: put as much as DAQ will allow into software triggers. Nevertheless "physics" and hardware constraints are leading to implementation differences.
<table>
<thead>
<tr>
<th></th>
<th>ALICE</th>
<th>LHCb</th>
<th>CMS</th>
<th>ATLAS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware trigger</strong></td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Software trigger input rate</strong></td>
<td>50 kHz Pb–Pb</td>
<td>30 MHz</td>
<td>500/750 kHz for PU 140/200</td>
<td>0.4 MHz</td>
</tr>
<tr>
<td></td>
<td>200 kHz p–Pb</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Baseline processing architecture</strong></td>
<td>CPU/GPU/FPGA/Cloud&amp;Grid</td>
<td>CPU farm (+coprocessors)</td>
<td>CPU farm (+coprocessors)</td>
<td>CPU farm (+coprocessors)</td>
</tr>
<tr>
<td><strong>Software trigger output rate</strong></td>
<td>50 kHz Pb–Pb</td>
<td>20–100 kHz</td>
<td>5–7.5 kHz</td>
<td>5–10 kHz</td>
</tr>
<tr>
<td></td>
<td>200 kHz p–Pb</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
DAQ take home message

<table>
<thead>
<tr>
<th></th>
<th>ALICE</th>
<th>LHCb</th>
<th>CMS</th>
<th>ATLAS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware trigger</strong></td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Software trigger input rate</strong></td>
<td>50 kHz Pb-Pb</td>
<td>200 kHz p-Pb</td>
<td>30 MHz</td>
<td>500/750 kHz for PU 140/200</td>
</tr>
<tr>
<td><strong>Baseline processing architecture</strong></td>
<td>CPU/GPU/FPGA/Cloud&amp;Grid</td>
<td>CPU farm (+coprocessors)</td>
<td>CPU farm (+coprocessors)</td>
<td>CPU farm (+coprocessors)</td>
</tr>
<tr>
<td><strong>Software trigger output rate</strong></td>
<td>50 kHz Pb-Pb</td>
<td>20–100 kHz</td>
<td>5–7.5 kHz</td>
<td>5–10 kHz</td>
</tr>
</tbody>
</table>

ATLAS, CMS, and LHCb will process the roughly the same number of collisions in their software triggers.

Opens up enormous potential for “light” physics, not just flavour but light exotics, dark matter, etc. in ATLAS/CMS.

Full exploitation of the physics potential will require the same approaches as I am presenting for LHCb!
Some example signal rates

ATLAS/CMS HL-LHC (?)

LHCb upgrade 100 fb⁻¹ (Multiply by 20 for ccbar)

LHCb 8 fb⁻¹

Belle II 50 ab⁻¹

B-factories

bbar pairs

1 × 10⁹  5 × 10¹⁰  2 × 10¹²  2 × 10¹³  ~10¹⁵
Typical GPD selection is by $P_T$ but this will kill beauty and especially charm $\Rightarrow$ exploit full power of software to create smarter classifiers, then save only part of event!
Microprocessor Transistor Counts 1971-2011 & Moore’s Law

The diagram shows a logarithmic graph of transistor count over time, with the x-axis representing the date of introduction and the y-axis representing the transistor count. The curve indicates that the transistor count doubles every two years, illustrating the exponential growth in microprocessor technology. The specific models are marked along the timeline, highlighting the advancements in transistor count from 1971 to 2011.
## Actually a bit more complicated

Stolen from Beat Jost

<table>
<thead>
<tr>
<th>Architectural change</th>
<th>Fabrication process</th>
<th>Microarchitecture</th>
<th>Codenames</th>
<th>Release date</th>
<th>Processors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>65 nm</td>
<td>P6, NetBurst</td>
<td>January 5, 2006</td>
<td>8/4P Server</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>45 nm</td>
<td>Core</td>
<td>November 11, 2007</td>
<td>Tigerton, Woodcrest, Clovertown</td>
</tr>
<tr>
<td>Took</td>
<td>New microarchitecture</td>
<td></td>
<td>Parnyn</td>
<td>November 11, 2007</td>
<td>Dunnington, Harpertown</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>32 nm</td>
<td>Nehalem</td>
<td>November 17, 2008</td>
<td>Bexley, Gainestown</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>22 nm</td>
<td>Sandy Bridge</td>
<td>January 9, 2011</td>
<td>(Skipped), Sandy Bridge-EP</td>
</tr>
<tr>
<td>Took</td>
<td>New microarchitecture</td>
<td></td>
<td>Ivy Bridge</td>
<td>April 29, 2012</td>
<td>Ivy Bridge-EX, Ivy Bridge-EP</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>14 nm</td>
<td>Haswell</td>
<td>June 2, 2013</td>
<td>Broadwell</td>
</tr>
</tbody>
</table>

We are here!
## Future microprocessor evolution?

<table>
<thead>
<tr>
<th>Architectural change</th>
<th>Fabrication process</th>
<th>Microarchitecture</th>
<th>Codename</th>
<th>Release date</th>
<th>Processors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>14 nm</td>
<td>Haswell</td>
<td>Broadwell</td>
<td>2014</td>
</tr>
<tr>
<td>Tock</td>
<td>New microarchitecture</td>
<td></td>
<td></td>
<td>Skylake</td>
<td>2015</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>10 nm</td>
<td>Skylake</td>
<td>Cannonlake</td>
<td>2016</td>
</tr>
<tr>
<td>Tock</td>
<td>New microarchitecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>7 nm</td>
<td></td>
<td></td>
<td>2017</td>
</tr>
<tr>
<td>Tock</td>
<td>New microarchitecture</td>
<td></td>
<td></td>
<td></td>
<td>2018</td>
</tr>
<tr>
<td>Tick</td>
<td>Die shrink</td>
<td>5 nm</td>
<td></td>
<td></td>
<td>2019</td>
</tr>
<tr>
<td>Tock</td>
<td>New microarchitecture</td>
<td></td>
<td></td>
<td></td>
<td>2020</td>
</tr>
</tbody>
</table>

**Take home message:** expect tick-tock and die shrinking to continue for the next years.
Extrapolating to the future

Clearly 25% performance improvement per year is not the same as doubling the performance every 2 years (more like 3). However also important to notice that this is a power law, so small changes in the assumed %/year lead to big differences on a 10-20 year timescale. CMS and LHCb somewhat more optimistic than CERN computing, backed up by observed performance improvements. But nobody betting the farm on ±5%.

Critical point: must fully exploit the new many core architectures!

<table>
<thead>
<tr>
<th></th>
<th>ALICE</th>
<th>LHCb</th>
<th>ATLAS</th>
<th>CMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assumed online performance gains</td>
<td>25%/year</td>
<td>35%/year</td>
<td>25%/year</td>
<td>35%/year</td>
</tr>
</tbody>
</table>

CMS observed performance improvements

- look at the power of the HLT nodes
  - and foreseen for 2015
- extrapolating to 2023 we could estimate increase by a factor $\times 10$
- this still leaves a factor $\times 2$ ($\times 4$)
ALICE DAQ

<table>
<thead>
<tr>
<th>Detector</th>
<th>Input to Online System (GByte/s)</th>
<th>Peak Output to Local Data Storage (GByte/s)</th>
<th>Avg. Output to Computing Center (GByte/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPC</td>
<td>1000</td>
<td>50.0</td>
<td>8.0</td>
</tr>
<tr>
<td>TRD</td>
<td>81.5</td>
<td>10.0</td>
<td>1.6</td>
</tr>
<tr>
<td>ITS</td>
<td>40</td>
<td>10.0</td>
<td>1.6</td>
</tr>
<tr>
<td>Others</td>
<td>25</td>
<td>12.5</td>
<td>2.0</td>
</tr>
<tr>
<td>Total</td>
<td>1146.5</td>
<td>82.5</td>
<td>13.2</td>
</tr>
</tbody>
</table>

**Input rate 1TByte/s**

**Goal is to achieve around 100x compression**

Later compression stages perform detector calibrations which are fed back into earlier stages. The compression explicitly preserves the ability to recalibrate offline.

---

ALICE performs event compression, not selection, in their software “trigger”
Data compression or real-time analysis?

The ALICE approach has many similarities with that of LHCb.

Once an analysis is defined, and the detector aligned and calibrated in real time, why isn’t the trigger writing your paper for you?

Would allow large volumes of data to be thrown away, reducing the pressure on this 2-10 Gb/second.

Could be applied at ATLAS/CMS to expand “light” physics programme.
Data compression or real-time analysis?

The ALICE approach has many similarities with that of LHCb.

Typical horrified responses:

Don’t throw my data away!

What if I make a mistake?!

How would you ever trust the results?!

The real choice is: throw the data away, or do analysis first and THEN throw the data away.
Backup
ALICE are fully committed to a GPU reconstruction for the TPC in particular. Already commissioned in Run I! Achieves a threefold increase in performance compared to CPU.
ATLAS/CMS reconstructions

Seed reconstruction by high momentum objects, then perform different reco inside/outside signal ROI.

Already used in RunI for brems/muon efficiency recovery. Expect to expand on these strategies.
LHCb’s 30 MHz reconstruction

Reconstruction timing is basically linear with number of pp interactions. Because we want to catch low momentum tracks crossing the full detector volume it is not trivial to parallelize the track finding, although a lot work is ongoing into GPU coprocessors.
Topological performance

Measured output is almost 100% consistent with bbar events!

See also LHCb public notes and trigger publications
LHCb-PUB-2011-002,003,016
http://arxiv.org/abs/1211.3055

Gligorov&Williams http://arxiv.org/abs/1210.6861
The challenges of running online

- If the keep regions are small relative to the resolution or stability of the detector, the signal could oscillate in and out of the keep regions. This would result in, at best, a less efficient trigger and, at worst, a trigger whose efficiency is very difficult to understand.

- In many cases the signal samples by necessity must come from simulations because the signals have, in fact, not yet been observed in data. In other cases the trigger is meant to be inclusive, i.e., the trigger is meant to select classes of signal types rather than one specific signal channel. In both cases, the signal PDFs might not be completely accurate or even available during the training process.

- Any HLT algorithm must run in the online environment; thus, it must be extremely fast.
HLT1 signal performance

LHCb trigger: hardware constraints

- **DAQ**
  - Readout rate: 1MHz
  - Total Event size: 50+kB
  - HLT output rate: 5000Hz
  - HLT output bandwidth: 250MB/s

- **Architecture**
  - Dual core routers
  - Data that can’t be processed by the HLT is temporarily stored on local HLT node discs for inter-fill processing
Because we cannot read the full LHCb detector at 40 MHz, we need a hardware trigger.

Implemented in custom electronics with 4μs latency

Triggers on calorimeter clusters and muon station segments.
Dramatis personae
What is a trigger?

trigger (trɪgər)
n.
1. a. The lever pressed by the finger to discharge a firearm.
   b. A similar device used to release or activate a mechanism.
2. An event that precipitates other events.
3. Electronics A pulse or circuit that initiates the action of another component.

tr.v. triggered, triggering, triggers
1. To set off; initiate: remarks that triggered bitter debates.
2. To fire or explode (a weapon or an explosive charge).
What is a trigger?

trigger  (trɪˈɡər)
n.

1. a. The lever pressed by the finger to discharge a firearm.
   b. A similar device used to release or activate a mechanism.

2. An event that precipitates other events.

3. *Electronics* A pulse or circuit that initiates the action of another component.

tr.v. triggered, triggering, triggers

1. To set off; initiate: *remarks that triggered bitter debates.*
2. To fire or explode (a weapon or an explosive charge).
# ATLAS/CMS vs. LHCb

<table>
<thead>
<tr>
<th></th>
<th>Rate of bunch crossings</th>
<th>Mean interactions per bunch crossing</th>
<th>Mean event size</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ATLAS/CMS</strong></td>
<td>15 MHz</td>
<td>~25</td>
<td>1500 kB</td>
</tr>
<tr>
<td><strong>LHCb</strong></td>
<td>15 MHz</td>
<td>1.5</td>
<td>50 kB</td>
</tr>
</tbody>
</table>

The data rates at ATLAS and CMS are 30 times greater than at LHCb. This drives a design in which much more work is done by hardware triggers which make their decisions based on information from only a part of the detector.
The confirmation strategy

For LHCb, the High Level Trigger ignores the hardware trigger

In ATLAS/CMS, in order to speed up execution, the high level trigger is set up to “confirm” the decision of the hardware trigger
The confirmation strategy

For LHCb, the High Level Trigger ignores the hardware trigger.

In ATLAS/CMS, in order to speed up execution, the high level trigger is set up to “confirm” the decision of the hardware trigger.
For LHCb, the High Level Trigger ignores the hardware trigger.

In ATLAS/CMS, in order to speed up execution, the high level trigger is set up to "confirm" the decision of the hardware trigger.

The hardware trigger has fired because of a muon identified in the muon system.
The confirmation strategy

For LHCb, the High Level Trigger ignores the hardware trigger.

In ATLAS/CMS, in order to speed up execution, the high level trigger is set up to “confirm” the decision of the hardware trigger.

The hardware trigger has fired because of a muon identified in the muon system.

The High Level Trigger “confirms” that this is a muon by finding it in the tracking system as well.

The region of interest for this search is defined by the detector geometry and the location of the hardware trigger candidate.
The confirmation strategy

For LHCb, the High Level Trigger ignores the hardware trigger.

In ATLAS/CMS, in order to speed up execution, the high level trigger is set up to “confirm” the decision of the hardware trigger.

The hardware trigger has fired because of a muon identified in the muon system.

The High Level Trigger “confirms” that this is a muon by finding it in the tracking system as well.

The region of interest for this search is defined by the detector geometry and the location of the hardware trigger candidate.

Confirmation is a very common strategy in triggering. It works if the early triggers fire predominantly because of the presence of signal.
Two parameters to consider

1) The frequency with which events occur

2) The complexity (size) of each event

Key concepts to remember from this seminar

Different triggers make their selection on different criteria, but they must always be independent of each other.

A trigger has a finite time to make its selection, so you need to optimize taking into account the time cost of obtaining information.

Multivariate selections are very powerful but usually need a simpler preselection to allow them the time to do their job.
A lot of time is spent in the trigger reconstructing charged particles.

If you know that you will cut on some minimum momentum of these particles, you can build this cut into the reconstruction.

The higher the momentum the straighter the charged particle path.

Can define a narrow path depending on momentum; saves a lot of time looking for fake paths.

Always look for ways to build a selection into your reconstruction!
Angular biases?

One of the key advantages of an inclusive trigger is that we minimally bias offline distributions, e.g. angular acceptances in $K^*\mu\mu$, or Dalitz acceptances in $KK\pi$, are kept as flat as possible.
Because we can reproduce the trigger decisions offline, we can measure lifetime biases in a data driven way offline.

Get an event-by-event acceptance by replaying the trigger decision for the full range of possible B/D lifetimes.

No trigger emulation needed, correct alignment and detector conditions automatically taken into account.
Lifetime biases?

Because we can reproduce the trigger decisions offline, we can measure lifetime biases in a data driven way offline.

Get an event-by-event acceptance by replaying the trigger decision for the full range of possible B/D lifetimes.

No trigger emulation needed, correct alignment and detector conditions automatically taken into account.
Because we can reproduce the trigger decisions offline, we can measure lifetime biases in a data driven way offline.

Get an event-by-event acceptance by replaying the trigger decision for the full range of possible B/D lifetimes.

No trigger emulation needed, correct alignment and detector conditions automatically taken into account.
Because we can reproduce the trigger decisions offline, we can measure lifetime biases in a data driven way offline.

Get an event-by-event acceptance by replaying the trigger decision for the full range of possible B/D lifetimes.

No trigger emulation needed, correct alignment and detector conditions automatically taken into account.
Lifetime biases?

Because we can reproduce the trigger decisions offline, we can measure lifetime biases in a data driven way offline.

Get an event-by-event acceptance by replaying the trigger decision for the full range of possible B/D lifetimes.

No trigger emulation needed, correct alignment and detector conditions automatically taken into account.
The LHCb physics programme...

- Charm Physics
- CPV in B decays
- Rare B decay searches
- Spectroscopy and Exotica

Note: clearly not the entire physics programme, see the [LHCb upgrade LOI](#) for more details.
...and its demands on the trigger

Charm Physics  CPV in B decays  Rare B decay searches  Spectroscopy and Exotica

10% of LHC interactions contain a charmed meson: keep the most interesting ones efficiently

Note: clearly not the entire physics programme, see the LHCb upgrade LOI for more details
10% of LHC interactions contain a charmed meson: keep the most interesting ones efficiently.

Trigger on any B decay into charged particles in an inclusive way, to minimize biases.

Note: clearly not the entire physics programme, see the LHCb upgrade LOI for more details.
...and its demands on the trigger

- Charm Physics
- CPV in B decays
- Rare B decay searches
- Spectroscopy and Exotica

10% of LHC interactions contain a charmed meson: keep the most interesting ones efficiently

Trigger on any B decay into charged particles in an inclusive way, to minimize biases

Maintain ~100% efficiency for rare muonic/photonic B decays

Note: clearly not the entire physics programme, see the LHCb upgrade LOI for more details
…and its demands on the trigger

- **Charm Physics**: 10% of LHC interactions contain a charmed meson: keep the most interesting ones efficiently.
- **CPV in B decays**: Trigger on any B decay into charged particles in an inclusive way, to minimize biases.
- **Rare B decay searches**: Maintain ~100% efficiency for rare muonic/photonic B decays.
- **Spectroscopy and Exotica**: Maintain a high rate of prompt and detached (di)muon triggers to enable datamining.

*Note: clearly not the entire physics programme, see the LHCb upgrade LOI for more details*
...and its demands on the trigger

<table>
<thead>
<tr>
<th>Charm Physics</th>
<th>CPV in B decays</th>
<th>Rare B decay searches</th>
<th>Spectroscopy and Exotica</th>
</tr>
</thead>
</table>

**And all this must fit into an output rate of ~4 kHz!**

**KEY CHALLENGE**: discriminate against prompt charm (300 kHz in the LHCb acceptance) while keeping the most interesting prompt charm!

- 10% of LHC interactions contain a charmed meson: keep the most interesting ones efficiently
- Trigger on any B decay into charged particles in an inclusive way, to minimize biases
- Maintain ~100% efficiency for rare muonic/photonic B decays
- Maintain a high rate of prompt and detached (di)muon triggers to enable datamining

Note: clearly not the entire physics programme, see the LHCb upgrade LOI for more details
Classifying triggered events: an interlude
All triggered events can be split into three categories
All triggered events can be split into three categories

**TOS** : The event would have triggered if only the particles belonging to the signal candidate were present.
All triggered events can be split into three categories

**TOS** : The event would have triggered if only the particles belonging to the signal candidate were present.

**TIS** : The event would have triggered if the particles belonging to the signal candidate were not present.
All triggered events can be split into three categories

TOS : The event would have triggered if only the particles belonging to the signal candidate were present.

TIS : The event would have triggered if the particles belonging to the signal candidate were not present.

TOB : neither TIS nor TOS
The overlap between TIS and TOS can be used to measure the trigger efficiency on data.

1) Select your signal events offline

2) Measure the fraction of TIS events which are also TOS of the trigger line which you are interested in

3) This gives the TOS efficiency of that line relative to the offline selection. The TIS efficiency can be similarly measured (fraction of TOS which are also TIS)
One big caveat in all of this

-- This whole concept relies on the fact that individual trigger decisions are independent of each other!
The LHCb trigger upgrade

The 1 MHz detector readout is the bottleneck in the current DAQ chain. Particularly limiting for hadronic decay modes, and would become more limiting as the luminosity rises due to pileup.

Therefore LHCb will upgrade all subdetectors to read out at 40 MHz.

And then scale the actual detector readout according to the available CPU capacity in the HLT farm.

Make the L0 (LLT) trigger less and less important as the upgrade progresses.
The hardware trigger stage

15 MHz collision rate

L0 Hardware Trigger: 1 MHz readout, high $E_T/P_T$ signatures

- 450 kHz $h^\pm$
- 400 kHz $\mu/\mu$
- 150 kHz $e/\gamma$
The calorimeter hardware trigger

Split the calorimeter into 2x2 cell clusters, and trigger on the sum transverse energy of the cluster
The muon hardware trigger

For muons search for track in first three stations
L0 trigger performance

The software trigger stage

15 MHz collision rate

L0 Hardware Trigger: 1 MHz readout, high $E_T/P_T$ signatures

- 450 kHz $h^2$
- 400 kHz $\mu/\mu$
- 150 kHz $e/\gamma$

Defer 20% to disk

Software High Level Trigger

- 29000 Logical CPU cores
- Offline reconstruction tuned to trigger time constraints
- Mixture of exclusive and inclusive selection algorithms

5 kHz Rate to storage
Corrected mass and vtx multiplicity
Why upgrade?

Only being able to read out the full detector at 1 MHz severely limits the event yields for hadronic modes.

To run at higher luminosity we must remove this bottleneck.

=> Full 40 MHz detector readout

=> All software trigger

=> Allows to run at a 5 times bigger proton-proton interaction rate compared to 2010–2012