Introduction to Trigger and DAQ systems

Sergo Jindariani (Fermilab)

HCP Summer School

August 24-25, 2022

About Me

- Joined Fermilab in 2008
- CDF Silicon Tracker Group Leader (2008-2009)
- USCMS Phase-1 Trigger Upgrade deputy project manager (2012-2013)
- Demonstration of Level-1 track trigger for CMS (2012-2016)
- LHC Physics Center Coordinator (2017-2021)

- Machine Learning in real-time data processing systems
- Unconventional trigger signatures
- Trigger/DAQ for future colliders

sergo@fnal.gov

Outline – Trigger

- What is a trigger and DAQ system, and why needed?
- Level-1 Triggers
- High Level Triggers
- Trigger Rate
- Trigger Efficiency

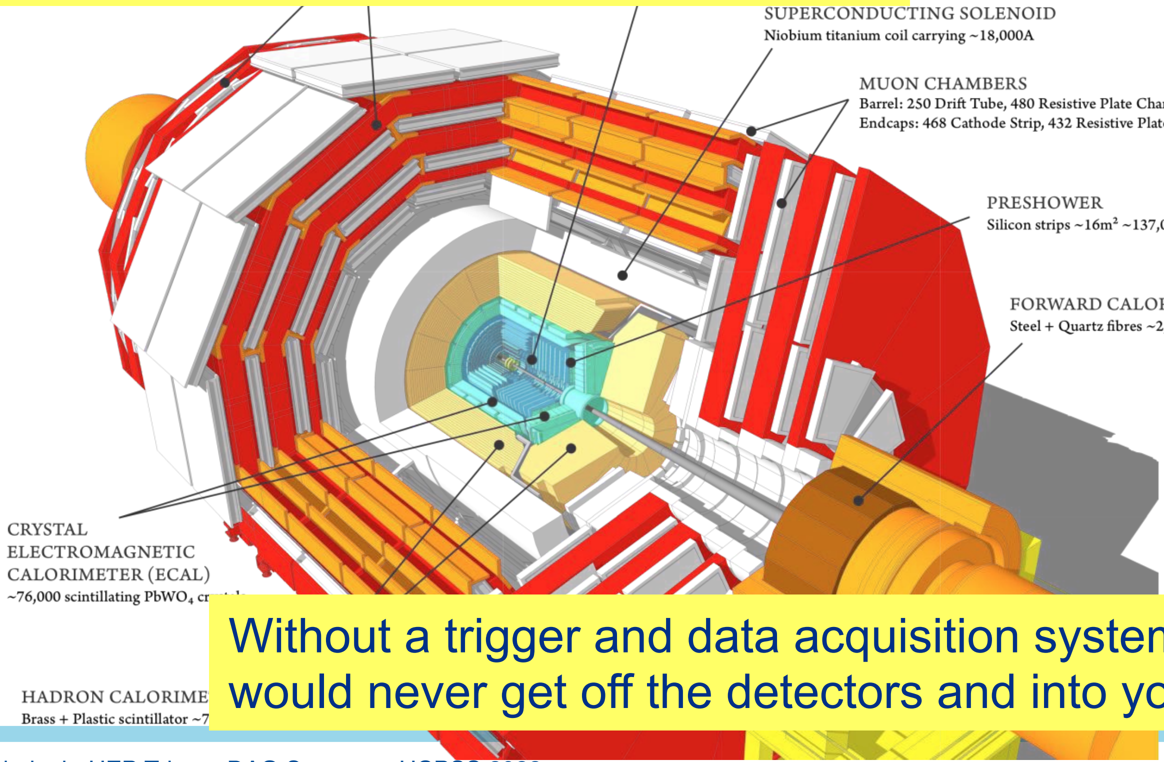
Day 1

- Trigger Menus
- Data parking and data scouting
- Future Outlook
  - HL-LHC trigger upgrade
  - AI in triggers
  - TDAQ for Future Colliders

Day 2

A Critical Piece of Experiment

When we show the canonical detector slide of an experiment, we usually omit a very important piece!

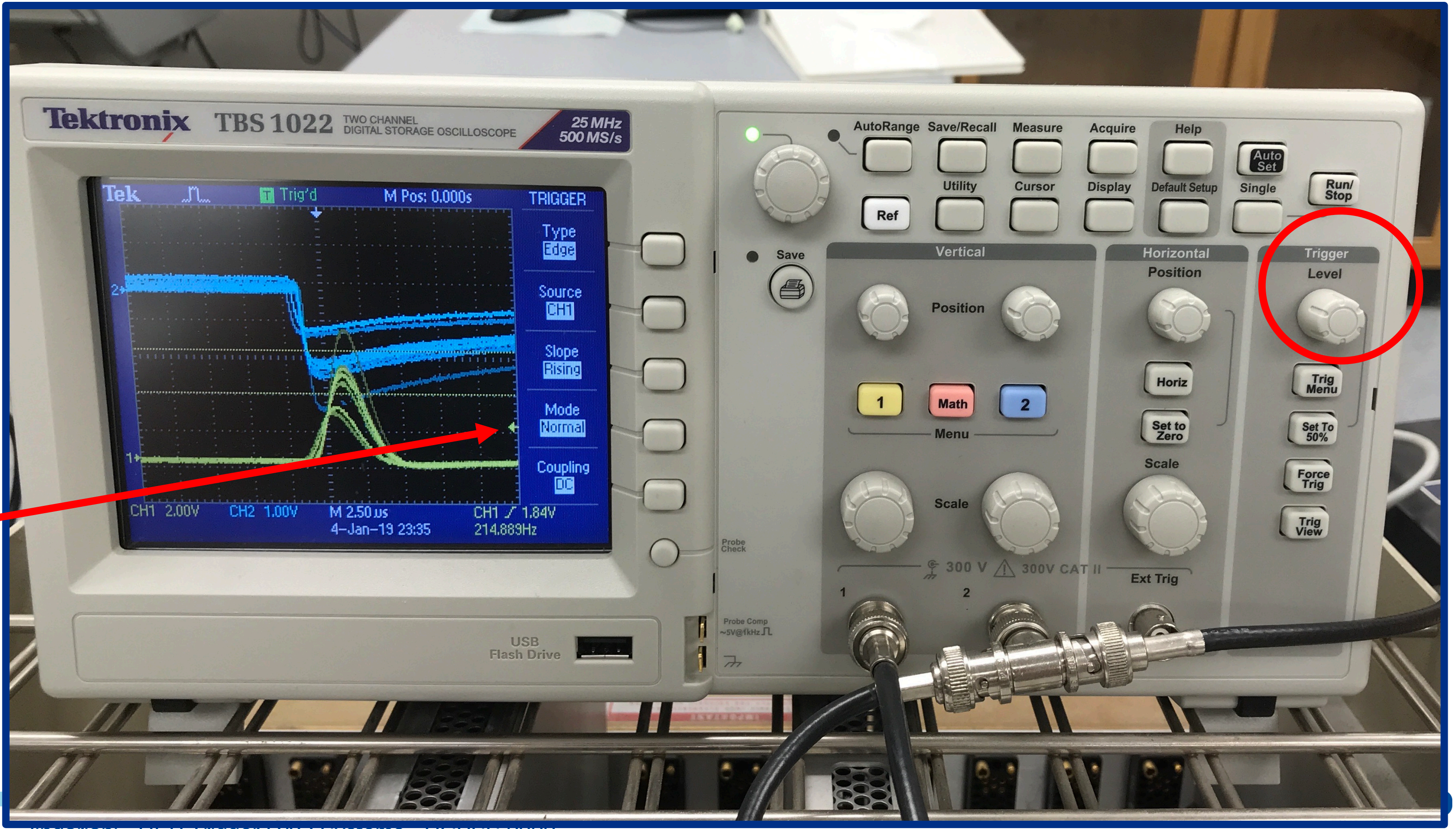


Without a trigger and data acquisition system, the data would never get off the detectors and into your analysis!

DAQ in HEP Experiments

- "Data acquisition" is the collection and storage of data recorded by a set of detectors comprising an experiment
- Often needs a "trigger", a set of conditions (a "menu") by which to initiate the sequence to digitize data from analog sensors (or read out from local buffers) and record to memory/disk/tape storage
  - Particle beam entering an experiment, or a collision taking place inside the experiment
- This trigger can be:
  - as simple as a single signal pulse indicating the presence of a particle passing or interacting in an experiment
  - or as complicated as partial reconstruction of a collision from detector data, and a long list of criteria to decide whether to store

Example: Trigger on a Scope



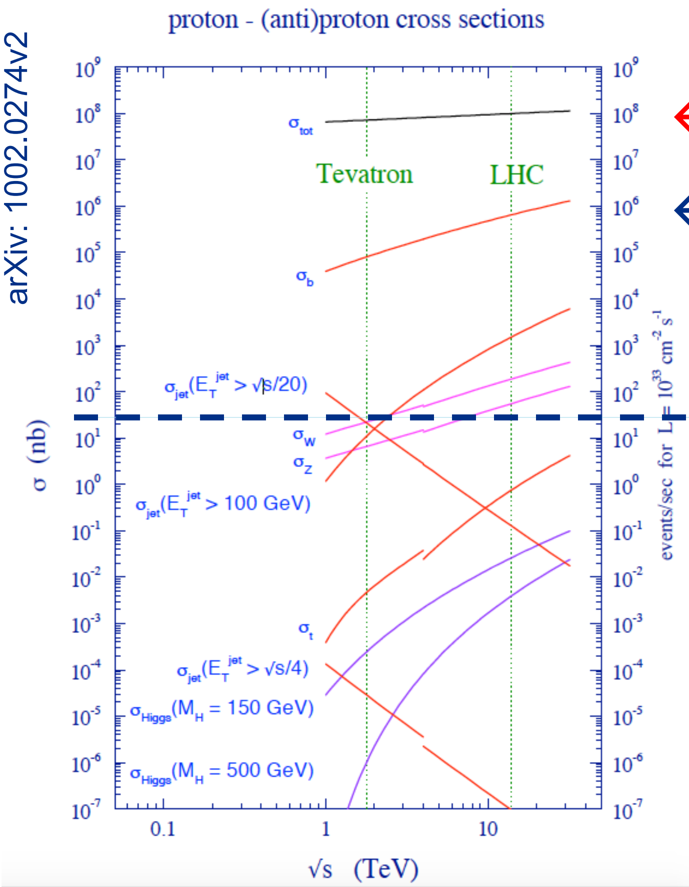
Pulse threshold for triggering readout

Why is Trigger Needed?

- Why is a trigger needed?
  - Too much data to continuously stream to disk for storage and/or computer processing reasons
    - For example , an LHC experiment could generate ~100 terabytes of data per second!
  - Electronics data digitization may induce "dead time"
    - A period of time when further data cannot be recorded because of processing, and thus the experiment is insensitive to new incoming data. This wastes collider luminosity.
- It is really the first step of a data analysis
  - An online data filtering system
  - But irreversible – can't go back to data you did not record
    - e.g. Throw away 99.998% of all LHC crossings
    - But, don't throw out the baby (Higgs) with the bath water!

LHC Cross-Sections and Rates

arXiv: 1002.0274v2



for L =2 x 10^{34} Hz/cm^2

← Total collision rate: 2 GHz

Billions of collisions per sec but at a beam crossing rate of 40 MHz

← b quark rate: 10 MHz

← W boson rate: 4 kHz

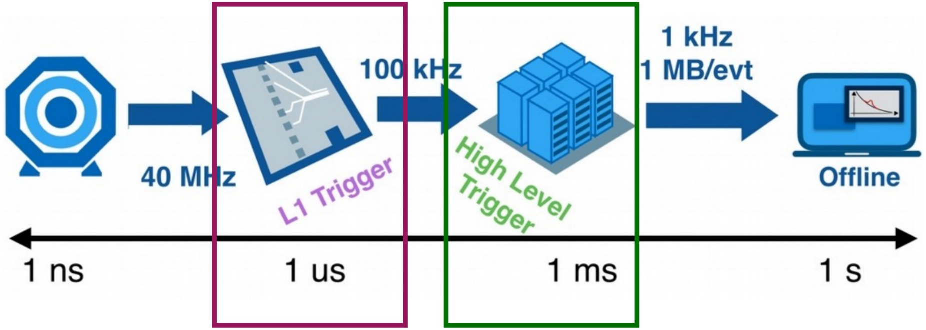
Keep

← Higgs boson rate: 1 Hz

We cannot keep all physics processes in order to collect enough data on interesting rare processes

Fermilab

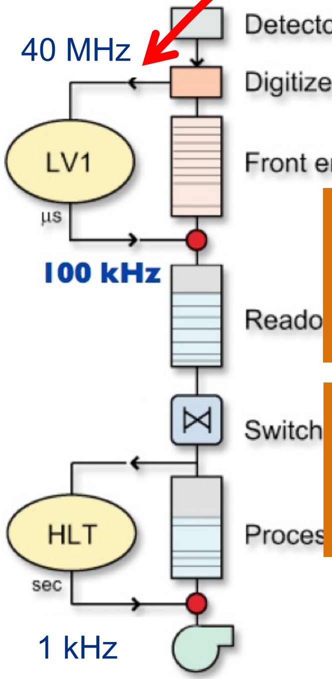
Trigger for Collider Experiments



- Typically segmented into multiple levels, with decreasing input rates and longer processing times (latencies)
- Level-1:
  - Custom electronic designs for maximum throughput (TB/s) and shortest latencies (microseconds). i.e. specialized computing
    - Custom chips (ASICs) and programmable logic (FPGAs)
  - Processing logic done in a maximally parallel way for shortest latency
  - Processing is pipelined, meaning processing is segmented into steps of a certain clock period (beam crossing), and intermediate results are registered after each step
- Level-2: (if needed…)
  - Combination of custom electronics and commercial computing equipment
- Level-3:
  - Commercial computing clusters of up to O(10^4) CPUs with up to ~second per event processing time
    - i.e. Parallelized with different collision events processed with different CPUs, but mostly sequential within a CPU core

Example: CMS Trigger Architecture

- Only two levels*:
  - Level-1: custom electronics to reduce the data from a collision rate of 40 MHz to no more than 100 kHz for the detector readout electronics, with only a 4 μs latency (buffer depth)
  - High Level Trigger (HLT): event filter farm comprised of commercial CPUs running software to further reduce event rate to storage to an average of ~1 kHz (for LHC Run 2)



Note that data path is split @ L1

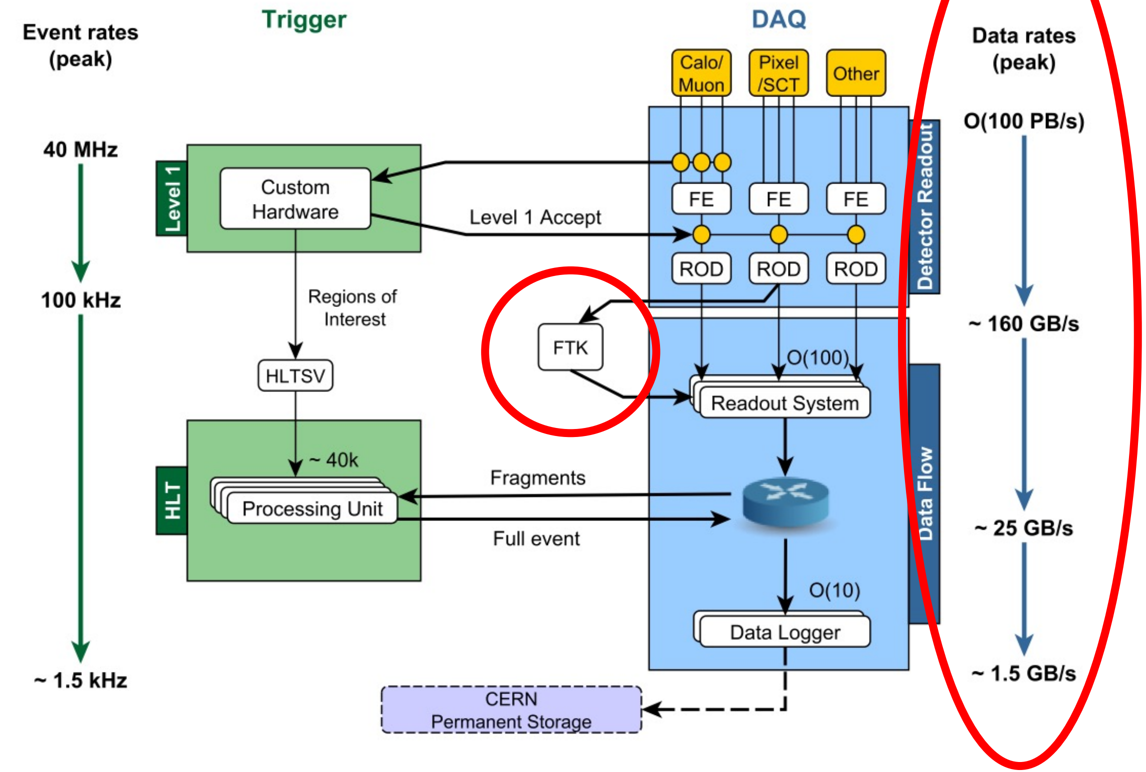
Data are sitting in buffers waiting for L1 decision

The L1 latency sets the depth of the buffers (and $)

*Historically, and for the ATLAS experiment initially, three levels were used. CMS was an adopter of a powerful HLT.

Fermilab

Example: ATLAS Trigger Architecture

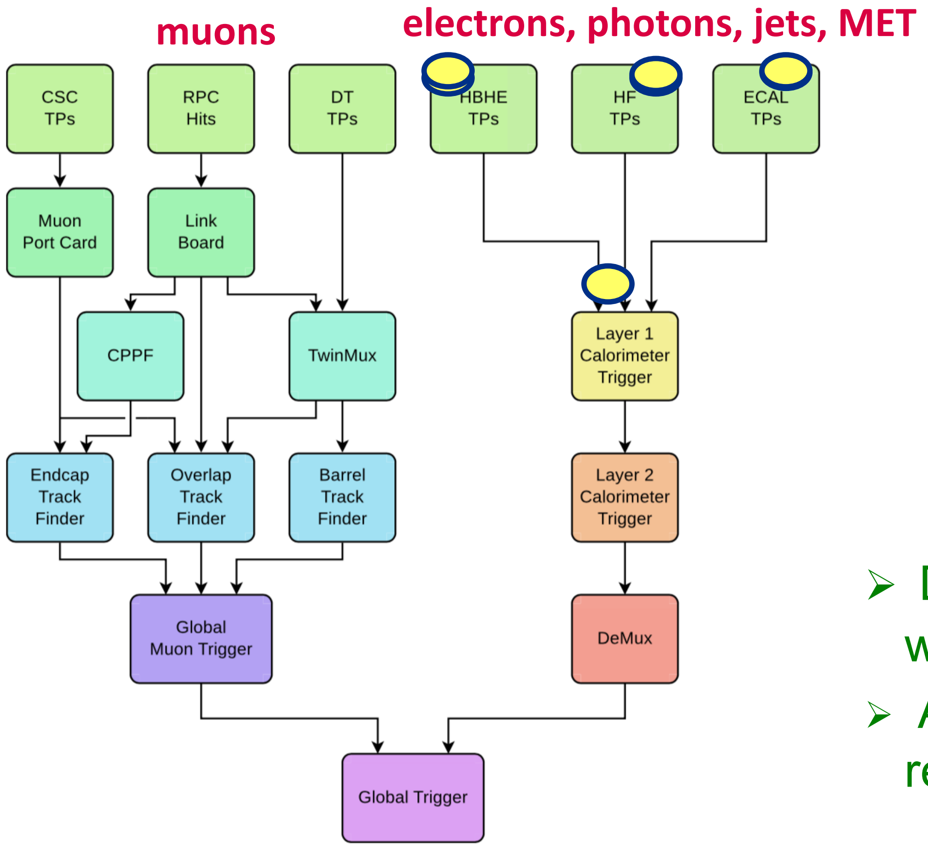


- Note the data rates (bandwidth) also gets reduced along with the event rate
- ATLAS also envisioned a custom accelerator for tracking at HLT

Evolution of Collider Triggers

	Tevatron / CDF (2004)		LHC / CMS (2018)
Beam Energy	1 TeV		6.5 TeV
Inst. Lumi. ($cm^{-2}s^{-1}$)	10^{32}	200X	$2x10^{34}$
Bunch xing freq / Time spacing	2.5 MHz / 400 ns	16X	40 MHz / 25 ns
L1 pipelined ?	No, initially		Yes
L1 output rate	25 kHz	4X	100 kHz
L2 output / HLT input	400 Hz	250X	100 kHz
L3 output rate	90 Hz	10X	1000 Hz
Event size	0.2 MB	5X	1 MB
Filter Farm	250 CPUs	40X	O(10 000) CPUs

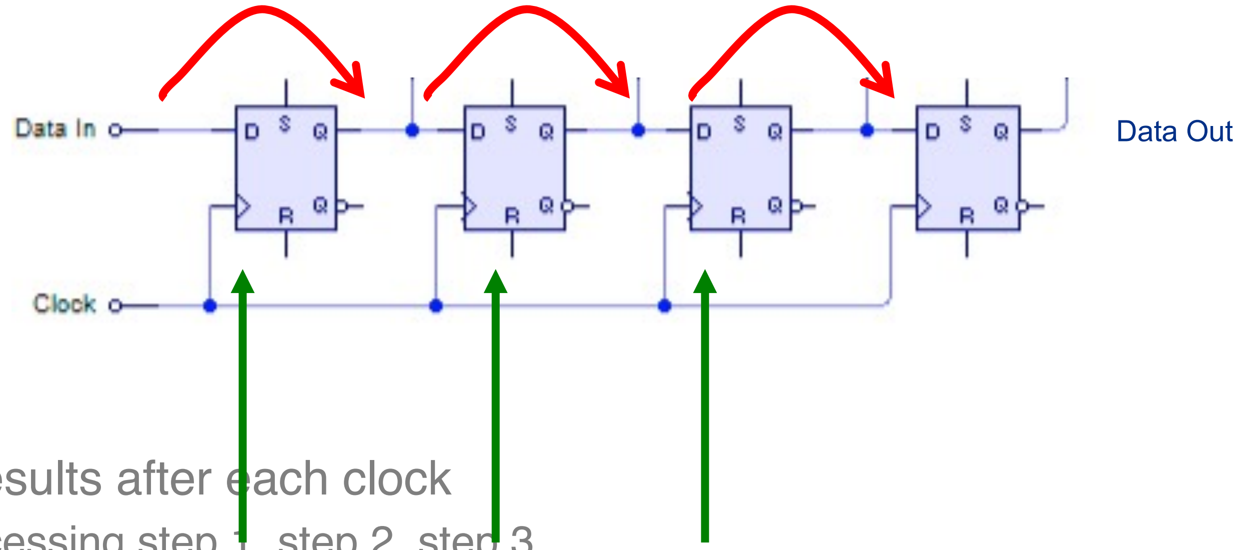
Pipelined Architecture



Processing takes longer than one beam crossing, so must register intermediate processing results along the way

- Data flows in a pipeline with a 40 MHz heartbeat
- Accept/reject decision reached in 4 μs

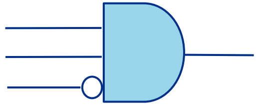
Pipeline



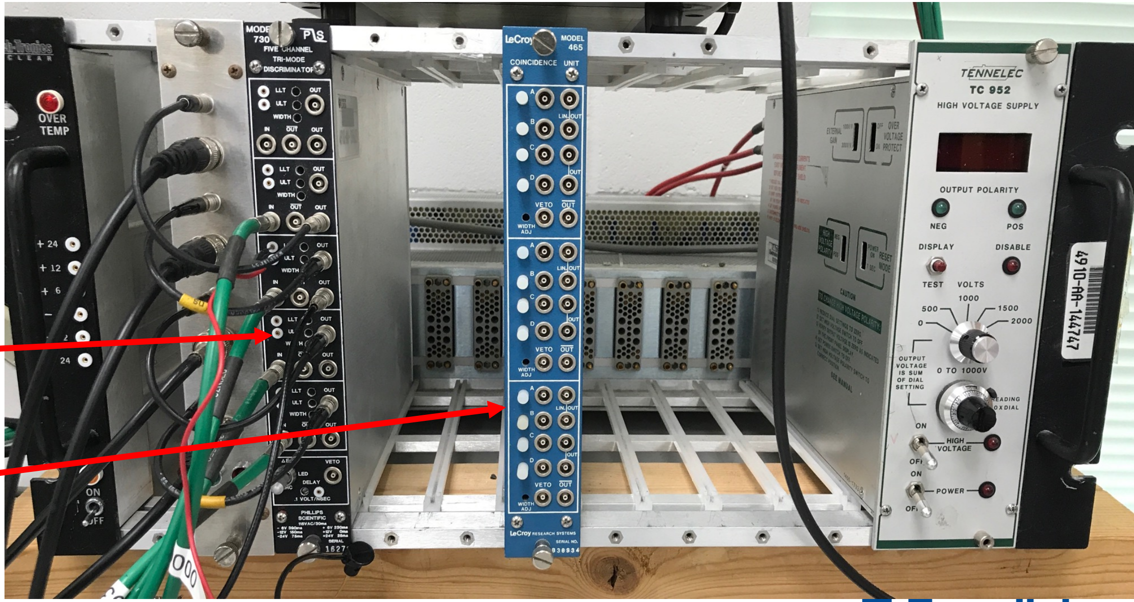
- Latch results after each clock
  - Processing step 1, step 2, step 3, …
- Implies your algorithm must be factorized into steps short enough to fit into 1 clock period

Early days of Trigger Implementation

- Simple logic coincidences
  - e.g. A and B and (not C)
  - Where, A, B, and C are digital signals coming from threshold discriminators on some detector signals (like scintillator paddles)



- NIM* modules could implement basic logical operations, including majority logic like 3 out of 4



Threshold discriminator, to convert analog to logic pulse

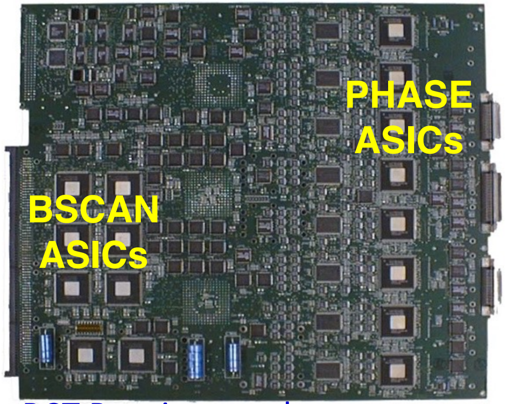
Coincidence unit

*Nuclear Instrumentation Module

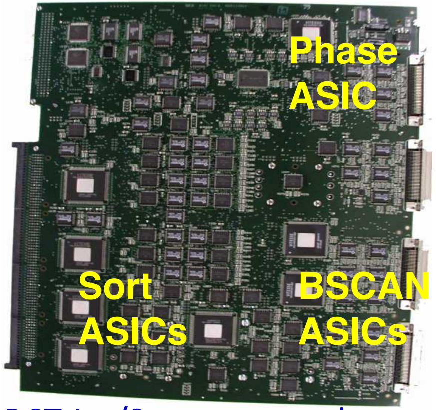
More recently: CMS Trigger System

Fermilab

Some L1 Hardware (2004)



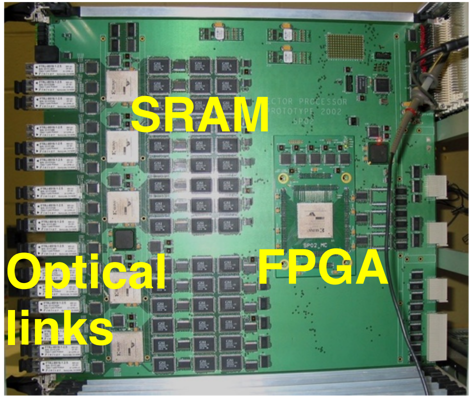
RCT Receiver card



RCT Jet/Summary card

RCT Electron isolation card

Muon Track-Finder

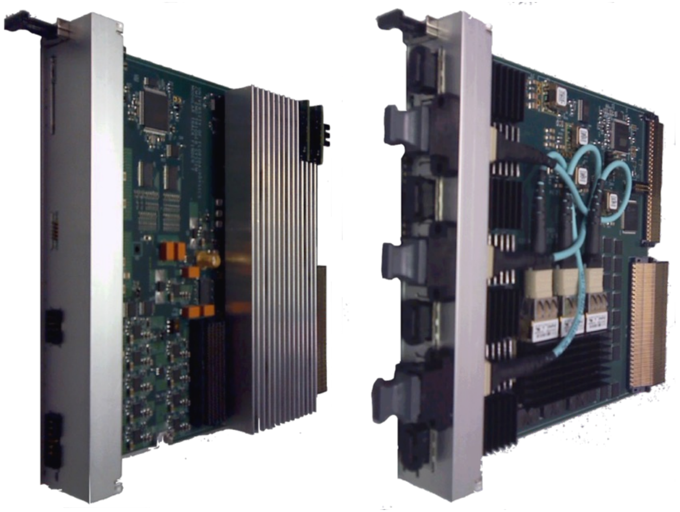


- **Custom chips (ASICs) – adders,** sorters...
- Programmable logic **(FPGAs)**
- **Memory (RAM)**
- **Copper** and Optical links, ~Gbit/s
- **VME chassis (early computer hw format)**

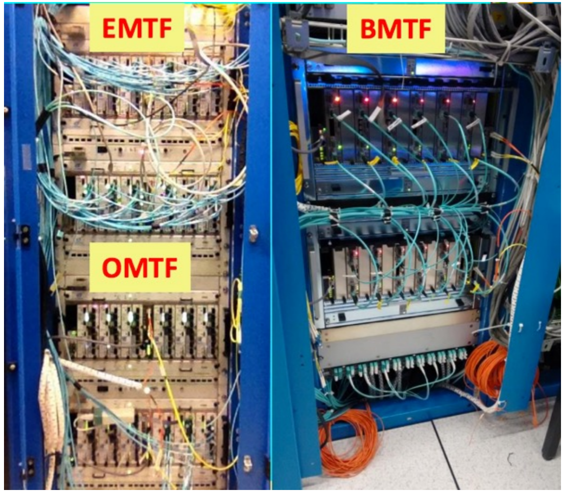
Fermilab

Some L1 Hardware (2016)

- CMS Phase-1 Trigger Upgrade
  - Larger Field Programmable Gate Arrays (FPGAs)
  - Up to 10 Gbit/s optical links
  - uTCA telecommunications infrastructure



Muon trigger racks



Calo trigger rack

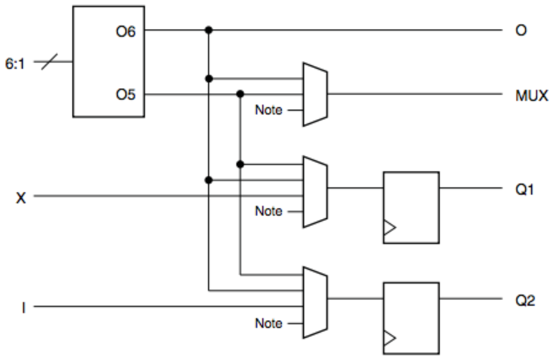
An Aside on FPGAs: Reprogrammable Silicon

- Field Programmable Gate Arrays (FPGAs)
  - Off the shelf component, not custom
  - A silicon "breadboard" of configurable logic gates, memories, transceivers, Digital Signal Processors (DSPs), registers
- Fast digital logic at a lower level than a CPU because the logic is not pre-specified
  - Although you could program an FPGA to become a CPU in principle...
- State of the art (Xilinx Ultrascale+, 16 nm technology):
  - Up to 1.7M Configurable Logic Blocks (CLBs) - 6 bit LUTs
  - Up to 12K DSP slices
  - Up to 128 transceivers at up to 32 Gbit/s
  - Up to ~500Mbit of RAM
- But there is also the Intel/Altera family of FPGAs

Millions of logic gates!

Terabits per second!

Xilinx Logic Block and DSP Slice



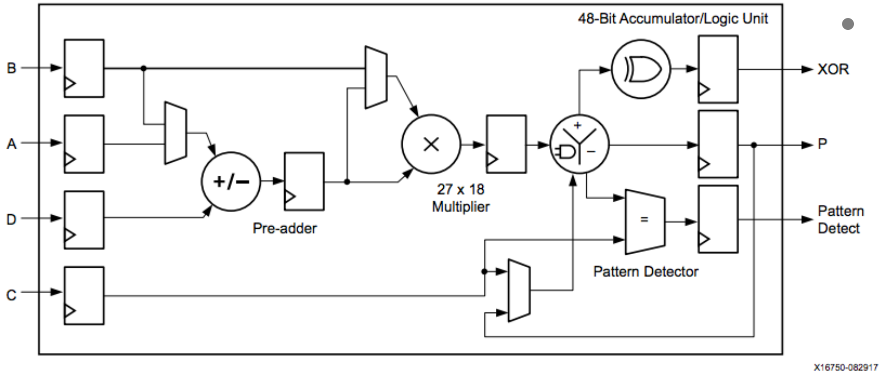


Figure 1-1: **Basic DSP48E2 Functionality**

- The Look-Up Table (LUT) can provide a 1 bit output for any collection of logic operations from 6 input bits
  - Like what a NIM module did
  - Millions of CLBs within large FPGA
- The DSP slice can specialize as a
  - 48 bit accumulator
  - 27 x 18 bit multiplier
  - 48 bit logic unit
    - Thousands within large FPGA

FPGA Programming Possibilities

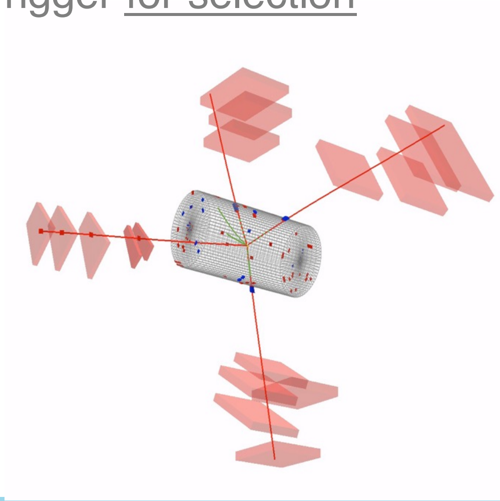
- These basic elements allow one to implement quite complex algorithms for a L1 trigger, such as
  - Massively parallel logic operations
    (remember, the L1 Trigger has only microseconds…)
    - pattern recognition for tracking
  - Matrix multiplication
    - Kalman filter track fitting
    - Artificial Intelligence: neural network implementation
- Programming an FPGA requires **firmware** to be written and synthesized into a "bit file" to load into the chip
  - But there are high-level languages to write the logic implementation, some very much like C/C++
    - VHDL, Verilog, Vivado HLS, OpenCL, …

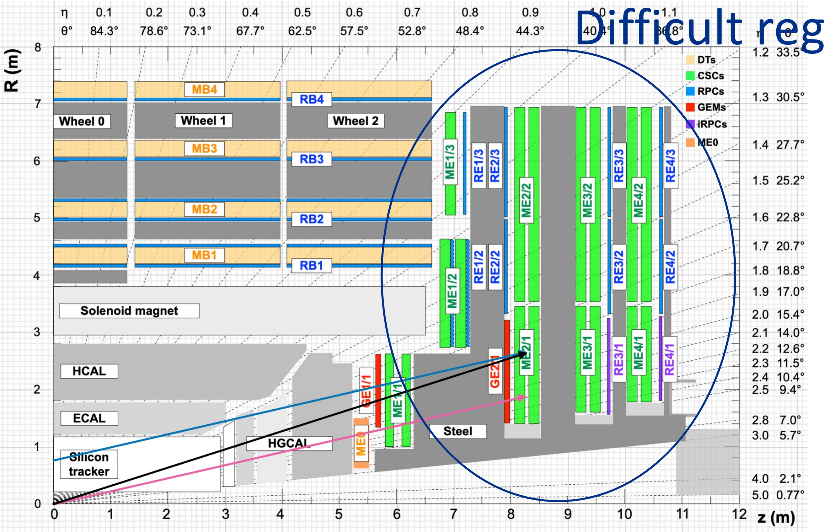
Algorithm: Level-1 Muon "Track-Finding"

– Track-Finder links receive track segments and FPGA builds into distinct tracks through pattern recognition logic in firmware
  - Algorithm implemented in a programmable FPGA

– Performs a momentum measurement using the deflection of muon in the magnetic field

– Transmit highest quality candidates to Global Trigger <u>for selection</u>

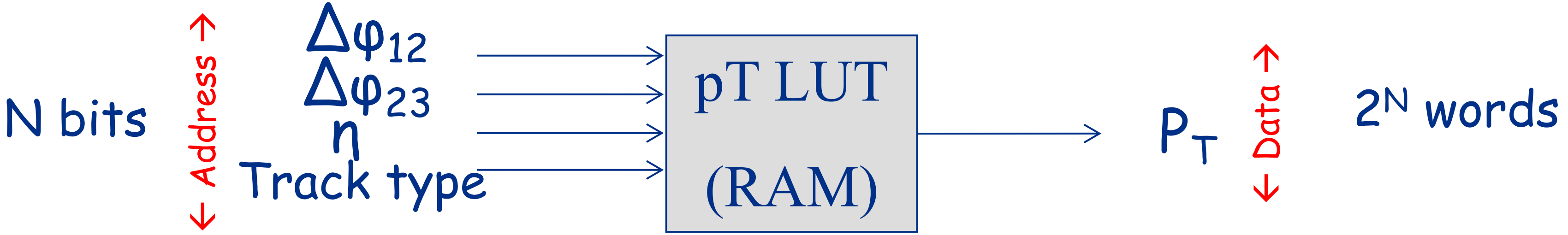




Fermilab

Triggers must be fast!

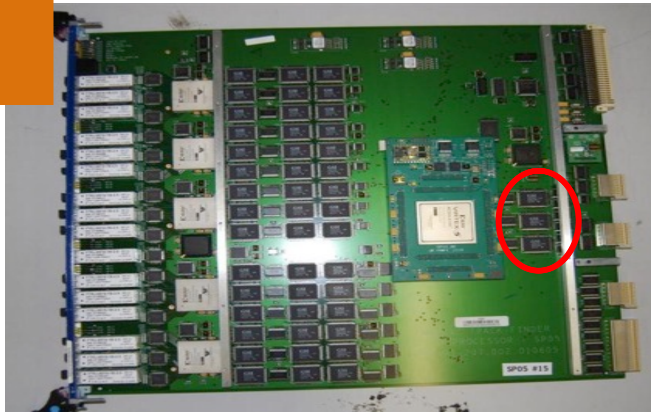
- The muon P_T for previous is calculated from a memory look-up table
  - A "cheat" to do the calculation quickly (~50ns) in the L1 trigger.
  - Don't really calculate it online at all (no CPU involved real-time)
  - Instead, pre-calculate offline the muon momentum using whatever algorithm you want and with however much computing resources you have!
    - But you must do this for every possible input to the memory
- The challenge:
  - You must squeeze all the data for your track fit into the memory address
    - N bits of of data requires a memory of 2^N addresses

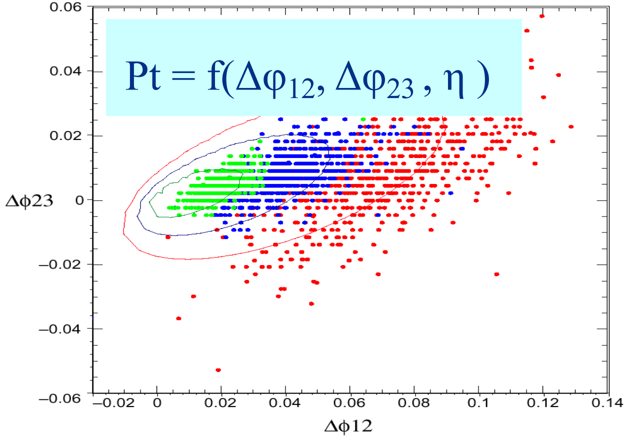


Version 1: CSC Track-Finder, 2005-2015

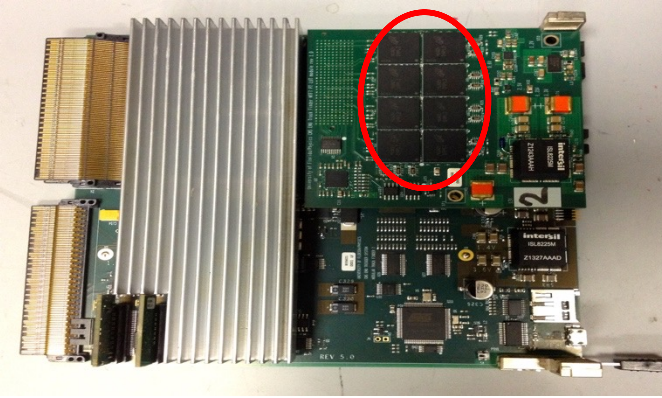
- 12 VME processors
  - Xilinx Virtex-5 FPGAs and memory
- P_T calculated from an external SRAM memory look-up table
  - Largest available at time to do the job: 4MB → 22 bit address space
- Algorithm
  - Likelihood-based fit using Δϕ bending between at most 3 detector stations to assign p_T
  - Multiple scattering in iron carries momentum information in addition to magnetic bending

Too large for FPGA logic





Version 2: Endcap Muon Track-Finder, 2016+



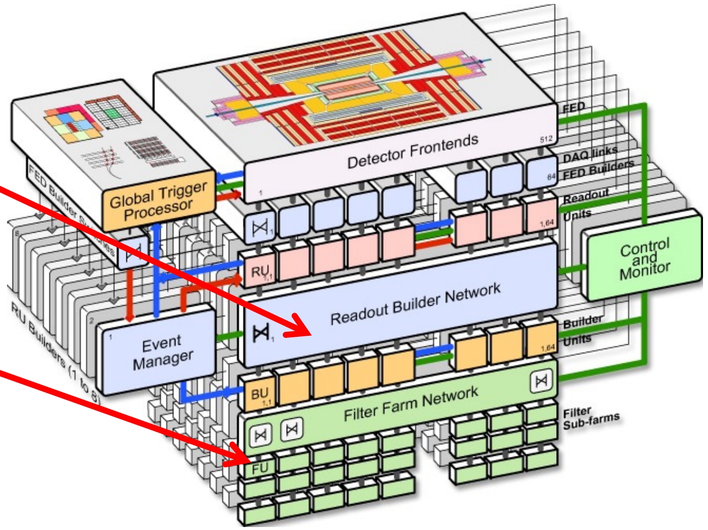
- 12 µTCA double-module processors
  - Xilinx Virtex-7 FPGA and memory
- P_T calculated from Reduced Latency DRAM
  - 1 GB → 30 bit address space
  - +8 address bits (only) over previous CSCTF
- Algorithm
  - Machine Learning: Boosted Decision Trees (BDTs) used for regression to assign P_T
  - More data: Can use Δφ bending between 4 detector stations, and Δη, and bend angle in first station
  - But note, as before, algorithm is run offline and stored in memory

ACAT17, CMS-CR-2017-35

Fermilab

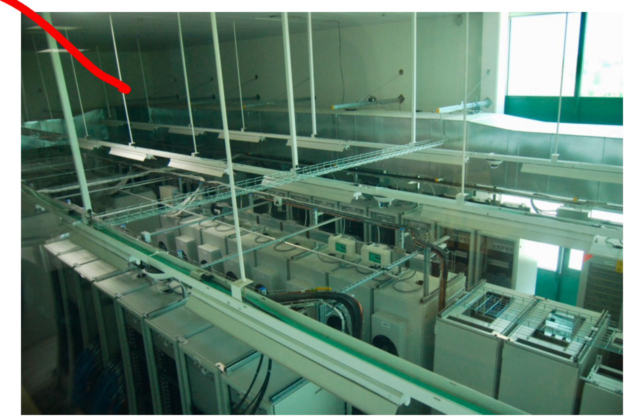
CMS High Level Trigger

- System accepts up to 100 kHz event rate (100 GB/s bandwidth) from detector electronics (“FEDs”)
- Event builder
  - Run 1: Myrinet network switch
  - Run 2: Infiniband network switches
- High Level Trigger
  - ~26 000 CPU cores
  - Software-based algorithms
  - Output bandwidth ~GB/s
  - Average selection rate ~1000 Hz
  - ~300 ms/event processing time available for 100 kHz input rate



Fermilab

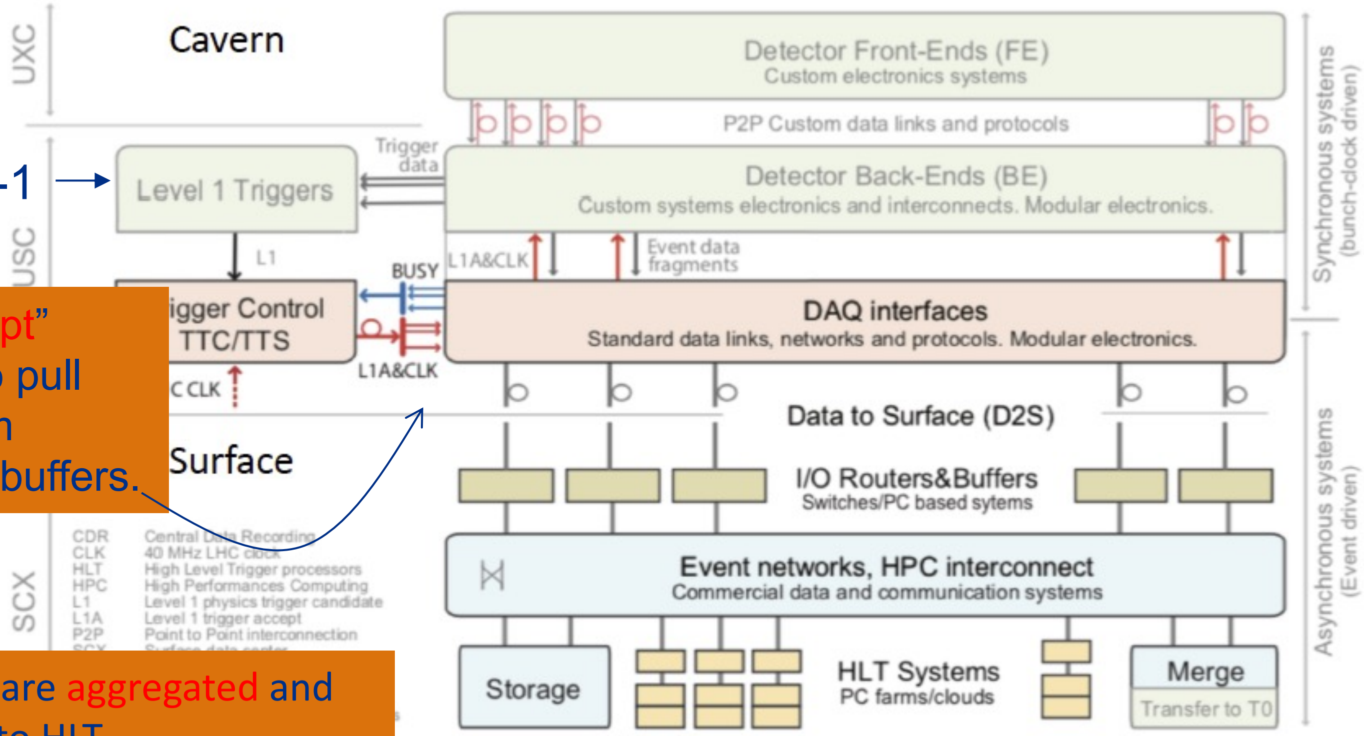
HLT Implemented in a Local Computer Center



Computers get the best seat in the (CMS) house!

CMS DAQ Architecture

Level-1 →



"L1 Accept" issued to pull data from detector buffers.

Data are aggregated and sent to HLT

When selected by an HLT node, event data sent to a storage system

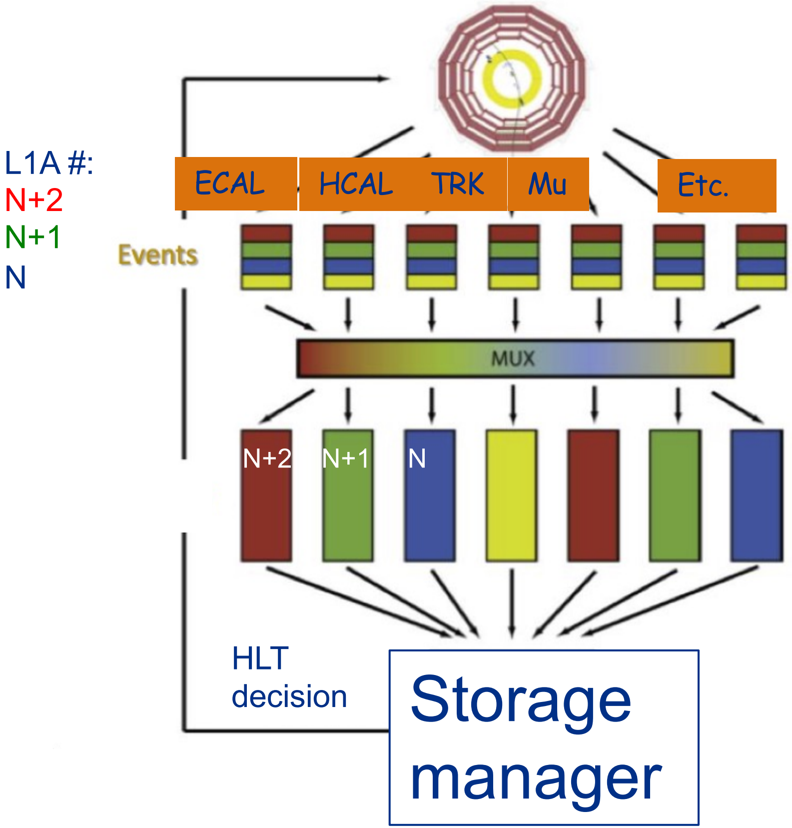
Detector data

Event Builder

HLT

Fermilab

DAQ Event Building



- Fragments of events from different detectors and regions received at front-end
- Must aggregate data sent from different detector readouts from a specific L1 triggered bunch crossing:
  - “event building”
- Send full event data to a target processor destination in a round-robin fashion
- This is an asynchronous process (unlike Level-1, which is synchronous).

Fermilab

What does HLT do?

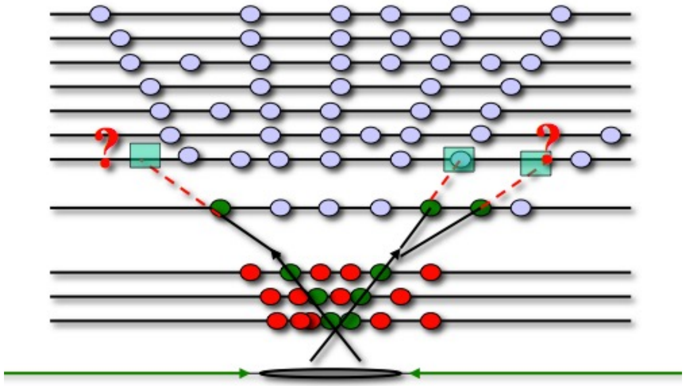
- Reduces rate from 100 kHz to ~1 kHz (LHC Run 2)
  - The output rate is set by offline computing data processing constraints, and data storage limits
- But the HLT starts with a data sample already enriched in physics!
  - Level-1 already applied a factor 400 background rejection, and still need to find another factor 100 reduction
- What else can HLT do?
  - Work with higher precision data than that used by Level-1
    - No I/O limitations, and uses same raw data as used offline
    - Algorithms very close to offline analyses
  - Include new detectors not used by L1 →

HLT Tracking

- New detector additions: the precision silicon pixel and strip tracking systems
  - Readout is too slow for the Level-1 trigger in CMS and ATLAS LHC experiments
    - A major “downgrade” from the previous Tevatron experiments, which did have tracking at Level-1 (but with looser timing constraints, and different trackers)
  - Much higher precision on the momentum of charged particles
    - 2% vs. 20% for muons
  - Much better rejection against fake “electrons” from $\pi^0 \rightarrow \gamma\gamma$ decays (with addition of a track requirement)
  - Better precision of jet energies (when using “Particle Flow”)

Fermilab

HLT Tracking at CMS

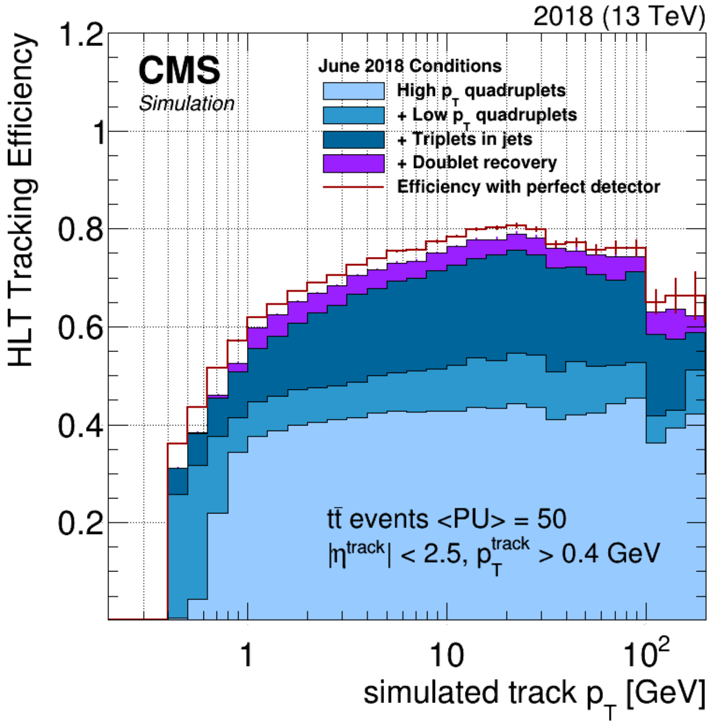


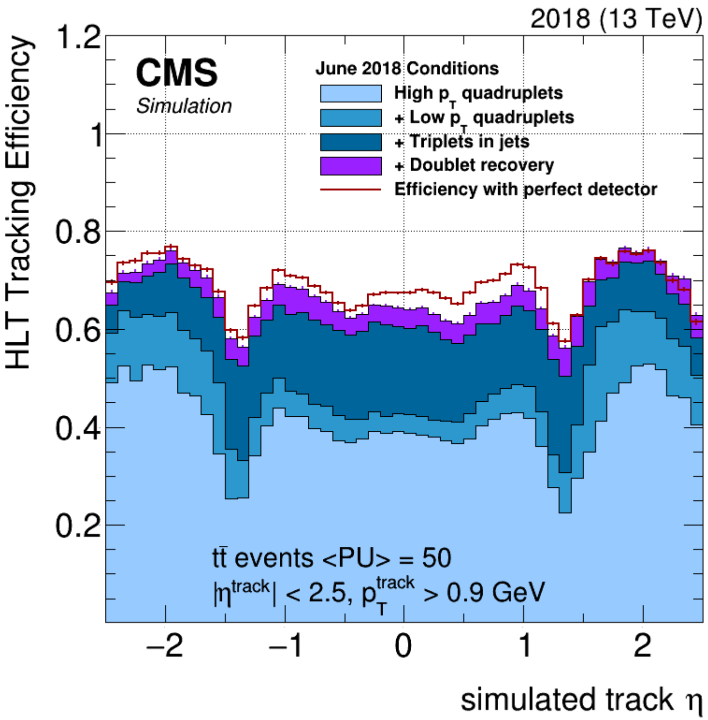
- Tracking in the HLT is performed iteratively, starting with tight requirements for the track seeds, which become looser for each subsequent iteration.
- Hits in the tracking detectors already used in a track are removed at beginning of the next it…
- HLT consists of three itera…
  - The first two re… …imum of four consecutive hits in the pixel… …eed the tracking.
  - T… …ration relaxes the requirement on the number of … the track seeds to three and is restricted to the vicinity of jet candidates identified from calorimeter information and the tracks reconstructed in the two previous iterations.

Emphasis on algorithm speed

HLT Tracking Efficiency at CMS

– Efficiency for simulated top pair events





“Particle Flow” at HLT

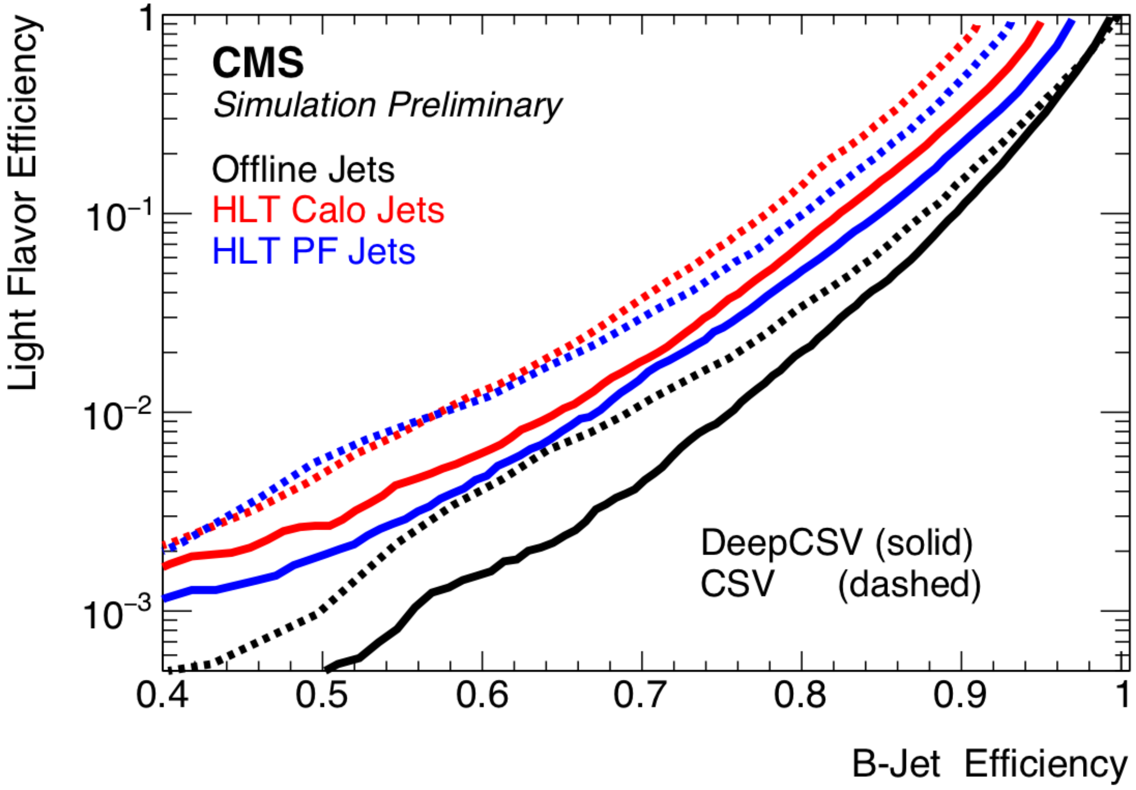
- The precision of charged particle tracking measurements of momentum exceeds that of calorimeter energy measurements at lower momenta
  - Therefore, combine them in an optimal way
- This allows for the best energy resolution on jets, and missing transverse momentum
- Requires mapping all charged tracks to energy clusters in the calorimeter to produce “particle flow” objects
  - ECAL only deposits → electrons
  - ECAL+HCAL → charged hadrons
  - ECAL only, no track → photons
  - ECAL+HCAL, no track → neutral hadrons
- This is now at HLT for CMS, as well as for offline analyses
  - Improves jet energy resolution (and combination of jets), and missing transverse energy resolution

Other HLT Algorithms

- More sophisticated algorithms can be run at HLT
  - B jet tagging (from track impact parameter measurements)
  - Jet algorithms (including jet substructure)
  - Lepton isolation using tracks from primary vertex
  - Topology and invariant mass cuts

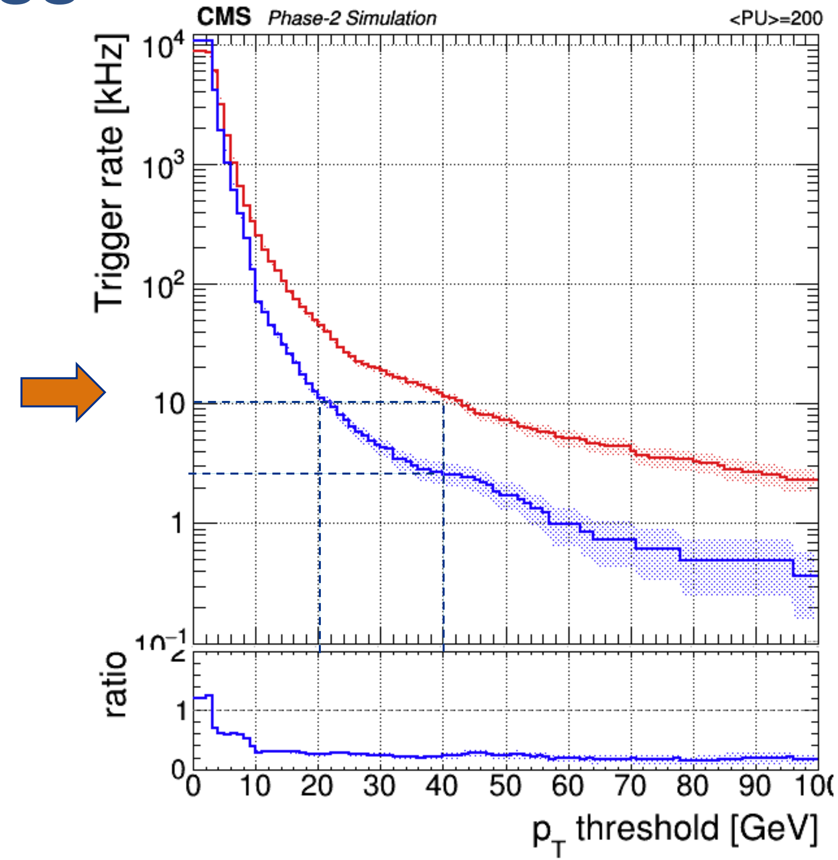
- However, note that the “KISS” principle applies here.
  - The simpler the trigger with which you can get by, the more efficient and more comprehensive it will be for physics!

B-tagging at HLT



Fermilab

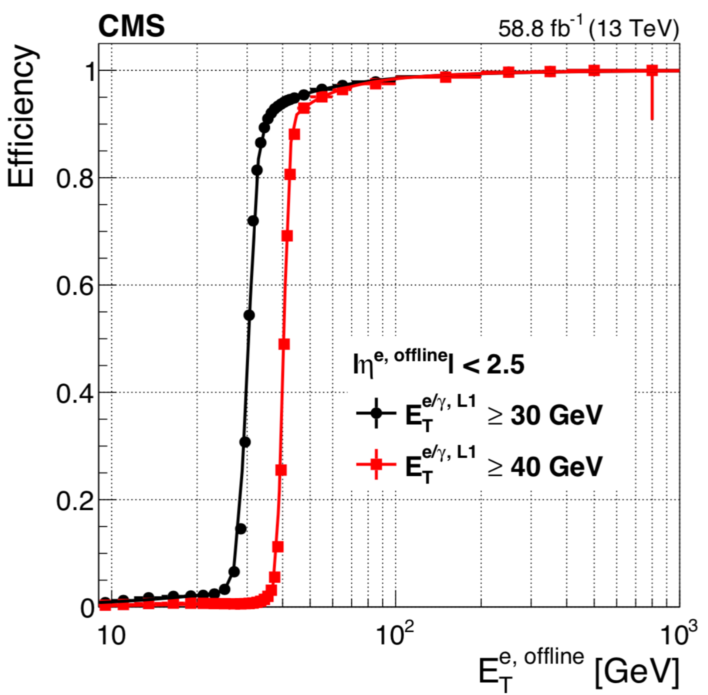
Trigger Rate



- Adjust a **threshold** for the rate you can afford to store
- Trigger rate is typically dominated by background
- These are simulated rate curves for a Level-1 single muon trigger at HL LHC
  - Current
  - Some improved algorithm
- Want a "knob" to be able to control the rate, albeit by raising the P_T threshold here
  - Thus need good P_T resolution

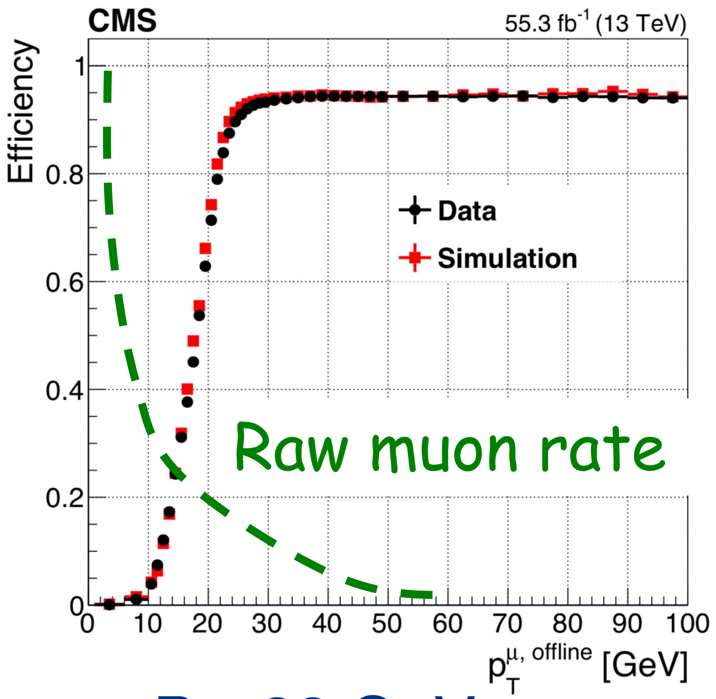
Trigger Efficiency for Electrons and Muons

electron/photon trigger



P_T >30 or 40 GeV

Muon trigger



P_T>22 GeV

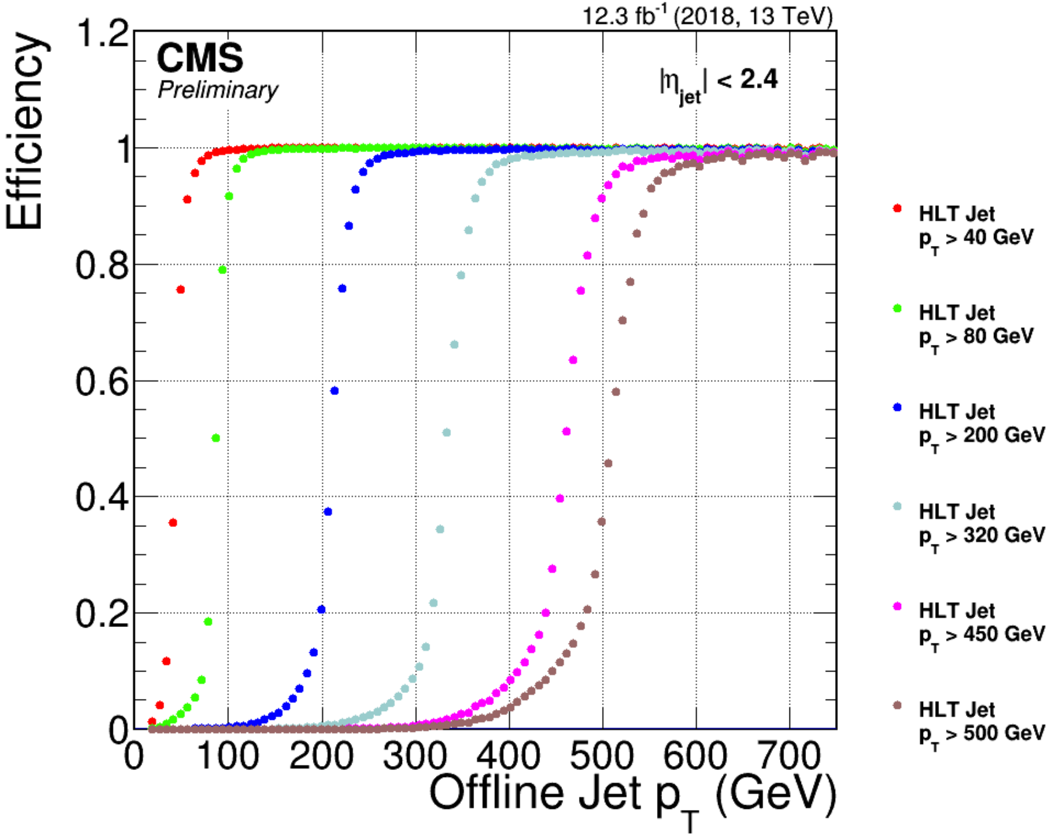
- Want sharp turn-on curves, and high efficiency plateaus that are flat
  - Why?

Sharper turn-on: less rate from mis-measured tracks at low PT

Flat, high efficiency plateau for event yield, less uncertainty

Fermilab

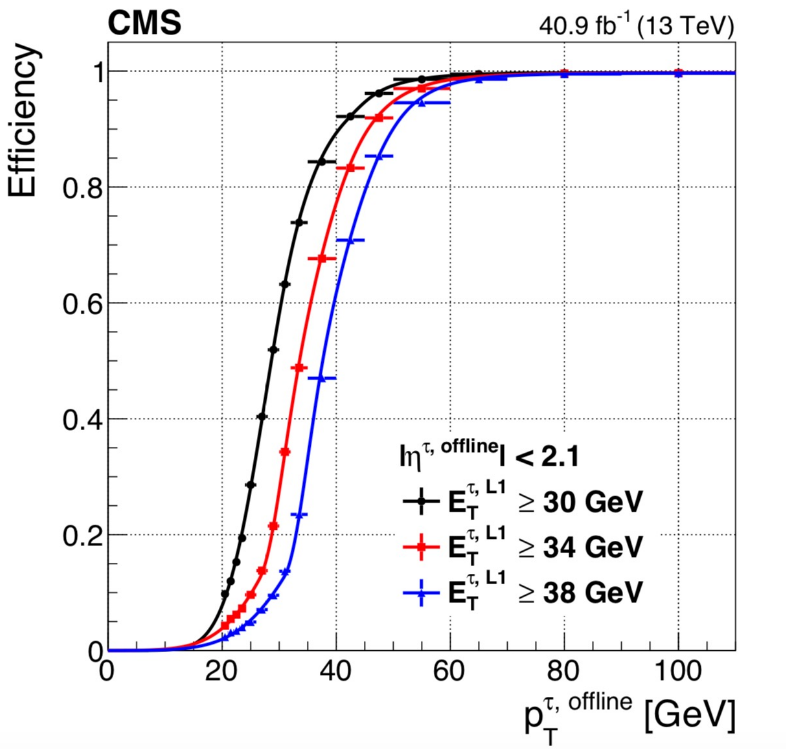
Trigger Efficiency for Jets



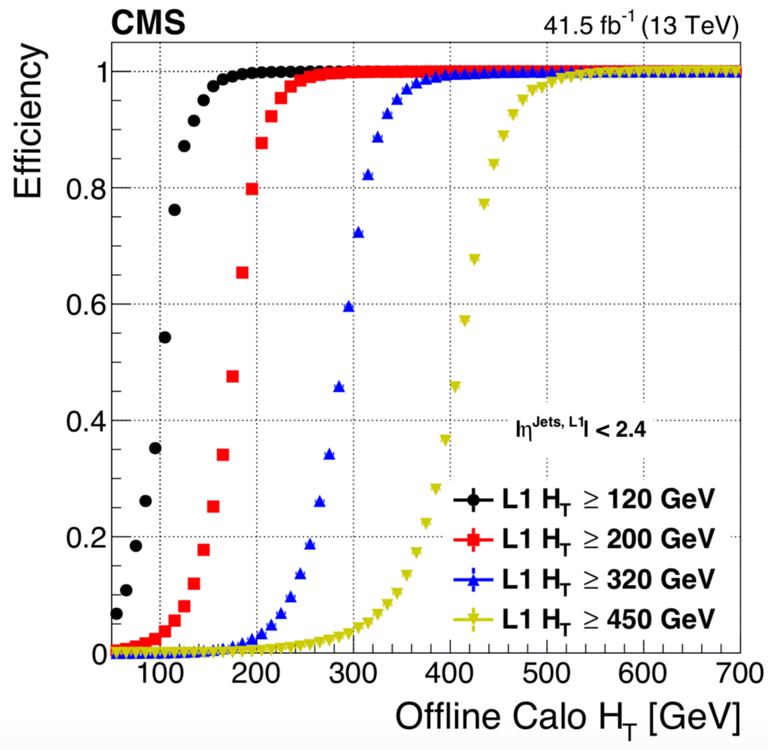
Fermilab

More Efficiency examples

Tau trigger

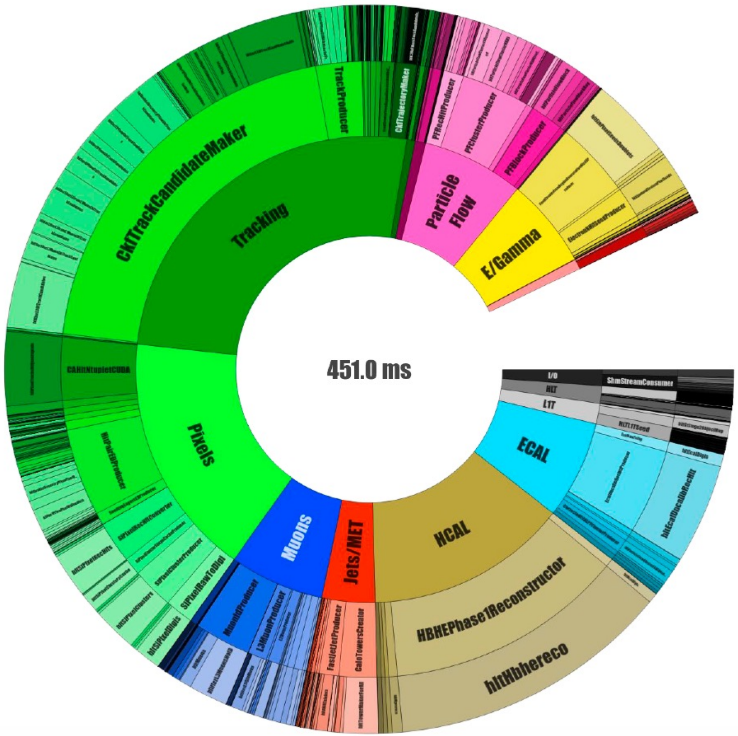


$H_T = \Sigma$ jet E_T trigger



HLT Timing

Timing of Run-3 HLT menu - CPU-only



Limits what can be done at HLT, so algorithms also must be fast

The pie-chart shows the distribution of CPU time in different instances of CMSSW modules (outermost ring), their corresponding C++ class (one level inner), grouped by physics object or detector (innermost ring). The empty slice indicates the time spent outside of the individual algorithms.

The HLT configuration is based on the 2018 definition, with minimal updates to the local reconstruction to reflect the ongoing developments foreseen for Run 3. Further updates and improvements are expected before the start of Run 3.

The timing is measured on pileup 50 events from Run2018D on a full HLT node (2x Intel Skylake Gold 6130) with HT enabled, running 16 jobs in parallel, with 4 threads each.

Tracking algorithms dominate

Measuring Trigger Efficiency

- Use events selected by an orthogonal trigger to measure performance of your trigger
  - e.g. select events satisfying requirements of a multijet trigger to measure the efficiency of a missing transverse momentum trigger
- ”Tag and probe” is often used to measure trigger efficiencies of leptons
  - e.g. select one lepton in a Z→ll decay to satisfy a trigger, measure efficiency of other lepton to fire a trigger
- Can also use looser, ”prescaled” triggers, to measure efficiency of a more complex trigger
  - Prescaling means only 1 in N events satisfying such a trigger are actually recorded (reduces rate by 1/N)
  - e.g. single lepton trigger to measure efficiency of leptons+jets
  - Disadvantage is collecting enough statistics, since effective luminosity is reduced

Trigger Menus

- A trigger "menu" represents a (large) set of selection criteria for the broad physics program of the experiment, e.g.
  - Entrée 1: 2 muons of opposite charge, one with P_T>17 GeV and one with P_T>8 GeV
  - Entrée 2: "jets" with a total scalar jet H_T > 500 GeV
    - These entrées are often referred to as "trigger lines"
- Crudely speaking, O(1) trigger per analysis topic
  - Some triggers are very general and serve many analyses (preference)
  - But often many "backup" triggers are required for the control regions of specific analyses, which increases the total count
- Separate L1 and HLT menus
  - L1 menu has ~300 items for CMS, ~500 for ATLAS
  - HLT menu has ~600 menu items for CMS, ~1500 for ATLAS
    - Each trigger item has prerequisite L1 "seeds"

Example: Atlas Menu

Trigger	Typical offline selection	Trigger Selection: L1 [GeV]	Trigger Selection: HLT [GeV]	L1 Peak Rate [kHz]	HLT Peak Rate [Hz]
				$L=2.0\times10^{34}$ $cm^{-2}s^{-1}$	
Single leptons	Single isolated μ, $p_T > 27$ GeV	20	26 (i)	16	218
	Single isolated tight e, $p_T > 27$ GeV	22 (i)	26 (i)	31	195
	Single μ, $p_T > 52$ GeV	20	50	16	70
	Single e, $p_T > 61$ GeV	22 (i)	60	28	20
	Single τ, $p_T > 170$ GeV	100	160	1.4	42
Two leptons	Two μ, each $p_T > 15$ GeV	2×10	2×14	[illegible]	[illegible]
	Two μ, $p_T > 23, 9$ GeV	20	22, 8	[illegible]	[illegible]
	Two very loose e, each $p_T > 18$ GeV	2×15 (i)	[illegible]	[illegible]	[illegible]
	One e & one μ, $p_T > 8, 25$ GeV	20 (μ)	[illegible]	[illegible]	[illegible]
	One loose e & one μ, $p_T > 18, 15$ GeV	15, 10	[illegible]	[illegible]	[illegible]
	One e & one μ, $p_T > 27, 9$ GeV	[illegible]	[illegible]	[illegible]	[illegible]
	Two τ, $p_T > 40, 30$ GeV	[illegible]	[illegible]	[illegible]	[illegible]
	One τ & one isolated μ, $p_T > 30, 15$ GeV	[illegible]	[illegible]	[illegible]	[illegible]
	One τ & one isolated e, $p_T > 30, 18$ G[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Three leptons	Three very loose e, $p_T > 25$[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	Three μ, each $p_T > 7$[illegible]	[illegible]	[illegible]	[illegible]	7
	Three μ, p_T[illegible]	[illegible]	[illegible]	[illegible]	9
	Two[illegible]	[illegible]	[illegible]	2.2	0.5
	[illegible]	[illegible]	[illegible]	2.3	0.1
Sig[illegible]	[illegible]	[illegible]	140	24	47
[illegible]	[illegible]	[illegible]	2×50	3.0	7
	[illegible]	[illegible]20	35, 25	3.0	21
	[illegible]	2×15 (i)	2×20 (i)	2.0	15
[illegible]	[illegible]	100	420	3.7	35
	[illegible]	111 (topo: $R = 1.0$)	460	2.6	42
	[illegible]$_{jet} > 45$ GeV	111 (topo: $R = 1.0$)	420, $m_{jet} > 35$	2.6	36
b-je[illegible]	[illegible]285 GeV	100	275	3.6	15
	[illegible]), $p_T > 185, 70$ GeV	100	175, 60	3.6	11
	[illegible]($\epsilon = 40\%$) & three jets, each $p_T > 85$ GeV	4×15	4×75	1.5	14
	Two b ($\epsilon = 70\%$) & one jet, $p_T > 65, 65, 160$ GeV	$2\times30, 85$	$2\times55, 150$	1.3	17
	Two b ($\epsilon = 60\%$) & two jets, each $p_T > 65$ GeV	4×15, $\lvert\eta\rvert < 2.5$	4×55	3.2	15
Multijets	Four jets, each $p_T > 125$ GeV	3×50	4×115	0.5	16
	Five jets, each $p_T > 95$ GeV	4×15	5×85	4.8	10
	Six jets, each $p_T > 80$ GeV	4×15	6×70	4.8	4
	Six jets, each $p_T > 60$ GeV, $\lvert\eta\rvert < 2.0$	4×15	6×55, $\lvert\eta\rvert < 2.4$	4.8	15
E_T^{miss}	$E_T^{miss} > 200$ GeV	50	110	5.1	94
B-physics	Two μ, $p_T > 11, 6$ GeV, $0.1 < m(\mu,\mu) < 14$ GeV	11, 6	11, 6 (di-μ)	2.9	55
	Two μ, $p_T > 6, 6$ GeV, $2.5 < m(\mu,\mu) < 4.0$ GeV	2×6 (J/ψ, topo)	2×6 (J/ψ)	1.4	55
	Two μ, $p_T > 6, 6$ GeV, $4.7 < m(\mu,\mu) < 5.9$ GeV	2×6 (B, topo)	2×6 (B)	1.4	6
	Two μ, $p_T > 6, 6$ GeV, $7 < m(\mu,\mu) < 12$ GeV	2×6 (Υ, topo)	2×6 (Υ)	1.2	12
Main Rate				86	1750
B-physics and Light States Rate					200

512 L1 items + O(1500) HLT chains to maximize the reach of ATLAS physics program

Emma Torró, “The ATLAS trigger menu: from Run 2 to Run 3”, LHCP2020 poster

Example: CMS HLT Menu snippet

	stream	dataset	path	group
1	stream	dataset	path	group
2				
490	PhysicsMuons	MuOnia	HLT_Mu7p5_Track7_Upsilon_v10	BPH
491	PhysicsMuons	MuOnia	HLT_Trimuon5_3p5_2_Upsilon_Muon_v4	BPH
492	PhysicsMuons	MuOnia	HLT_TrimuonOpen_5_3p5_2_Upsilon_Muon_v2	BPH
493	PhysicsMuons	MuonEG	HLT_DiMu9_Ele9_CaloIdL_TrackIdL_DZ_v15	SUS,HIG,SMP,EXO
494	PhysicsMuons	MuonEG	HLT_DiMu9_Ele9_CaloIdL_TrackIdL_v15	SUS,HIG,SMP
495	PhysicsMuons	MuonEG	HLT_DoubleMu20_7_Mass0to30_L1_DM4EG_v6	SMP,BPH
496	PhysicsMuons	MuonEG	HLT_DoubleMu20_7_Mass0to30_L1_DM4_v6	SMP,BPH
497	PhysicsMuons	MuonEG	HLT_DoubleMu20_7_Mass0to30_Photon23_v6	SMP,BPH
498	PhysicsMuons	MuonEG	HLT_Mu12_DoublePhoton20_v3	SMP
499	PhysicsMuons	MuonEG	HLT_Mu12_TrkIsoVVL_Ele23_CaloIdL_TrackIdL_IsoVL_DZ_v13	TOP,SUS,SMP,EXO
500	PhysicsMuons	MuonEG	HLT_Mu12_TrkIsoVVL_Ele23_CaloIdL_TrackIdL_IsoVL_v5	TOP,SUS,SMP,EXO
501	PhysicsMuons	MuonEG	HLT_Mu17_Photon30_IsoCaloId_v4	SUS
502	PhysicsMuons	MuonEG	HLT_Mu23_TrkIsoVVL_Ele12_CaloIdL_TrackIdL_IsoVL_DZ_v13	TOP,SUS,SMP,B2G,EXO
503	PhysicsMuons	MuonEG	HLT_Mu23_TrkIsoVVL_Ele12_CaloIdL_TrackIdL_IsoVL_v5	TOP,SUS,SMP,B2G,EXO
504	PhysicsMuons	MuonEG	HLT_Mu27_Ele37_CaloIdL_MW_v3	B2G
505	PhysicsMuons	MuonEG	HLT_Mu37_Ele27_CaloIdL_MW_v3	B2G
506	PhysicsMuons	MuonEG	HLT_Mu43NoFiltersNoVtx_Photon43_CaloIdL_v4	EXO
507	PhysicsMuons	MuonEG	HLT_Mu48NoFiltersNoVtx_Photon48_CaloIdL_v4	EXO
508	PhysicsMuons	MuonEG	HLT_Mu8_DiEle12_CaloIdL_TrackIdL_DZ_v16	HIG,SUS,SMP,EXO
509	PhysicsMuons	MuonEG	HLT_Mu8_DiEle12_CaloIdL_TrackIdL_v16	HIG,SUS,SMP
510	PhysicsMuons	MuonEG	HLT_Mu8_Ele8_CaloIdM_TrackIdM_Mass8_PFHT350_DZ_v17	SUS
511	PhysicsMuons	MuonEG	HLT_Mu8_Ele8_CaloIdM_TrackIdM_Mass8_PFHT350_v17	SUS
512	PhysicsMuons	MuonEG	HLT_Mu8_TrkIsoVVL_Ele23_CaloIdL_TrackIdL_IsoVL_DZ_v11	HIG,SUS,SMP,B2G,TOP,EXO
513	PhysicsMuons	MuonEG	HLT_Mu8_TrkIsoVVL_Ele23_CaloIdL_TrackIdL_IsoVL_v9	HIG,SUS,SMP,B2G,TOP,EXO
514	PhysicsMuons	SingleMuon	HLT_IsoMu20_eta2p1_LooseChargedIsoPFTau27_eta2p1_CrossL1_v10	TAU,HIG,SUS
515	PhysicsMuons	SingleMuon	HLT_IsoMu20_eta2p1_LooseChargedIsoPFTau27_eta2p1_TightID_CrossL1_	TAU,HIG,SUS
516	PhysicsMuons	SingleMuon	HLT_IsoMu20_eta2p1_MediumChargedIsoPFTau27_eta2p1_CrossL1_v10	TAU,HIG,SUS
517	PhysicsMuons	SingleMuon	HLT_IsoMu20_eta2p1_MediumChargedIsoPFTau27_eta2p1_TightID_CrossL	TAU,HIG,SUS

Outline – Trigger

- What is a trigger and DAQ system, and why needed?
- Level-1 Triggers
- High Level Triggers
- Trigger Rate
- Trigger Efficiency

Day 1

- Trigger Menus
- Data parking and data scouting
- Future Outlook
  - HL-LHC trigger upgrade
  - AI in triggers
  - TDAQ for Future Colliders

Day 2

Fermilab

Designing a Trigger Menu

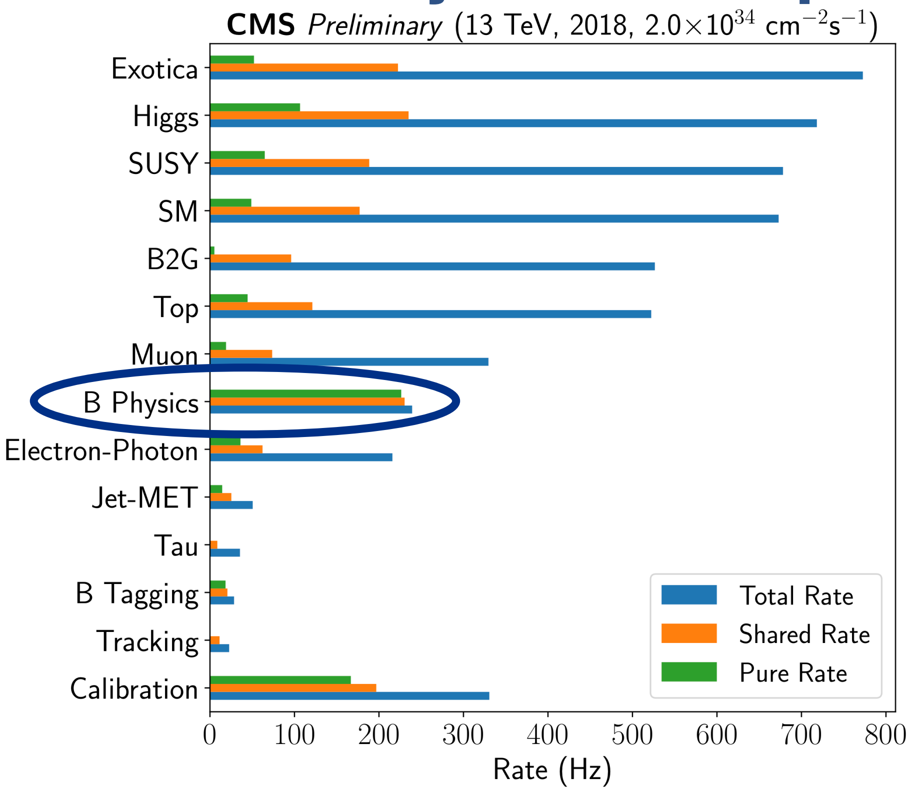
- **Goal**: Highest efficiency for your overall physics program at a recording rate that fits your budget
- Simple inclusive triggers could catch the most range of physics topics (e.g. single lepton trigger), but will have large rates for low thresholds
- Complex triggers (e.g. two leptons and two jets) will have lower rates at low thresholds, but also lower efficiency. Also it may be difficult to measure their efficiency
  - e.g. efficiency = $\varepsilon^2_{lep} * \varepsilon^2_{jet}$
  - Simulations are not perfect, so it's important to validate in data the efficiency of your trigger
- Menu composition is often an art more than an exact science, with multiple ways to successfully construct. Also can evolve with time and priorities

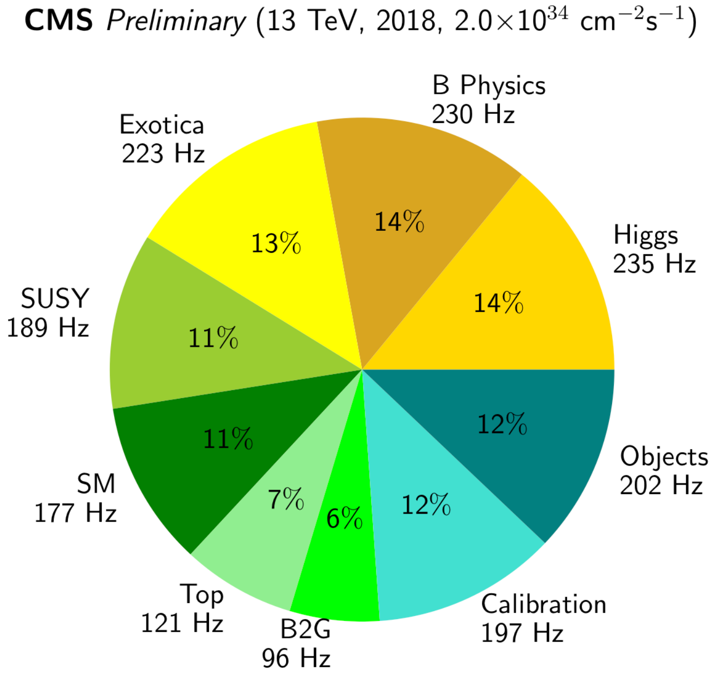
Specific Triggers from CMS

- Muons
  - Single isolated muon: $P_T > 24$ GeV
  - Double muon: $P_T > 17$, $P_T > 8$ GeV
  - Triple muon: $P_T > 12$, $P_T > 10$, $P_T > 8$ GeV
- Electrons
  - Single isolated electron: $P_T > 35$ GeV
  - Double electron: $P_T > 23$, $P_T > 12$ GeV
  - Triple electron: $P_T > 16$, $P_T > 12$, $P_T > 8$ GeV
- Jets
  - Single jet: $E_T > 200$ GeV
  - Quadjet: $E_T > 110$, $E_T > 90$, $E_T > 80$, $E_T > 15$ GeV
  - Sum of jet E_T : $H_T > 500$ GeV
- MET
  - Missing $E_T > 250$ GeV

Why multi-object triggers?

Menu and Physics Groups





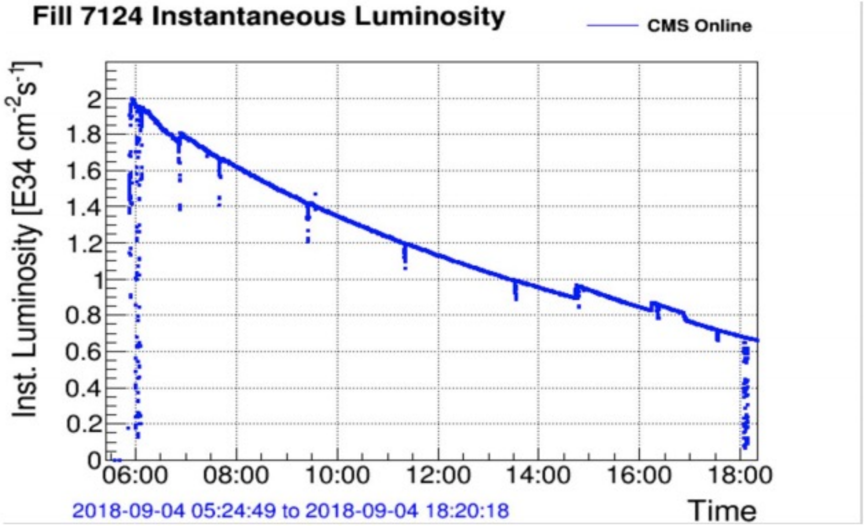
Note also rate for calibration and object studies

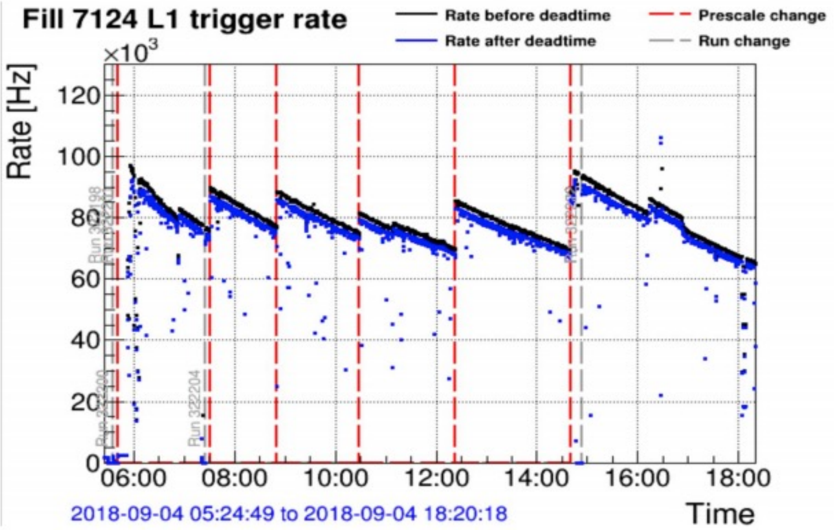
Total rate: If group triggered alone
Shared: Estimate of shared amount
Pure: Exclusive rate not used by another

Pre-scaling

- One solution to reduce the trigger rate is to just not accept every even that satisfies a trigger condition
- A “prescale” factor N means that only 1 of every N events selected by an algorithm is actually recorded
  - Essentially the Nth event satisfying a trigger
- What is the downside? Well, the collected luminosity is only L_{int}/N. You don’t get full delivered luminosity.
- But sometimes that is okay!
  - Maybe you take what you can get if there is no other way to reduce the rate and keep all your events
  - e.g. B physics triggers at a hadron collider

Pre-scaled Triggers



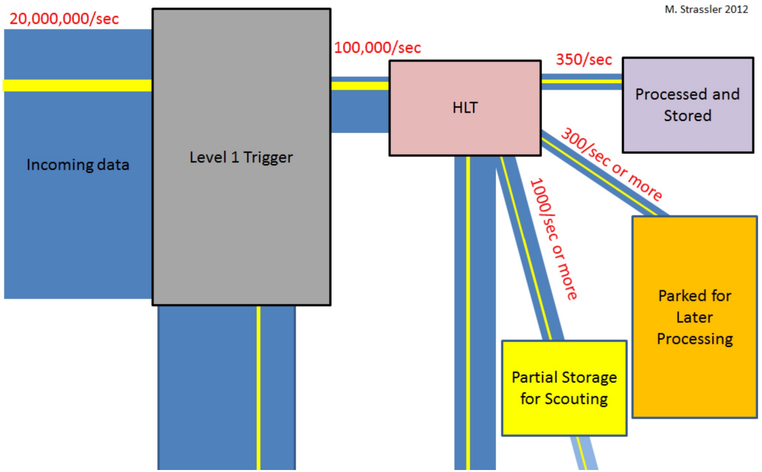


- As luminosity drops, one can afford to loosen triggers by reducing/removing prescale factors on some trigger lines
  - Increase trigger rate back to DAQ limit

Fermilab

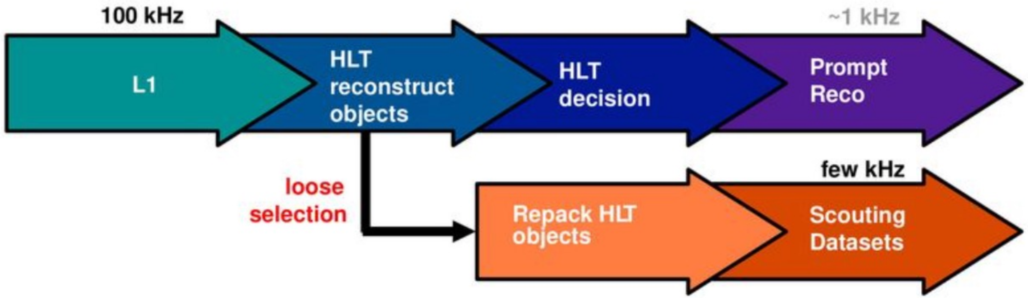
Data Parking

- Recall why a trigger is needed
  - Too much data to continuously stream to disk for storage and/or computer processing reasons
- If we can at least handle the storage rate, maybe we can record data at an even higher rate to disk, but postpone processing that data until experiment is no longer running
- This is known as “data parking”
  - CMS does this trick to record more data for B physics studies
- Can take advantage of long shutdown periods
  - For example, during the LHC 2-year long shutdowns



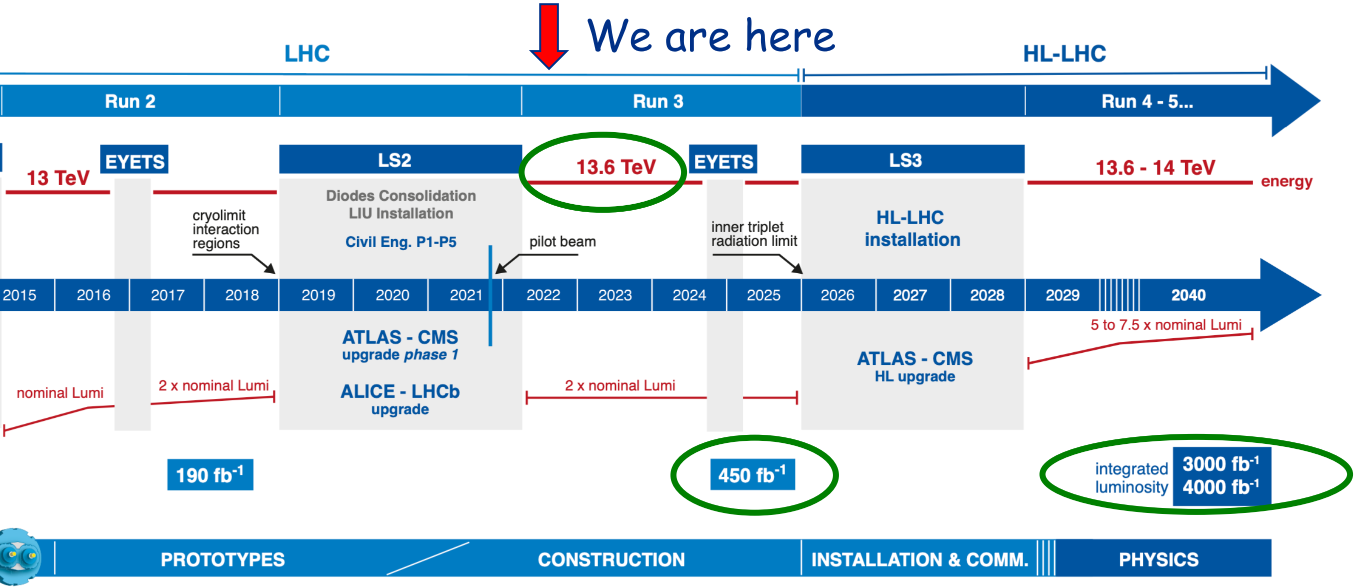
Data Scouting

- The limit on DAQ is the bandwidth of the data to record to disk (how many GB/s) not really the event rate per se
- "Data scouting" is a recent invention by the LHC experiments to store only a small summary of reconstructed event quantities, and not all the raw data, in order to record a higher rate of events
  - Allows much lower trigger thresholds, and thus a higher acceptance of a physics process
  - Generally does not allow reprocessing data afterward (from new calibrations or alignments)



FUTURE OF TRIGGER-DAQ

The High Luminosity LHC



$L \rightarrow 7.5 \times 10^{34}$ Hz/cm^2
Pileup: 140-200

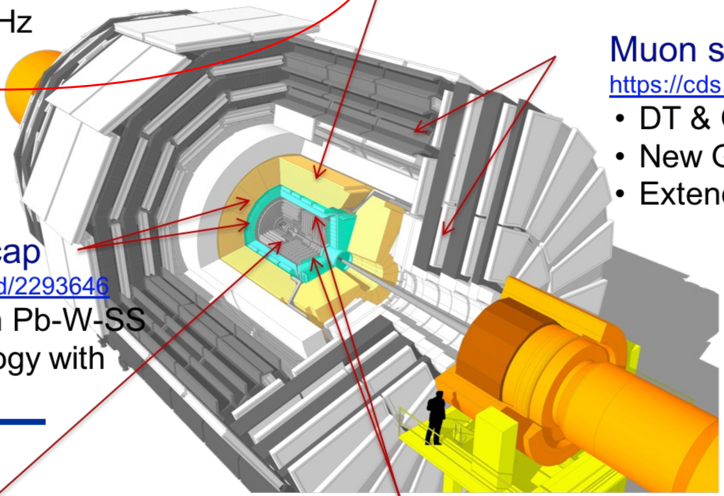
4X higher luminosity and pileup, accrue 10X larger data sample, add new high granularity detectors to expts.

HL-LHC CMS Upgrade Plans

L1-Trigger/HLT/DAQ
https://cds.cern.ch/record/2283192
https://cds.cern.ch/record/2283193
- Tracks in L1-Trigger at 40 MHz for 750 kHz PFlow-like selection rate
- HLT output 7.5 kHz

Barrel Calorimeters
https://cds.cern.ch/record/2283187
- ECAL crystal granularity readout at 40 MHz with precise timing for e/γ at 30 GeV
- ECAL and HCAL new Back-End boards

Muon systems
https://cds.cern.ch/record/2283189
- DT & CSC new FE/BE readout
- New GEM/RPC 1.6 < η < 2.4
- Extended coverage to η ≃ 3



Calorimeter Endcap
https://cds.cern.ch/record/2293646
- Si, Scint+SiPM in Pb-W-SS
- 3D shower topology with precise timing

Beam Radiation Instr. and Luminosity, and Common Systems and Infrastructure
https://cds.cern.ch/record/2020886

Tracker https://cds.cern.ch/record/2272264
- Si-Strip and Pixels increased granularity
- Design for tracking in L1-Trigger
- Extended coverage to η ≃ 3.8

MIP Timing Detector
https://cds.cern.ch/record/2296612
- ≃ 30 ps resolution
- Barrel layer: Crystals + SiPMs
- Endcap layer: Low Gain Avalanche Diodes

New detector systems with higher granularity to handle unprecedented pileup and radiation

Endcap calorimeter, Triggerable silicon tracker, Timing detector, more muon coverage, electronics

HL-LHC Trigger Challenge

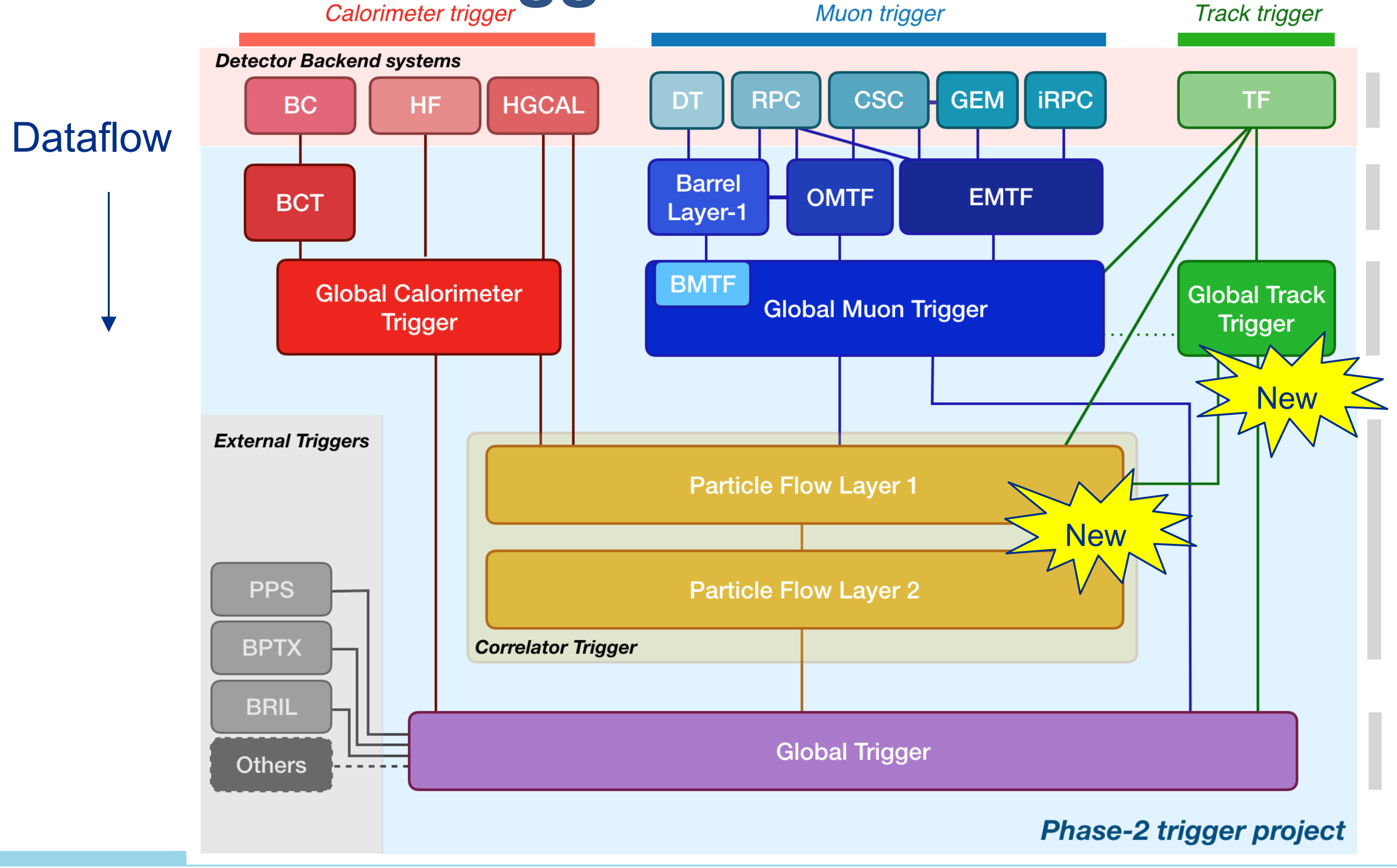
- Cope with higher collision rate and particle densities, yet maintain thresholds in trigger menu similar to LHC in order to retain sensitivity to electroweak scale physics
  - Want to benefit from the increased luminosity
  - Must improve rate reduction at no cost to efficiency
    - Need better trigger algorithms
- Add sensitivity to new physics scenarios, such as from long-lived heavy particles
  - Displaced “muons” from new particle decays
    - Particles displaced from the nominal collision vertex
  - Heavy stable charged particles
    - Muon-like particles traveling with $\beta < c$ and highly ionizing

Fermilab

CMS HL-LHC Trigger Requirements

- Increase Level-1 Trigger output rate: 0.1 →0.75 MHz
  - Better sharing of rate reduction between Level-1 and HLT
- Increase Level-1 decision latency: 4μs → 12.5μs
  - More processing time for more complex algorithms
    - Both requirements imply changes to detector front-end electronics (deeper data buffers, higher bandwidth)
- Adding tracking (silicon strips) to Level-1
  - Better performing Level-1 Trigger
  - Recovers capability that Tevatron experiments had a generation ago (albeit with much less data throughput demands)
  - Major new change to make silicon detector systems fast enough for trigger purposes, and yet still able to power and cool
- Increase HLT DAQ storage rate to disk: 1 → 7.5 kHz

CMS HL-LHC Trigger Architecture



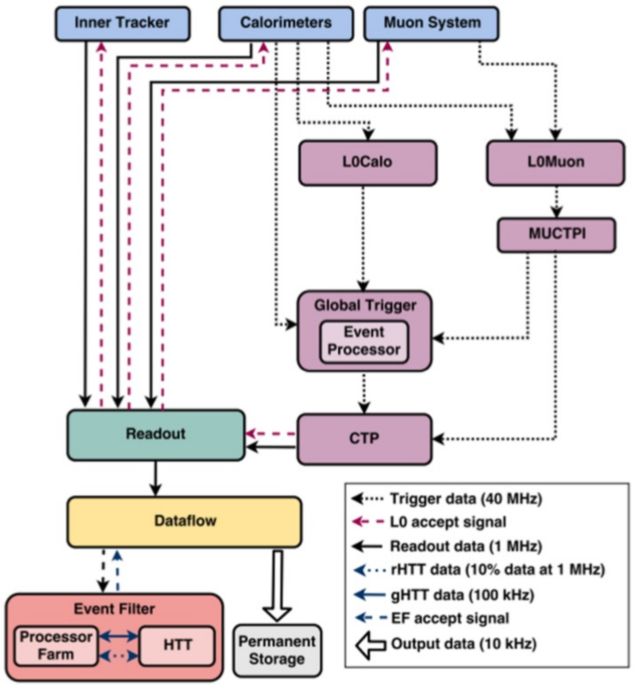
← Detector upgrades

Match tracks with calorimeter clusters and muons

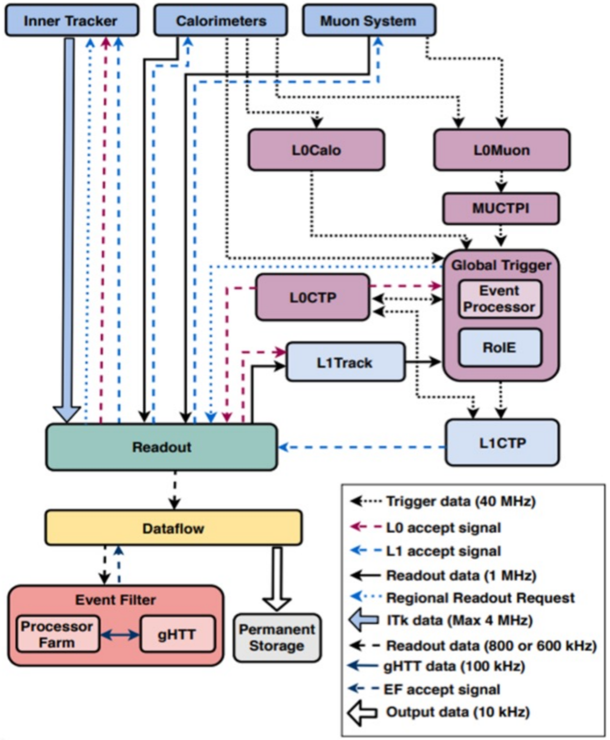
CMS TDR 021

ATLAS HL-LHC Trigger Architecture

Baseline Architecture

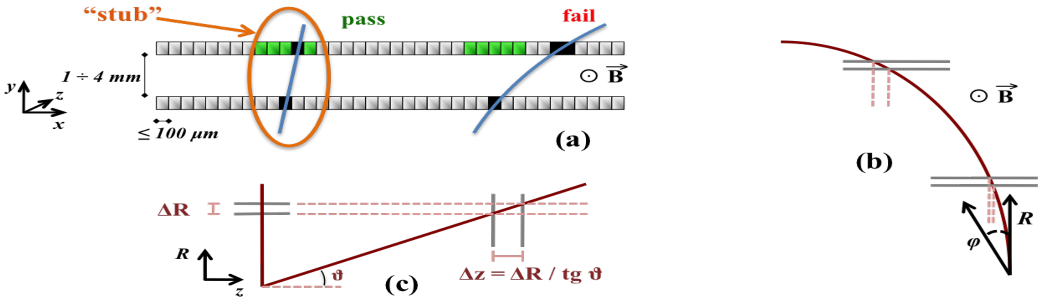


Evolved Architecture



Using Tracker “Stubs”

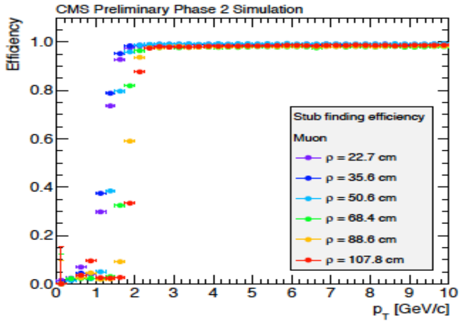
- Coincidence of hits in a double layer of silicon greatly reduces data bandwidth from detector
- Require a coincidence within a pattern, limits $P_T > 2$ GeV
  - < 3% of tracks
- “Push” design: all found stubs forwarded
  - ~10K stubs/BX
  - O(50) Tbps

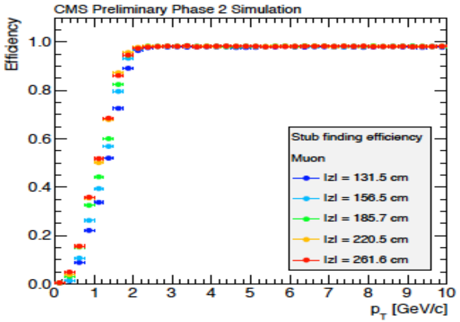


Barrel

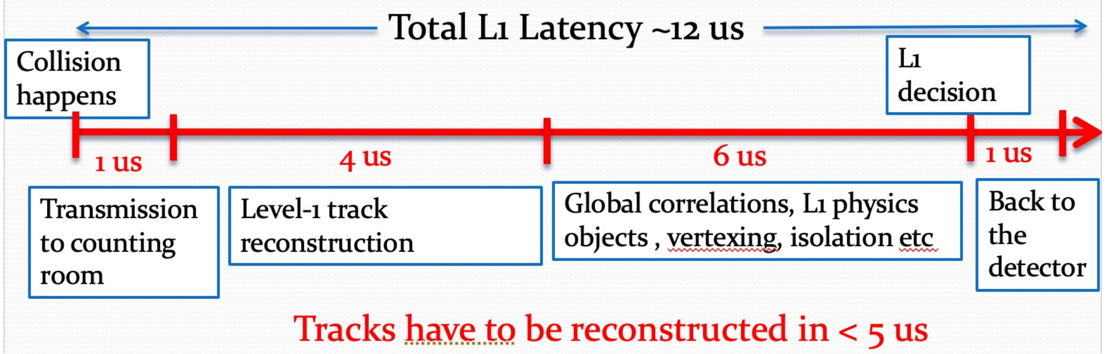
Disks

Stub efficiency vs. P_T for various layers and disks →

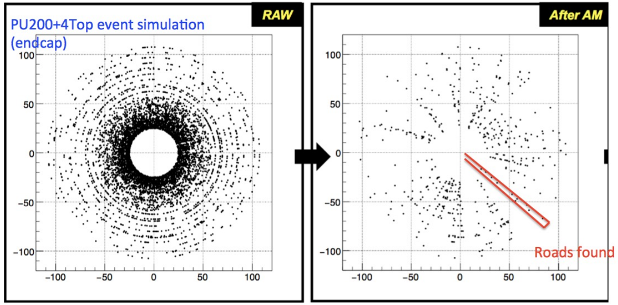




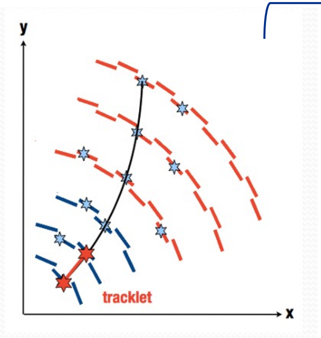
CMS Example: 40 MHz Tracking at L1



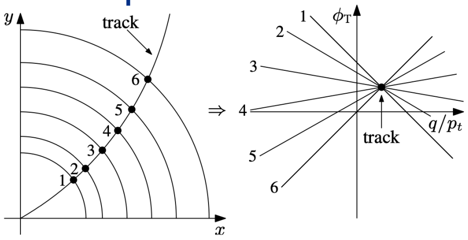
CMS options



Pattern ASIC based Approach



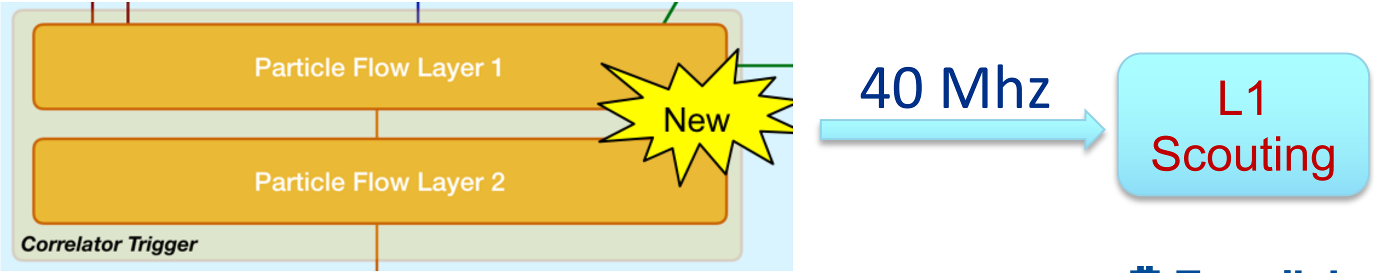
Tracklet Based



Hough Transform Based

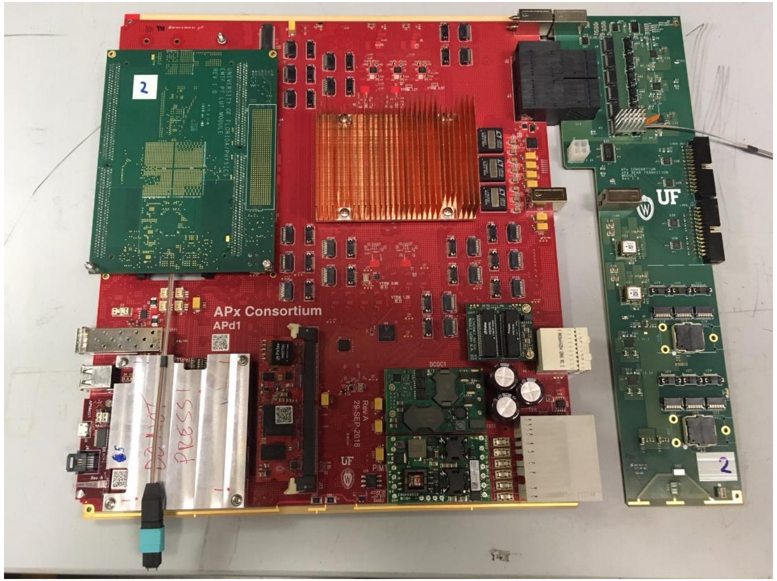
Particle Flow at Level-1

- As discussed, Particle Flow is the optimum combination of tracking momentum and calorimeter energy measurements
  - Used successfully in CMS HLT and offline analyses
- Full exploitation of tracking at Level-1 for HL LHC proposed
  - Requires matching all calorimeter clusters to all tracks, removing clusters not attached to the primary vertex, and replacing cluster energy with track parameters for rest
  - Significantly improves the jet and hadronic energy flow measurements at CMS. Also reduces pile-up dependence.



Technologies for HL-LHC

- Optical data links @ bandwidths up to 28 Gbit/s, and possibly 56 Gbit/s (from 10 Gbit/s currently)
  - Much more data to ship around
- Ultrascale+ FPGAs
  - O(100) data link receivers and transceivers
  - Factor 4 or more logic resources per chip than currently, to tackle more complex algorithms
- Large Memory banks (DDR4) – 128 GB
- Advanced Telecommunication Computing Architecture (ATCA)
  - Standardized shelf technology
    - Current CMS trigger uses µTCA, a daughter card of ATCA



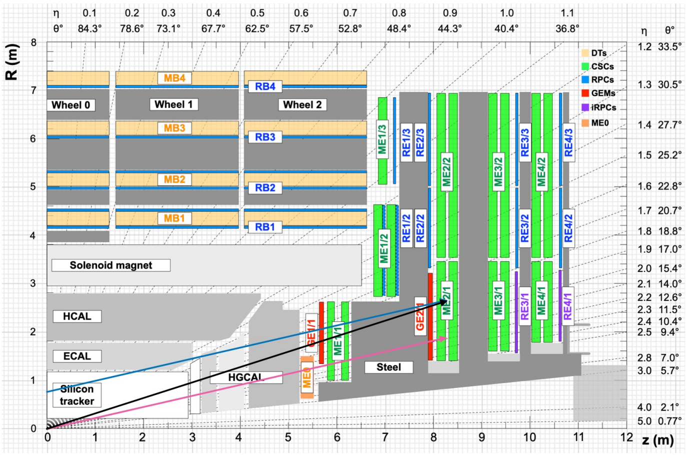
APx: Xilinx VU9P FPGA, and 100 high-speed optical links at 28 Gbit/s, 128 GB RAM

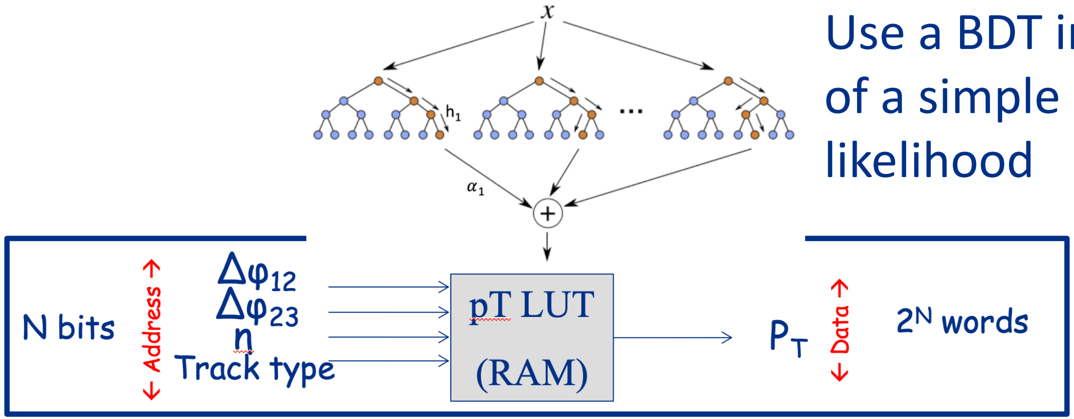
Artificial Intelligence in Trigger Systems

- Machine learning algorithms (aka Artificial Intelligence) have been used in HEP analyses for decades
  - First in a limited capacity, but now extensively! Most Higgs boson measurements make extensive use of machine learning.
- Starting to be incorporated into the Level-1 Trigger
  - For example, a Boosted Decision Tree algorithm was used to train the muon momentum regression in the CMS endcap muon Level-1 trigger.
    - Results were precalculated offline and stored in a memory look-up table. (Logic not implemented directly)
  - But with current technology, neural networks (and BDTs) are possible to implement directly into the FPGA logic fabric, opening many possibilities!

Boosted Decision Trees

- A BDT is a machine learning algorithm used for classification and regression tasks
- A decision tree repeatedly splits a dataset into smaller subregions based on features in that dataset
  - Similar to what particle physicists were doing already by hand ("cuts" on the data set)

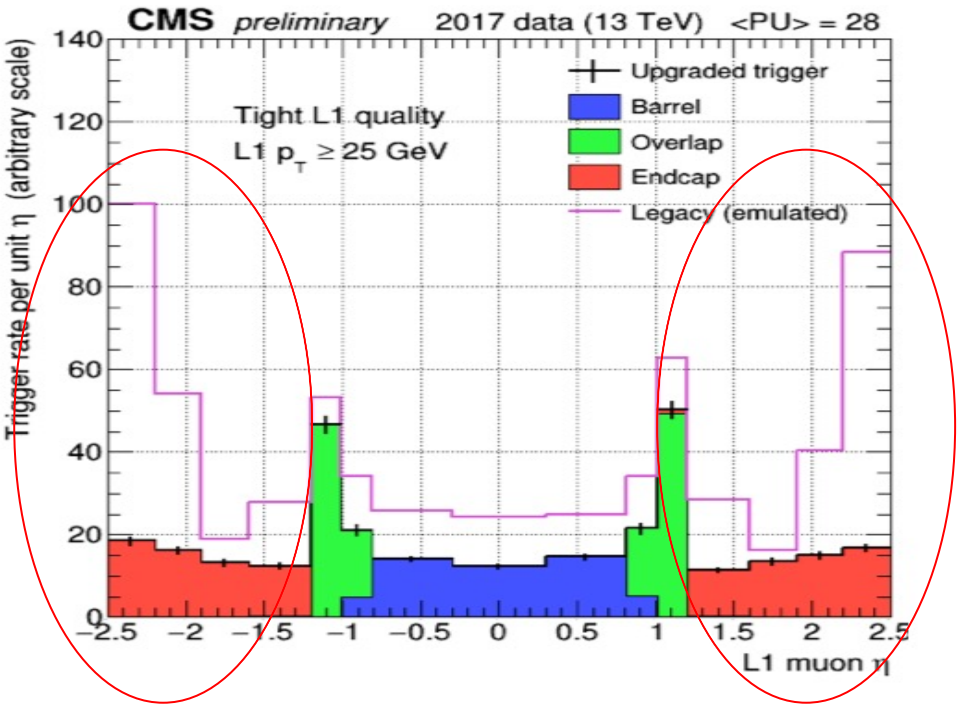




Use a BDT instead of a simple likelihood

Likelihood vs BDT for L1 muons

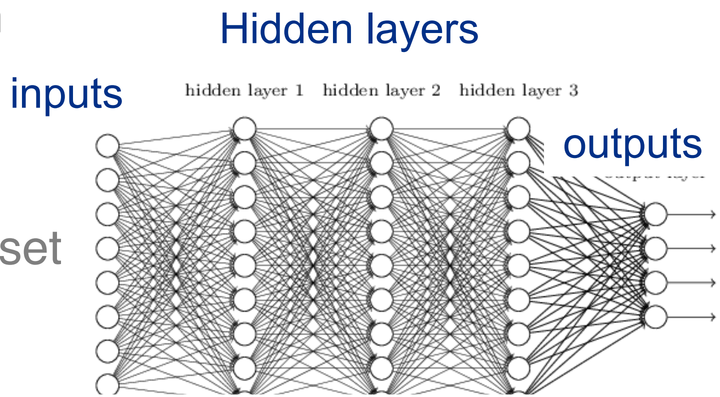
Endcap regions significantly improved with AI algorithm over previous algorithm

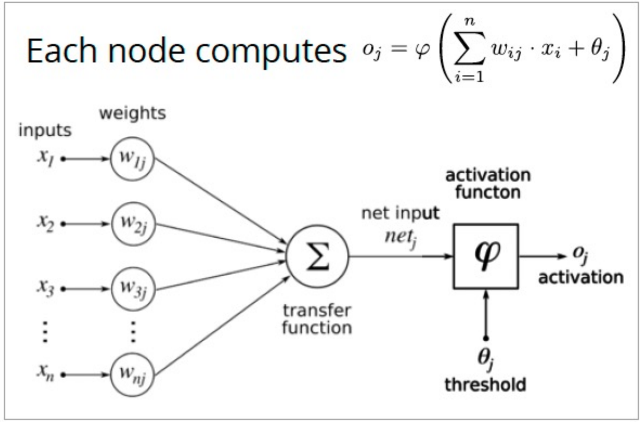


Fermilab

Neural Networks

- Loosely inspired by how neurons work in the brain
  - Neurons fire signals to other connected neurons, amplifying the signal to some degree in the process
- In a neural network, the inputs are multiplied by a set of weights, and the product is sent to a nonlinear activation function
  - e.g. tanh, sigmoid, etc
- A Deep Neural Network has many hidden layers
  - e.g. Convolutional neural nets for image recognition





Neural Networks

- The Neural Network must be trained with large sample of examples of desired classification (just as with the BDT algorithm)
  - Cat vs. not a cat;   Higgs boson vs. not a Higgs boson; momentum =10 vs. momentum =100
  - Weights are determined from back propagation and using a specific loss function (penalty)
- The application of trained network is known as inference

NN Toolkits

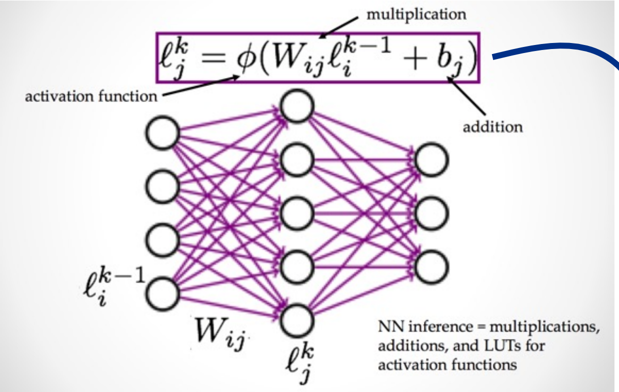
- The TMVA library of Root
  - root.cern.ch : https://root.cern.ch/download/doc/tmva/TMVAUsersGuide.pdf
- Keras: a python deep learning library
  - Runs on top of TensorFlow, CNTK, or Theano (all open source)
- PyTorch

Jupyter notebook snippet:

```
In [7]: # Training with Batch Normalization
        # 'model' is a densely connected NN with 3 hidden layers and 2 output nodes, q/pT and PU discriminator

        if training_bn:
          assert(keras.backend.backend() == 'tensorflow')

          if add_noise:
            x_train_new, y_train_new = mix_training_inputs(x_train, y_train, pu_x_train, pu_y_train, pu_aux_train, discr_pt_cut=di
          else:
            raise Exception('add_noise must be set to True')

          model = create_model_bn(nvariables=nvariables, lr=learning_rate, clipnorm=gradient_clip_norm, l1_reg=l1_reg, l2_reg=l2_r
                                  nodes1=50, nodes2=30, nodes3=20)

          logger.info('Training model with l1_reg: {0} l2_reg: {0}'.format(l1_reg, l2_reg))

          normal_epochs = 300
          normal_batch_size = 256*4*2

          history = train_model(model, x_train_new, y_train_new,
                                model_name='model', epochs=normal_epochs, batch_size=normal_batch_size,
                                callbacks=[lr_decay,modelbestcheck,modelbestcheck_weights], validation_split=0.1, verbose=1)

          metrics = [len(history.history['loss']), history.history['loss'][-1], history.history['regr_loss'][-1], history.history[
                     history.history['val_loss'][-1], history.history['val_regr_loss'][-1], history.history['val_discr_loss'][-1]]
          logger.info('Epoch {0}/{0} - loss: {1} - regr_loss: {2} - discr_loss: {3} - val_loss: {4} - val_regr_loss: {5} - val_dis
```

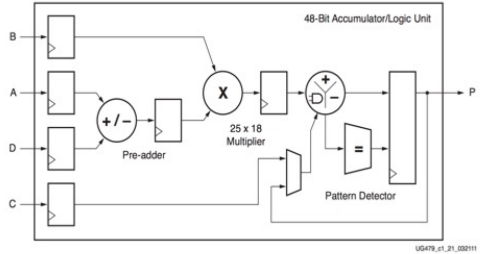
NN in FPGA

- The hls4ml toolkit
  - Implementation of fast neural network inferences into FPGAs:
    - arXiv:1804.06913v2
  - Converts results of a trained NN into Xilinx Vivado HLS firmware
  - The Xilinx DSP slice offers a 27 x 18 bit multiplier
  - HLS4ML optimizes use of FPGA resources
    - #bits, sequential re-use of slice, etc.



Device	# of DSPs
Kintex-7 325T	840
Virtex-7 690T	3600
Kintex UltraScale KU115	5500
Virtex UltraScale+ VU9P	6800

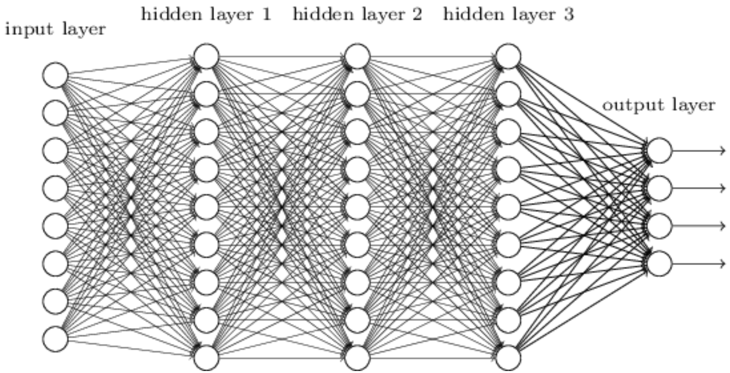
Basic DSP slice



Fermilab

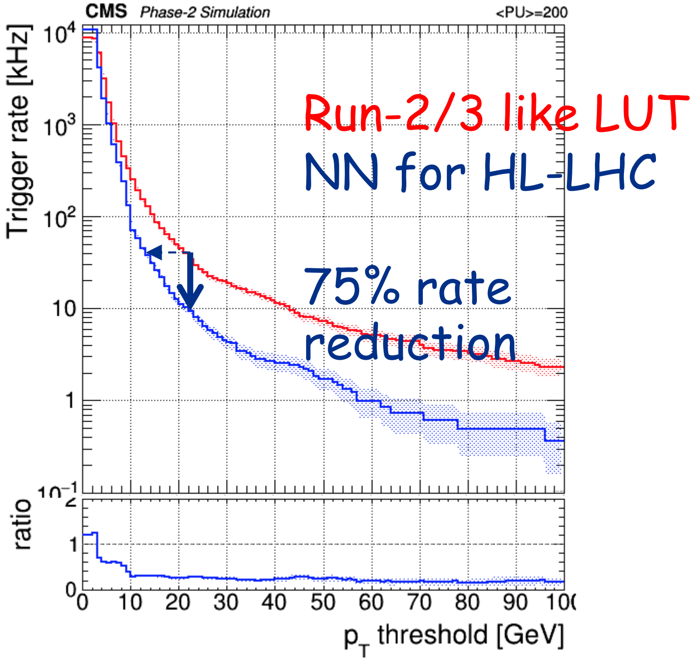
NN for Momentum Regression

- The Level-1 trigger needs fast inferences
- A neural network implemented within an FPGA alleviates the bottleneck of the memory look-up table approach
  - No loss of input variable precision
  - Can achieve better performance, and still fit into target FPGA
- Prototyped one for CMS muon trigger
  - 3 hidden layer, not too deep
  - Momentum regression output



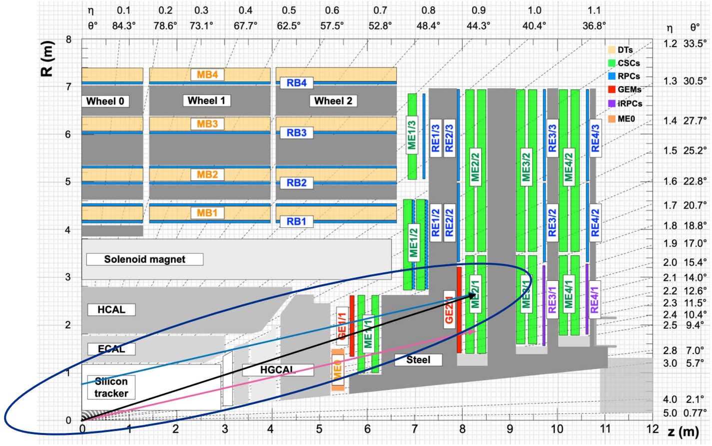
Performance?

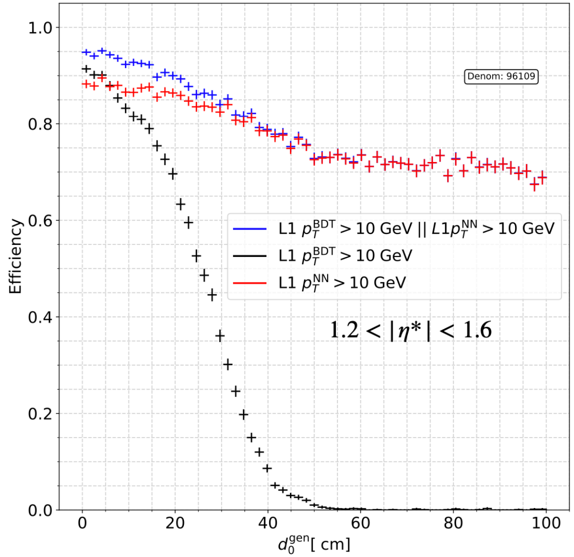
- No input compression in NN (vs BDT LUT)
- NN algorithm achieves lower trigger rate than previous BDT LUT approach without losing efficiency for high momentum tracks
  - 4 times rate reduction
    - Allows us to lower momentum threshold
    - Or take higher collision rate
  - But algorithm also made use of more data from future detectors, so not quite a fair comparison yet…
  - Should retrain BDT for LUT also to compare

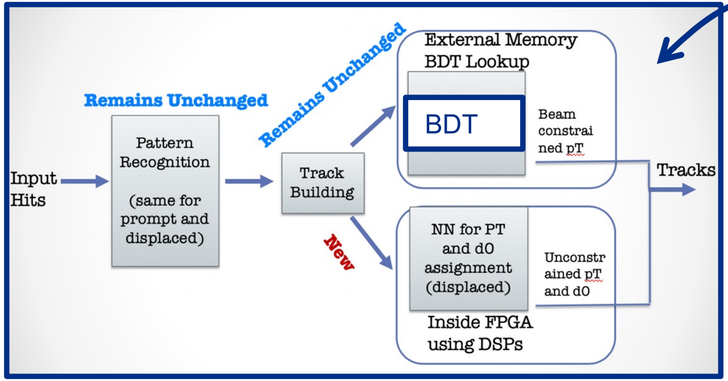


Fermilab

Can we use a NN in Run-3? Yes!







A great example how different board resources can be complementary and maximize physics

Fermilab

hls4ml FPGA Implementation

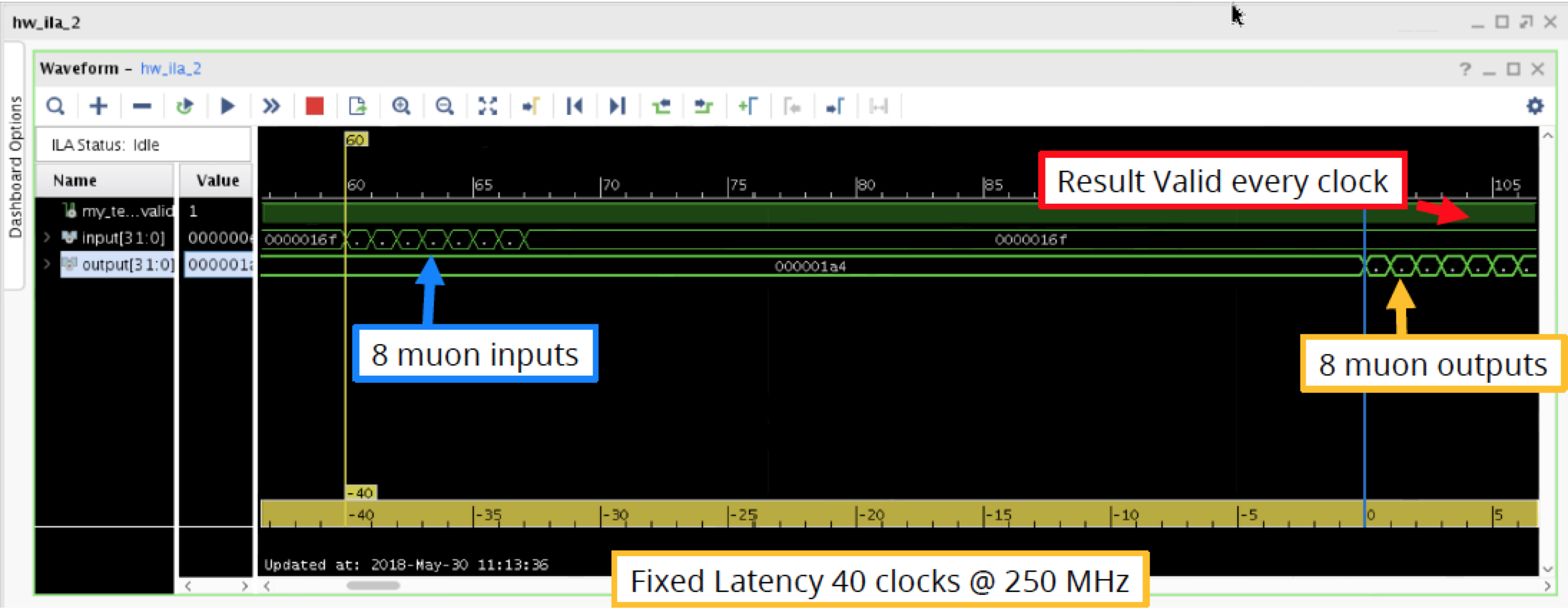
- Take results from trained NN and use HLS4ML to generate firmware
  - Use 18 bit precision for inputs & output
  - Xilinx Virtex-7 690T target FPGA
  - Clock of 125 MHz
- Fits within FPGA resources, and latency ~160ns
- Tested in hardware, results agree with expectation

Utilization Estimates

Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	-	0	6
FIFO	-	-	-	-
Instance	56	2822	315515	112745
Memory	-	-	-	-
Multiplexer	-	-	-	36
Register	-	-	4689	-
Total	56	2822	320204	112787
Available	2940	3600	866400	433200
Utilization (%)	1	78	36	26

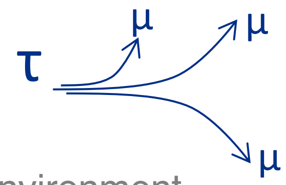
Test Patterns



Fermilab

Other ideas for NN

- Current application applies to reconstructing muon momenta
- But there are also pattern recognition tasks with unique signatures:
  - Displaced muon-like particles
    - Identify tracks that do not project to IP, and measure momentum without beam constraint
  - $\tau \rightarrow 3\mu$ (New Physics?!)
    - Muons are collimated (in η) and soft in p_T. May not penetrate full muon spectrometer
    - Train to identify this signature within (HL)LHC environment
    - Access full luminosity with near zero p_T thresholds?



  - Muon (Lepton) jets, possibly displaced
    - Generalized collimated muons signature

Other ideas for NN

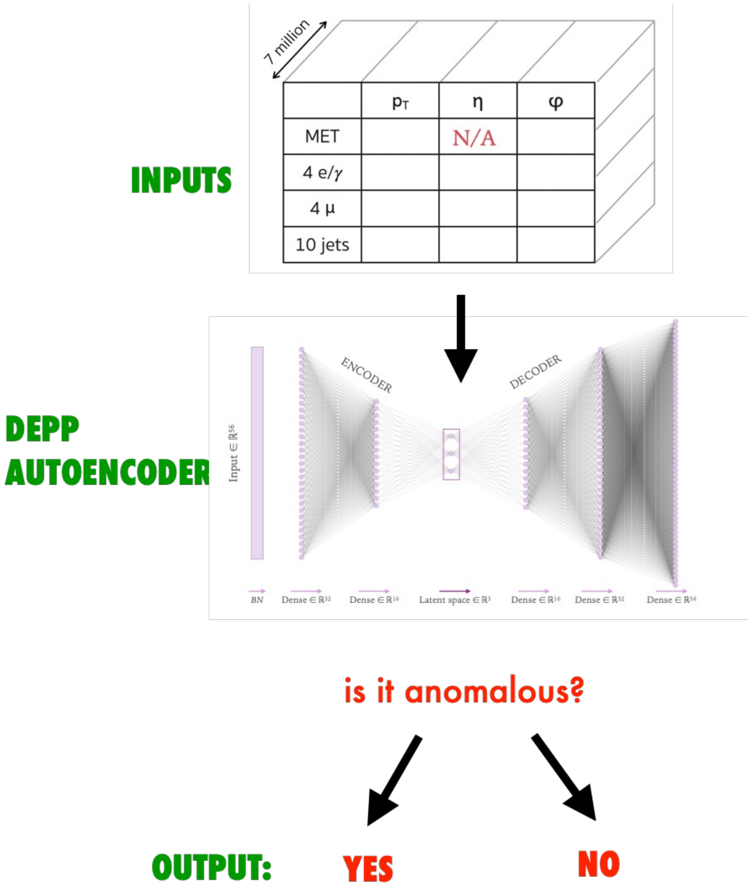
Main idea: try to detect signatures that we have not theorized yet:

- be free from model-dependent thresholds or some combinations of objects and thresholds

How: with unsupervised learning
the only ML@L1 algo proposed so far that will be trained directly on data!

Idea of deploying deep autoencoders in the Global Trigger

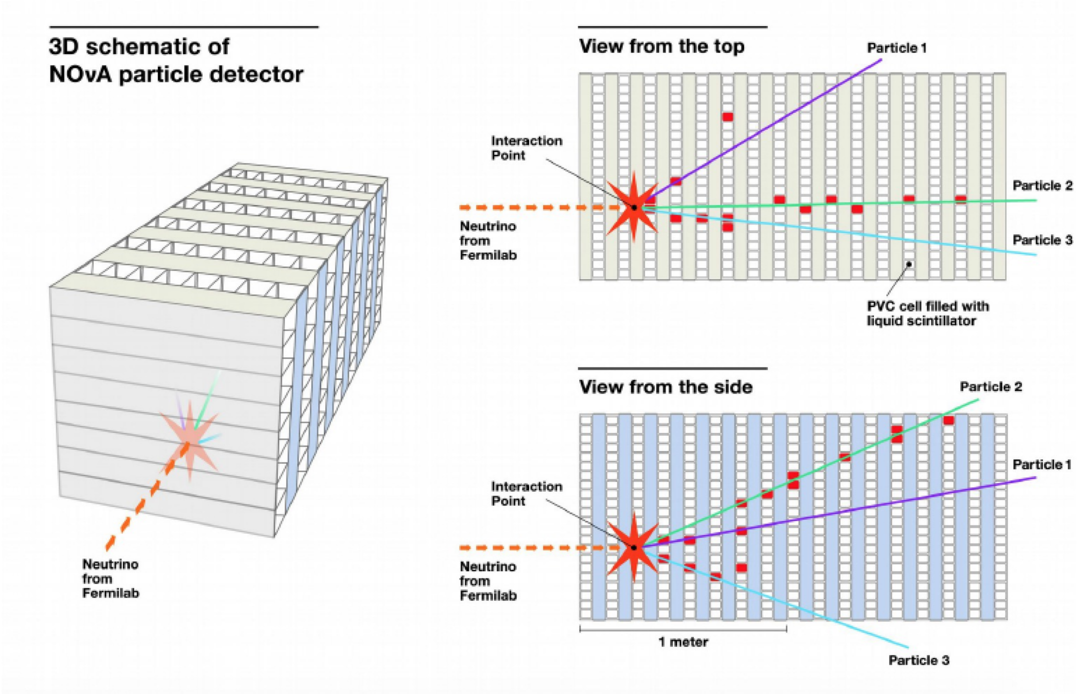
- inputs: same as global trigger (jets, electrons, muons, MET)
- output: trigger decision based on some anomaly metric(latent space most promising in terms of latency)

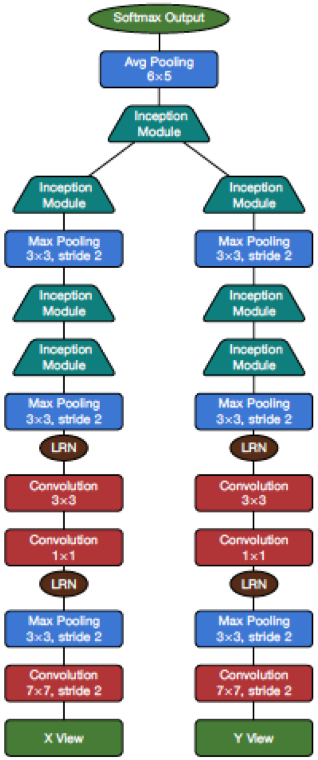


Applications Beyond Colliders

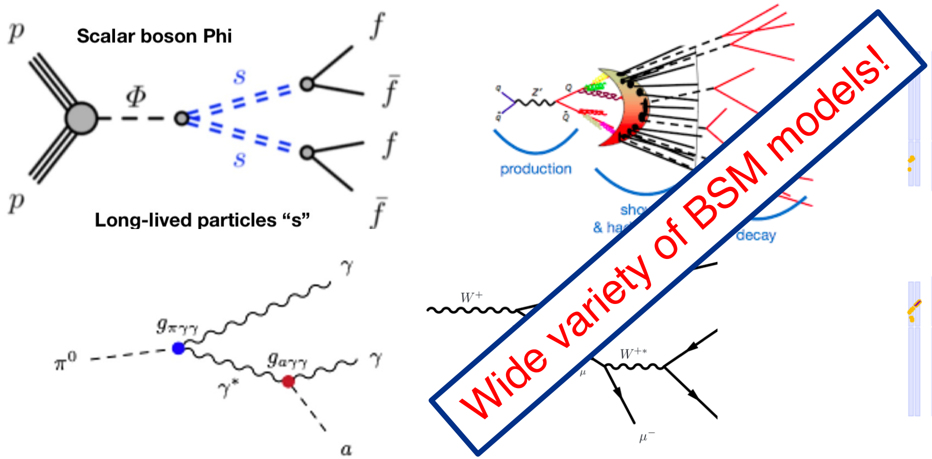
arXiv:1604.01444v3

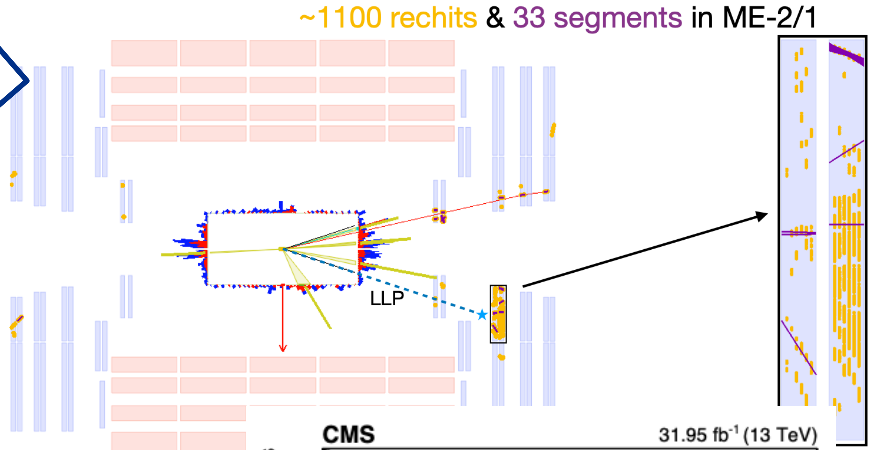
– Convolutional Neural Net to determine what type of neutrino interaction

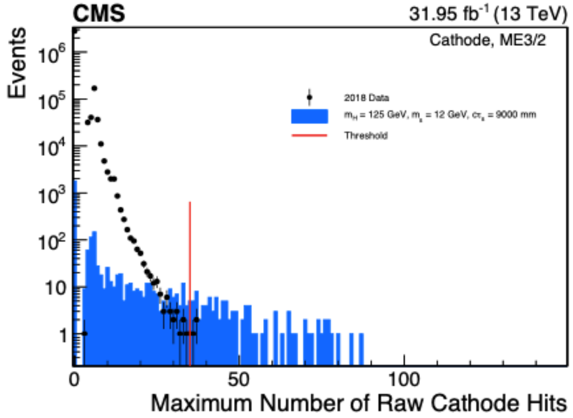




Innovation does not have to come with AI







- Hadronic shower in muon system is an exciting unexplored detector signature
- Steel between muon stations can act as absorbers in a sampling calorimeter
- Unique feature of CMS muon system
- L1 and HLT triggers developed, being commissioned

Fermilab

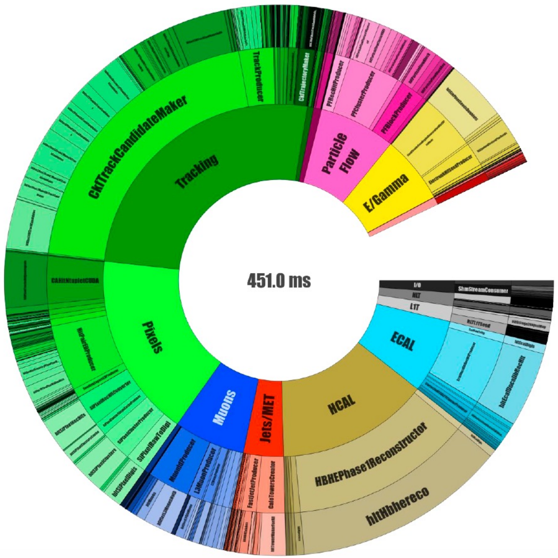
HLT in HL-LHC

- If we have full tracking at Level-1, what is left for HLT to achieve another factor 100 reduction in rate reduction?
- Tracking
  - Has access to pixel hits in addition to strips → b jet tagging
  - HLT so far has not had the resources to perform global tracking for the entire Level-1 bandwidth
  - But with the Track-Trigger, the full collection of tracks can be accessed by HLT for every event and used even if Level-1 did not need to for all triggers to achieve rate reduction
    - This is what ATLAS experiment planned with its Fast Tracker (a custom "accelerator" for it computing processors)

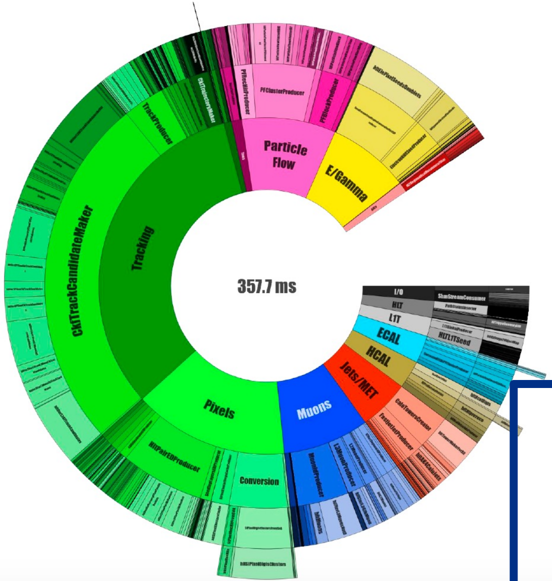
GPU Acceleration by CMS

21% processing time reduction

Timing of Run-3 HLT



Timing of Run-3 HLT menu with GPU acceleration



The pie-chart shows the distribution of CPU time in different instances of CMSSW modules (outermost ring), their corresponding C++ class (one level inner), grouped by physics object or detector (innermost ring). The empty slice indicates the time spent outside of the individual algorithms.

The time spent in the conversion of GPU-friendly *Structure of Arrays* data formats to legacy data formats is indicated by "Conversion" in the extra internal ring.

The timing is measured on pileup 50 events from Run2018D on a full HLT node (2x Intel Skylake Gold 6130) with HT enabled, running 16 jobs in parallel, with 4 threads each - equipped with an NVIDIA T4 GPU.

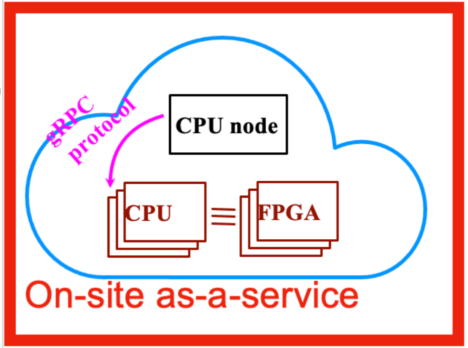
Using the GPU to accelerate:

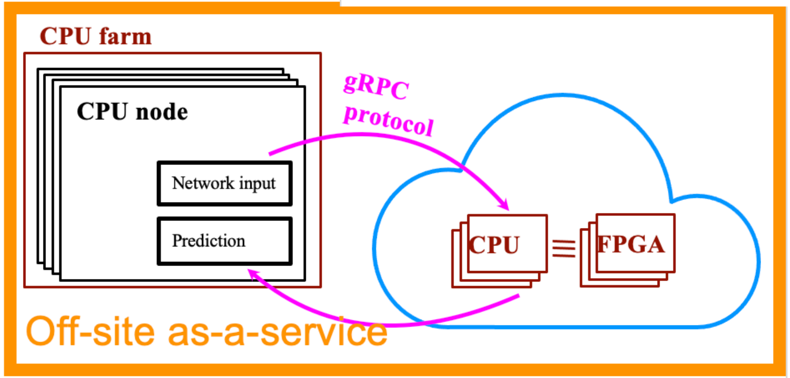
- pixel local reconstruction, track and vertex reconstruction
- HCAL local reco (MAHI)
- ECAL unpacking and local reconstruction (multifit)

reduces the CPU usage by 21%, increasing the throughput by 26%.

Heterogeneous Computing Platforms

- Acceleration from GPUs helps CPUs in some compute-intensive tasks
  - e.g. machine-learning training and inference (Artificial Intelligence), but not only. Other algorithms also have been ported to GPUs
- FPGAs are also available as accelerators
  - Particular interest also for machine-learning
  - For example:
    - Microsoft Azure cloud computing
    - Amazon Web Services (AWS) cloud – F1 instance
- Possibilities to improve HLT processing time
- One node for multiple trigger levels?

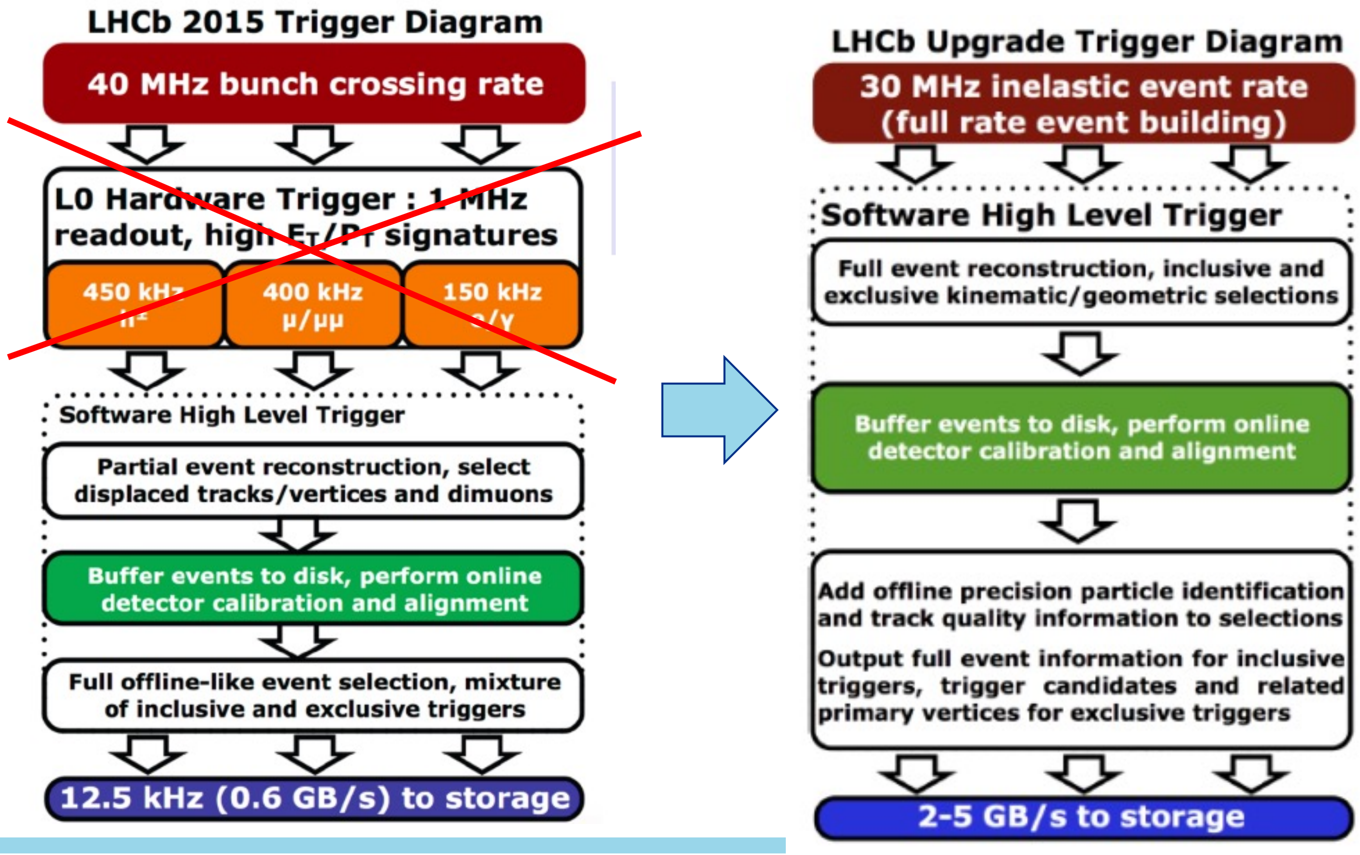




Fermilab

Trigger-less → Streaming DAQ

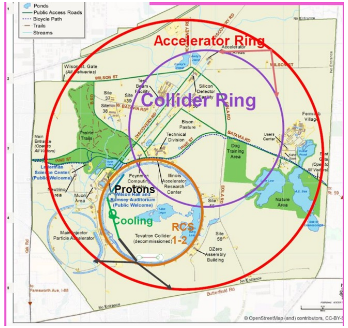
- Replace all front-end electronics to be able to send all data to software-based High Level Trigger
  - Current LHCb upgrade
  - Would be much more challenging for a large experiment like ATLAS or CMS



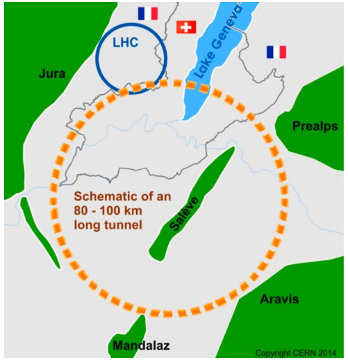
Example future "Energy" machines

- FCC-hh:
  - Higher energy ~100 TeV
  - Higher luminosity: 5-30 x 10^{34} Hz/cm^2
- Muon Collider:
  - 10 TeV leptonic collisions
  - No QCD backgrounds --> energy and precision in one collider

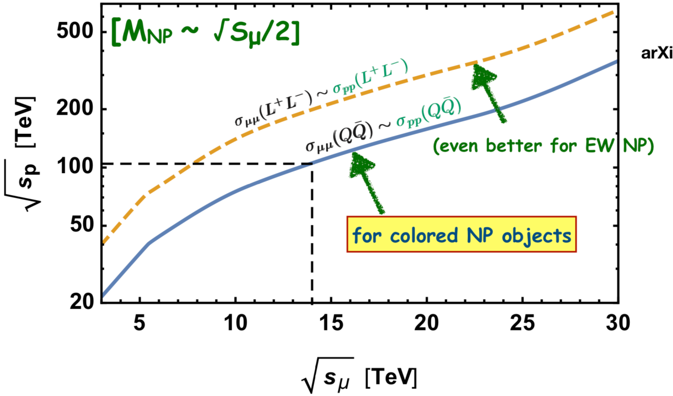
MuC sketch



FCC sketch

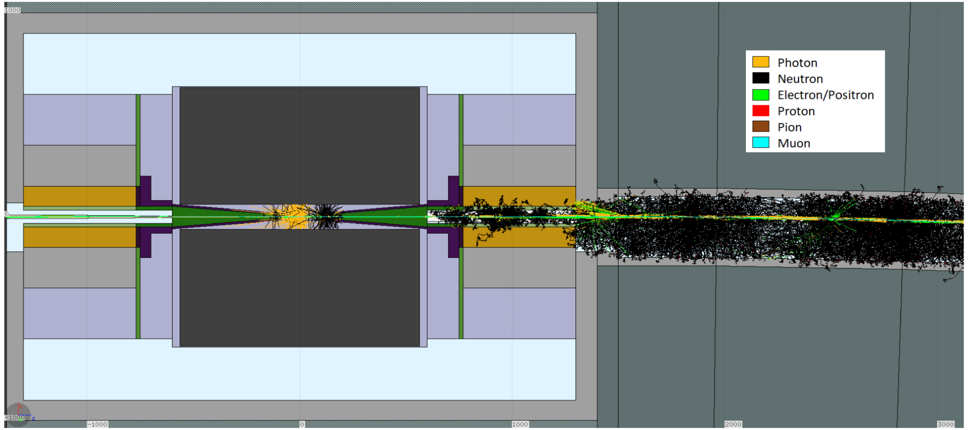


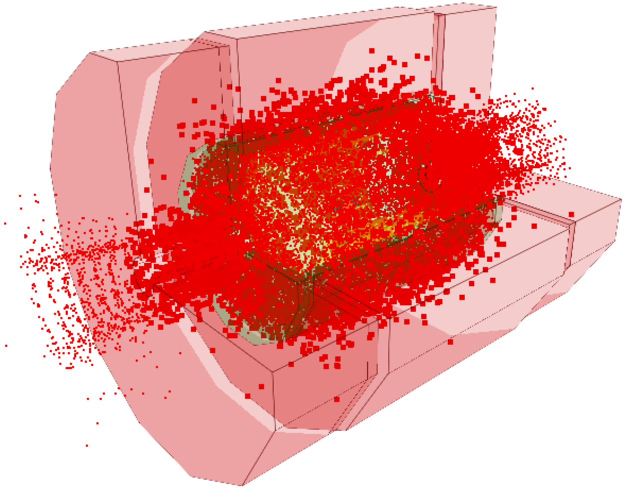
- Trigger Challenges:
  - Pileup or Beam background
  - Higher detector channel count from increased granularity
  - More energy → more particles
  - Radiation levels for front-end electronics



arXiv:1901.06150

Muon Collider Detectors





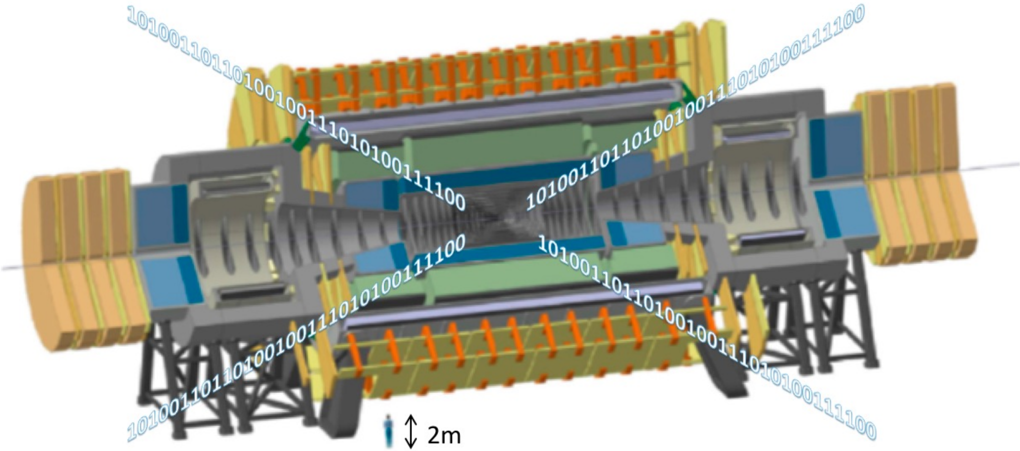
- Estimated Event Size is 50 MB to be compared to ~ 7 MB for HL-LHC
- But events are happening every 10 microseconds vs 25 ns at the LHC
- Background typically non-pointing and out-of-time – use in trigger algorithms

- Can we use a Streaming system like LHCb?

Fermilab

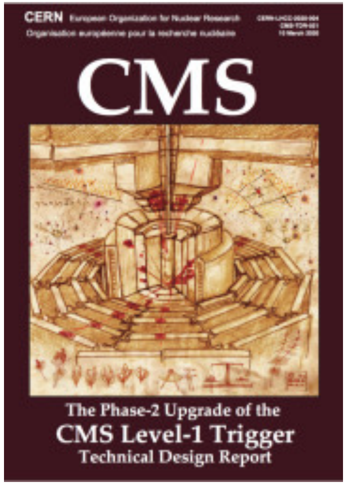
FCC-hh is a Data Collider

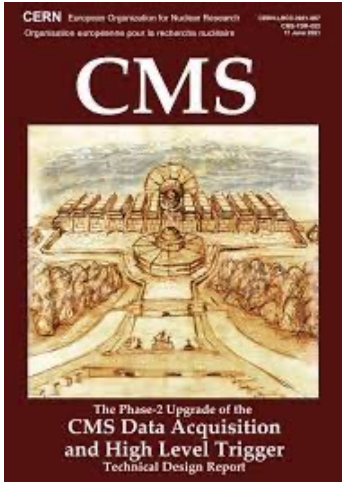
- Exabytes per second before zero suppression and data reduction!

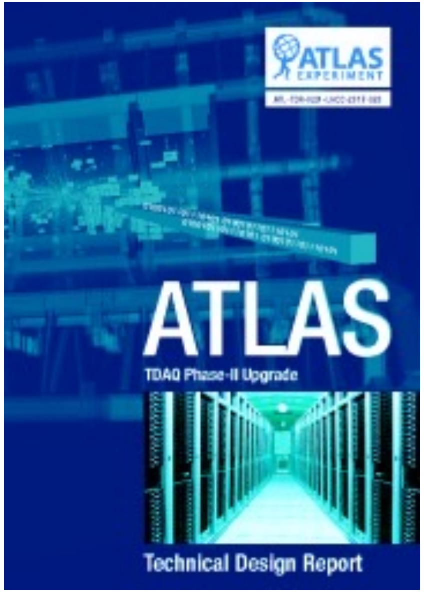


- Requires new ideas and new technologies to tackle next order of magnitude!
  - **And you!**

Further Reading







Fermilab

Further Reading

- CMS Collaboration, “Performance of the CMS Level-1 trigger in proton-proton collisions at $\sqrt{s}$ = 13 TeV”, arXiv:2006.10165
- D. Acosta, C. Foudas and D. Newbold, “CMS gears up for the LHC data deluge”, CERN Courier, Volume 56, Number 7, September 2016
- CMS Collaboration, “The CMS trigger system”, Journal of Instrumentation 12 (2017) no.01, P01020
- CMS Collaboration, “The Phase-2 Upgrade of the CMS L1 Trigger Interim Technical Design Report”, CERN-LHCC-2017-013; CMS-TDR-017
- CMS Collaboration, “The Phase-2 Upgrade of the CMS DAQ Interim Technical Design Report”, CERN-LHCC-2017-014; CMS-TDR-018
- N.P. Ghanathe et al., “Software and firmware co-development using high-level synthesis”, JINST 12 (2017) C01083.
- CMS Collaboration, “CMS Technical Design Report for the Level-1 Trigger Upgrade”, CERN-LHCC-2013-011; CMS-TDR-012
- ATLAS Collaboration, “Performance of the ATLAS Trigger System in 2010”, Eur. Phys. J. C 72 (2012) 1849

BACKUP

Machine Learning for Regression

- Our trigger application is somewhere between a classification problem and a regression
  - We want to know the particle momentum above or below a specific threshold, but for multiple thresholds (just not an infinite set)
- We use a transformation + loss function to focus on low momentum events (whose mismeasurement to high momentum drives the rate)
  - Target= $1/p_T$ makes differences in low p_T count more in loss
  - Loss = $|1/p_{T,meas} - 1/p_{T,true}|^2$, but studied other loss functions
    - Focus on low p_T more → lower rate (good), lower effic. (bad)
    - Focus on low p_T less → higher rate (bad), higher effic. (good)
- With redundant measurements (4 detector stations), ML can identify outliers (e.g. high momentum muon showering) and reject them to keep efficiency high
  - We used to have to introduce ad hoc "human" algorithms to recover efficiency

Encoding and Training

ACAT2017

- Next pack (compress!) your input data in an optimal way if using a LUT
  - This is a data science in itself!
  - Here are muon track variables for the CMS trigger, for example:

Many variables, but few bits each

Appendix B - Schematic of 2017 PT LUT address bits

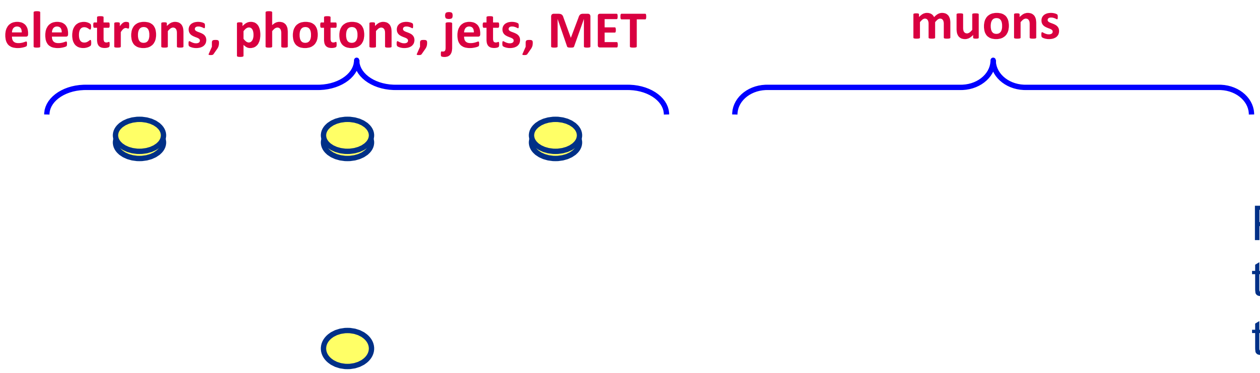
PT LUT address bits	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	4	4	3	2	1	0
Two-station tracks	0	0	0	0	mode2			5b_theta					3b_clctB			3b_clctA			frB/A		3b_dThAB			7b_dPhAB						
Station 2-3-4 tracks	0	0	0	1	5b_theta					2b_rpc		clct2		fr	3b_dTh24			s	5b_dPh34					7b_dPh23						
Three-station tracks	0	mod3		5b_theta					2b_rpc		clctA		frB/A		3b_dThAC			s	5b_dPhBC					7b_dPhAB						
Four-station tracks	1	8b_theta_rpc_clct1								fr	dTh14		s34-23		4b_dPh34				5b_dPh23					7b_dPh12						

*** Some names truncated for space. **Two-station:** [frB/A] = [frB][frA]. **Station 2-3-4:** [fr] = [fr2], [s] = [sph34].
Three-station: [mod3] = [mode3], [frB/A] = [frB][frA], [s] = [sphBC]. **Four-station:** [fr] = [fr1], [s34-23] = [sph34][sph23], [dTh14] = [2b_dTh14].

- Train the AI algorithm
  - For this example, used a Boosted Decision Tree for a regression on the momentum of the track
    - TMVA package of root.cern.ch
  - Trained on simulated track trajectories in the experiment
  - Care taken on choice of a loss function (penalty) so that performance is optimized for application (efficiency, rate)
- Apply trained algo to all 2^{30} input combinations and store in file

Fermilab

CMS Level-1 Trigger Pipeline Scheme



Processing takes longer than one beam crossing, so must register results along the way

- Data flows in a pipeline with a 40 MHz heartbeat
- Accept/reject decision reached in 4 µs

The Trigger, an Online Filtering System

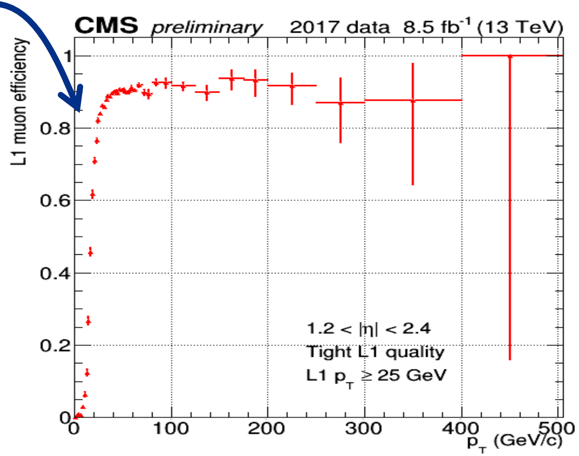
- The detector data payload from a single beam crossing (called an "event") , after zero suppression and compression, is roughly 1 MB
- Beams cross at a rate of up to 40 MHz
  - But Higgs production rate is only 0.1 Hz…
- Potential data bandwidth is therefore O(40) TB/s !
- The "trigger system" is an online filtering system to reduce this to manageable levels for offline storage and processing
  - Throw away 99.998% of all crossings… (keep ~1000 Hz)
  - But, don't throw out the baby (Higgs) with the bath water!
- We still record a couple petabytes (PBs) of data annually…
  - Which was foreseen in the design, even though the experiments were conceived in the early 1990s!
  - Set up a tiered level of global computing for data processing and analysis (grid computing) in the 2000s

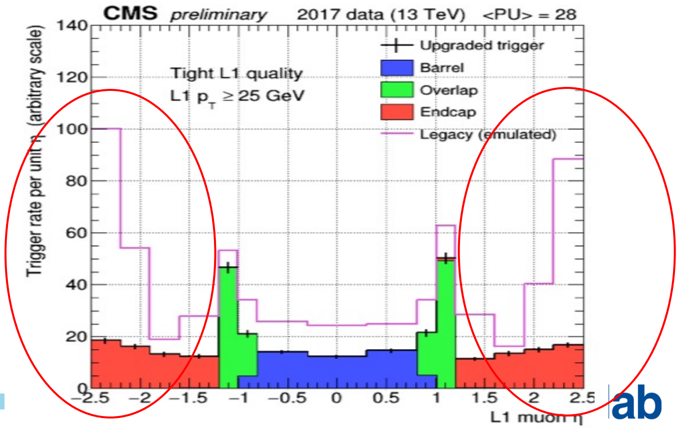
Endcap Muon Trigger Performance

25 GeV threshold

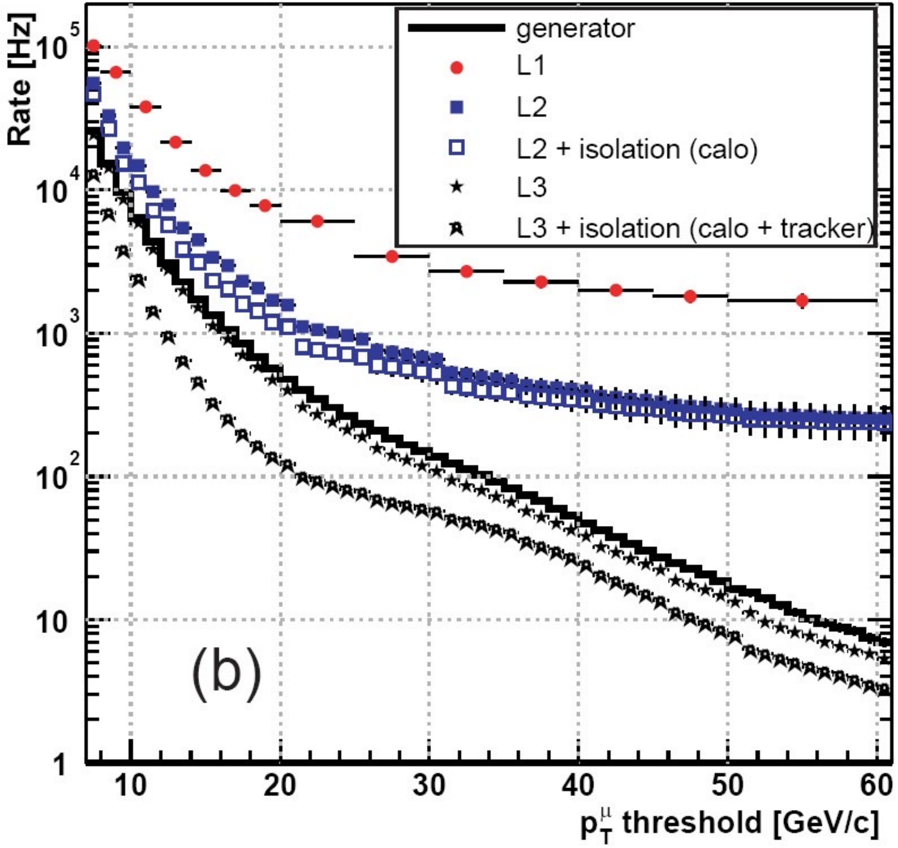
- Efficiency is high even to highest p_T (TeV-scale)

- Rate suppressed 3X in forward region relative to previous trigger, and comparable to barrel rate despite much less magnetic bending and high backgrounds





Addition of Tracking at HLT

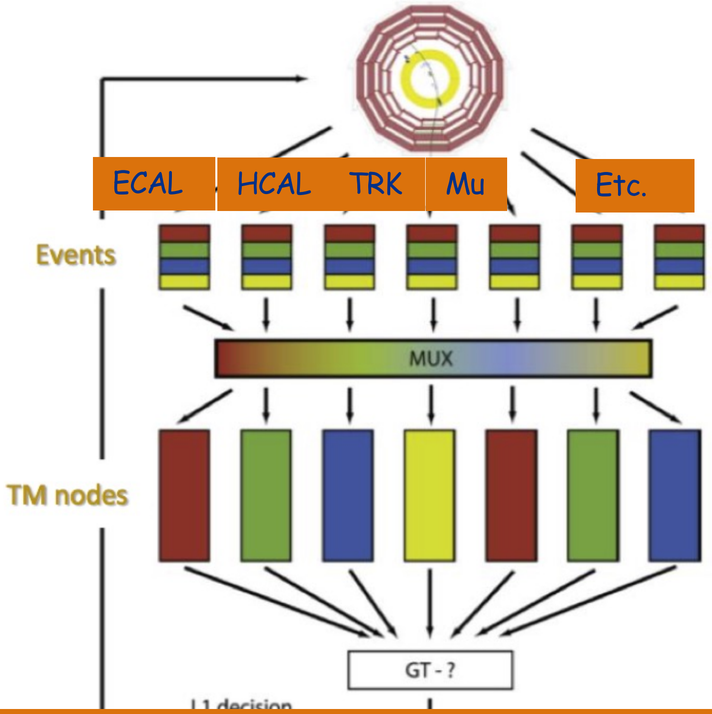


- Better momentum precision leads to steeper rate curve
  - Able to reduce rate better at high momentum

← Level-1 Muon Trigger rate vs. P_T (coarse resolution)

← HLT Muon Trigger rate vs. P_T with tracking (higher resolution)

CMS Level-1 Trigger "Time Multiplexing"



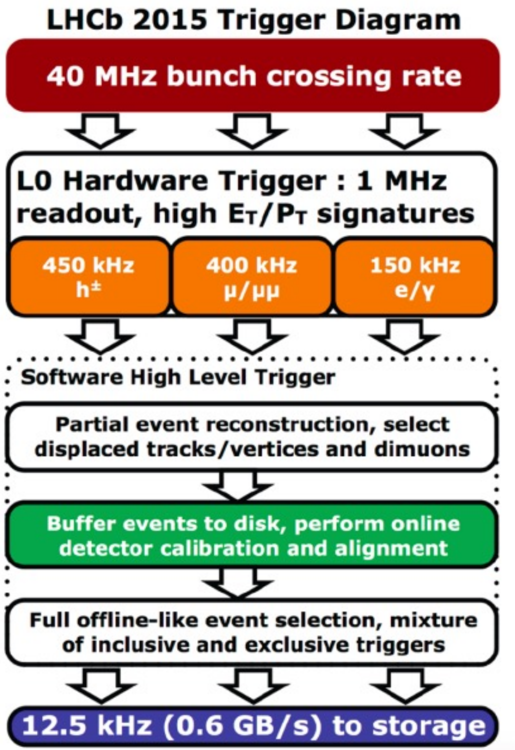
- Fragments of events from different detectors and regions received at front-end
- Data specific to a particular bunch crossing from a specific detector readout are sent to a target destination in a round-robin fashion
- This is an asynchronous process (unlike Level-1). Data get tagged with an event label

Also now done for part of CMS Level-1 Trigger

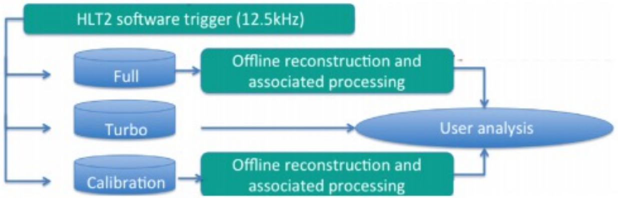
Another Two-Level Trigger System

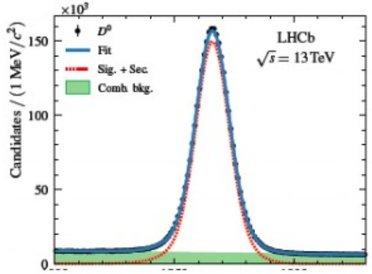
LHCb is a smaller collider experiment instrumented in forward direction only to specialize in B physics

Trigger



- ➢Detector alignment, calibration and Particle ID at the trigger level
- ➢Same reconstruction in online and offline

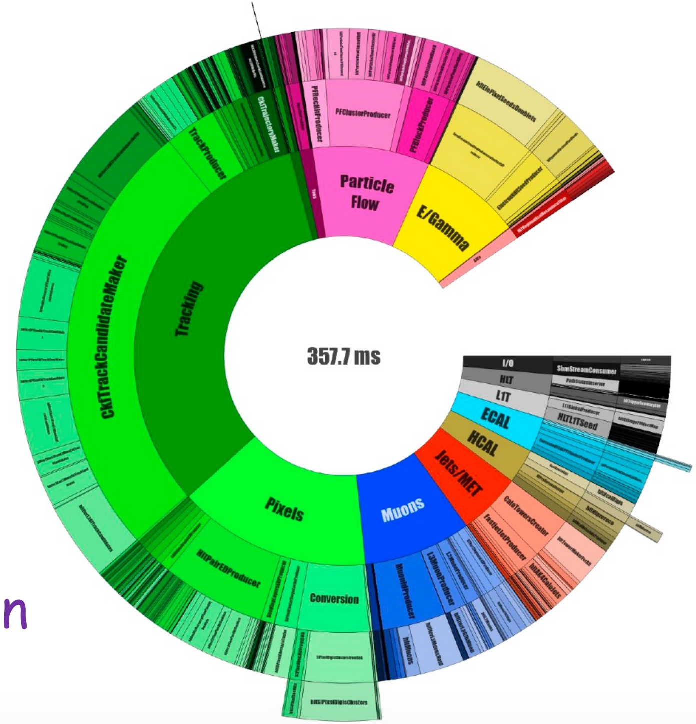




And is now going to only one Level for Run 3!

Example Acceleration by CMS

Timing of Run-3 HLT menu with GPU acceleration



21% reduction

The pie-chart shows the distribution of CPU time in different instances of CMSSW modules (outermost ring), their corresponding C++ class (one level inner), grouped by physics object or detector (innermost ring). The empty slice indicates the time spent outside of the individual algorithms.

The time spent in the conversion of GPU-friendly *Structure of Arrays* data formats to legacy data formats is indicated by “Conversion” in the extra internal ring.

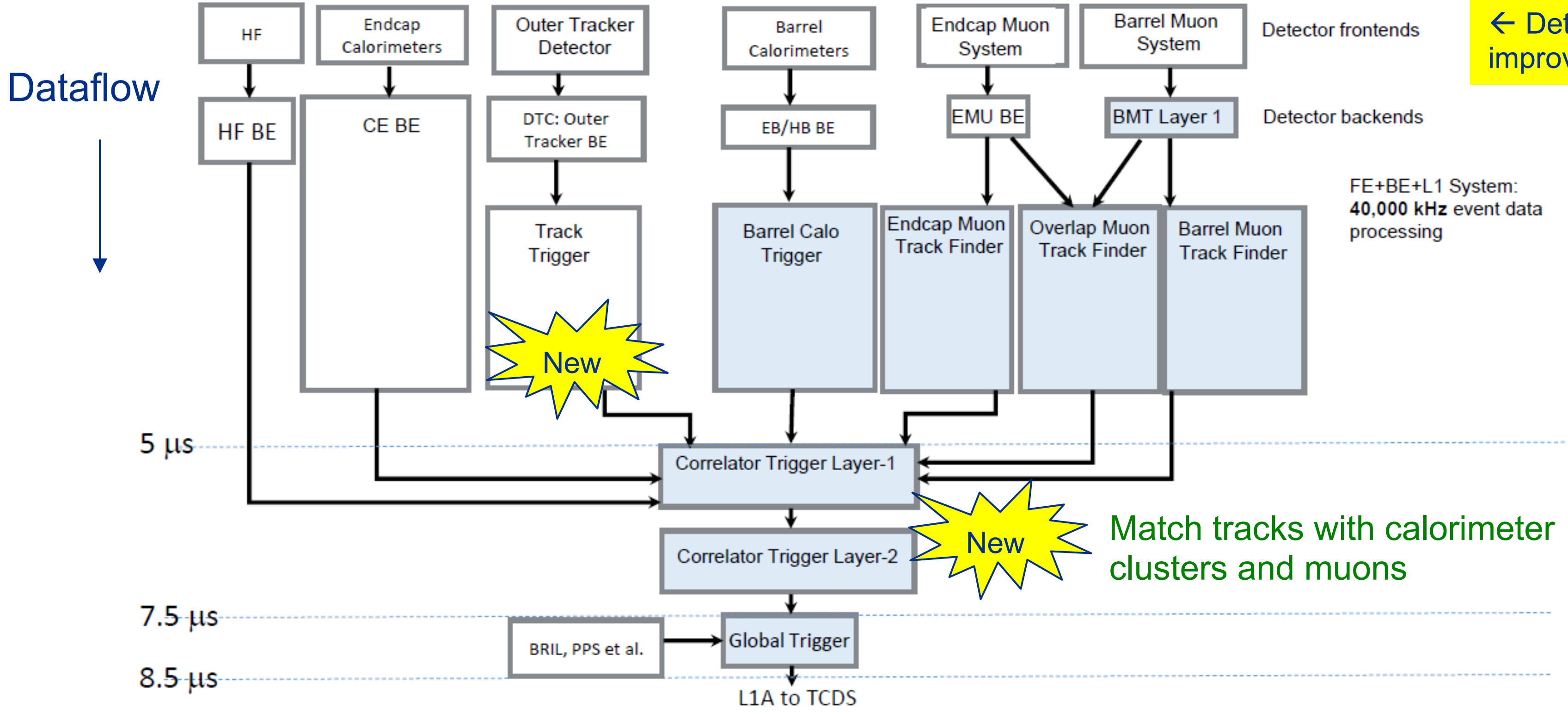
The timing is measured on pileup 50 events from Run2018D on a full HLT node (2x Intel Skylake Gold 6130) with HT enabled, running 16 jobs in parallel, with 4 threads each - equipped with an NVIDIA T4 GPU.

Using the GPU to accelerate:

- pixel local reconstruction, track and vertex reconstruction
- HCAL local reco (MAHI)
- ECAL unpacking and local reconstruction (multifit)

reduces the CPU usage by 21%, increasing the throughput by 26%.

CMS Proposed Level-1 Trigger Architecture



← Detector improvements

Match tracks with calorimeter clusters and muons

Fermilab