Operation of the CMS Level-1 calorimeter trigger in high pileup conditions & motivations for Phase 2

Dr. Aaron Bundock

Imperial College London

TWEPP 2018

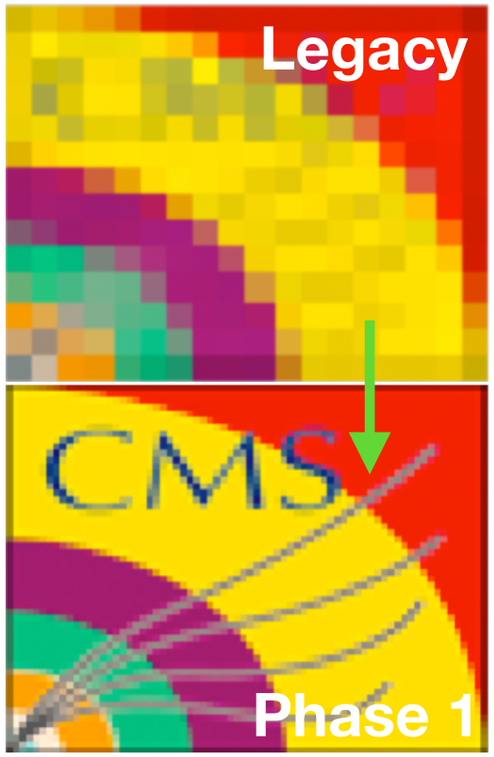
21st September 2018

In this talk:

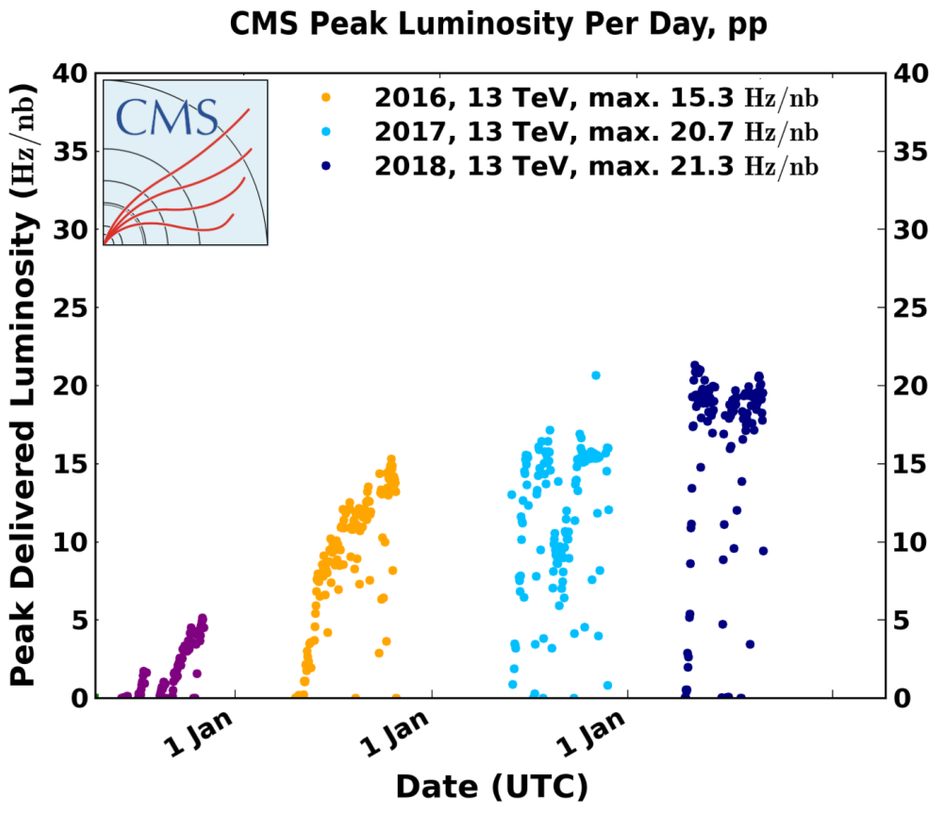
1. Introduction
2. CMS Phase 1 calorimeter trigger upgrade
3. Run 2: operating in high pileup conditions
4. Phase 2: much higher pileup conditions
5. Summary

Introduction

- CMS Phase 1 calorimeter trigger running successfully since 2016
- μTCA time-multiplexed system with Virtex-7 FPGAs, 10Gb/s links
- Increase in calorimeter tower resolution by factor 4 in $\eta \times \varphi$
- More sophisticated algorithms → improved trigger performance
- Last few years challenging for L1: LHC delivering higher luminosity
  - increase in number interactions per bunch crossing ("pileup")



- Some L1 algorithms exhibit non-linear dependence of rate with pileup, especially MET
- Flexibility of calorimeter trigger has enabled us to adapt quickly to changing beam conditions
- Lessons learned during from Phase 1 will inform design choices and algorithms for Phase 2



1. Introduction

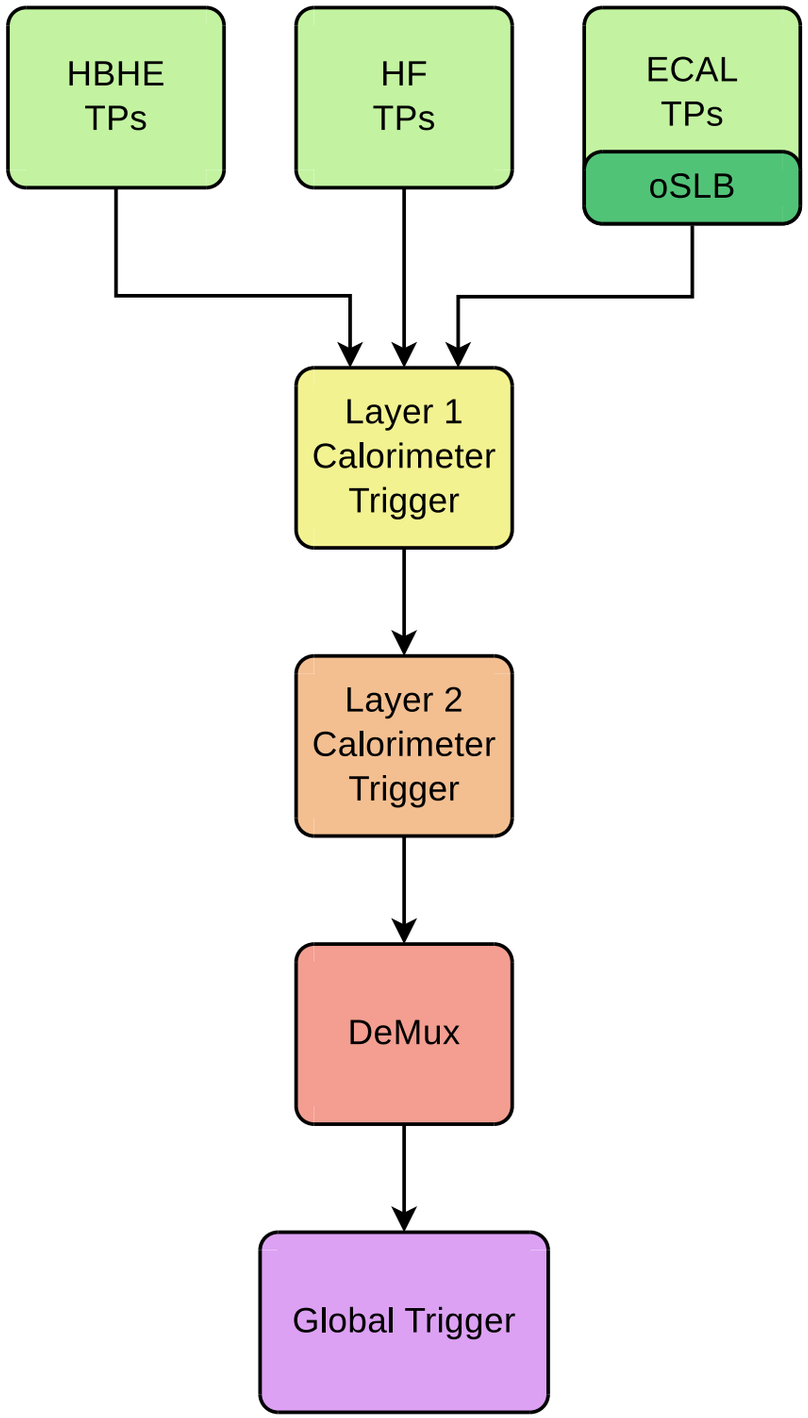
2. CMS Phase 1 calorimeter trigger upgrade

3. Run 2: operating in high pileup conditions

4. Phase 2: much higher pileup conditions

5. Summary

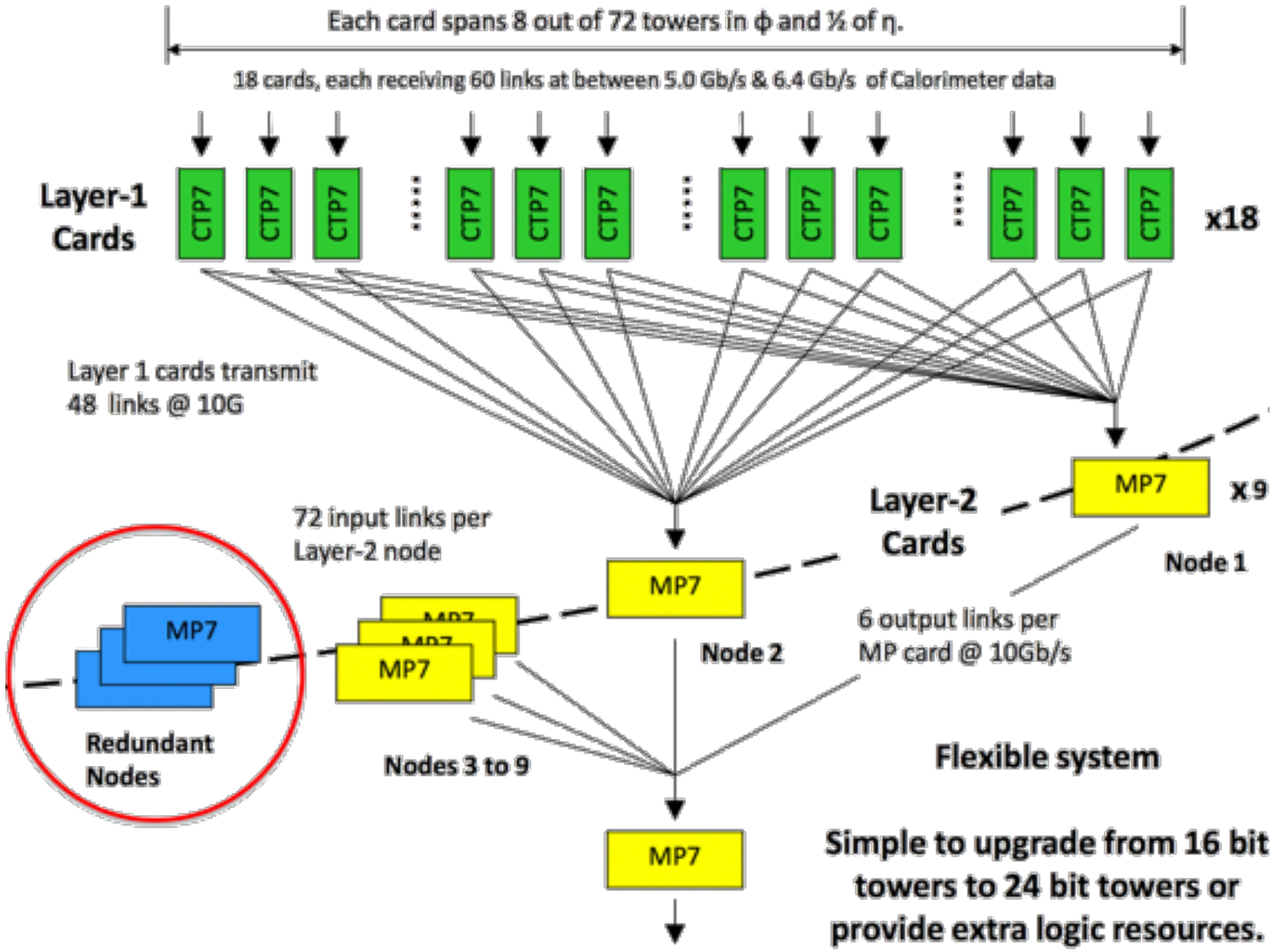
CMS Level-1 calorimeter trigger



- L1 calorimeter trigger receives trigger primitives from ECAL, HCAL, and HF (forward calorimeter)
- Layer 1 builds towers from TPs by adding ECAL+HCAL energies and setting feature bits
- ECAL & HCAL TPs have particle response level calibrations applied at Layer 1 to account for losses
- 40 towers in $\pm\eta$ and 72 in φ, with max. E_T of 256 GeV
- Layer 2 runs jet, egamma & tau algorithms, and constructs ETT, HTT, MET & MHT global energy sums
- Pileup subtraction applied to jet, egamma, tau & MET
- Object level energy calibrations applied at Layer 2, also egamma & tau isolation, shape ID
- Layer 2 is then demultiplexed, with 12 highest energy objects + global energy sums sent to global trigger
- Global trigger issues L1 accept if calorimeter objects (or muons) pass selection defined in the L1 menu
- GT applies trigger rules and receives external conditions

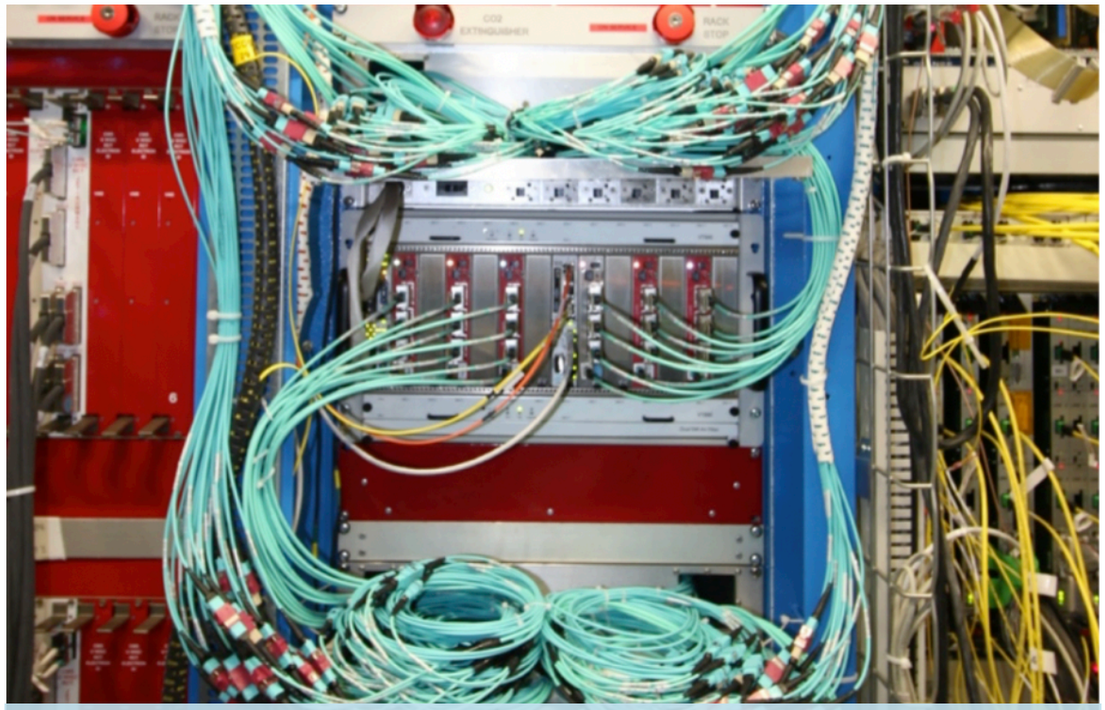
CMS Level-1 calorimeter trigger

- 18 × Layer 1 pre-processor CTP7 cards each receive a calorimeter region in φ for + or - η and send data to Layer 2 via molex flexplane optical patch panel
  - Input: 60 links/board @ ~6 Gb/s
- 9 × Layer 2 main processor MP7 cards receive data from full calorimeter: time multiplexed system
  - Input: 72 links/board @ 10 Gb/s
- No need for overlap regions: no data sent between MPs
- Redundant nodes ready to receive and send data after central trigger reconfiguration if a board fails

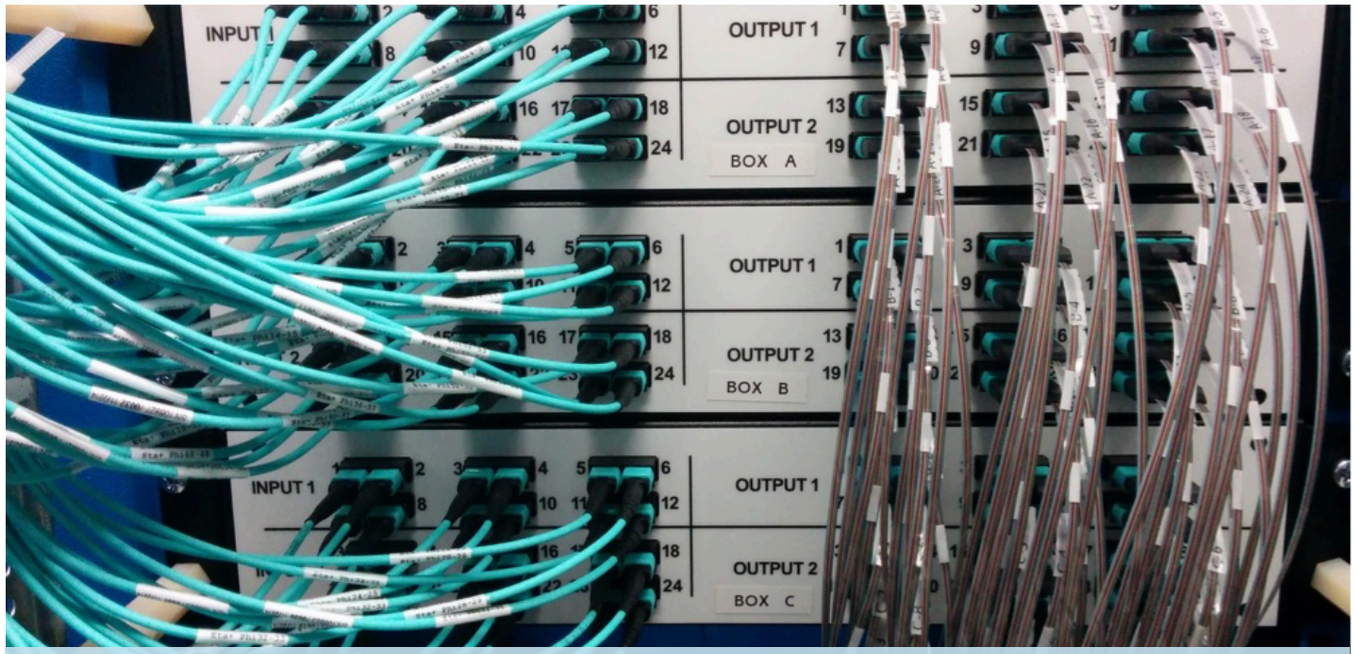


CMS Level-1 calorimeter trigger

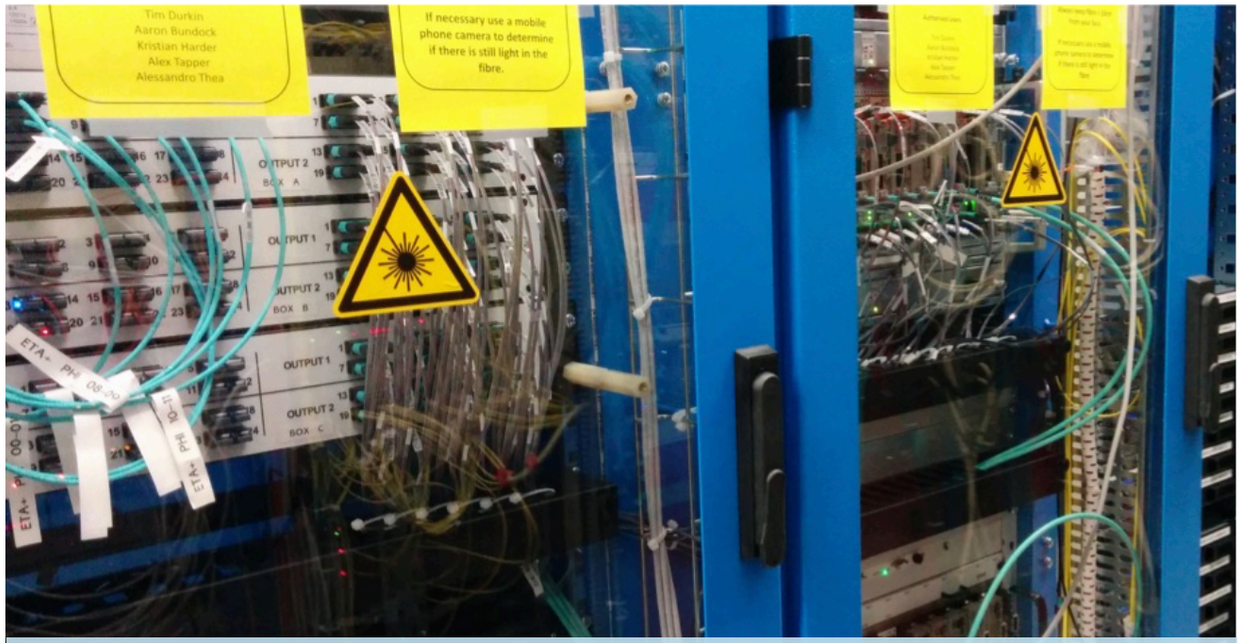
Imperial College London



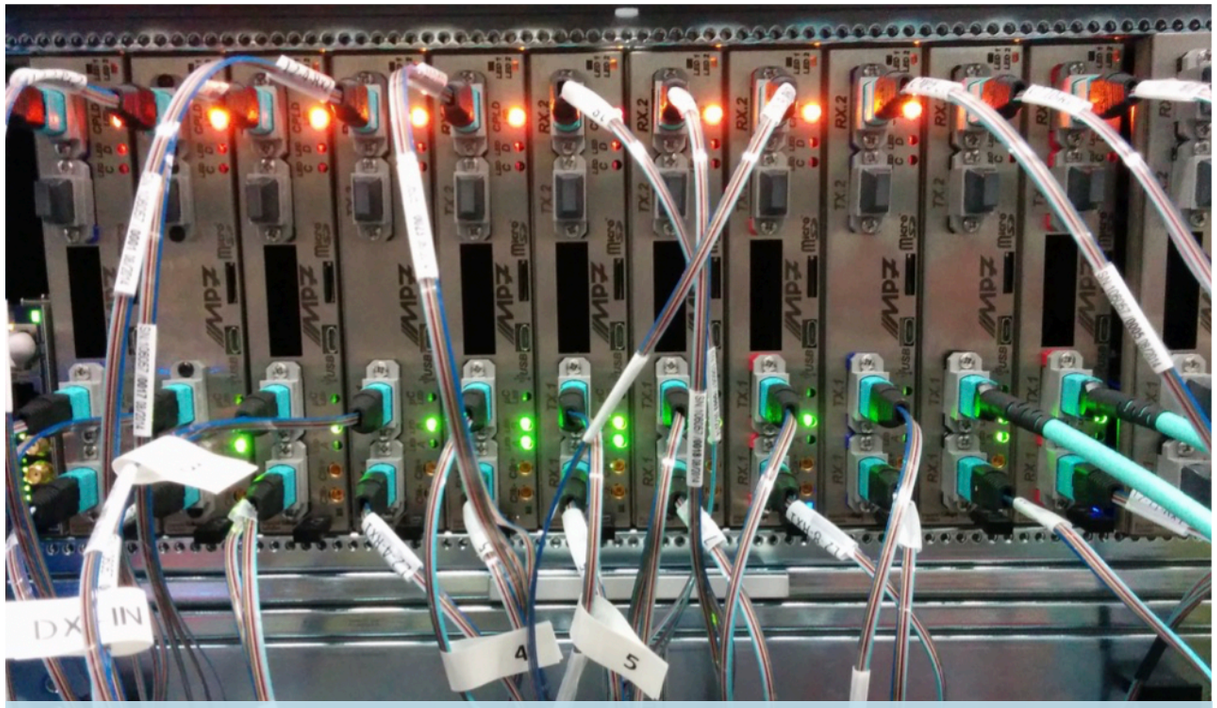
Layer 1 pre-processor cards



Layer 1 → Layer 2 patch panel



MP7 Rack



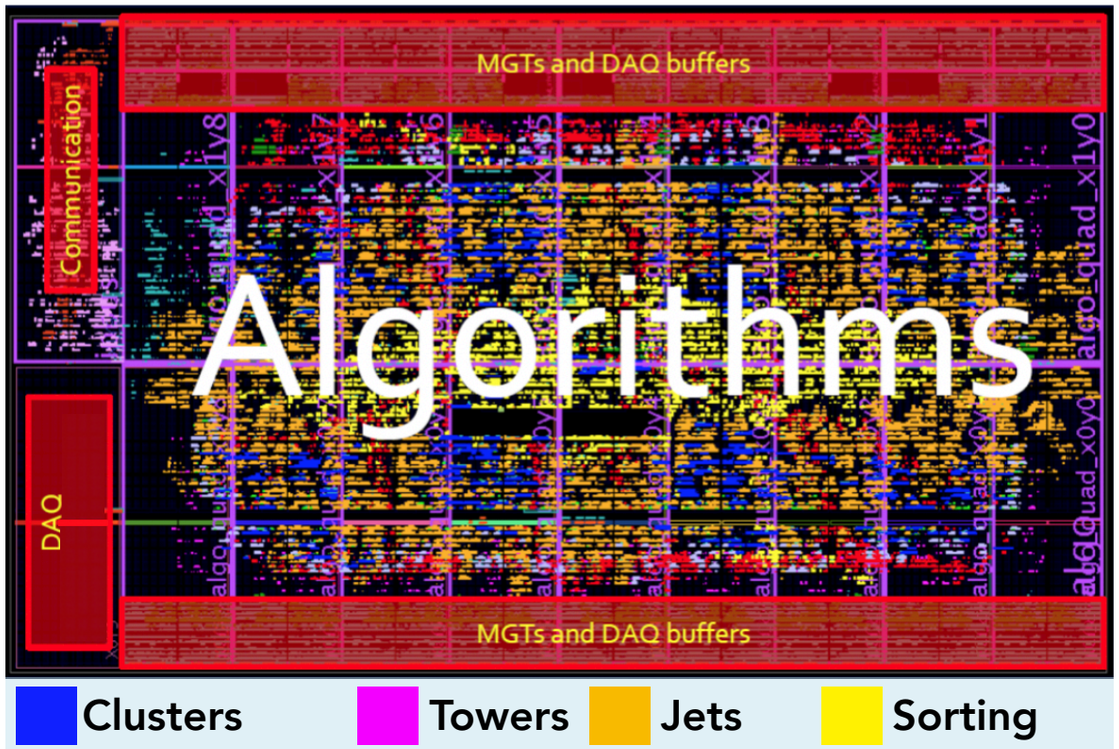
Layer 2 crate: MP7 main processor boards

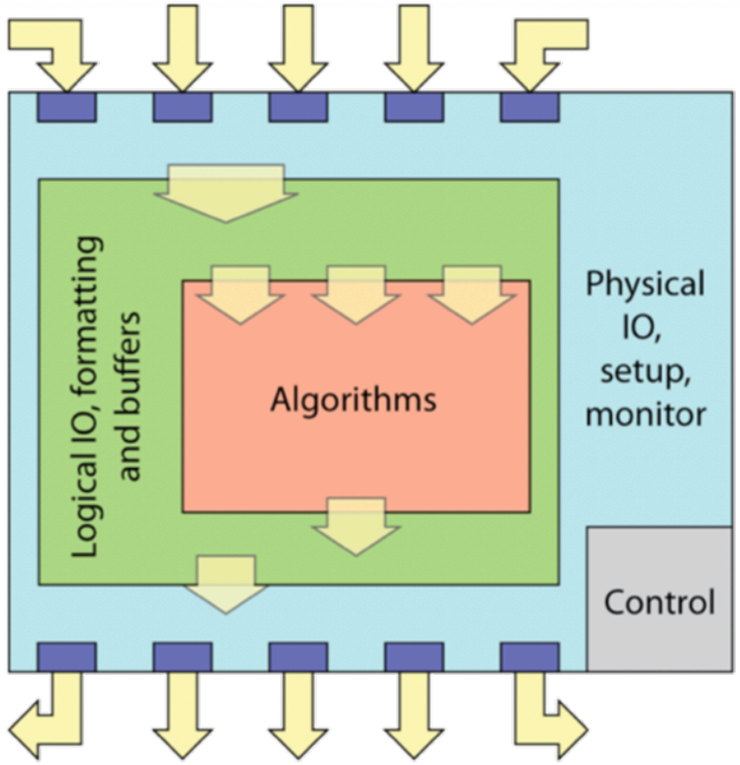
MP7 main processor board

Master Processor Virtex 7

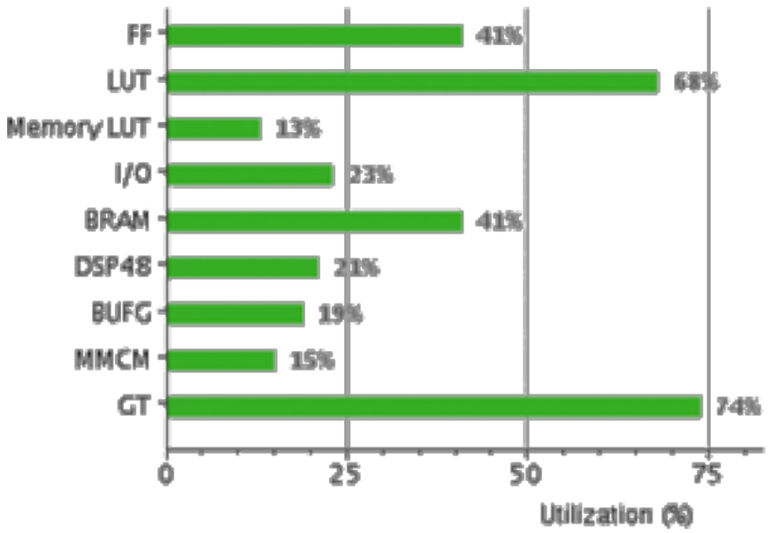
- Main workhorse for L1 calorimeter trigger upgrade
- uTCA form factor
- Also used in BMTF, uGMT, uGT
- 1.5 Tb/s signal processor board Xilinx XC7VX690T FPGA
- 72 Tx + 72 Rx links @ 10 Gb/s Avago MiniPOD optics
- GbE, AMC13/TTC/TTS, PCIe, SAS, SATA, SRIO backplane links
- AVR UC3A3256 microcontroller
- MicroSD for FPGA booting, used as firmware repository

MP7 firmware architecture





- Modular firmware design, 240 MHz clock
  - split into algorithm and infrastructure
  - allows for independent development
- High resource usage due to complex algorithms
- Significant care required for timing constraints and clock distribution



Run-time configuration: Funky Minibus

- Funky minibus provides firmware interface for run-time configuration of algorithm parameters
- Consists of clock line, data line, and instruction specifying what to do with data
- Bus is self-identifying: reports ordering, names and sizes of endpoints
- All data treated as BLOBs (LSB first): endpoint receives data, fills up, ignores remaining bits
- Endpoints joined together in daisy chain
- 6 types of instructions: ignore, data, read, write, lock, unlock
- Typical use case: send unlock instructions to all endpoints, send write instruction, send data for filling the first endpoint, send lock instruction → loop until all endpoints have been written
- 4 IPbus endpoints:
  - register for sending instruction down bus
  - register for loading data for sending down bus
  - register reporting bus size
  - RAM containing list of endpoints ("infospace")

- All calorimeter trigger object level calibrations and algorithm parameters use this bus
  - Has been critical part of flexibility of calorimeter trigger firmware to adapt quickly

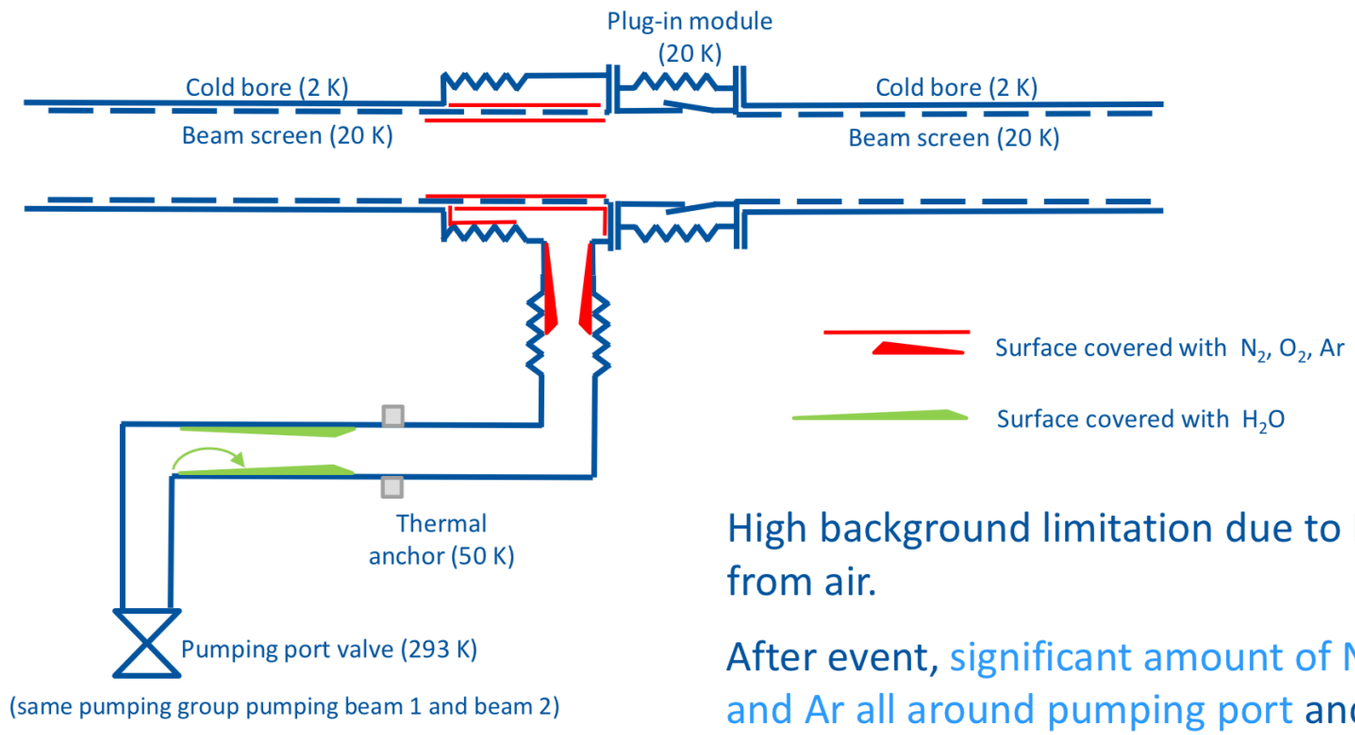
1. Introduction
2. CMS Phase 1 calorimeter trigger upgrade
3. **Run 2: operating in high pileup conditions**
4. Phase 2: much higher pileup conditions
5. Summary

L1 Trigger: pileup dependence of rates

Imperial College London

- Towards end of 2016 pp running, LHC raised instantaneous luminosity by increasing number of protons per bunch → significant increase in number of interactions per crossing
- This increase in pileup led to (very!) non-linear increase in rates for some trigger algorithms
  - particularly MET and multi-jet triggers
- Significant contribution to this pileup dependence driven by larger towers at high η in endcap
  - For HTT, simply η-restricted to same as offline
- For MET solution was not so simple, need full calorimeter
  - required event level pileup mitigation
- Flexibility of calorimeter trigger system allowed implementation of run-time configurable LUT
  - excludes towers from MET calculation below E_T threshold determined from η of tower & estimation of event pileup
- Significantly reduced pileup dependence of MET triggers
  - kept rates manageable in high pileup conditions
  - maintained low thresholds and high trigger efficiencies

2017 p-p running: a challenge for L1T



Courtesy of '16L2 task-Force'

High background limitation due to H_2O from air.

After event, significant amount of N_2, O_2 and Ar all around pumping port and beam screen bellows. However, most likely most of those gases stick on the pumping port bellows.

- During 2017 running, a problem in one of the LHC sectors produced frequent beam dumps
  - due to gas contamination around pumping port
  - beam interaction with gas produces electron clouds
  - electron clouds cause beam dump

- Mostly fixed by warming to 90K and pumping out, only a few 16L2 beam dumps this year

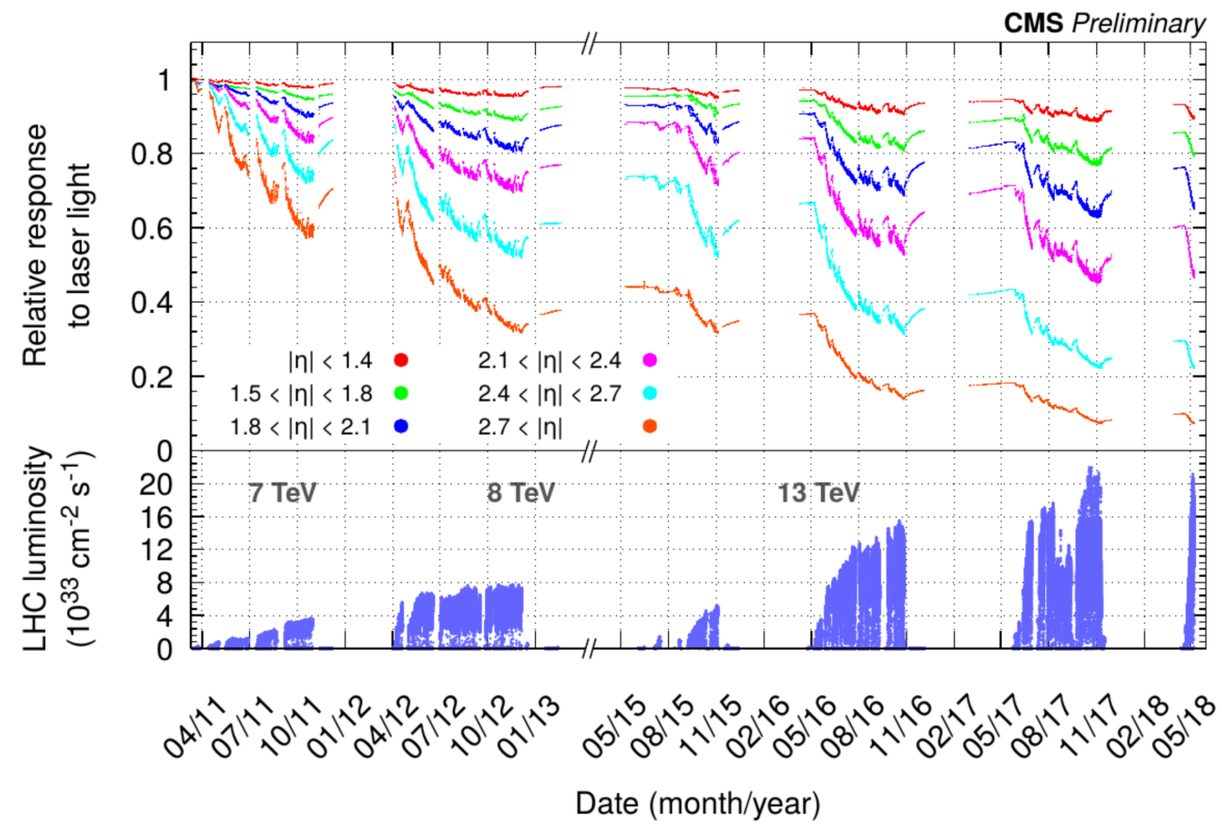
2017 p-p running: a challenge for L1T

- Mitigated by changing fill scheme: 48 bunch trains → 8b4e (8 filled, 4 empty)
- To maintain instantaneous luminosity, bunch intensity increased
- Increased overall level of out of time pileup, which is larger for first few bunches of train
- Calorimeter front end dynamic pedestal subtraction not as effective for first 2 filled bunches
  - need to measure pedestal to subtract over first few bunches in train
- Leads to further increase in the pileup dependence various L1 algorithm rates
- Mitigated by tuning the MET pileup subtraction LUT to increase tower thresholds
- Important to consider how changes in fill scheme can affect TPs and trigger rates for Phase 2

Responding to changes in detector response

- Electromagnetic & hadronic calorimeter response changes significantly over various timescales
- Generally response degrades as it is irradiated → increase channel gains
- ECAL exhibits significant recovery during technical stops / shutdown (even inter-fill!)

- Heavily irradiated high η towers in endcap suffer significant loss of transparency, do not recover well
  - need to amplify channels by up to 50×
  - noise floor raised above minimum tower E_T for some towers in η-φ

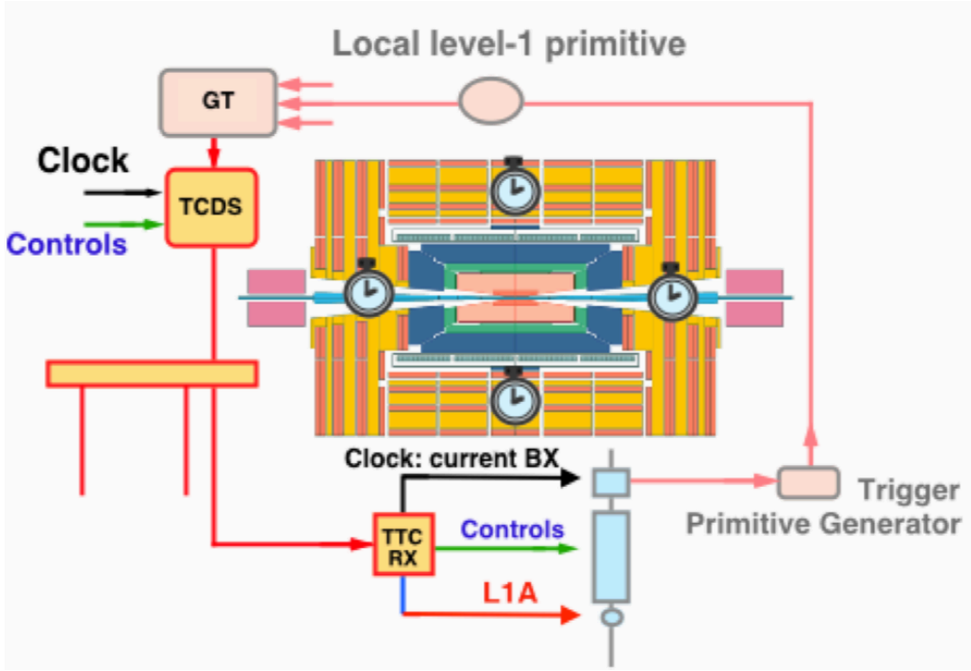


- To avoid including ECAL noise in L1T algorithms, implemented η-dependent zero suppression
- Possible using ECAL TP calibration LUTs: scale factors applied for given η and E_T
- Significantly reduced rate for algorithms using towers at high η, particularly jets & MET

Measuring the trigger latency

- Measuring latency of various trigger subsystems not necessary for legacy trigger
  - just needed to time in final L1 accept decision with detector

- Phase 1 trigger upgrade much more complex: improved algorithms & additional stages → latency increased
- Now important to understand every BX of latency in system, all the way from trigger primitives to TCDS, including fibre lengths, serialisation/de-serialisation etc.
- Find out where latency can be saved → any algorithm or system improvements that require additional latency can be implemented if/when necessary



- For Phase 1 far from a trivial process: no "standard" method built into system to measure latency of various trigger sub-components
  - took a lot of time and effort to understand and measure trigger subsystem latencies
- For Phase 2, an order of magnitude more complex than Phase 1, extremely useful to have built-in ability to accurately measure latency at each subsystem
- In addition to easily determining where latency can be saved for more important uses, will facilitate timing in different trigger paths that arrive at a common subsystem, e.g. correlator

Some lessons from Phase 1

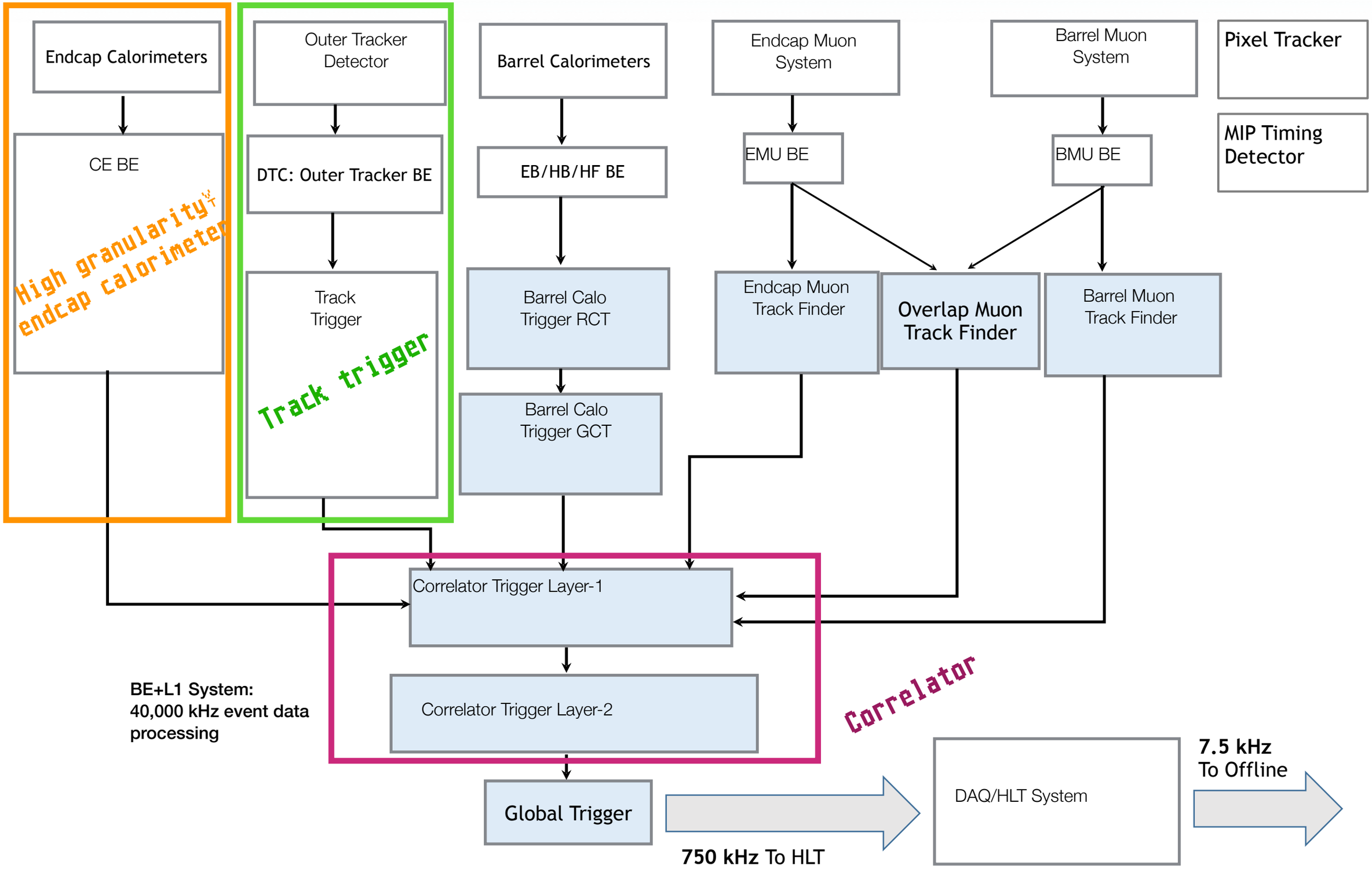
- Always prepare for LHC to deliver more than expected
  - has sometimes been difficult to maintain rates within the L1 bandwidth
  - either due to increase in instantaneous luminosity, increase in pileup, or both
  - rates can sometimes behave unpredictably with changes in beam conditions

- Make algorithms as flexible and configurable as possible
  - ability to update and tune algorithms easily has enabled us to maintain stability in the L1 trigger throughout many changes in the LHC beam conditions

- Understand calibrations well
  - complex detector and sophisticated algorithms require significant care with calibrations
  - small change in calibrations can have large effect on rates and efficiencies
  - must maintain consistency from detector → trigger primitives → towers → algorithms
  - wherever possible use data driven calibrations, try not to rely on MC, or at least cross-check with data

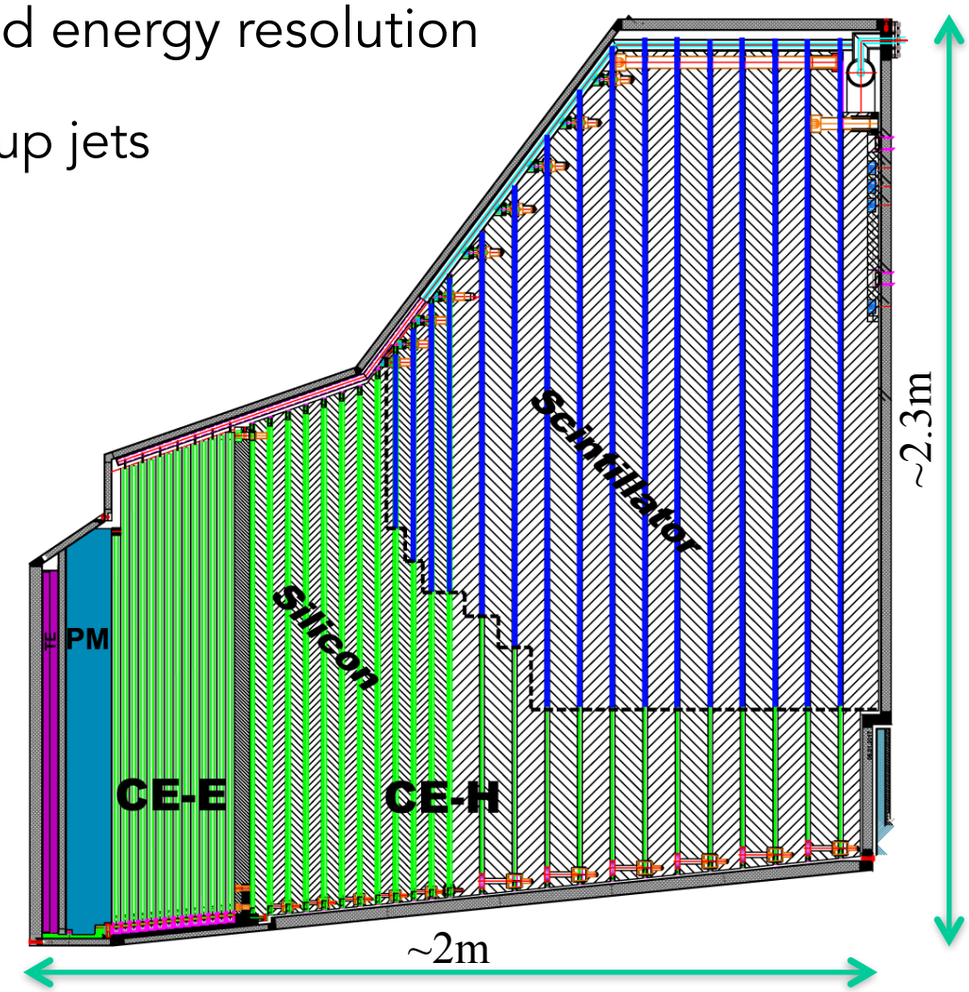
1. Introduction
2. CMS Phase 1 calorimeter trigger upgrade
3. Run 2: operating in high pileup conditions
4. Phase 2: much higher pileup conditions
5. Summary

Phase 2 L1T: architecture (WIP!)



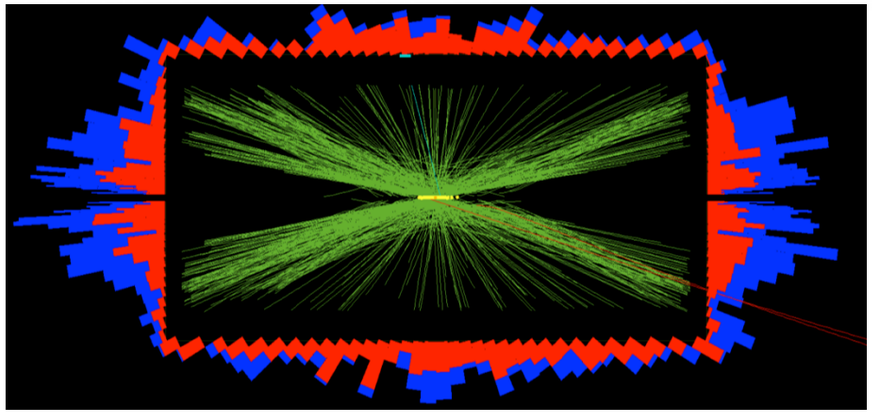
Phase 2 pileup mitigation: HGCAL

- Current ECAL + HCAL calorimeter endcaps will be replaced during Phase 2 upgrade [1]
- High granularity calorimeter endcap will measure electromagnetic + hadronic components
- Built to facilitate particle flow: 3D separation of individual particle showers
- Fine lateral granularity limits energy measurement region → minimise inclusion of pileup energy
- Fine sampling of longitudinal shower development: good energy resolution
  - significantly improve ability to discriminate against pileup jets
- Precision time measurement of high energy showers
  - helps with rejection of energy from pileup
- Fast shaping time: peaking-time 20ns
  - minimise out-of-time pileup



[1] https://indico.cern.ch/event/697988/contributions/3075473/attachments/1717249/2771593/HighGranularityCalorimeter_TWEPP_200918.pdf

Phase 2 pileup mitigation: Track Trigger



- For Phase 2 CMS will have a track trigger
  - provide tracks to the L1 trigger algorithms
- Crucial in mitigating high pileup of HL-LHC
  - up to 200 interactions per bunch
- Run vertex finding and associate L1 objects to primary vertex
- Run Particle Flow and PUPPI (Pileup Per Particle Identification) as is currently done offline
- Will still of course need robust standalone calorimeter and track only algorithms

- Ongoing work to characterise vertex finding with L1 tracks
  - provide necessary info to PF and PUPPI algorithms
- Eventually train machine learning algorithm
  - identify leading vertices in the event
  - utilise HLS4ML resources

Summary

- Many lessons learned from Phase 1 upgrade that will inform design choices for Phase 2
- Up to 200 pileup from HL-LHC: need to be prepared for this at L1 trigger
- Build system that can maintain high trigger efficiencies whilst keeping rates within bandwidth
- Need as much flexibility as possible to allow L1 to respond effectively to beam conditions
- Important for L1 to be able to deal with fluctuations in the response of the detectors
- Require both robust standalone calorimeter & track algorithms in addition to complex high performance algorithms that use particle flow, PUPPI, machine learning etc.