TDAQ for the LHC experiments

Niko Neufeld
CERN/EP Department

ISOTDAQ 2016, Israel

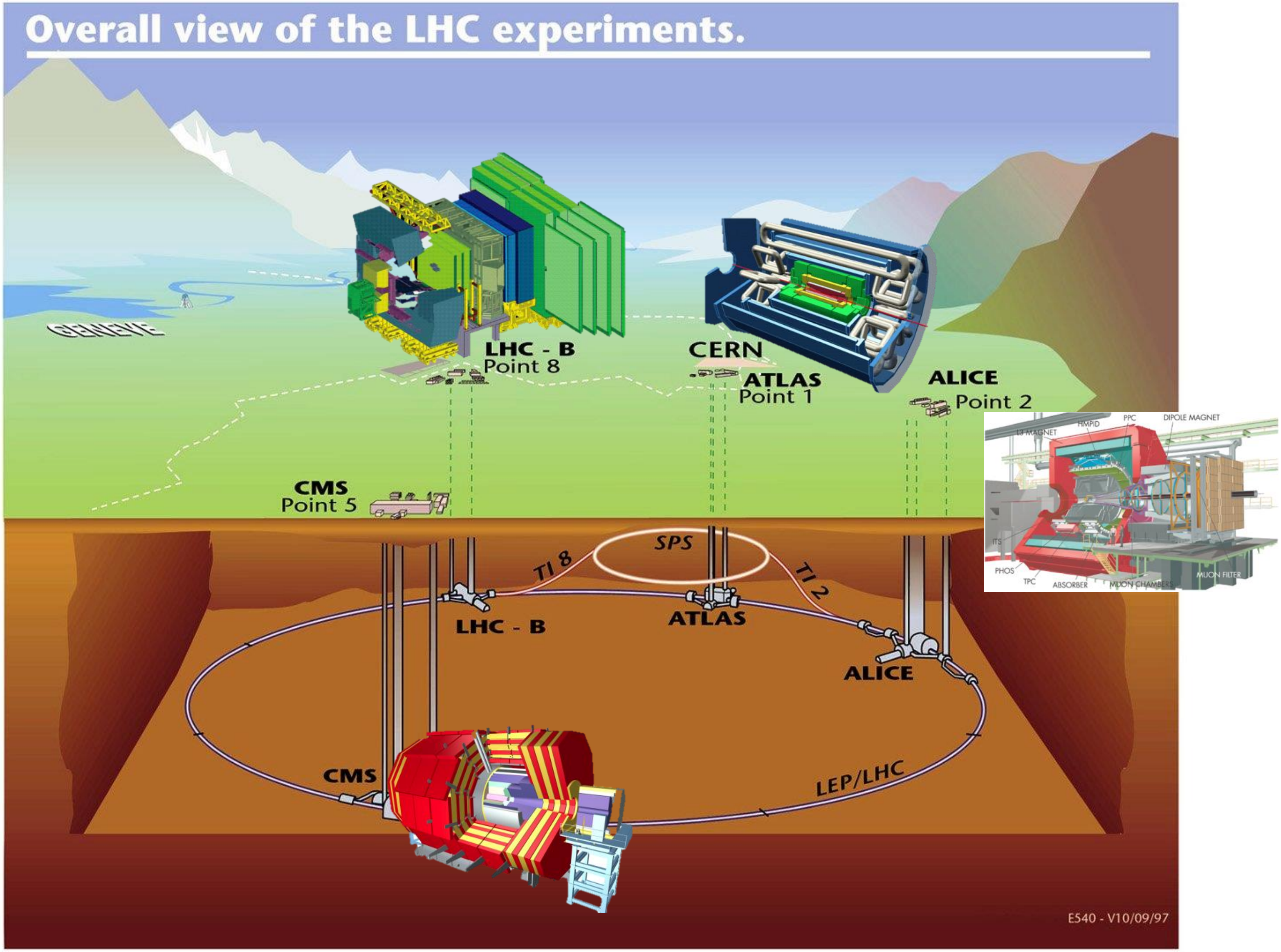
Thanks & disclaimer

- I am indebted to the DAQ and trigger teams of ALICE, ATLAS, CMS and LHCb for material and helpful comments. In particular:
  - Pierre vande Vyvre, David Francis, Brian Martin, Frans Meijers, Francois Vasey, Beat Jost, Giovanna Lehmann, Will Buttinger, Nick Ellis, Hannes Sakulin
- This tries to be a balanced, fair, pedagogical overview – however I have been working too long and too much on these systems to claim a completely objective view!
- Lots of important material missing (run-control)

Overview

- The challenge of the LHC
  - Large amount of data / high rate of collisions
- How do we cope with many, many Terabits / s - the LHC Trigger systems
  - L1 Trigger (high pT) in CMS and ATLAS
  - Timing and Trigger distribution
  - Flexible electronics (FPGAs)
- After the L1 trigger – Data Acquisition
  - Networked DAQ
  - Staged read-out (ATLAS)
  - Full readout LHCb & CMS
- High Level Trigger
  - Event selection in software in big server farms (ATLAS, CMS, LHCb)
  - Data-reduction & high-speed storage (ALICE)
- Where do we go from here? The future: new links, faster networks, GPUs, etc…
- Additional material (which I probably can't cover)
  - Controlling it all: State-machines, expert-systems and all that

The Large Hadron Collider



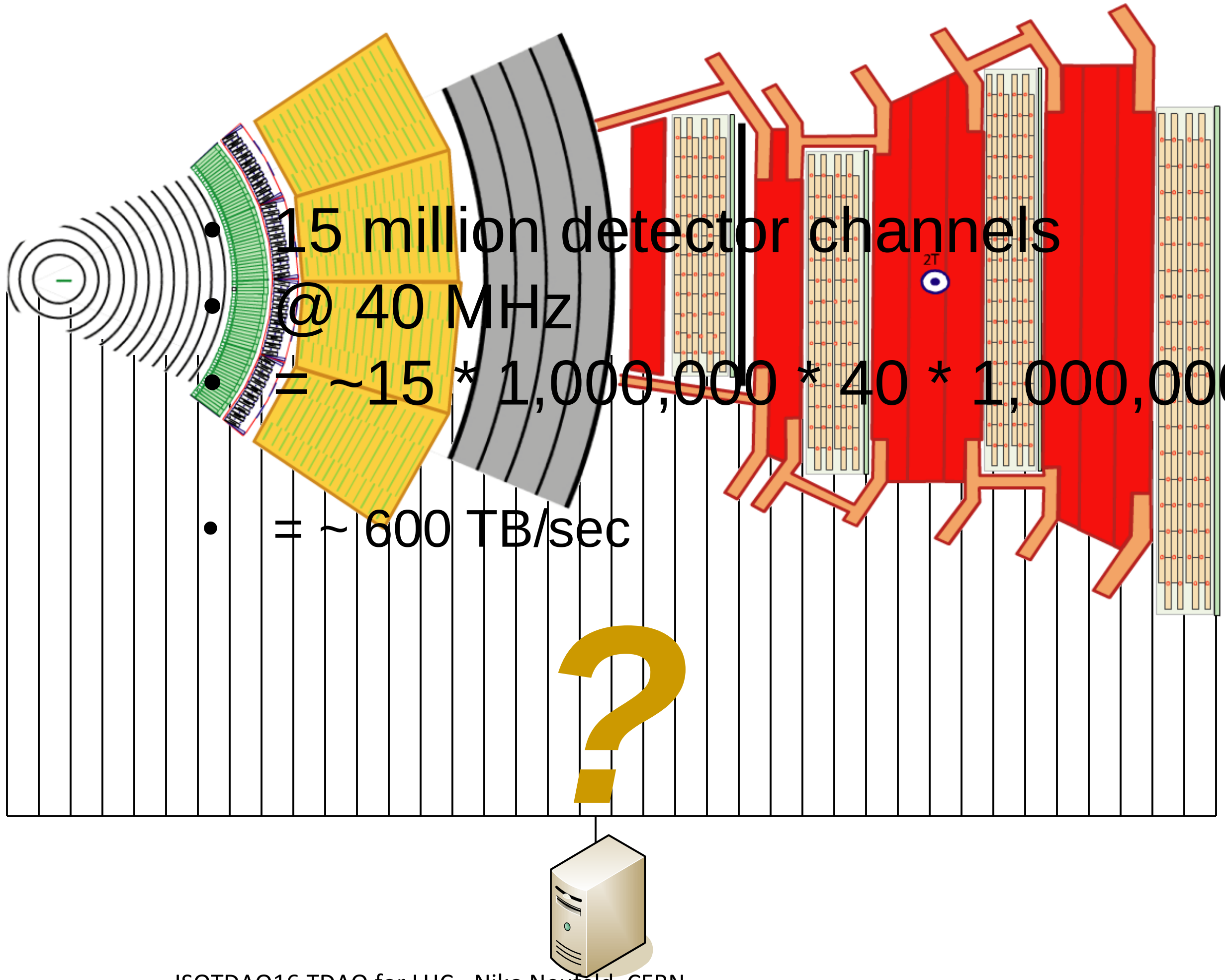
The Challenge

- The goal of the LHC is clear: find the Higgs and (at least hints of) new physics →
  - Lots of collisions (to find rare events)
  - High energy (to find heavy particles)

$$L = \frac{N^2 f}{4ps_x s_y} = \frac{N^2 f}{4pe_n b^*}$$

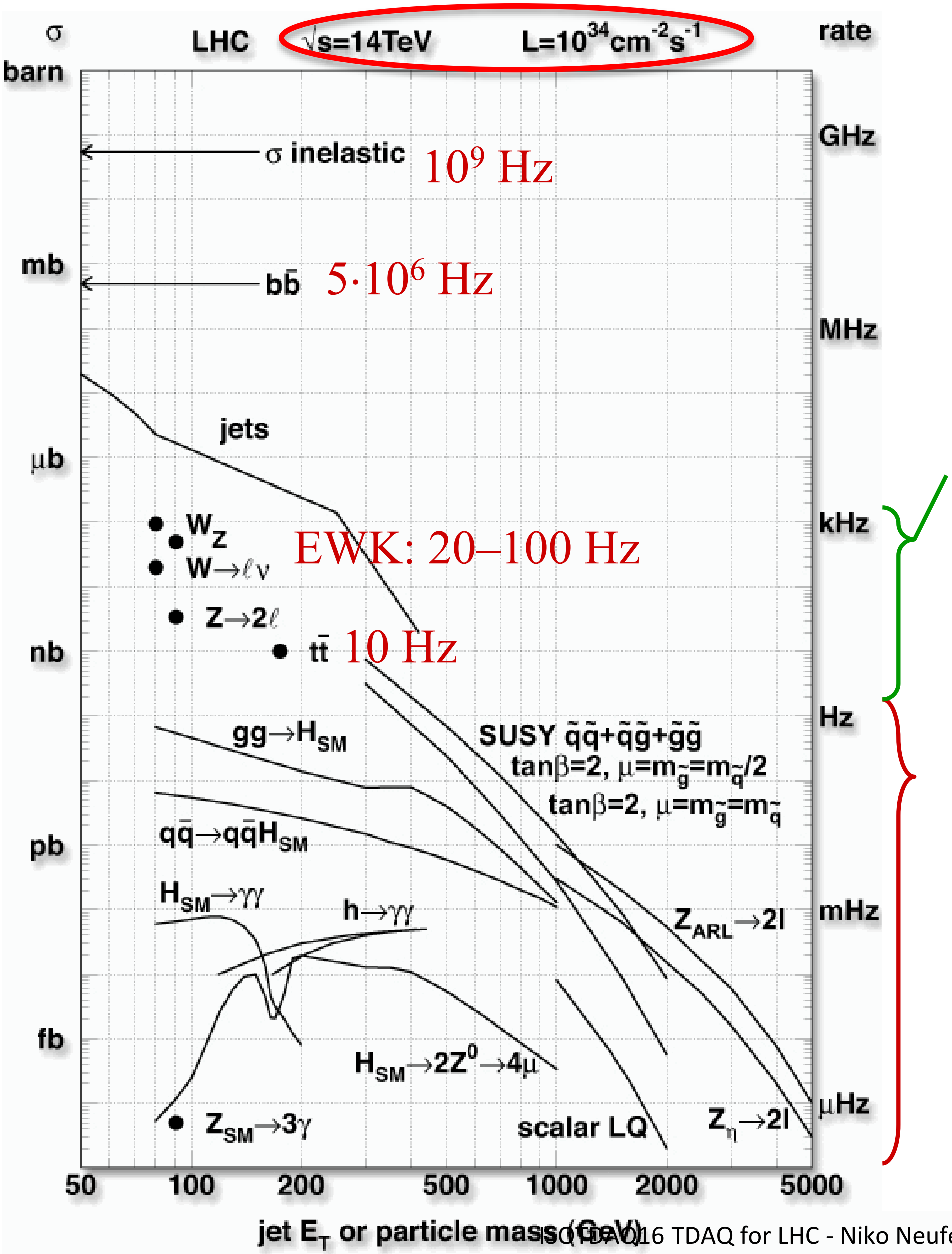
- It turns out that the best way to get the most collisions is to increase N and keep f constant
- In 2011 and 2012: f = 20 MHz, N = 1.5 10^{11} , from 2015 f = 40 MHz (max!) but N will still go up
- This has a price → high number of interactions per crossing ("pile-up") → large occupancy in the detectors → *big events*

The Data Acquisition Challenge at LHC



- 15 million detector channels
- @ 40 MHz
- = ~15 * 1,000,000 * 40 * 1,000,000 bytes
- = ~ 600 TB/sec

Should we read everything?

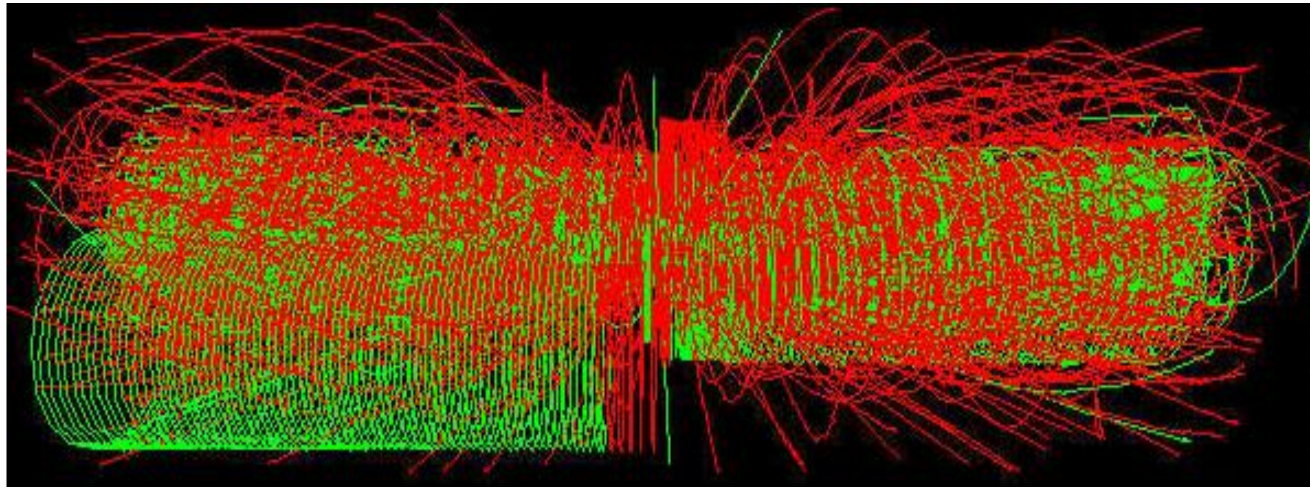


- A typical collision is “boring”
  - Although we need also some of these “boring” data as cross-check, calibration tool and also some important “low-energy” physics
- “Interesting” physics is about 6–8 orders of magnitude rarer (EWK & Top)
- “Exciting” physics involving new particles/discoveries is ≥ 9 orders of magnitude below σ_{tot}
  - 125.09 GeV Higgs 0.1 Hz*
- We *just* ☺ need to efficiently identify these rare processes from the overwhelming background <u>before</u> reading out & storing the whole event

***Note: this is just the production rate, actual detection is more rare!**

Know Your Enemy: pp Collisions at 14 TeV at 10^{34} $cm^{-2}s^{-1}$

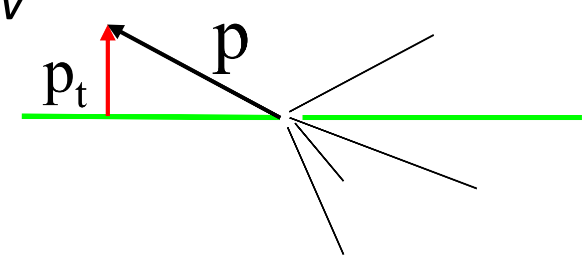
- σ(pp) = 70 mb --> >7 x 10^8 /s (!)
- In ATLAS and CMS* 20 – 30 min bias events overlap
- H→ZZ
  Z →μμ
  H→ 4 muons: the cleanest (“golden”) signature



*)LHCb @4x10^{33} cm^{-2}-1 isn’t much nicer and in Alice (PbPb) is even more busy

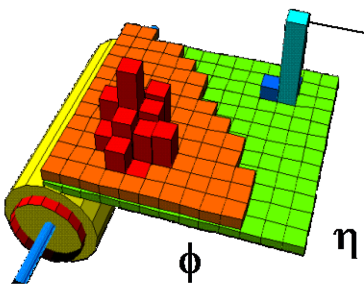
Mother Nature is a kind woman (after all)

- pp collisions produce mainly hadrons with transverse momentum "p_t" ~1 GeV
- Interesting physics (old and new) has particles (leptons and hadrons) with large p_t:
  - $W \rightarrow e\nu$: $M(W)=80$ GeV $p_t(e) \sim 30\text{-}40$ GeV
  - $H(125\ \text{GeV}) \rightarrow \gamma\gamma$: $p_t(\gamma) \sim 50\text{-}60$ GeV
  - $B \rightarrow \mu D^{*+}\nu$ $p_t(\mu) \sim 1.4$ GeV
- Impose high thresholds on the p_t of particles
  - Implies distinguishing particle types; possible for electrons, muons and "jets"; beyond that, need complex algorithms
- **Conclusion: We need to watch out for high transverse momentum electrons, jets or muons**

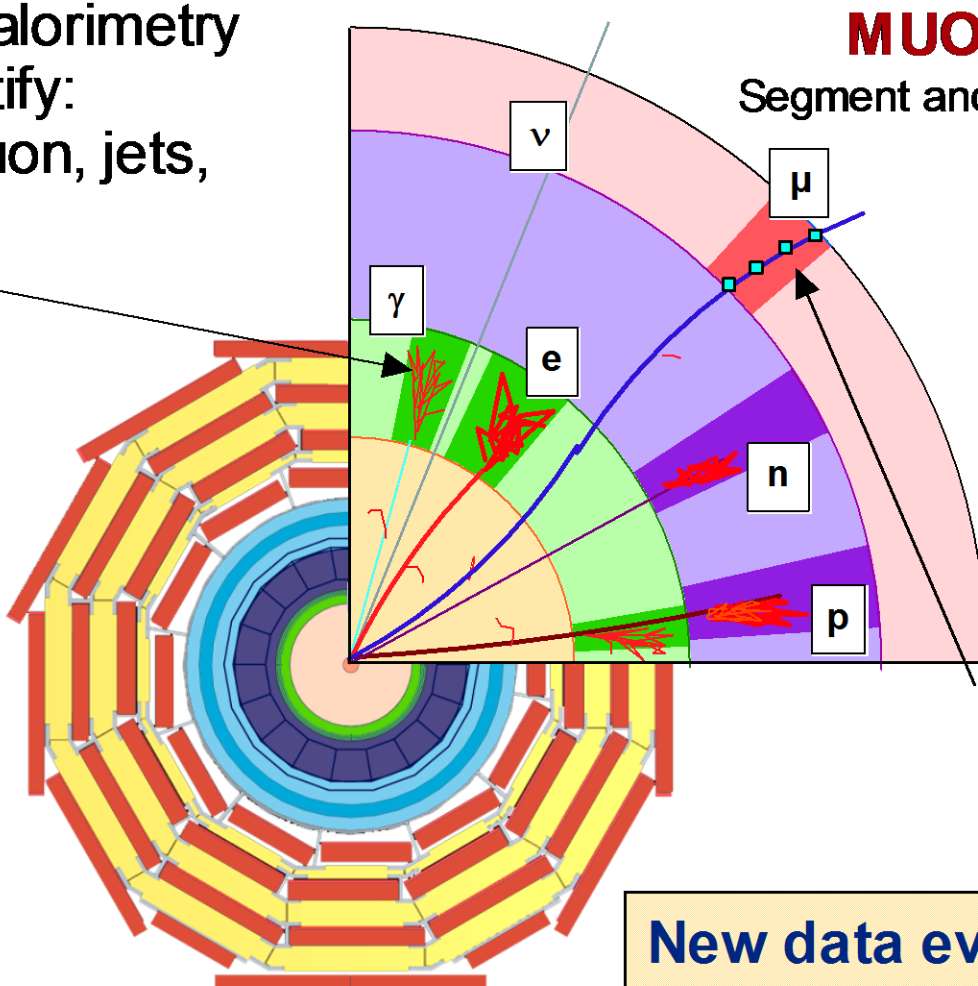


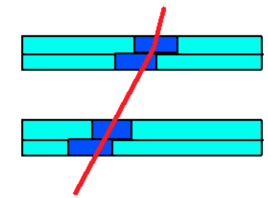
How to defeat minimum bias: transverse momentum p_t

Use prompt data (calorimetry and muons) to identify:
High p_t electron, muon, jets, missing E_T



MUON System
Segment and track finding

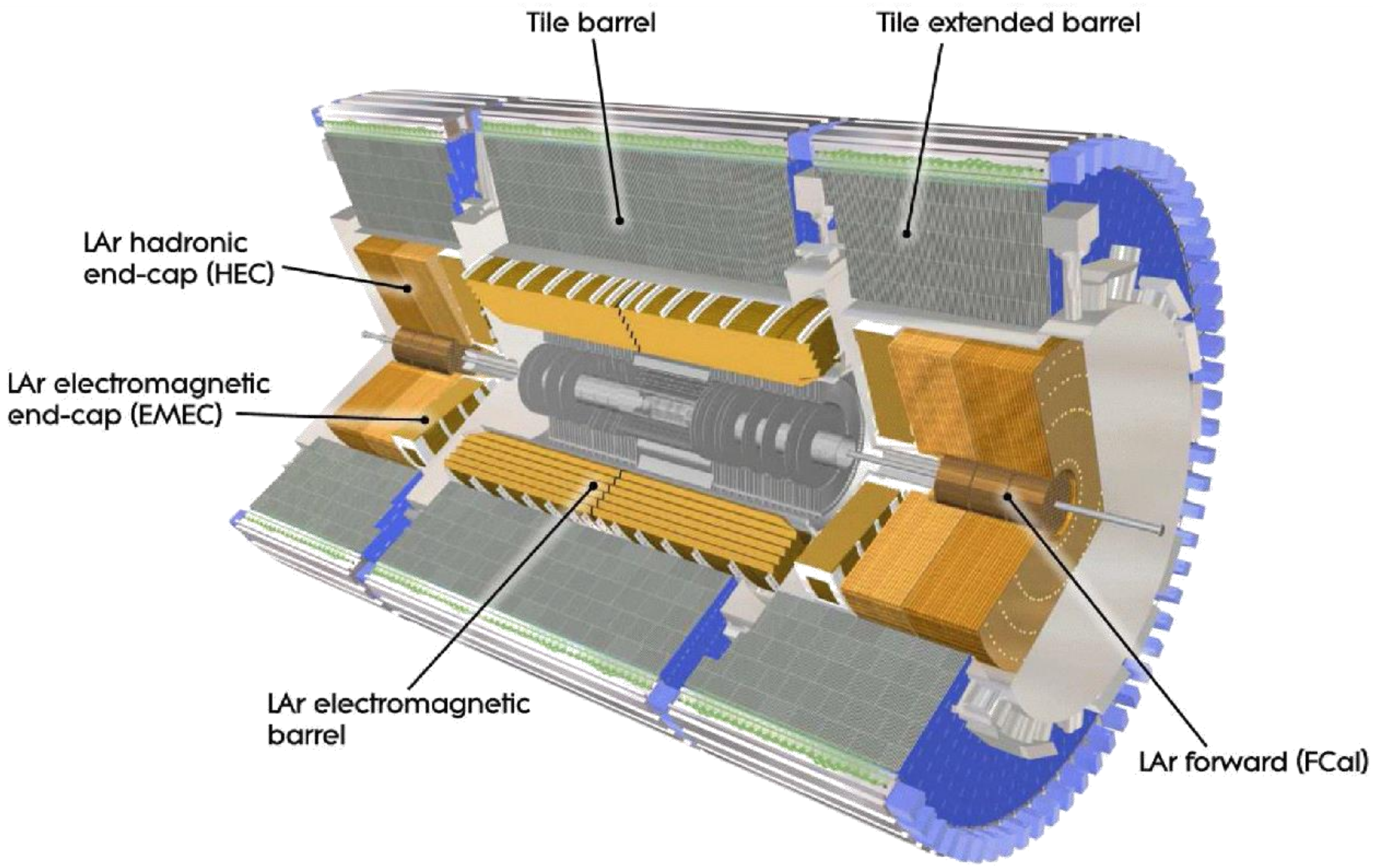




CALORIMETERs
Cluster finding and energy deposition evaluation

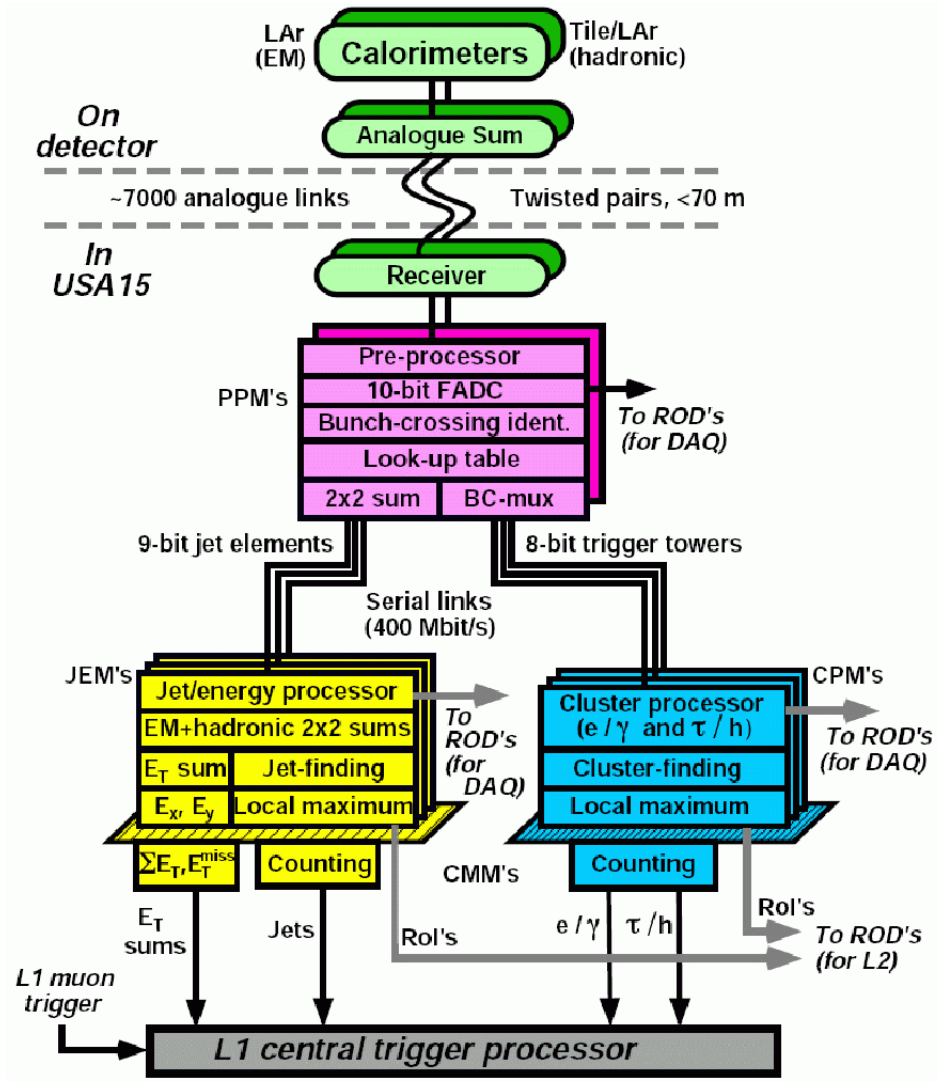
New data every 25 ns
Decision latency ~ µs

L1 trigger in practice: ATLAS Calorimeters



ATLAS L1 Calo Trigger

- Form analogue towers 0.1 x 0.1 (δη x δϕ)
- digitize, identify
- bunch-xing, Look-Up Table (LUT) → E_T
- Duplicate data to Jet/Energy-sum
- (JEP) and Cluster (CP) processors
- Send to CTP 1.5 µs after bunch-crossing (“x-ing”).
- Store info at JEP and CP to seed next level of trigger



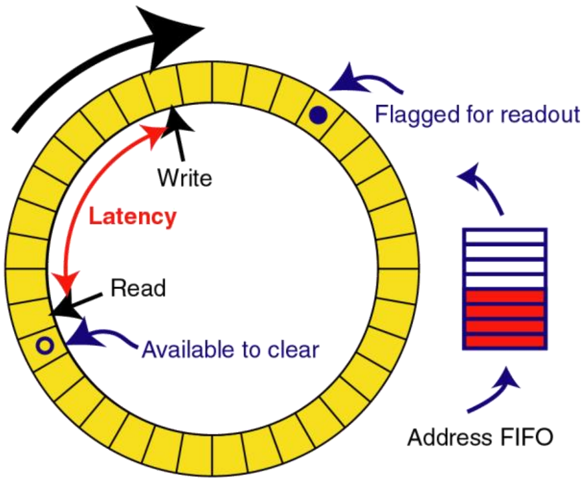
It's complicated…

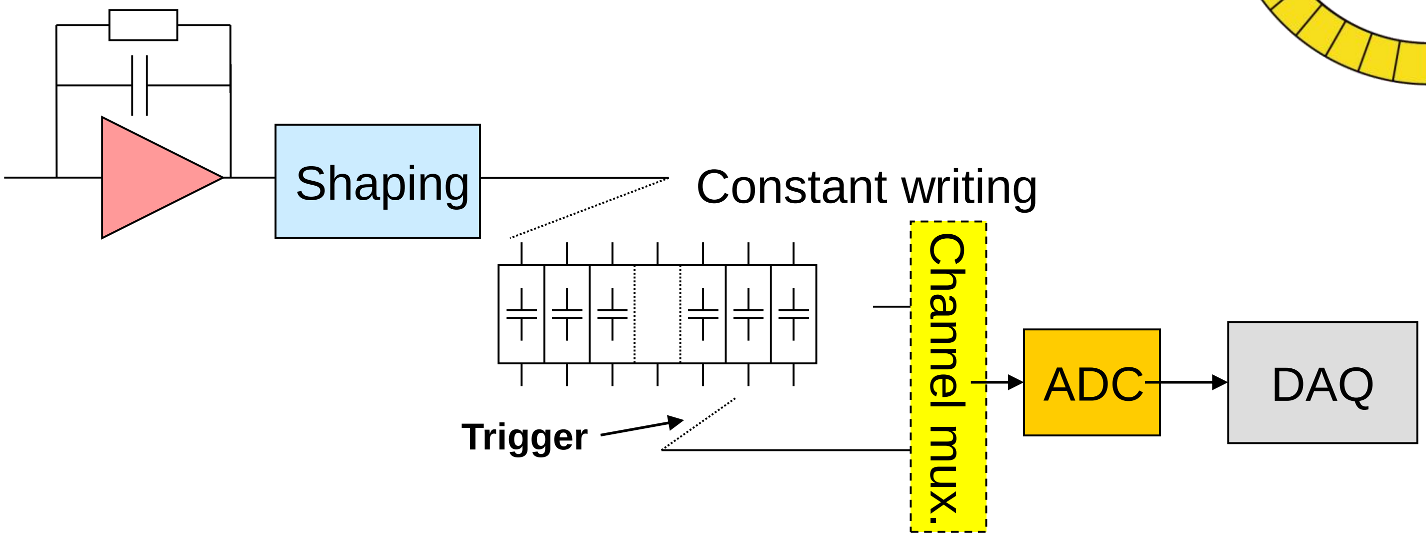
- The array of E_T values computed in the Preprocessor has to be transmitted to the CP
  - Use digital electrical links to CP modules (LVDS)
    - ~5000 links @ 400 Mbps
    - Convert to 160 Mbps single-ended signals on CP modules (LVDS rx; serializer FPGA)
  - Fan out data to neighbouring modules over very high density custom back-plane
    - ~800 pins per slot in 9U crate
    - 160 Mbps point-to-point
  - Fan out data to 8 large FPGAs in each CP module
    - Receive data at 160 Mbps in FPGAs that implement the algorithms

- The e/γ (together with the τ/h) algorithm is implemented in FPGAs
  - This has only become feasible with recent advances in FPGA technology
    - Require very large and very fast devices
  - Each FPGA handles 4×2 windows
    - Needs data from 7×5×2 towers ($\eta\times\phi\times${E/H})
  - Algorithm is described in a language (VHDL) that can be converted into the FPGA configuration file
    - Flexibility to adapt algorithms in the light of experience
  - Parameters of the algorithms can be changed easily
    - e.g. cluster-E_T thresholds are held in registers that can be programmed without reconfiguring the FPGAs

Triggered read-out

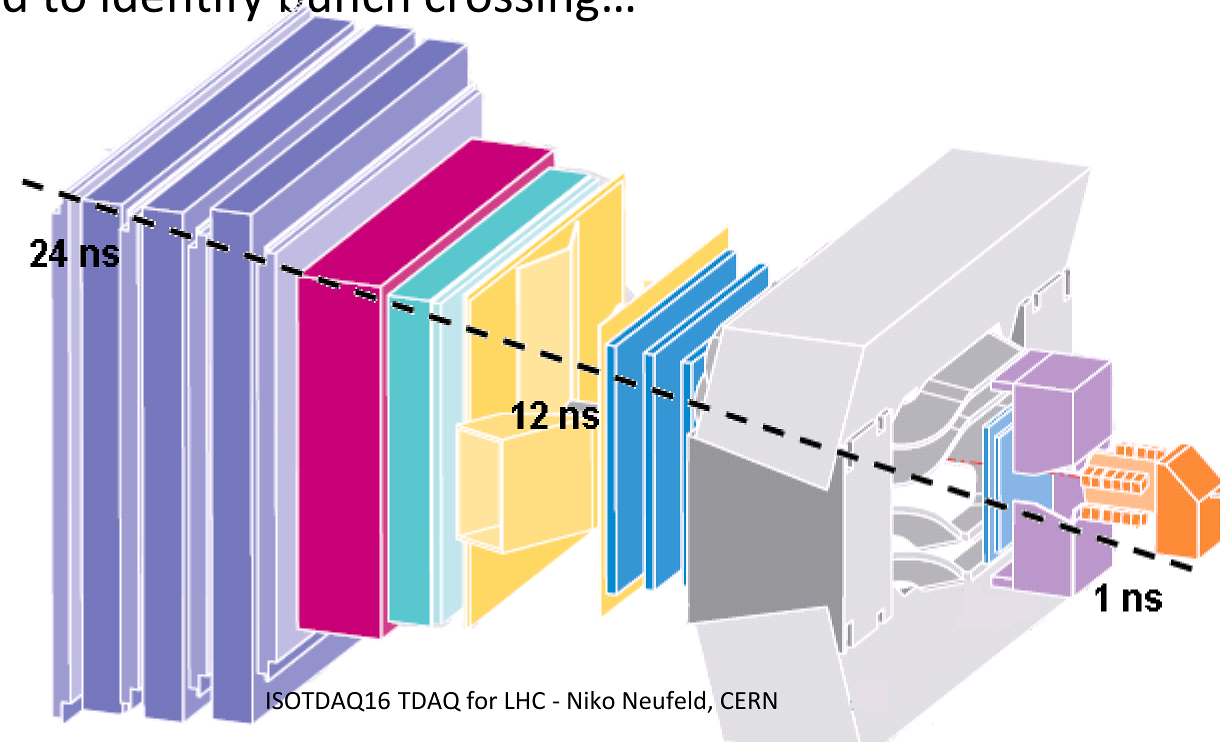
- Trigger processing requires some data transmission and processing time to make decision so front-ends must buffer data during this time. This is the trigger latency
- For constant high rate experiments at LHC "pipeline" buffer is needed in all front-end detector channels: (analog or digital) (e.g. circular buffer →)





Challenges for the L1 at LHC

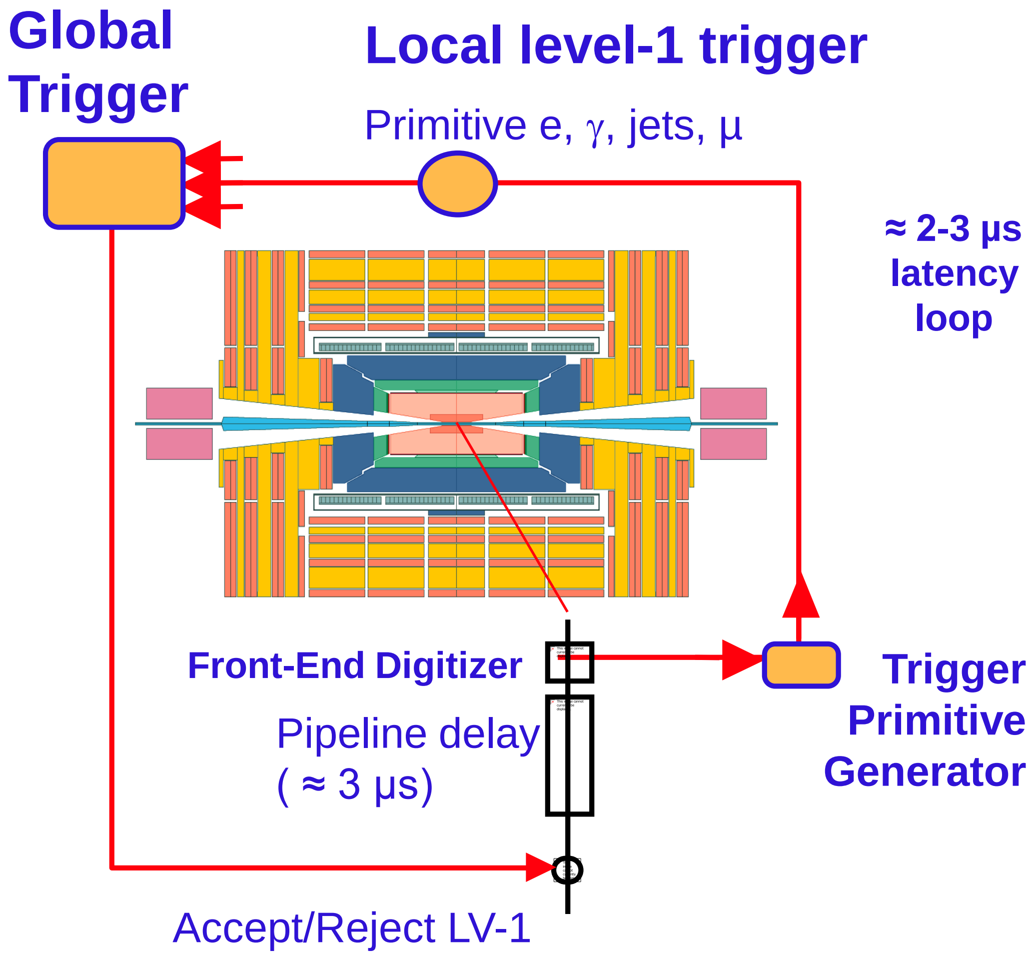
- N (channels) ~ O(10^7); ≈20 interactions every 25 ns
  - need huge number of connections
- Need to synchronize detector elements to (better than) 25 ns
- Detector signal/time of flight can be > 25 ns
  - integrate more than one bunch crossing's worth of information
  - need to identify bunch crossing...



Distributing the L1 Trigger

- L1 decision has to be brought for each crossing to all the detector front-end electronics elements so that they can send of their data or discard it
- All experiments use the common TTC system (c.f. Paolo Durante: Optical links , Sunday

L1 trigger latencies	
ALICE	No pipeline
ATLAS	2.5 us
CMS	3 us
LHCb	4 us



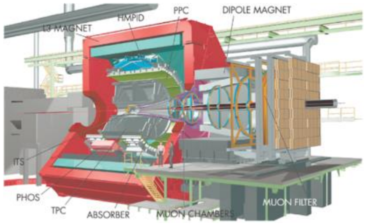
Trigger - Summary

- Lots of custom electronics
- Massive use of parallel processing in FPGAs (and some ASICs)
- Part of the electronics in radiation area
- Processing time typically 1 – 2 us (limited by buffer memory on the detector)
- Common system for reliable transmission of timing and trigger decisions (TTC) → in the future (partially) replaced by the versatile link / GBT (c.f. Paolo Durante: "Optical links", Sunday)

After the Trigger
Detector Read-out and DAQ

Readout Links of LHC Experiments

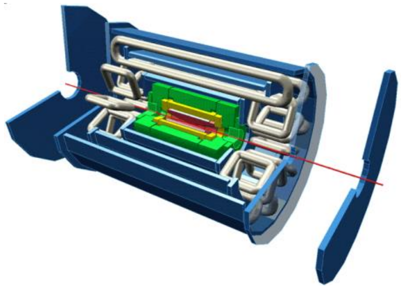
Flow Control



DDL

Optical 200 MB/s ≈ 400 links
Full duplex: Controls FE (commands, Pedestals, Calibration data)
Receiver card interfaces to PC

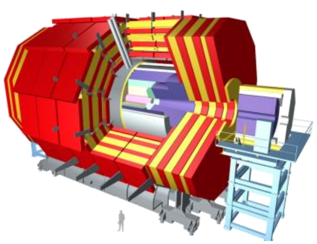
yes



SLINK

Optical: 160 MB/s ≈ 1600 Links
Receiver card interfaces to PC.

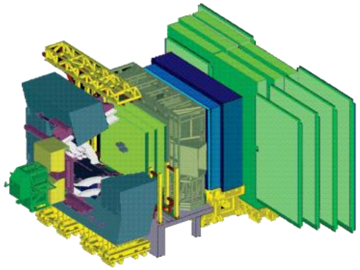
yes



SLINK 64

LVDS: 200 MB/s (max. 15m) ≈ 500 links
Peak throughput 400 MB/s to absorb fluctuations
Receiver card interfaces to 10 G Ethernet driven by an FPGA

yes



Glink (GOL)

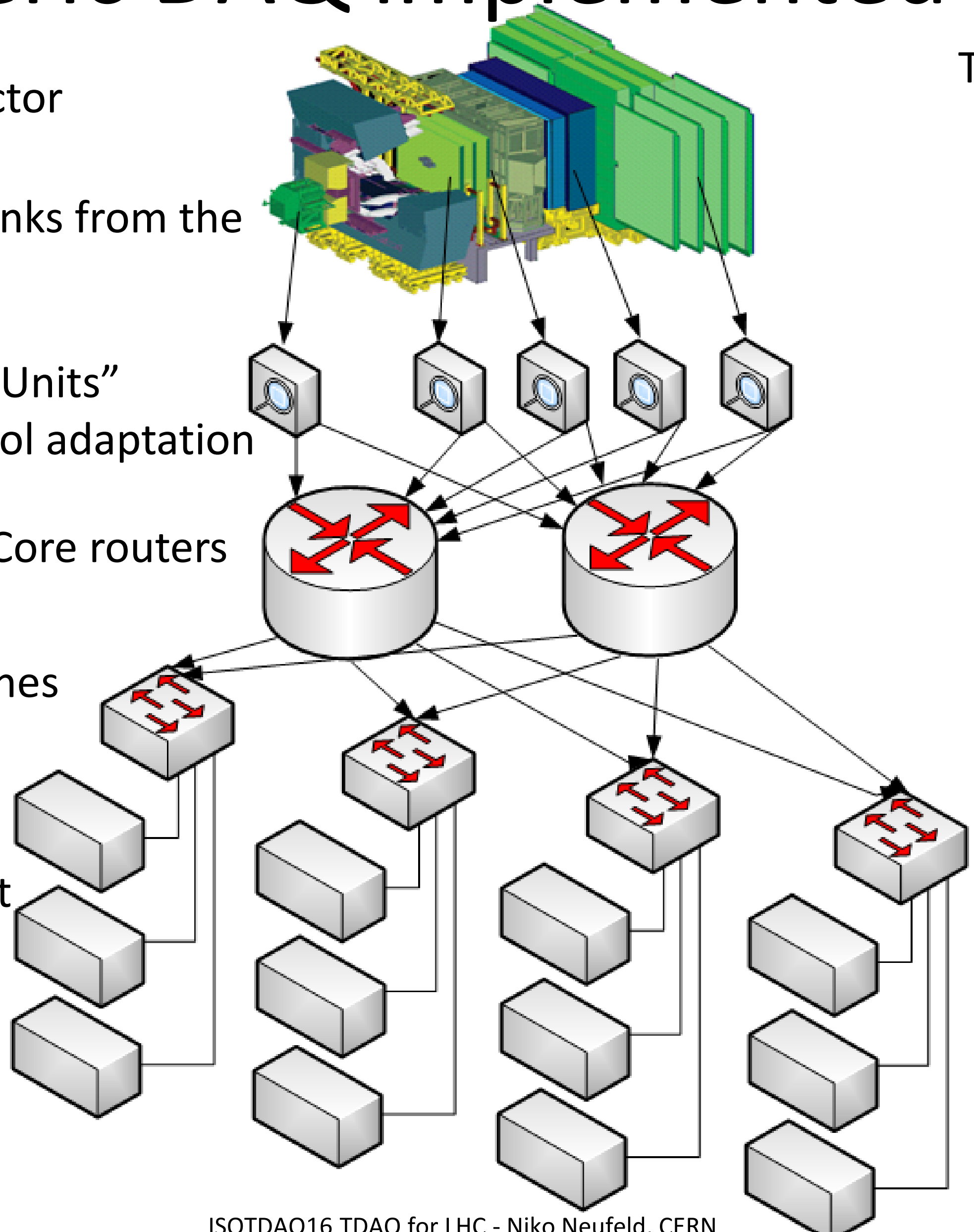
Optical 200 MB/s ≈ 400 links
Receiver card interfaces to custom-built Ethernet NIC (4 x 1 Gbit/s over copper)

(no)

DAQ design guidelines

- Scalability – change in event-size, luminosity (pileup!)
- Robust (very little dead-time, high efficiency, non-expert operators) → intelligent control-systems
- Use industry-standard, commercial technologies (long-term maintenance) → PCs, Ethernet
- Low cost ☺ → PCs, standard LANs
- High band-width (many Gigabytes/s) → use local area networks (LAN)
- "Creative" & "Flexible" (open for new things) → use software and reconfigurable logic (FPGAs)

Generic DAQ implemented on a LAN



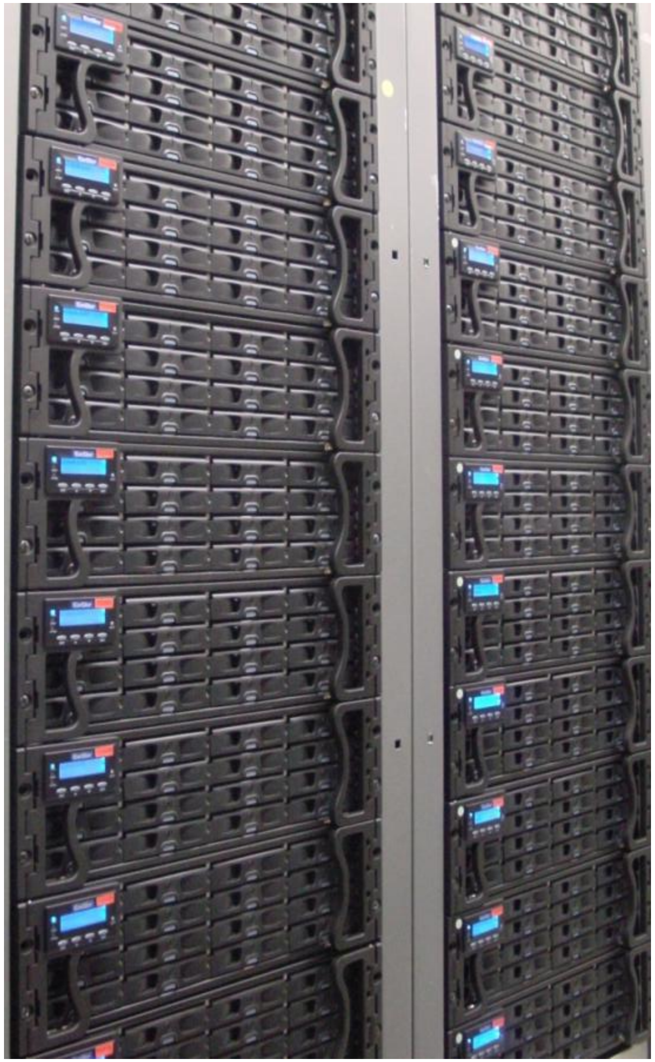
	Typical number of pieces
Detector	1
Custom links from the detector	1000
“Readout Units” for protocol adaptation	100 to 1000
Powerful Core routers	2 to 8
Edge switches	50 to 100
Servers for event filtering	> 1000

AACL

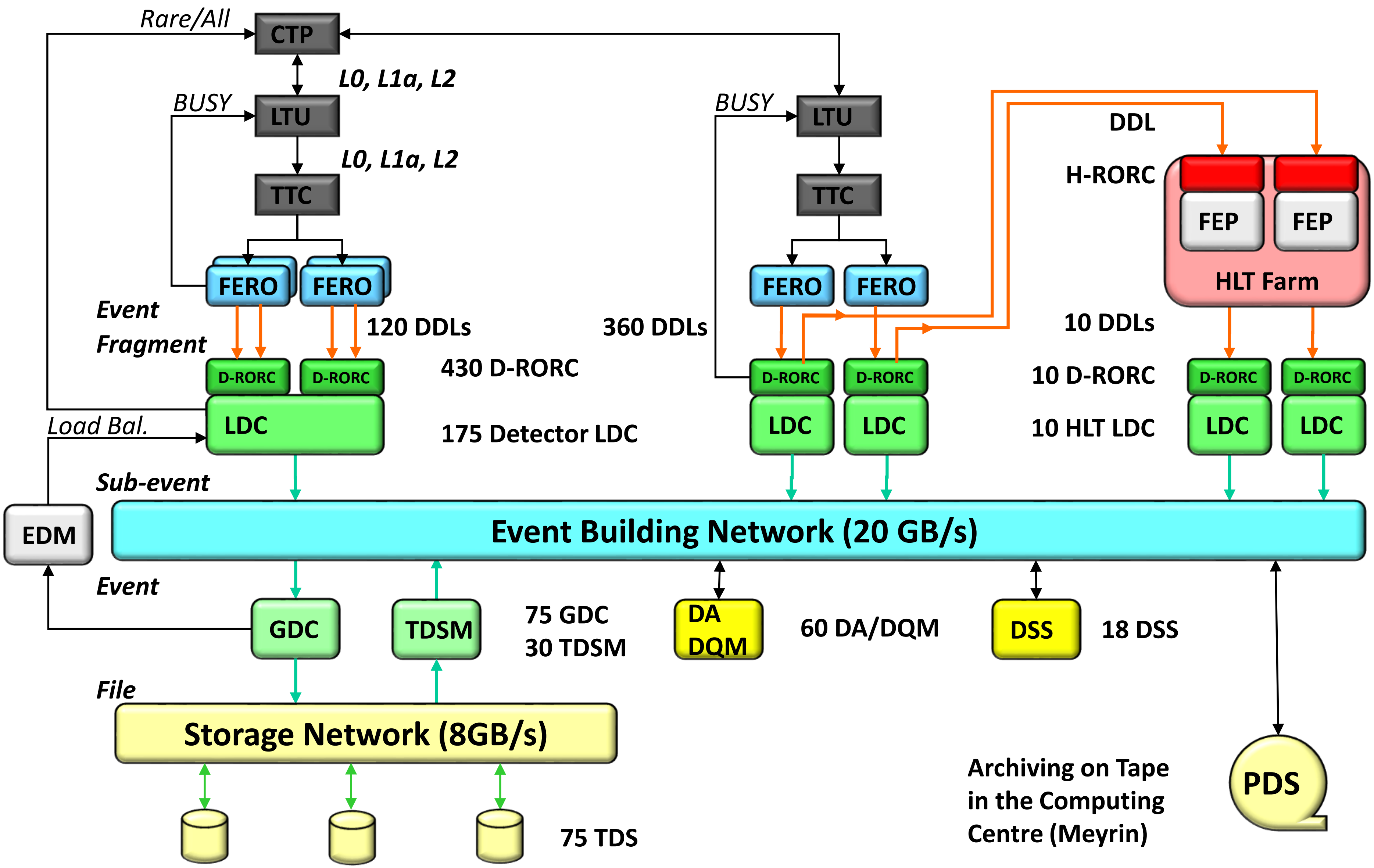
ALICE, ATLAS, CMS, LHCb DAQs

ALICE DAQ

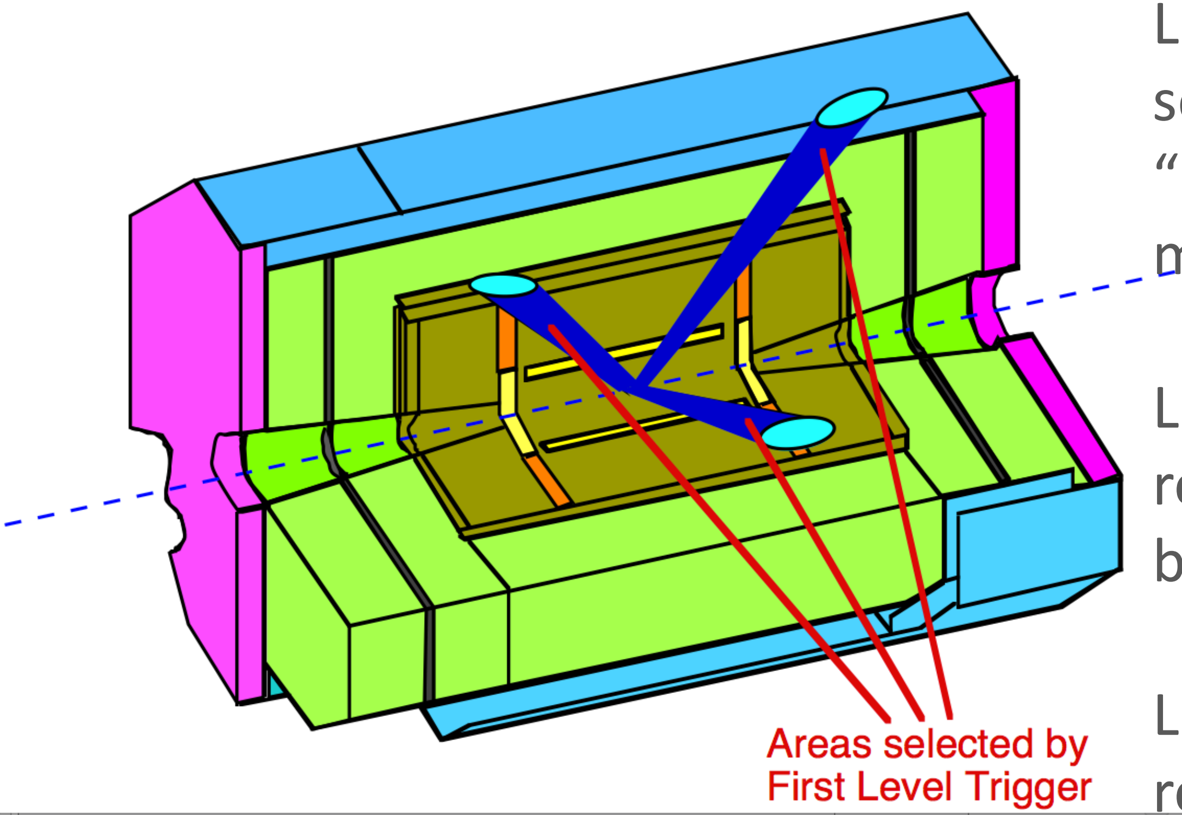
- Large event-size (up to 1 GB for central Pb-Pb collisions)
  - Low (compared to pp) interaction rate → software trigger HLT used for compression rather than rejection
- Design requirements (minimum)
  - Detector readout bandwidth: ~ 50 GB/s
  - DAQ storage bandwidth:
    - pp run: 100 MB/s
    - HI run: 1250 MB/s
- Observed
  - DAQ bandwidth:
    - pp run 2010 : 300-500 MB/s
    - Test in view of the HI run:
      - Bandwidth to local DAQ storage: 4.5 GB/s (!)
      - Bandwidth to central CERN storage (CASTOR): 2.5 GB/s
- **High-level Trigger uses an interesting combination of FPGA pre-processor cards and Nvidia GPUs for track-reconstruction connected by an InfiniBand network**



ALICE Trigger – DAQ – HLT '11



ATLAS: Region of Interest



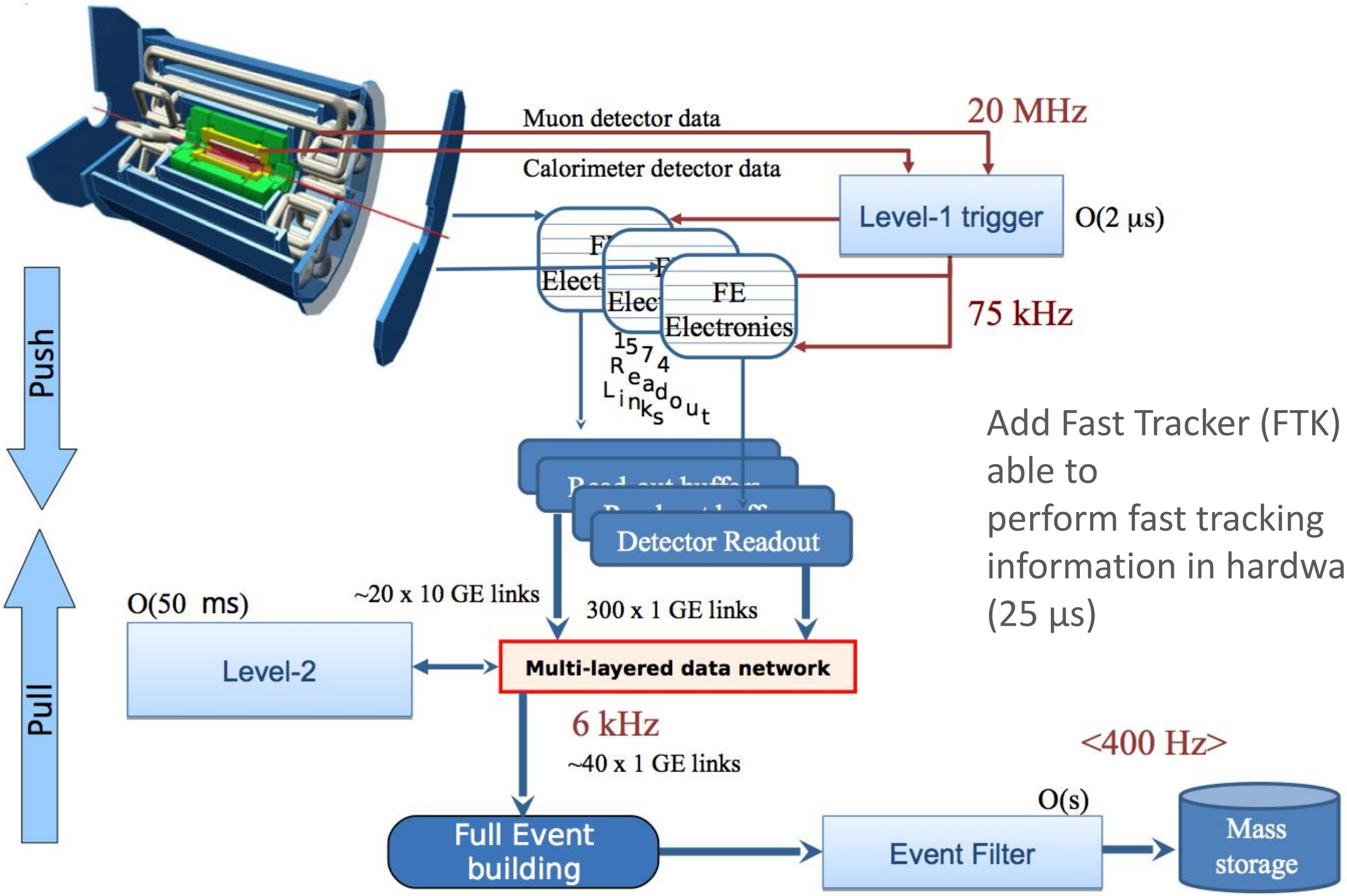
L2 trigger only selects data based on "Regions of Interest" marked by L1

L2 runs at 75 kHz, rejects > 90% of events based on ~10% of data

L3 runs at ~6 kHz, rejects >90% of events based on full reconstruction

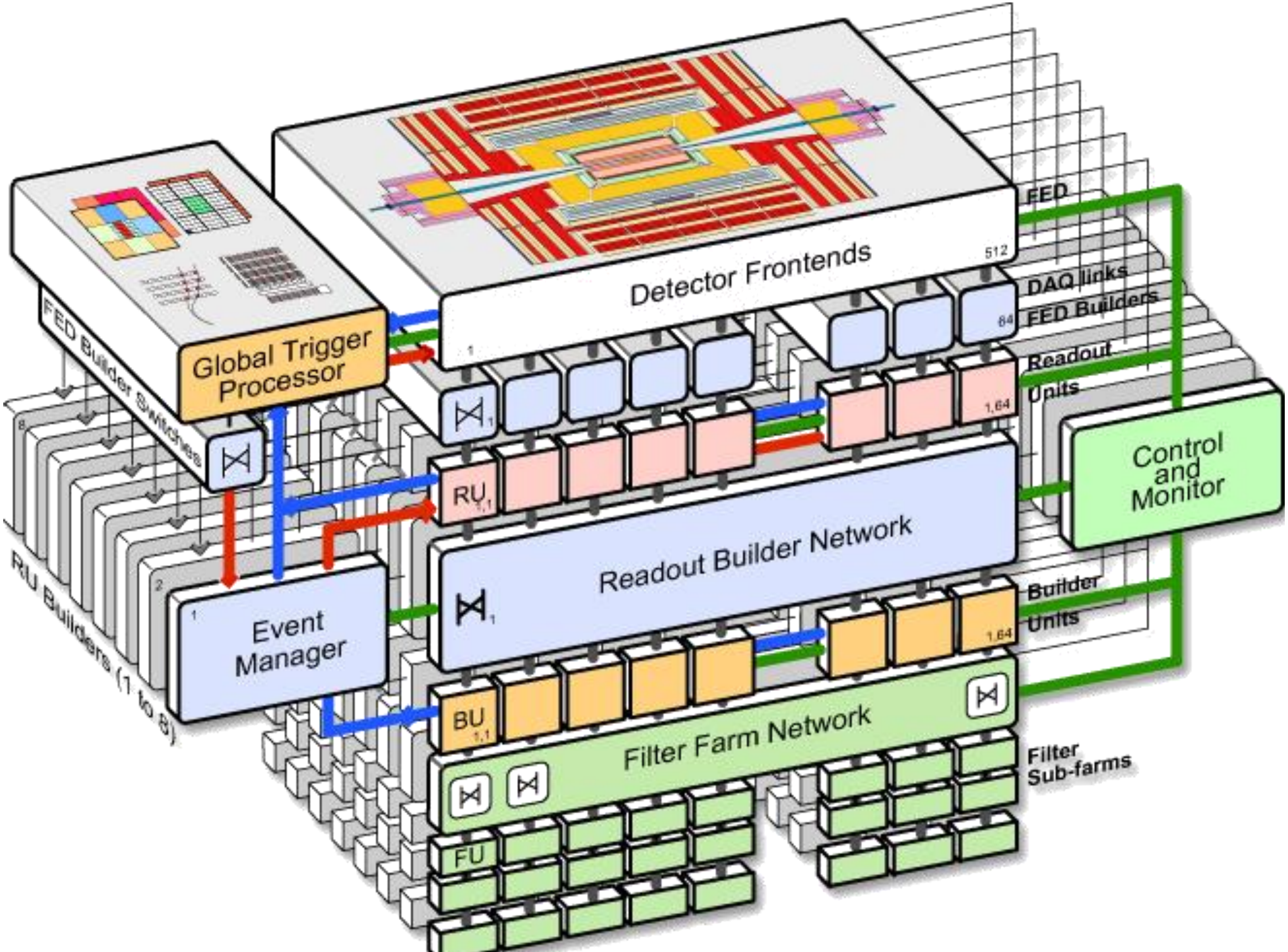
Overall network bandwidth: ~10 GB/s !

ATLAS DAQ



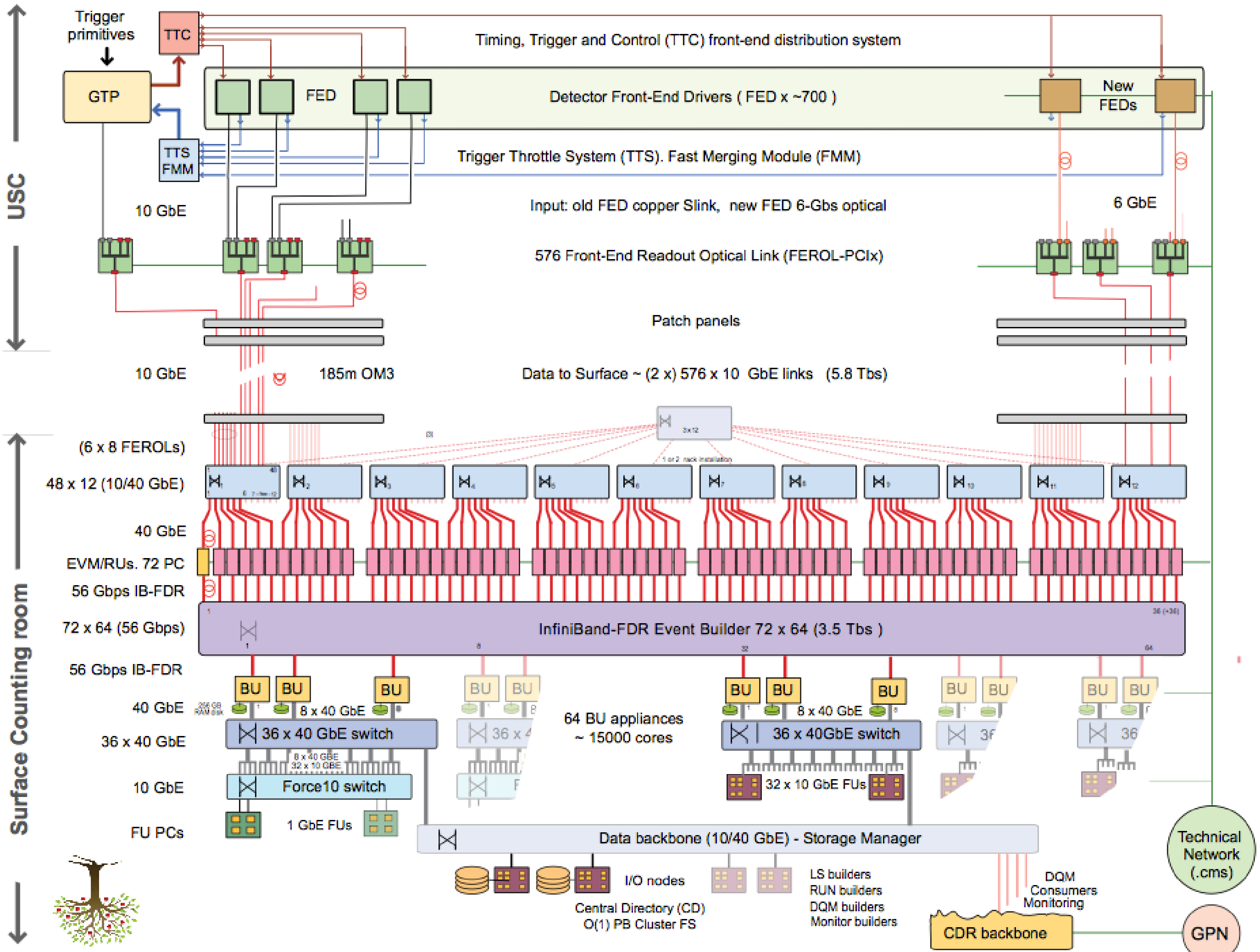
Add Fast Tracker (FTK) to be able to
perform fast tracking
information in hardware
(25 μs)

CMS DAQ



Global Trigger Processor
Detector Frontends
512
FED
DAQ links
64
FED Builders
FED Builder Switches
Readout Units
1,64
RU 1,1
Readout Builder Network
Control and Monitor
Event Manager
RU Builders (1 to 8)
Builder Units
BU 1,1
Filter Farm Network
Filter Sub-farms
FU

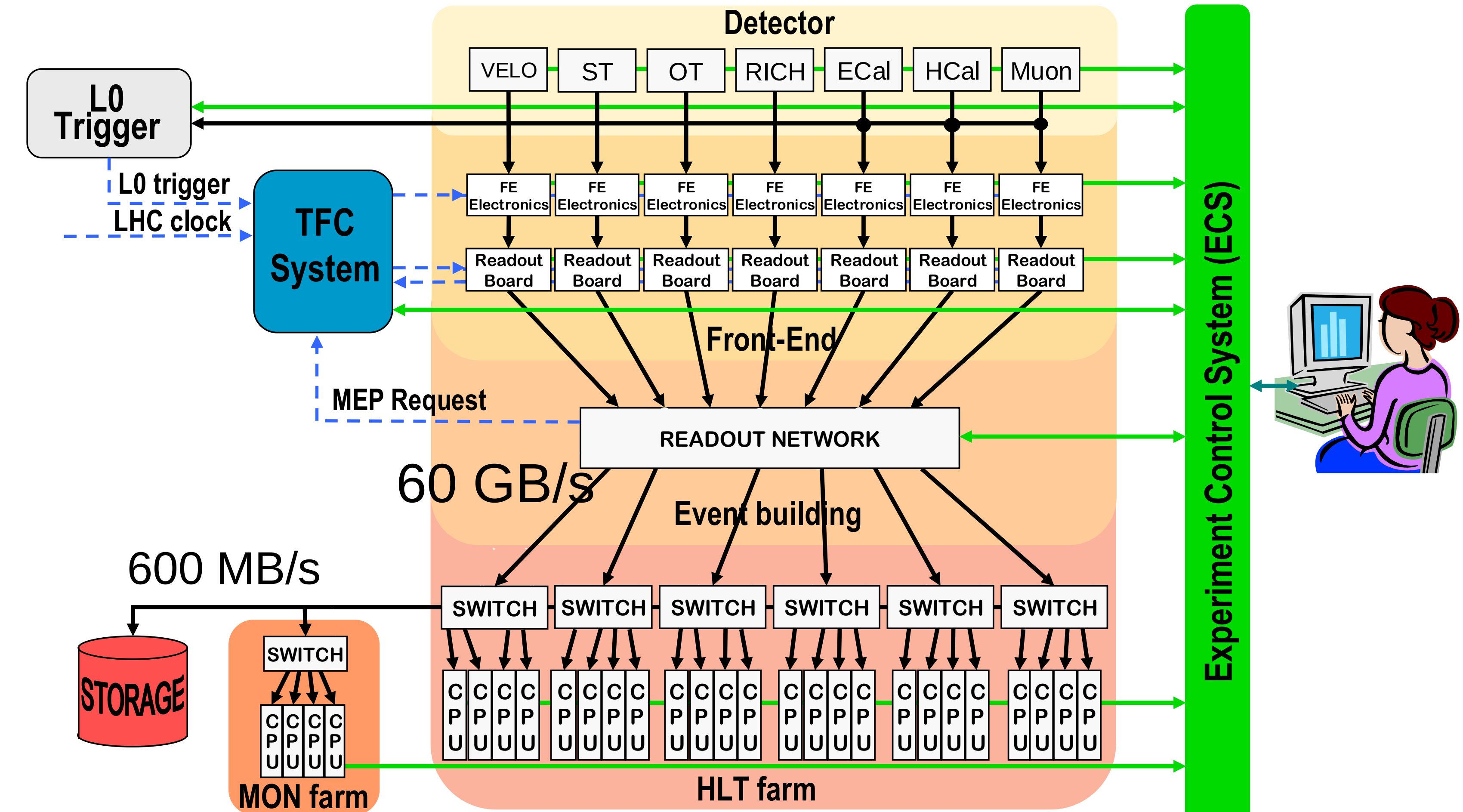
CMS DAQ Run 2



CMS DAQ:
Single event builder InfiniBand Clos network (200 GB/s)

DAQ and HLT decoupled via intermediate shared temporary storage!

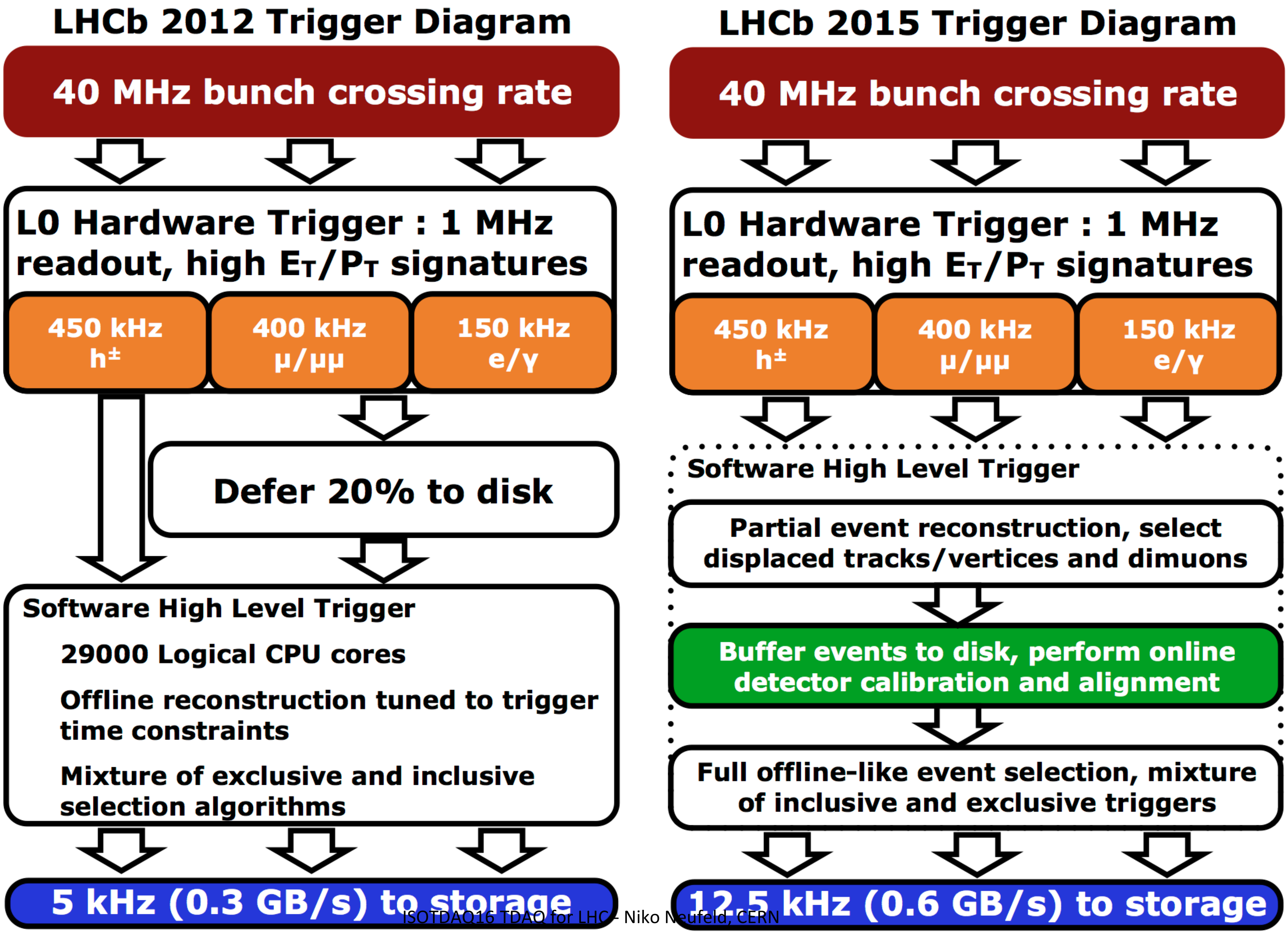
LHCb DAQ



Event data
Timing and Fast Control Signals
Control and Monitoring data

Average event size 60 kB
Average rate into farm 1 MHz
Average rate to tape 12 kHz

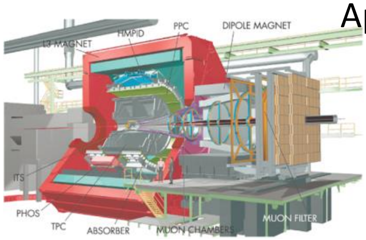
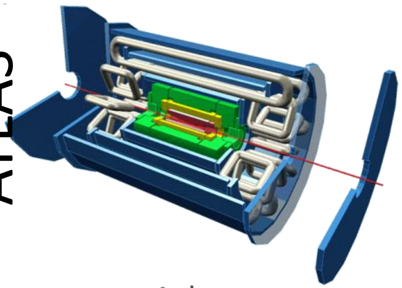
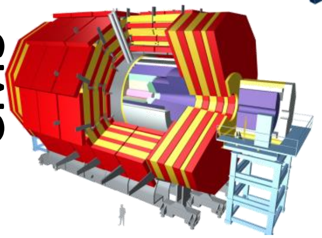
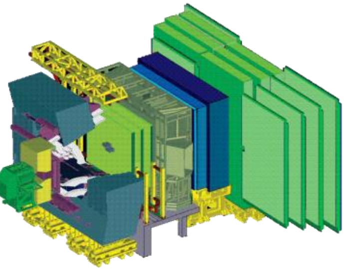
LHCb TDAQ


LHCb 2012 Trigger Diagram
40 MHz bunch crossing rate
L0 Hardware Trigger : 1 MHz readout, high E_T/P_T signatures
450 kHz h±
400 kHz μ/μμ
150 kHz e/γ
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 (0.3 GB/s) to storage
LHCb 2015 Trigger Diagram
40 MHz bunch crossing rate
L0 Hardware Trigger : 1 MHz readout, high E_T/P_T signatures
450 kHz h±
400 kHz μ/μμ
150 kHz e/γ
Software High Level Trigger
Partial event reconstruction, select displaced tracks/vertices and dimuons
Buffer events to disk, perform online detector calibration and alignment
Full offline-like event selection, mixture of inclusive and exclusive triggers
12.5 kHz (0.6 GB/s) to storage

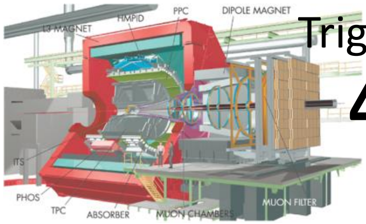
LHCb DAQ

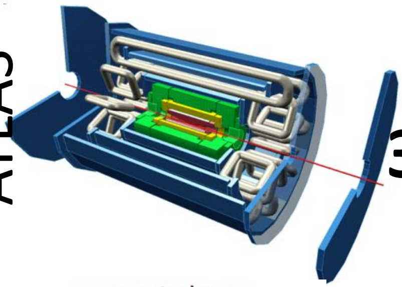
- Events are very small (about 55 kB) total →
  - each read-out board contributes about 200 bytes (only!!)
  - A UDP message on Ethernet takes 8 + 14 + 20 + 8 + 4 = 52 bytes → 25% overhead(!)
- LHCb uses coalescence of messages, packing about 10 to 15 events into one message (called MEP) → message rate is ~ 80 kHz (c.f. CMS, ATLAS)
- Protocol is a simple, single stage push, every farm-node builds complete events, the TTC system is used to assign IP addresses coherently to the read-out boards

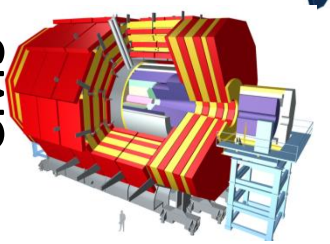
DAQ network parameters

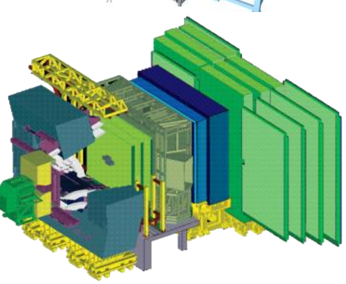
	Link load Approx [%]	Technology	Protocol	Eventbuilding
ALICE 	30%	Ethernet InfiniBand (HLT)	TCP/IP	pull pull (RDMA)
ATLAS 	20% 10 Gbit/s (L2) 50% (Event-collection)	Ethernet	TCP/IP	pull / push
CMS 	50% 40 – 80%	InfiniBand Ethernet	InfiniBand TCP/IP	pull (RDMA) pull
LHCb 	40 - 80%	Ethernet	UDP	push

LHC Trigger/DAQ parameters







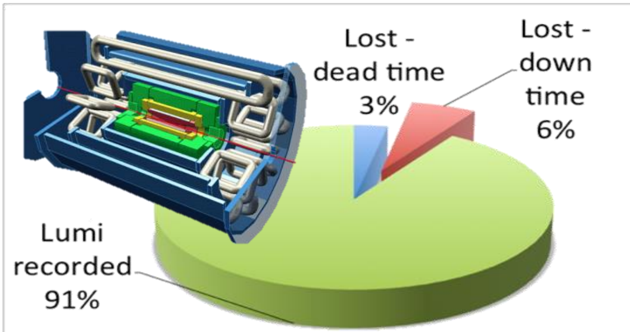


	# Trigger	Level-0,1,2 Rate (kHz)	Event Size (Byte)	Network Bandw.(GB/s)	Storage MB/s (Event/s)
ALICE	**4**	Pb-Pb **5**	**5x10^7**	**25**	**4000** (10^2)
		p-p **1**	**2x10^6**		**200** (10^2)
ATLAS	**3**	LV-1 **75**	**1.5x10^6**	**10**	**~1000** (>400)
		LV-2 **6**			
CMS	**2**	LV-1 **75**	**10^6**	**200**	**~1000** (10^3)
LHCb	**2**	LV-0 **1000**	**6x10^4**	**55**	**>600** (1.2x10^4)

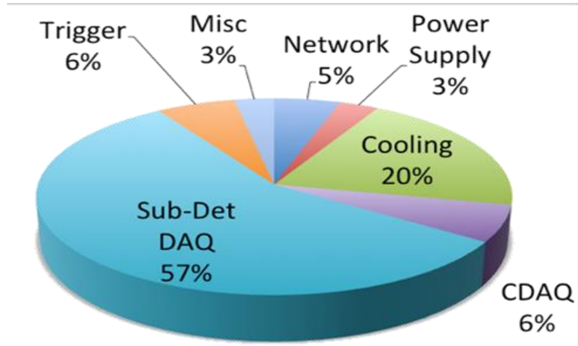
The LHC DAQ systems

- Mostly commercial hardware (Switches, PCs)
- Large numbers
  - → need robust systems, which can cope with hardware failure
  - → time-consuming operation (hardware installation, failures, follow-up, etc…)
- Mostly Ethernet (Gigabit and a little 10 Gigabit), rather static networks, operated at much higher link-load than typical campus LANs

How well do the DAQs dodo?

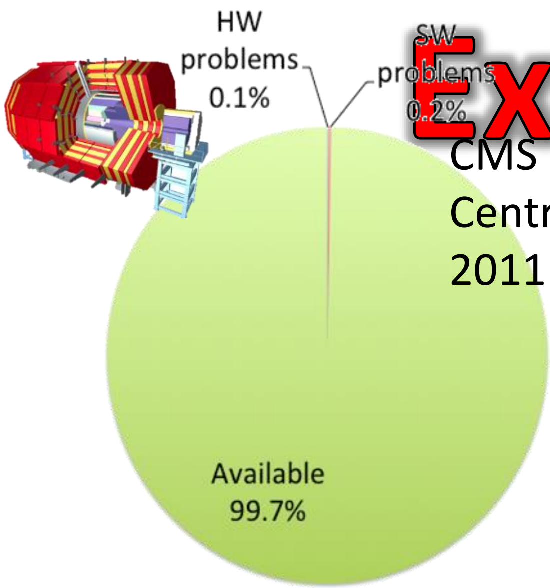


ATLAS central DAQ

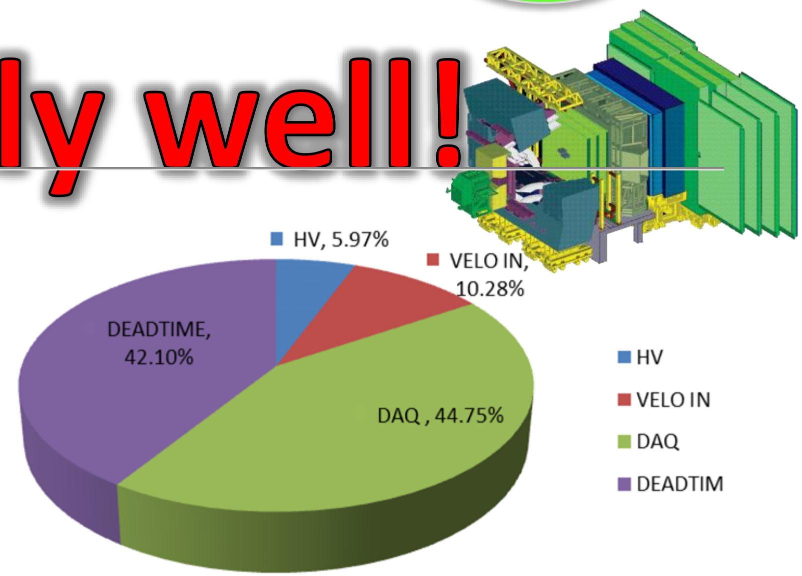


97% DAQ efficiency at the end of HI

Extremely well!



CMS Central DAQ 2011 (pp)



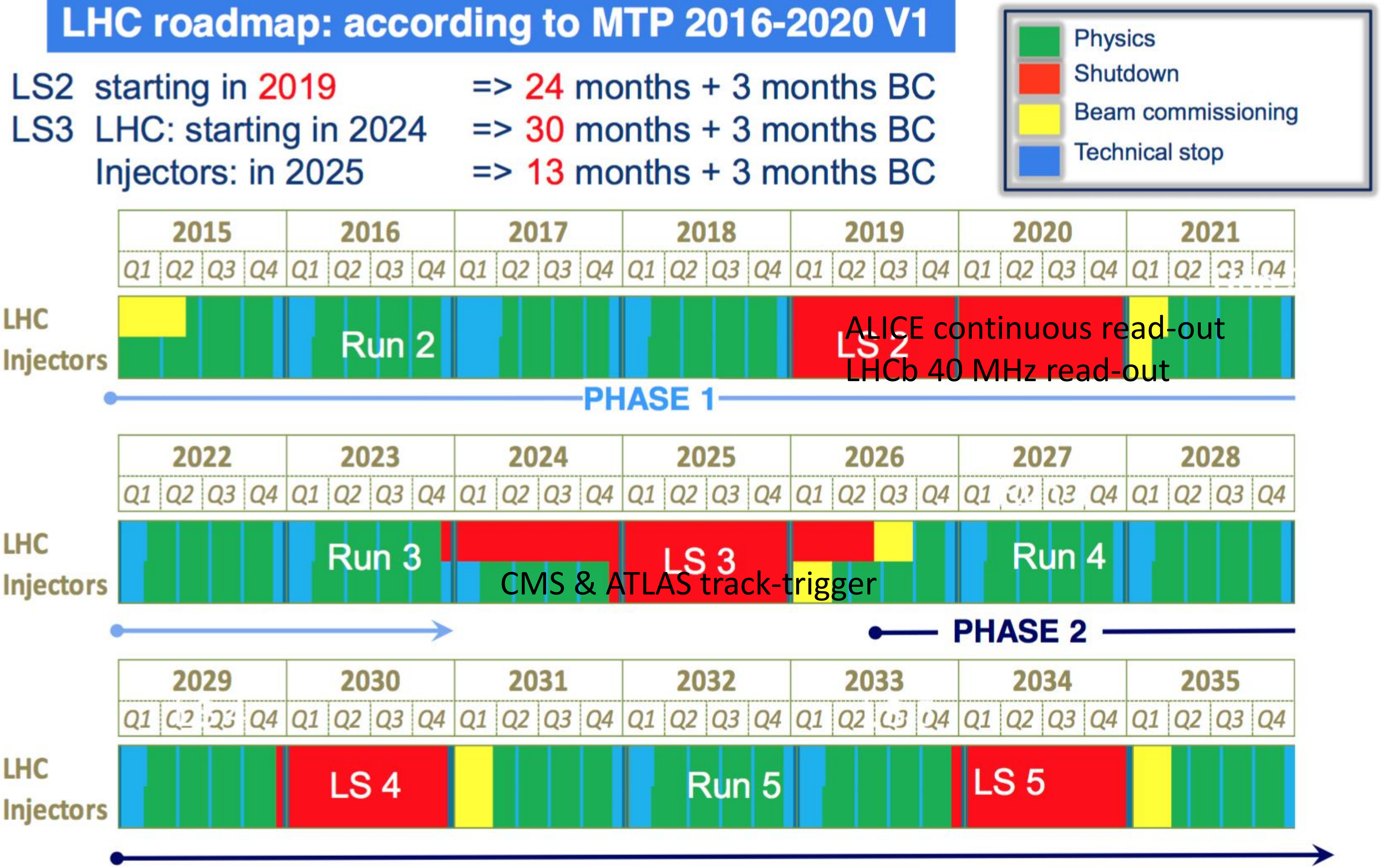
Central DAQ available > 98%

What’s next

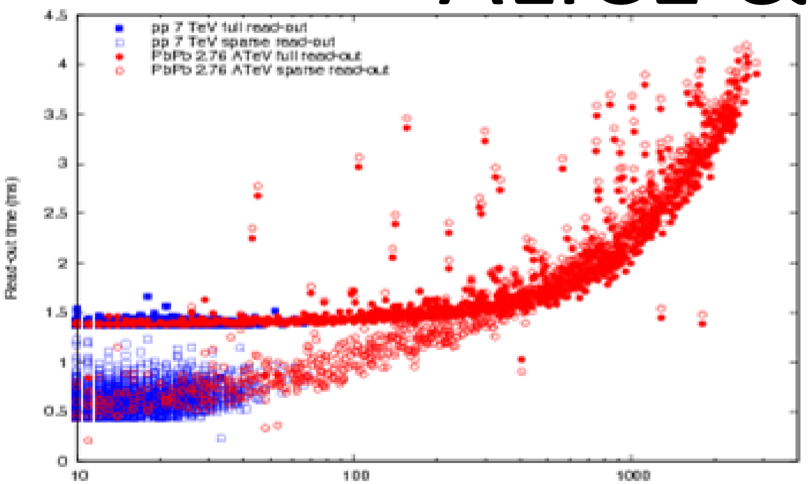
LHC planning

LHC roadmap: according to MTP 2016-2020 V1

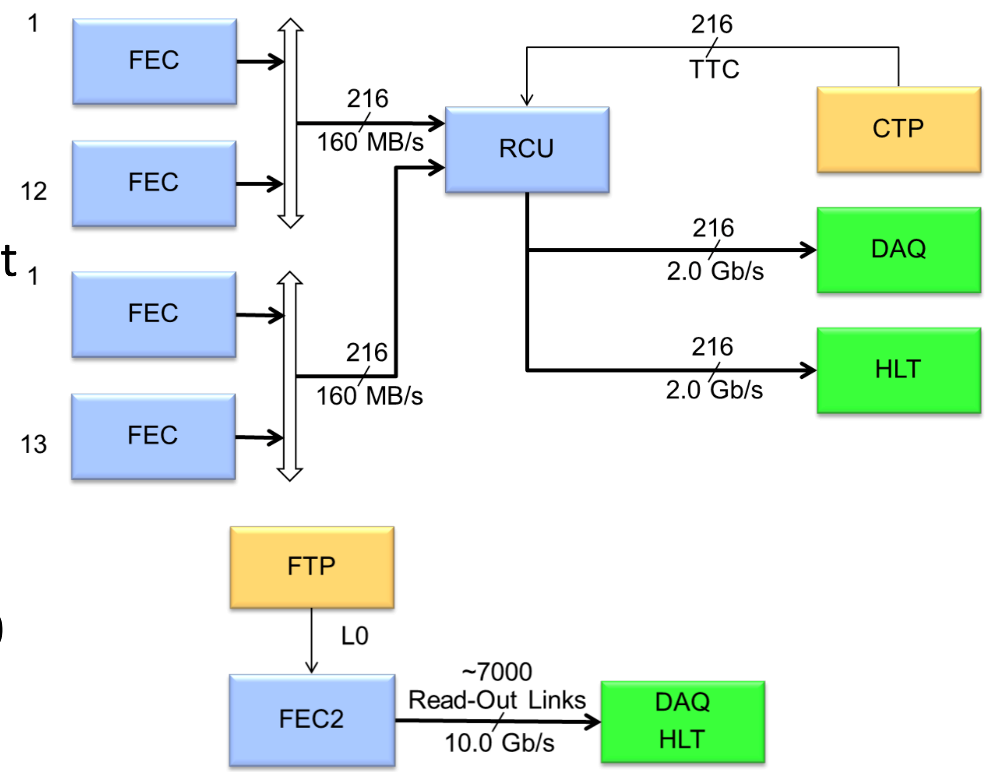
LS2 starting in 2019 => 24 months + 3 months BC
LS3 LHC: starting in 2024 => 30 months + 3 months BC
Injectors: in 2025 => 13 months + 3 months BC



ALICE & LHC after LS2

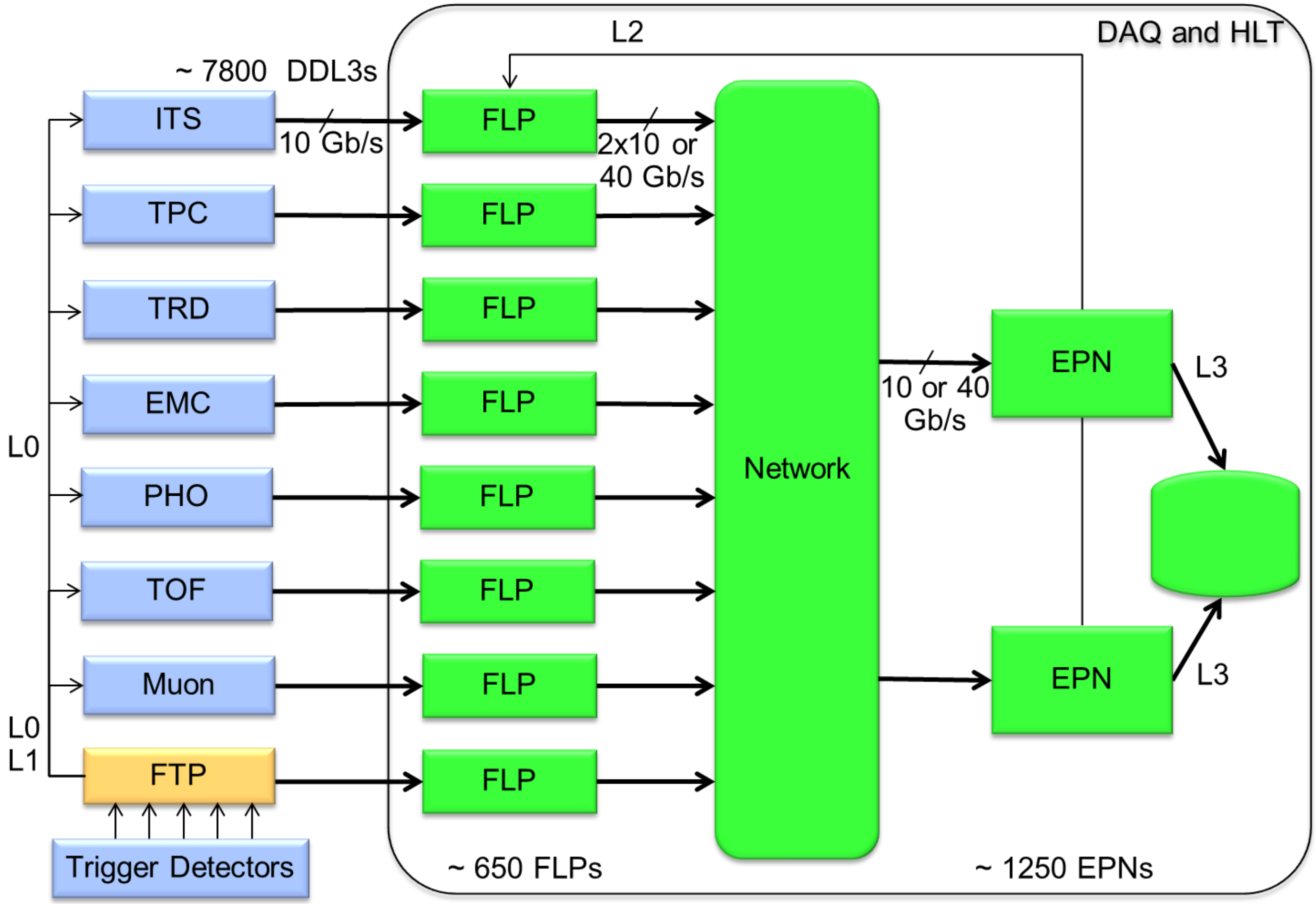


Present and future TPC readout system



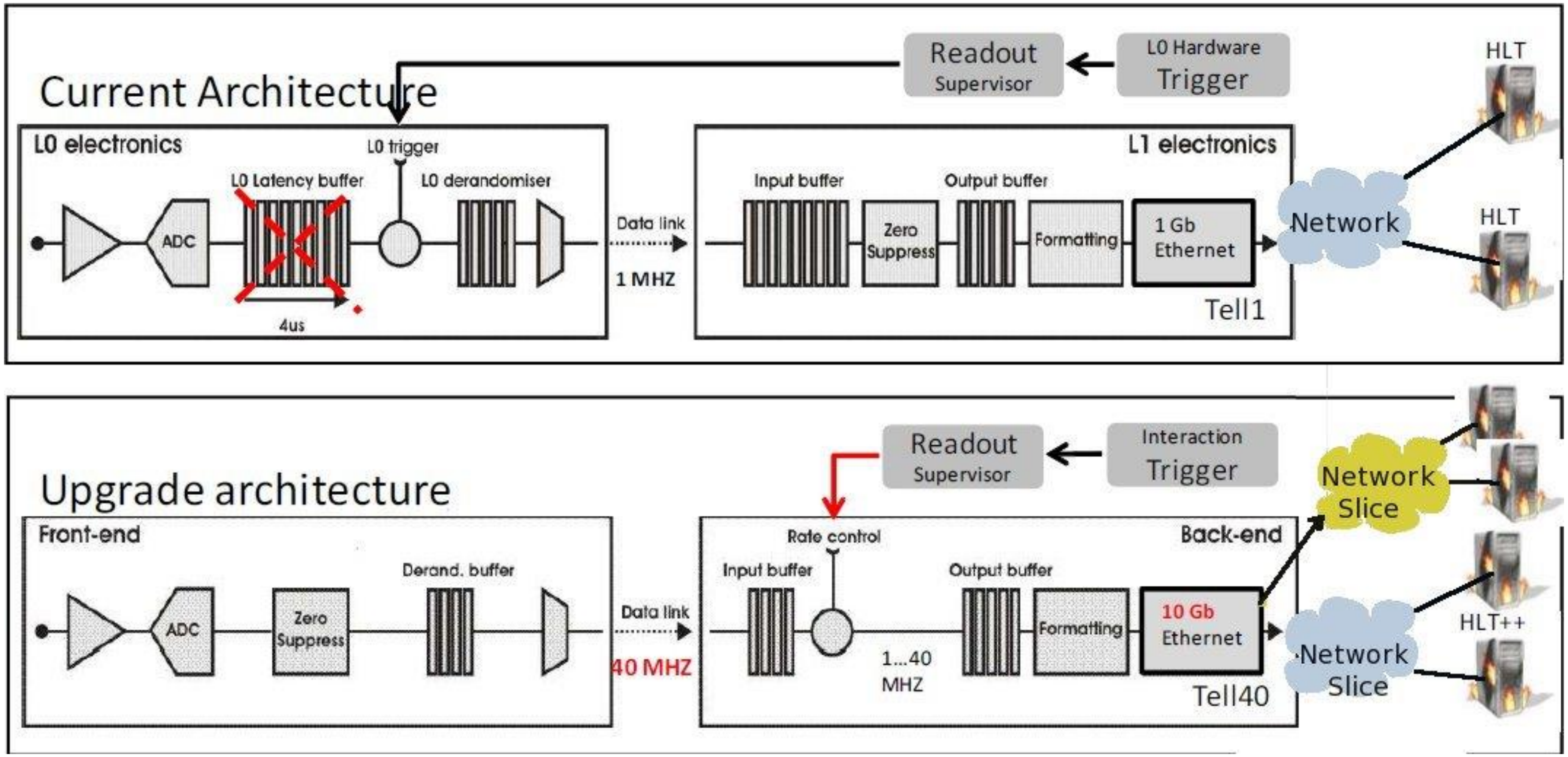
- LHC will deliver Pb beams colliding at an interaction rate of about **50 kHz** and an instantaneous luminosity of L=6x10^{27} cm^{-2}s^{-1}
- ALICE strategy: upgrade detectors and online systems to be able to inspect the **50 kHz** minimum bias interaction rate and accumulate ~10 nb^{-1}
- This means going up from a maximum read-out speed of less than 1 kHz today!
- Main challenge: TPC read-out

ALICE DAQ & HLT after 2018



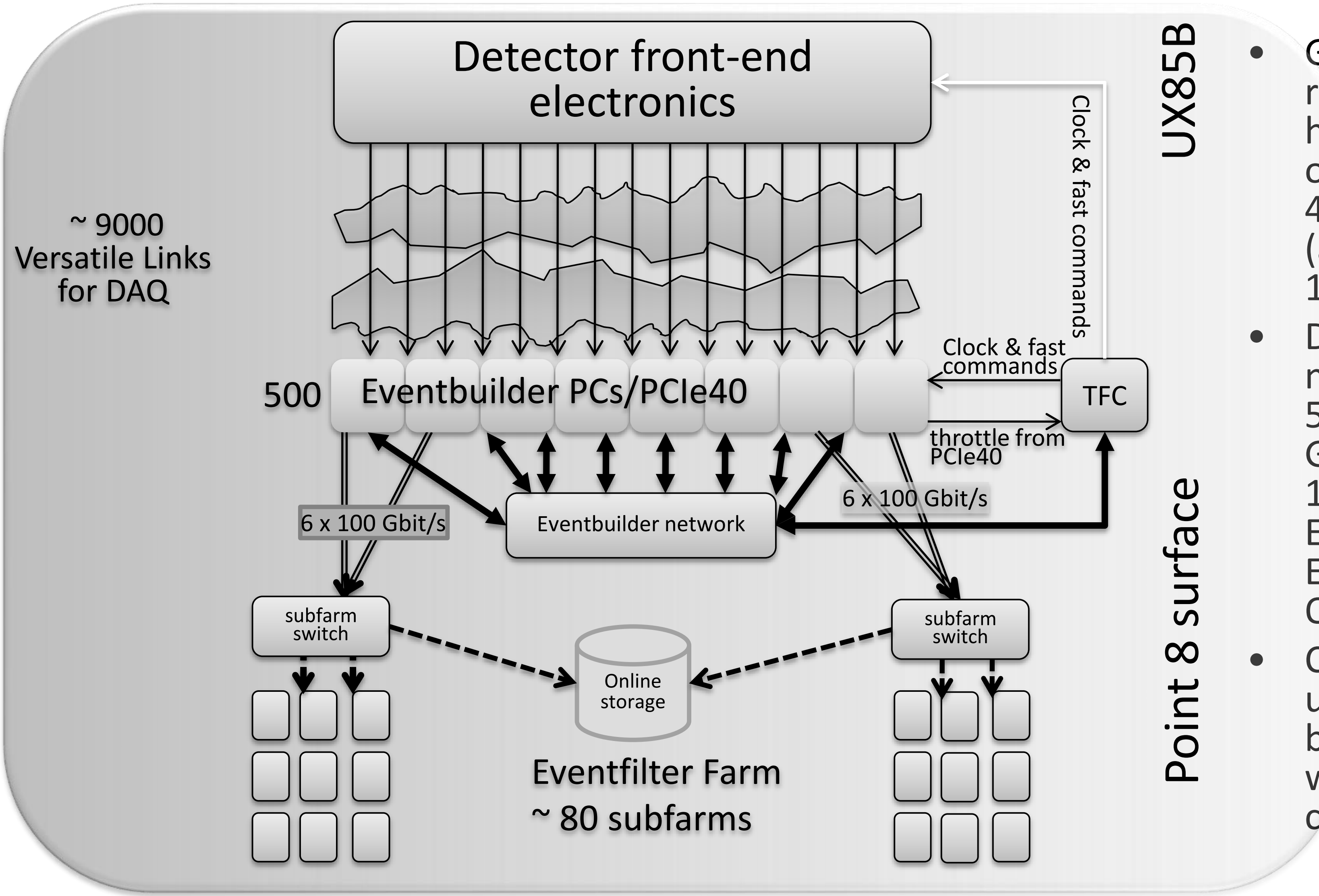
Network options:
InifniBand or (some)
Ethernet

LHCb after LS2



- Ready all software trigger (resources permitting)
- 0-suppression on front-end electronics mandatory!
- Event-size about 100 kB, readout-rate 40 MHz
- Will need a network scalable up to 32 Tbit/s: InfiniBand, 10/40/100 Gigabit Ethernet?

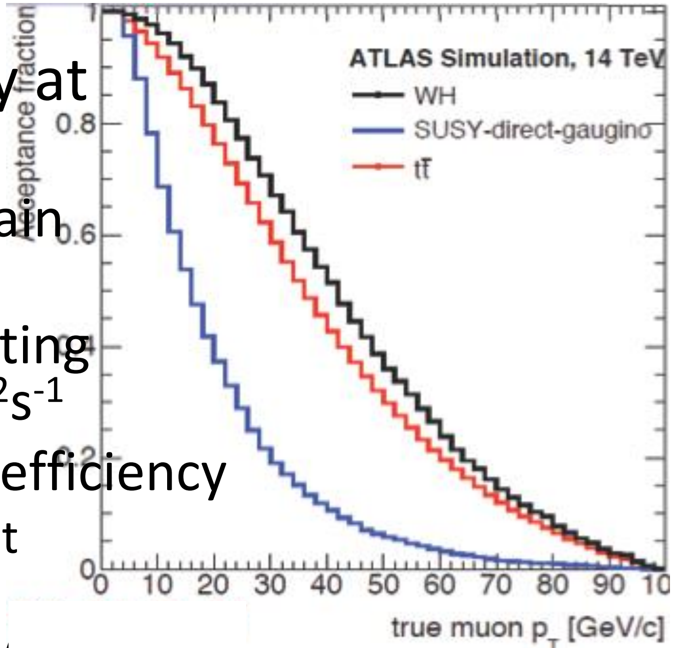
LHCb DAQ for Run3



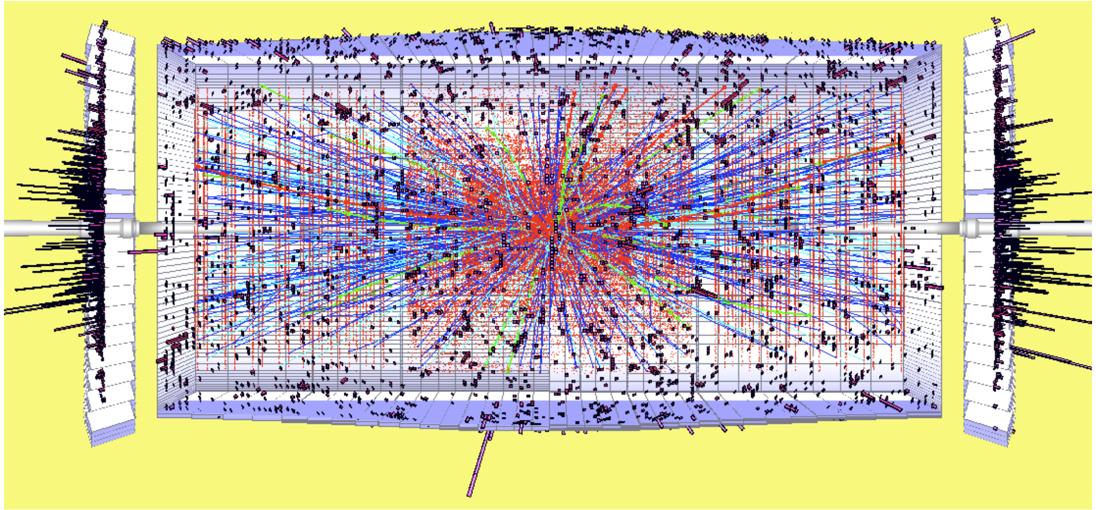
- GBT: radiation-hard link over MMF, 4.8 Gbit/s (about 10000)
- DAQ network 500 x 100 Gbit/s links: 100 Gbit Ethernet / EDR IB, Intel OPA)
- Compute units could be servers with coprocessors

ATLAS & CMS for Run 4

- Maintaining current physics sensitivity at HL-LHC challenging for trigger
  - EWK, top (and Higgs) scale physics remain critical at HL-LHC
  - 100kHz L1 bandwidth cannot fit interesting physics events at 13-14 TeV, $5\times10^{34}cm^{-2}s^{-1}$
  - Increasing p_T thresholds reduces signal efficiency
    - Trigger on lepton daughters from H->ZZ at p_T ~ 10-20 GeV
    - Thresholds risk to increase beyond energy scale of interesting processes
  - Backgrounds from HL-LHC pileup reduces the ability to trigger on rare decay products
    - Leptons, photons no longer appear isolated and are lost in QCD backgrounds
    - Increased hadronic activity from pileup impacts jet p_T and MET measurements

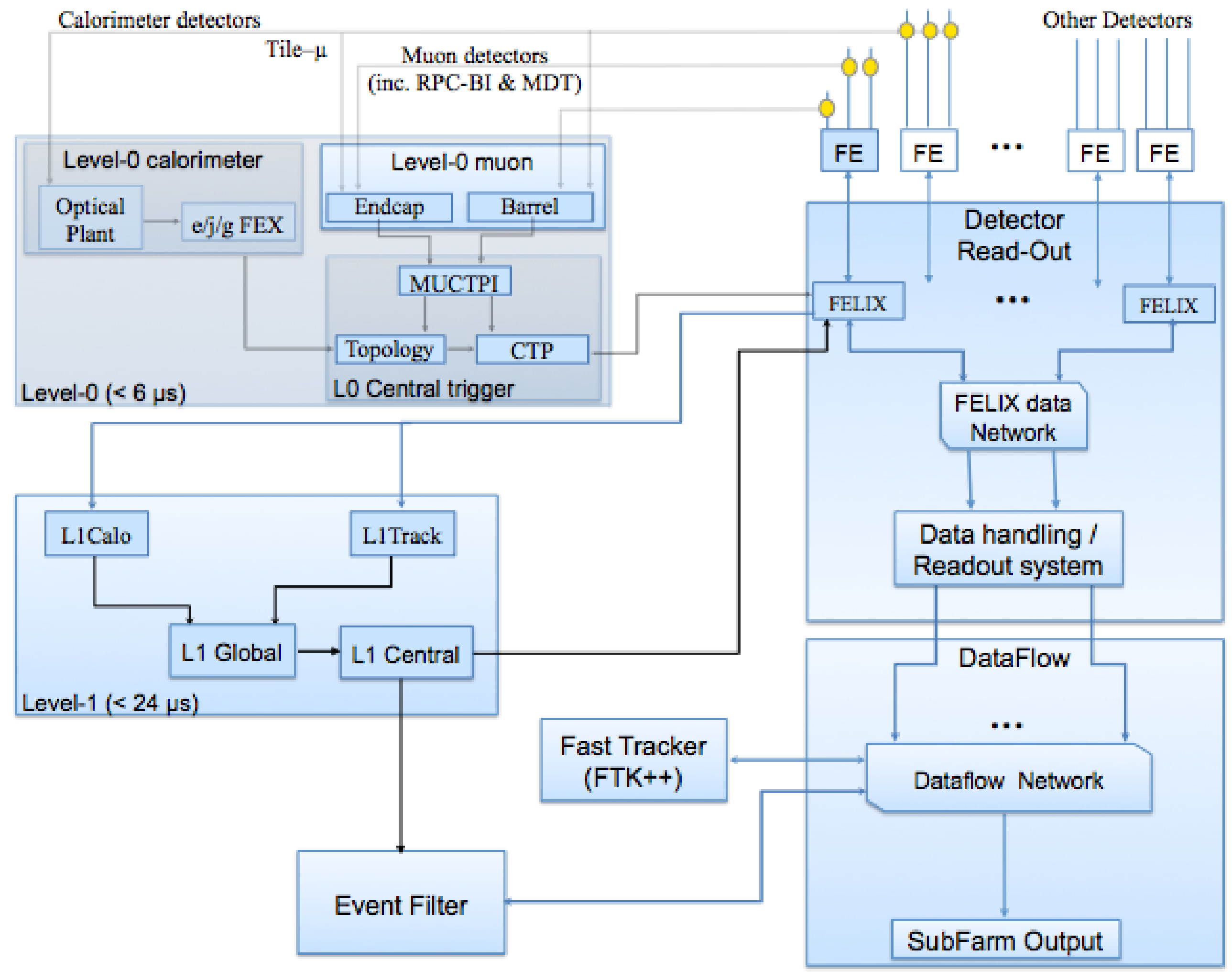


A Track-Trigger at 40 MHz 2020++



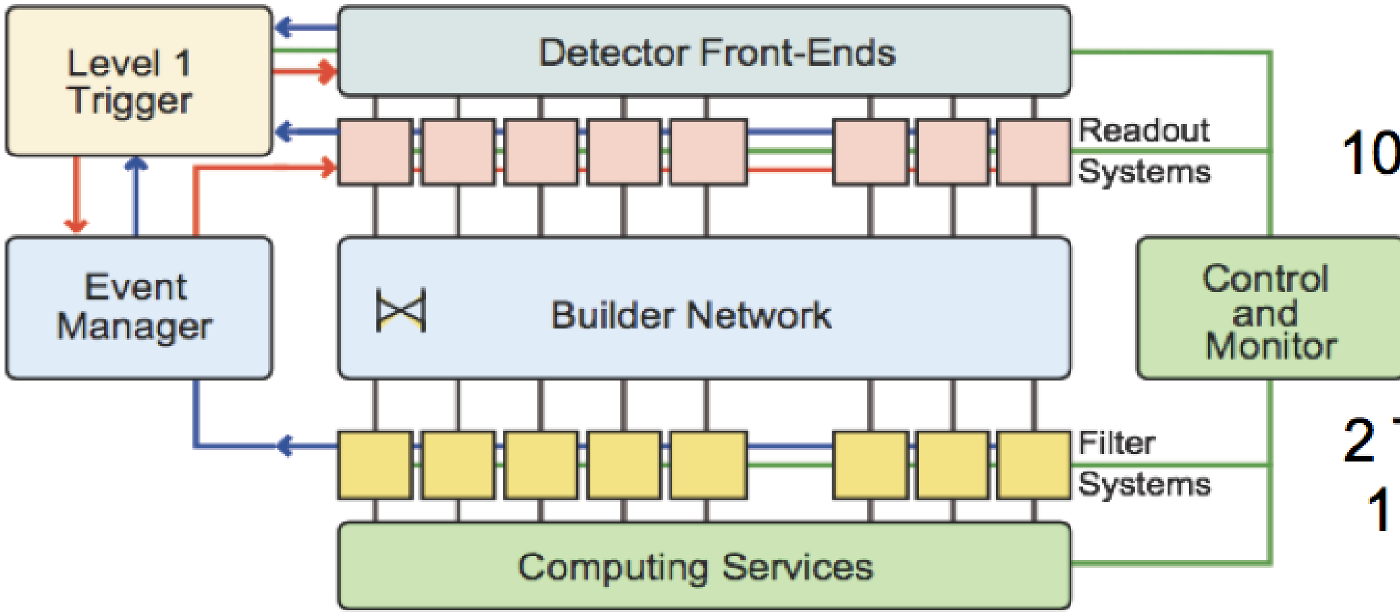
- Goals:
  - Resolve up to 200÷250 collisions per bunch crossing
  - Maintain occupancy at the few % level
  - Maintain overall L1 rate within 100 KHz
  - Keep latency within ~ 6 μs (ECAL pipeline 256 samples = 6.4 μs)
    - The current limit is the Tracker
- L1 tracking trigger data combined with calorimeter & muon trigger data
  - With finer granularity than presently employed.
- Physics objects made from tracking, calorimeter & muon trigger data transmitted to Global Trigger.

ATLAS



- Divide L1 Trigger into L0/L1 of latency 6/30 μsec;
  rate ≤ 1 MHz/400 kHz
- HLT output 5-10 kHz
- L0 uses cal. & μ Triggers, which generate track trigger seeds
- L1 uses Track Trigger and more fine-grained calorimeter trigger information.

CMS



40 MHz
100 kHz → 750 kHz

2 Tbps → ~50 Tbps
1 kHz → ~7.5 kHz

- L1 tracking trigger calculated stand-alone, combined with calorimenter & muon trigger data regionally
- After regional correlation stage, physics objects transmitted to global trigger
- L1 trigger latency = 12.5 μs

Summary

- The LHC trigger and DAQ systems are designed to select events out of up to 40 Million bunch-crossings every second. This is done in two basic steps:
  - Many detector channels are analyzed in 0.5 to 2 microseconds using ASICs and FPGAs looking for high transverse momentum objects
  - Large farms of PCs select from the remaining events. They are connected with very large, high-speed local area networks
- These systems are the biggest – and among the most complex DAQ systems every built, and they work remarkably well with only 0.5 - 2% loss-rate
- In the future more information will be included in the triggers, events will be bigger, rates will be higher, networks and PC-farms will grow in size and speed

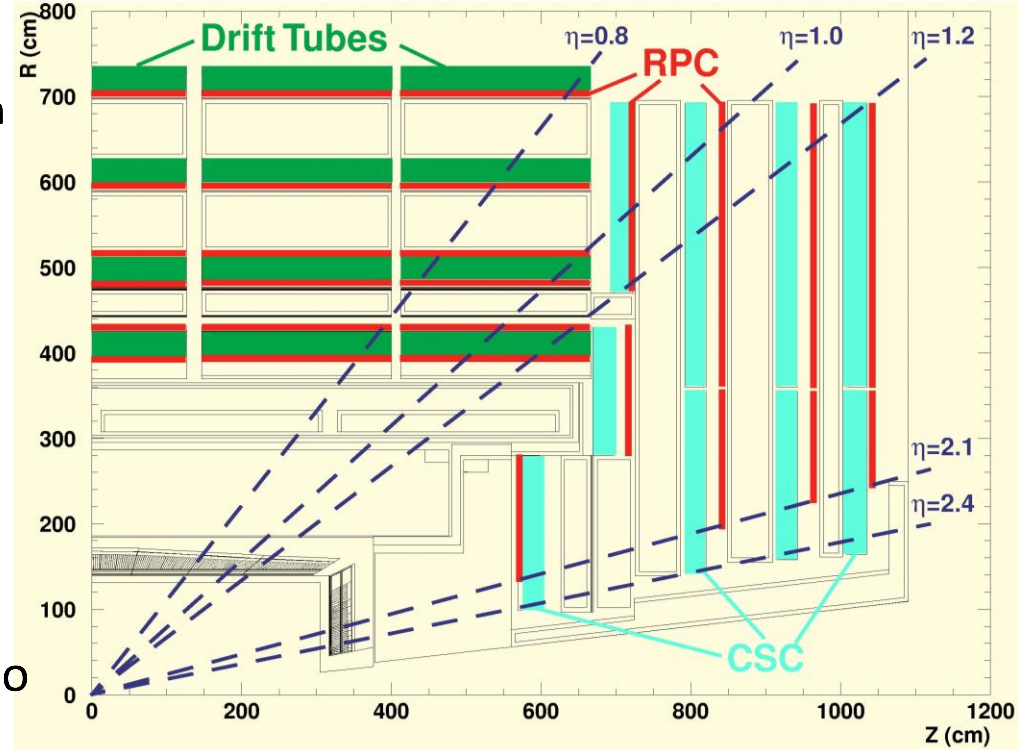
Further Reading

- The Online sessions of the recent CHEP conferences: e.g. J. Phys.: Conf. Ser. (JPCS), Volume 664, 2015 http://iopscience.iop.org/1742-6596/664
- The IEEE RealTime conference series
- DAQ@LHC https://indico.cern.ch/event/217480/ (2013) and (upcoming) https://indico.cern.ch/event/471309/other-view?view=standard (April 2016)

More Stuff

CMS muon system

- CMS muon system includes three detector technologies
  - RPC and DT in barrel
  - RPC and CSC in end-caps
- All three detector systems participate in the LVL1 trigger
  - Specific logic for each system
  - Global logic that combines all the muon information
- Will focus on information from four DT muon stations (see figure)
- Some of the detectors used in the triggers have a response time below 25 ns (e.g. RPCs)
- For slower detectors, information from several chamber layers has to be combined to identify locally which bunch crossing gave rise to the hits, as well as giving the position of the muon in the chambers



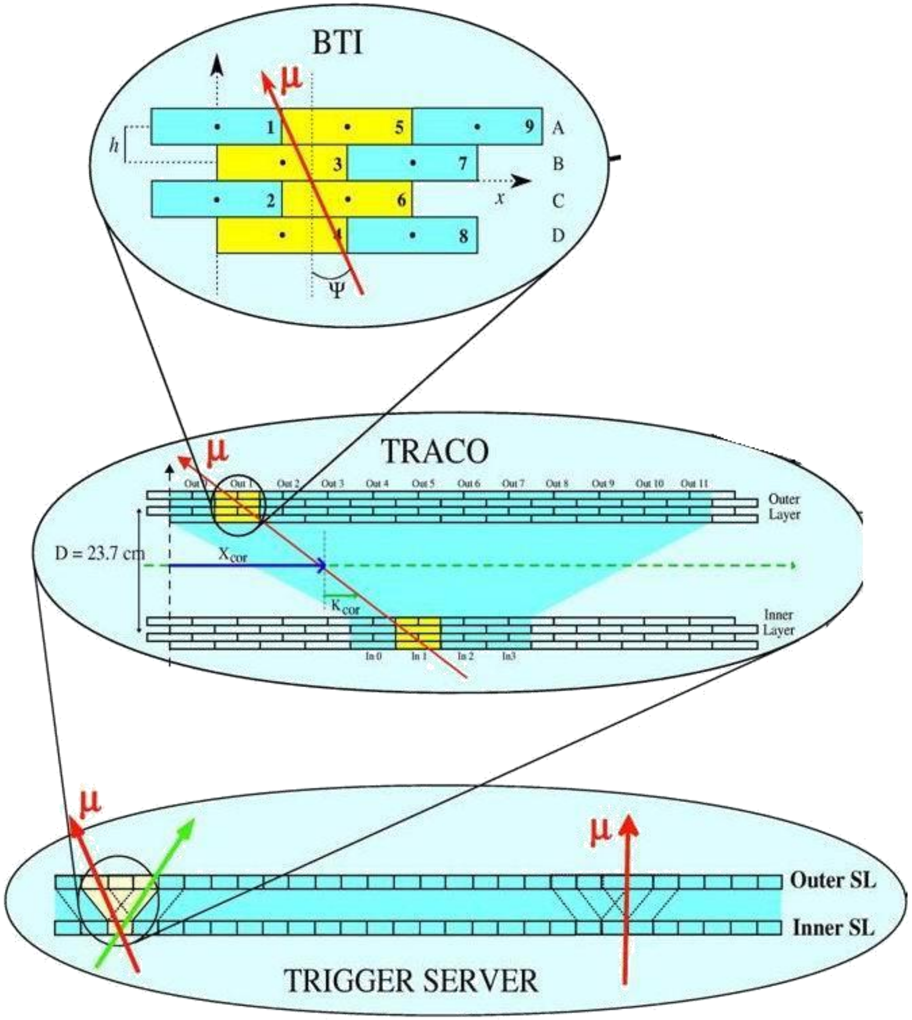
Maximum drift time 380 ns >> 25 ns

CMS local Drift Tube muon trigger

Bunch & **T**ime **I**dentification

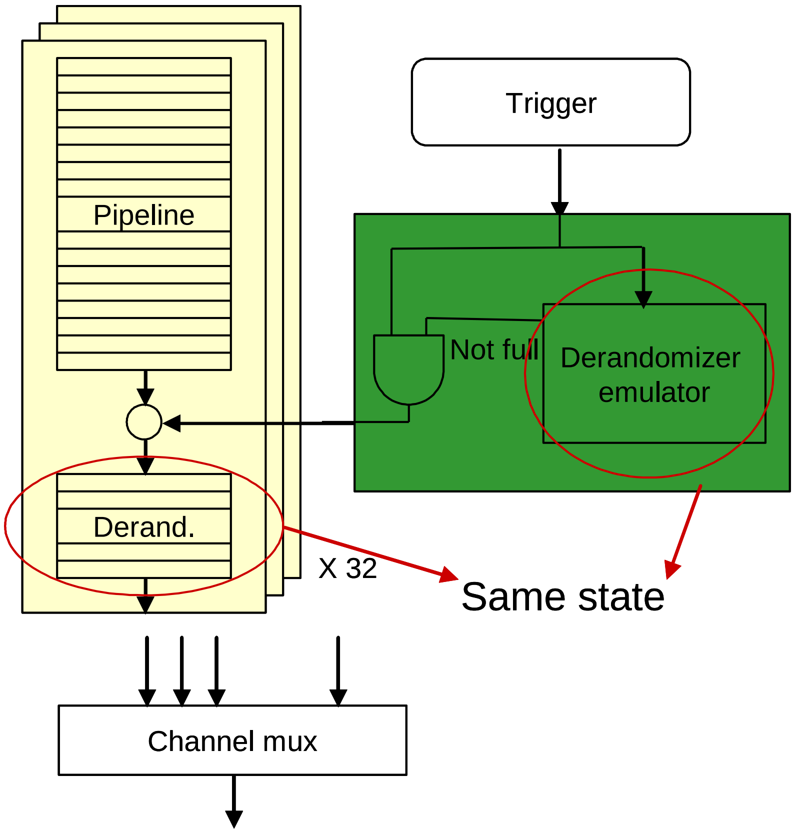
TRAck **CO**rrelator

Local trigger electronics associated with each Super Layer is mounted on the detector and implemented using ASICs



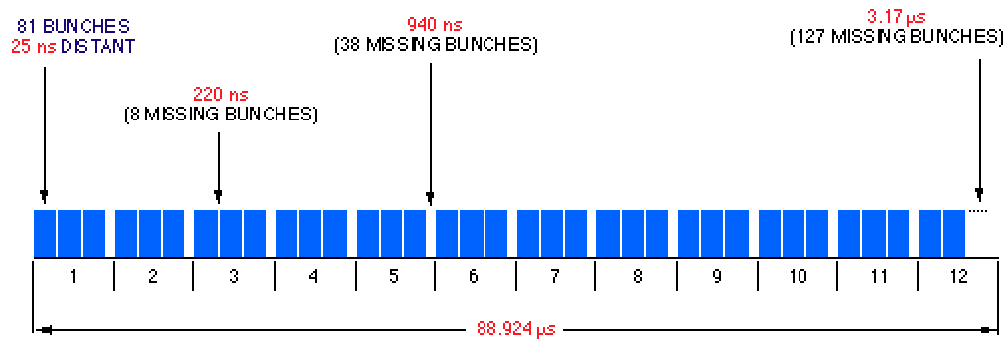
Trigger rate control

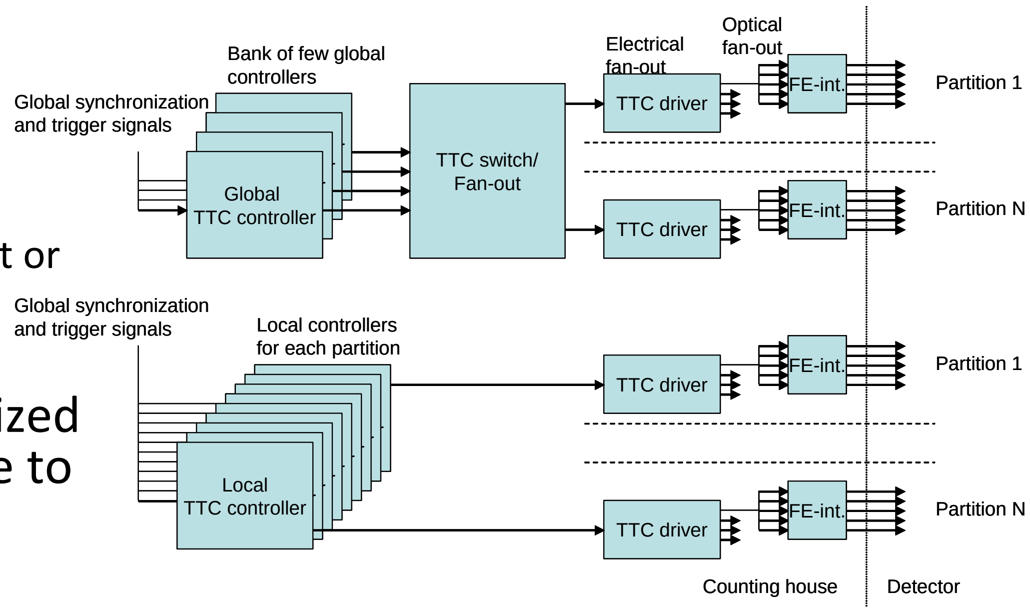
- Trigger rate determined by physics parameters used in trigger system:
  1 kHz – 1MHz for LHC experiments
  - The lower rate after the trigger allows sharing resources across channels (e.g. ADC and readout links)
- Triggers will be of random nature i.e. follow a Poisson distribution → a burst of triggers can occur within a short time window so some kind of rate control/spacing is needed
  - Minimum spacing between trigger accepts → dead-time
  - Maximum number of triggers within a given time window
- Derandomizer buffers needed in front-ends to handle this
  - Size and readout speed of this determines effective trigger rate



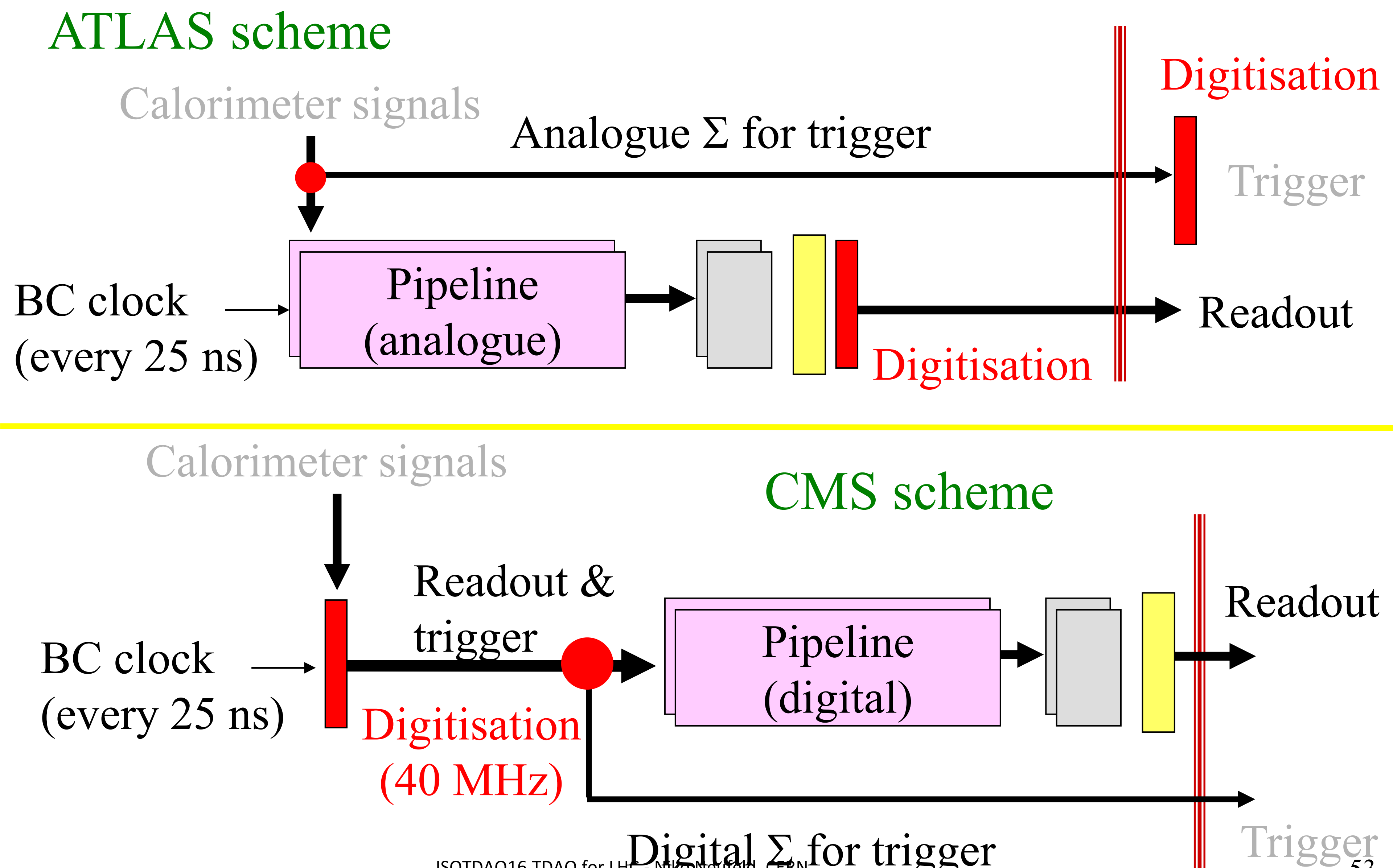
Timing & sync control

- Sampling clock with low jitter
- Synch reset
- Synchronization with machine bunch structure
- Calibration
- Trigger (with event type)
- Time align all the different sub-detectors and channels
  - Programmable delays
- Fan-out – unidirectional
  - Global fan-out to whole experiment or
  - Sub-detector fan-out
- Must be reliable as system otherwise may get de-synchronized which may take quite some time to correct

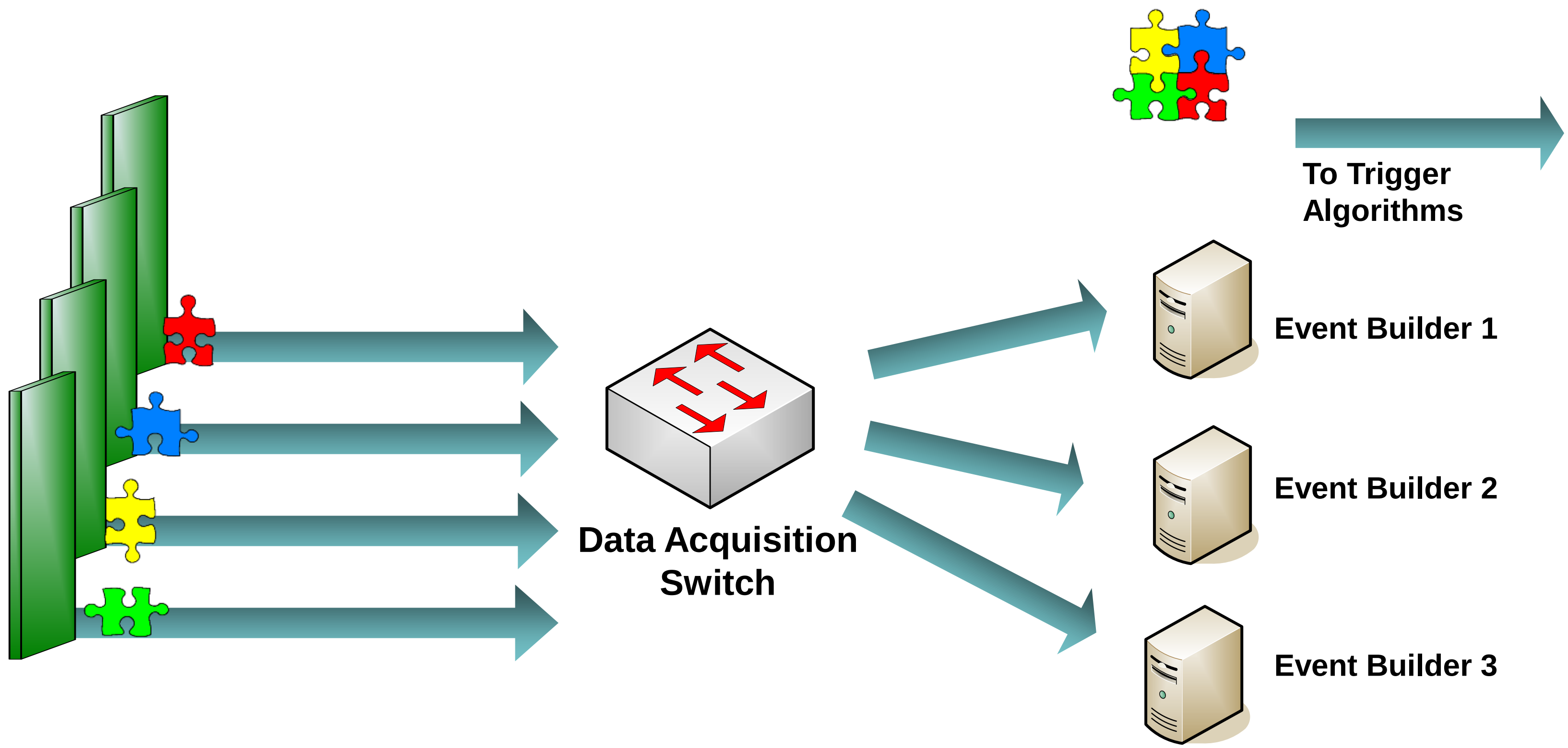




Digitisation options (ATLAS c.f. CMS)



Event Building over a LAN



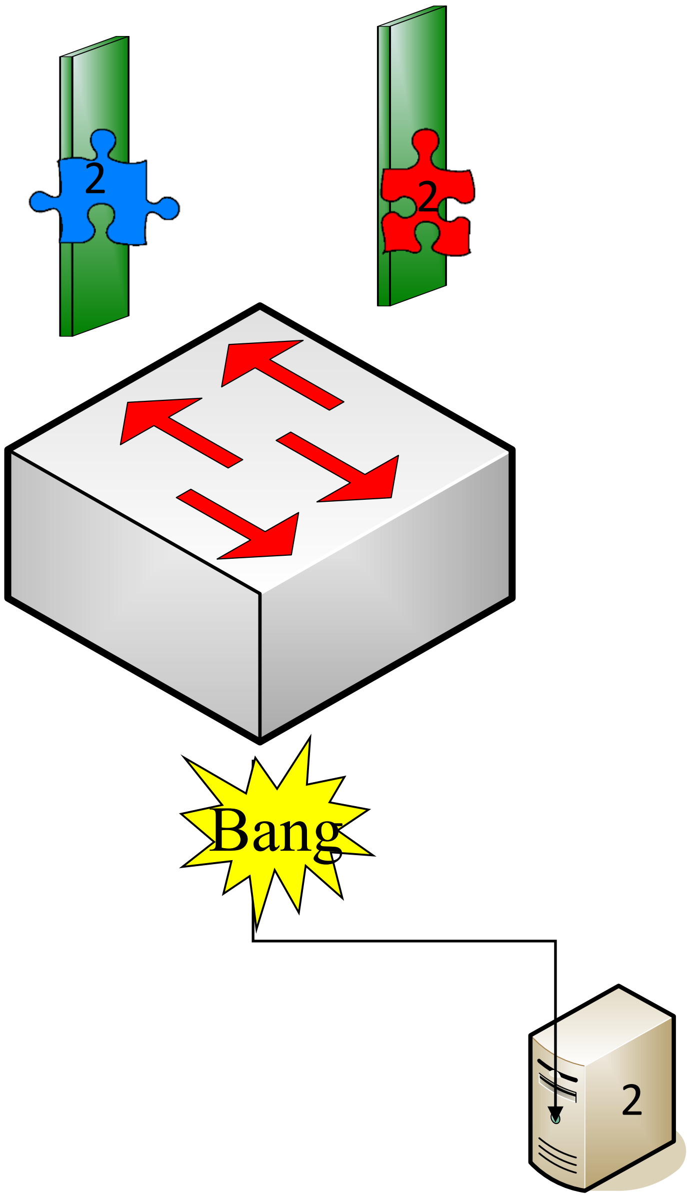
1 Event fragments are received from detector front-end

2 Event fragments are read out over a network to an event builder

3 Event builder assembles fragments into a complete event

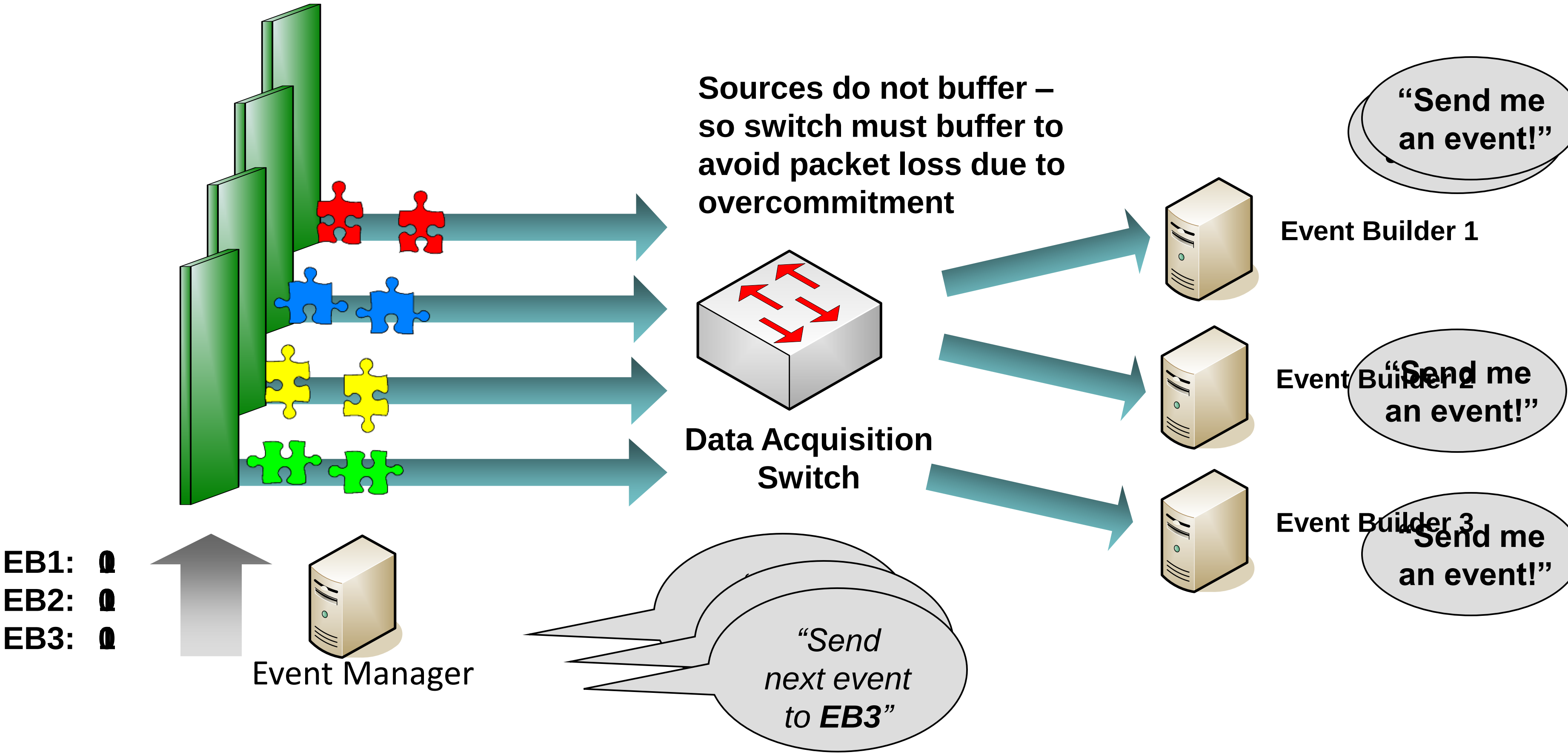
4 Complete events are processed by trigger algorithms

Congestion



- "Bang" translates into random, uncontrolled packet-loss
- In Ethernet this is perfectly valid behavior and implemented by many low-latency devices
- This problem comes from ***synchronized*** sources ***sending*** to the same destination at the ***same time***
- Either a higher level "event-building" protocol avoids this congestion or the switches must avoid packet loss *with deep buffer memories*

Push-Based Event Building with store& forward switching and load-balancing

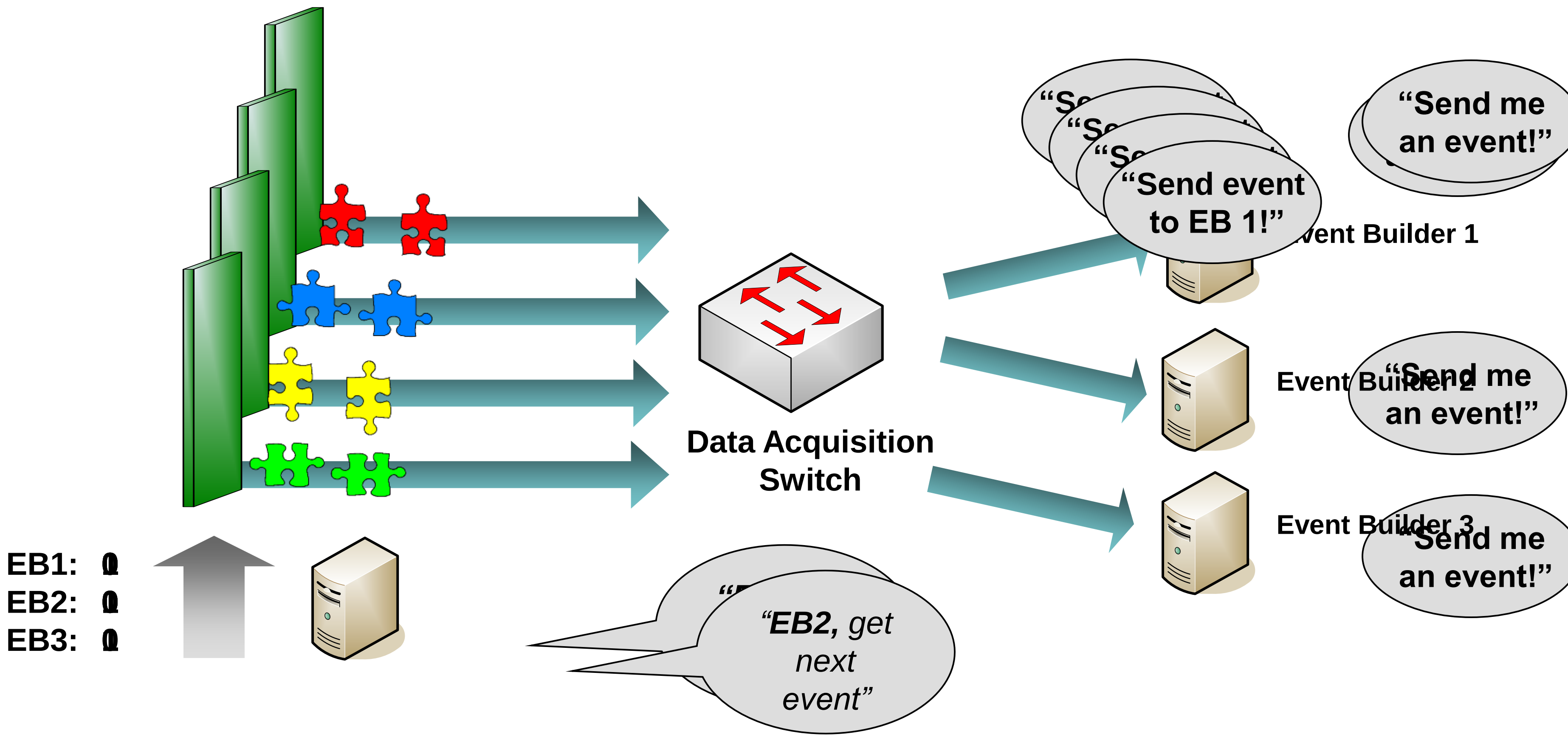


1 Event Builders notify Event Manager available capacity

2 Event Manager ensures that data are sent only to nodes with available capacity

3 Readout system relies on feedback from Event Builders

Pull-Based Event Building



1 **Event Builders notify Event Manager of available capacity**

2 **Event Manager elects event-builder node**

3 **Readout traffic is driven by Event Builders**

Using more of that bandwidth

- Cut-through switching is excellent for low-latency (no buffering) and reduces cost (no buffer memories), but "wastes" bandwidth
  - It's like building more roads than required just so that everybody can go whenever they want immediately
- For optimal usage of installed bandwidth there are in general two strategies:
  - Use store-and-forward switching (next slide)
  - Use traffic-shaping / traffic-control
    - Different protocols ("pull-based event-building"), multi-level readout
    - end-to-end flow control
    - virtual circuits (with credit-scheme) (InfiniBand)
    - Barrel-shifter

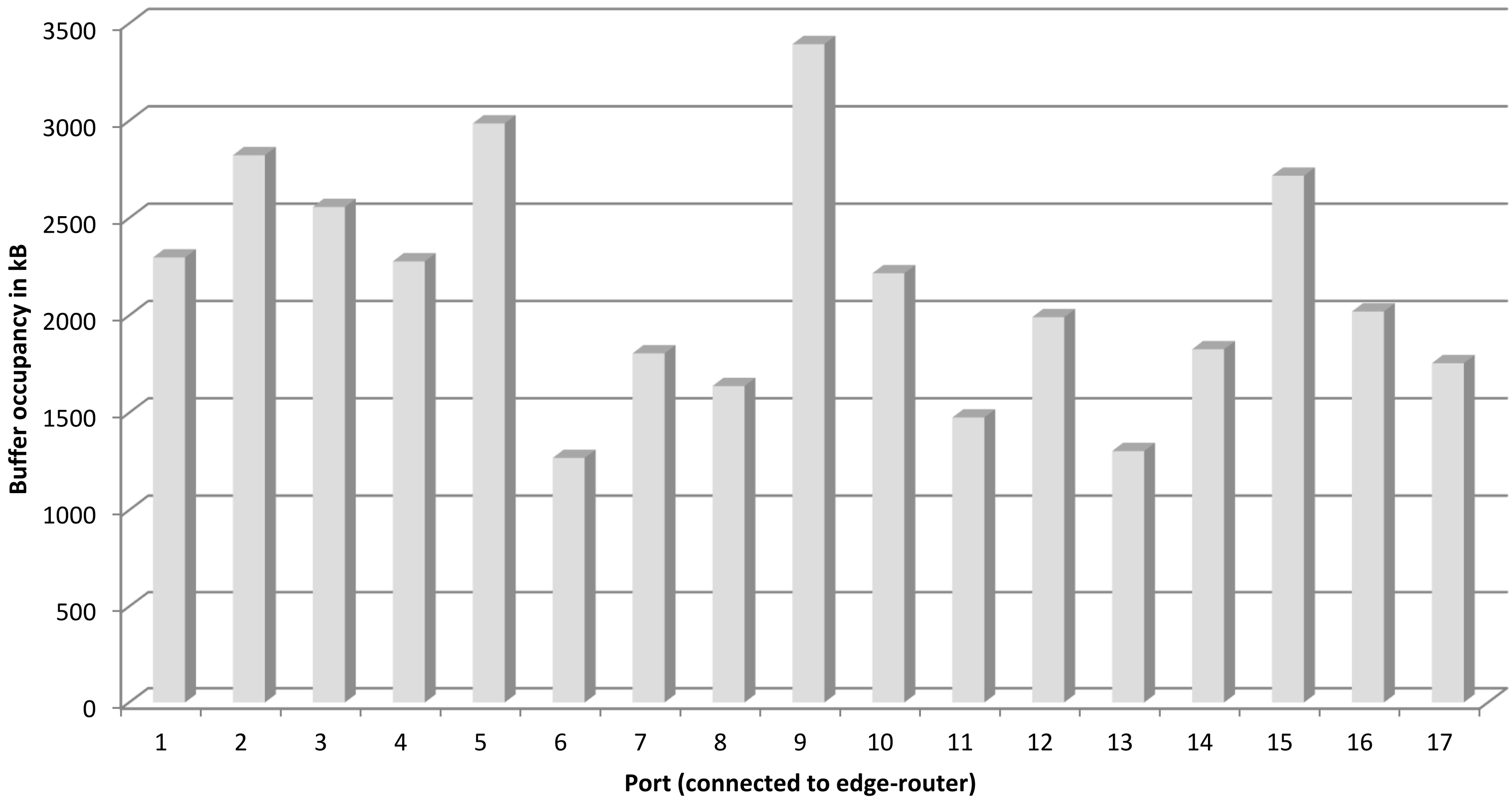
Store-and-Forward in the LHC DAQs

- **256 MB shared memory / 48 ports**
- Up to 1260 ports (1000 BaseT)
- Price / port ~ 500 - 1000 USD
- Used by all LHC experiments
- 6 kW power, 21 U high
- Loads of features (most of them unused ☺ in the experiments)

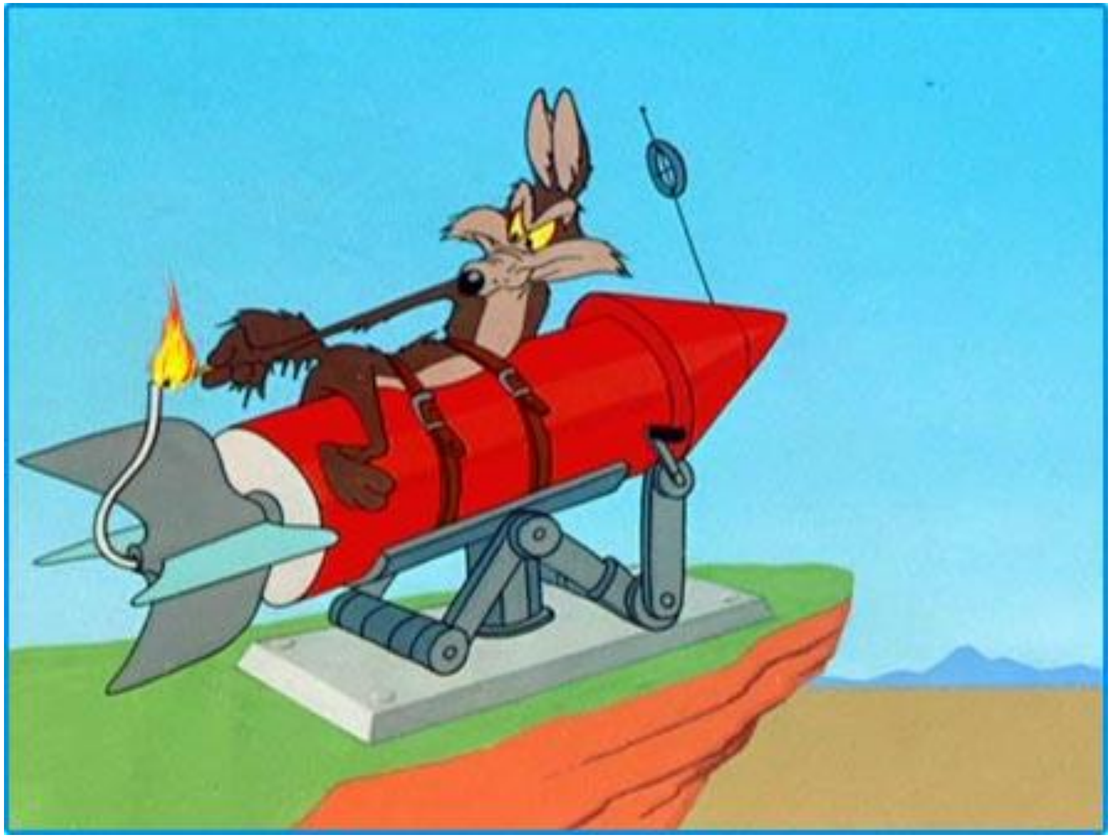
Buffer Usage in F10 E1200i

Buffer usage in core-router with a test using 270 sources @ 350 kHz event-rate

- 256 MB shared between 48 ports
- 17 ports used as “output”
- **This measurement for small LHCb events (50 kB) at 1/3 of nominal capacity**



Runcontrol



Runcontrol challenges

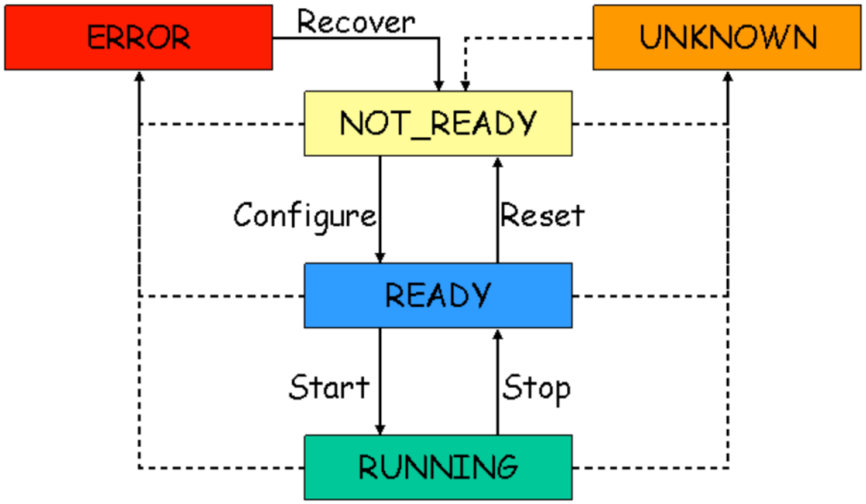
- Start, configure and control O(10000) processes on farms of several 1000 nodes
- Configure and monitor O(10000) front-end elements
- Fast data-base access, caching, pre-loading, parallelization and all this 100% reliable!

Runcontrol technologies

- Communication:
  - CORBA (ATLAS)
  - HTTP/SOAP (CMS)
  - DIM (LHCb, ALICE)
- Behavior & Automatisation:
  - SMI++ (Alice)
  - CLIPS (ATLAS)
  - RCMS (CMS)
  - SMI++ (in PVSS) (used also in the DCS)
- Job/Process control:
  - Based on XDAQ, CORBA, …
  - FMC/PVSS (LHCb, does also fabric monitoring)
- Logging:
  - log4C, log4j, syslog, FMC (again), …

Run Control

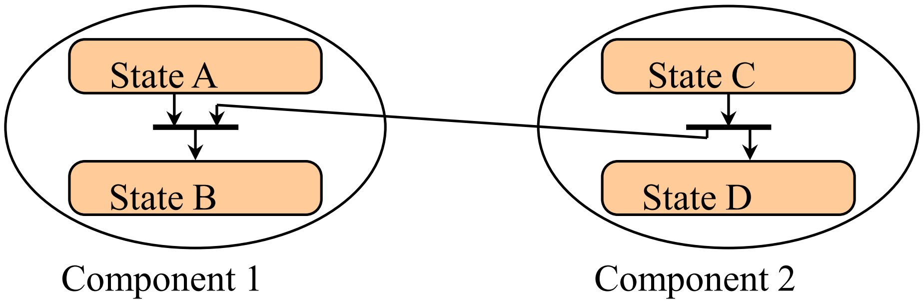
- The run controller provides the control of the trigger and data acquisition system. It is the application that interacts with the operator in charge of running the experiment.
- The operator is not always an expert on T/DAQ. The **user interface** on the Run Controller plays an important role.
- The complete system is modeled as a **finite state machine**. The commands that run controller offers to the operator are state transitions.



LHCb DAQ /Trigger Finite State Machine diagram (simplified)

Finite State Machine

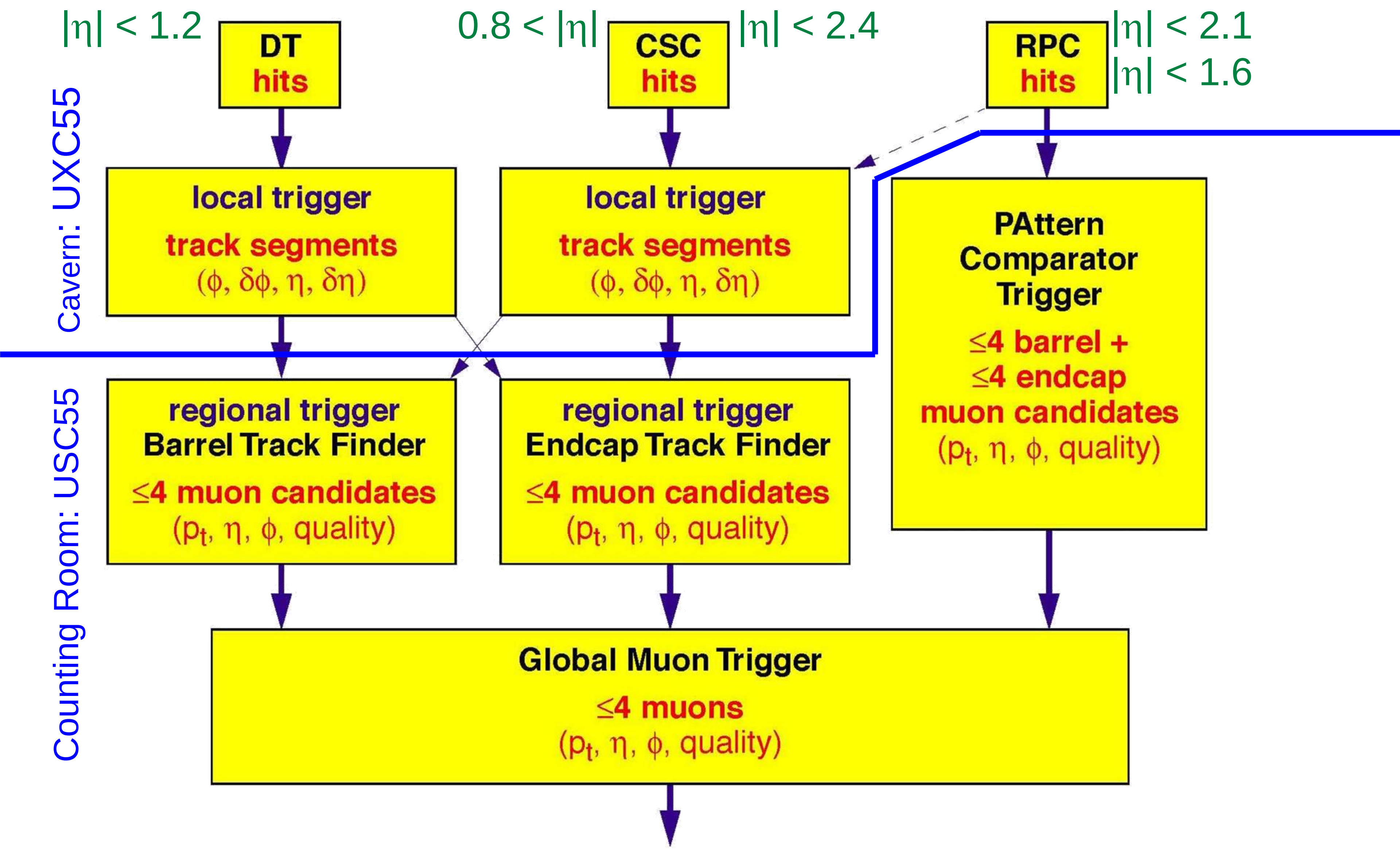
- Each component, sub-component of the system is modeled as a *Finite State Machine (FSM)*. This abstraction facilitates the description of each component behavior without going into detail
- The control of the system is realized by inducing transitions on remote components due to a transition on a local component



Component 1 can only complete the transition to State B if Component 2 is in state D.

- Each transition may have actions associated. The action consist of code which needs to be executed in order to bring the component to its new state
- The functionality of the FSM and state propagation is available in special software packages such as SMI or frameworks XDAQ

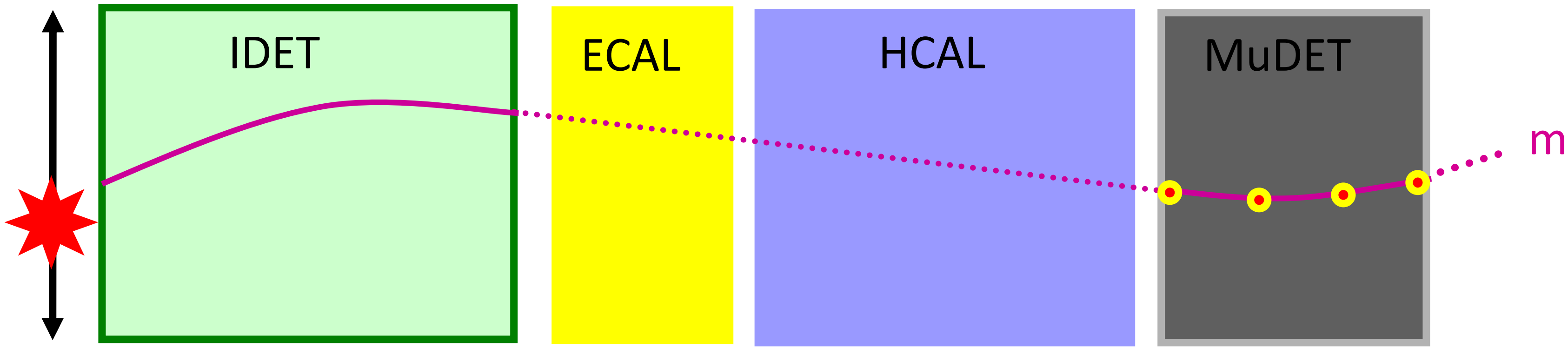
CMS muon trigger overview



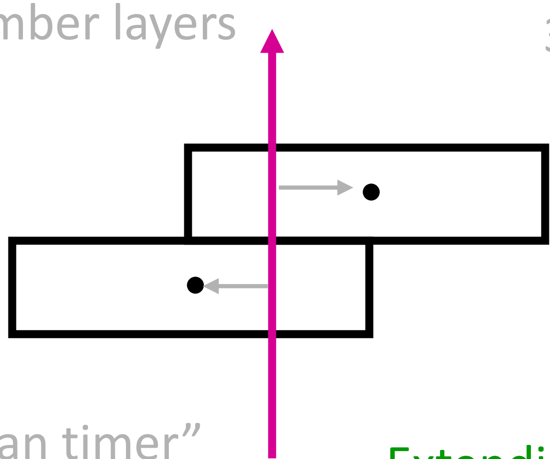
p_T, h, f of candidates sent to CMS Global Trigger

(ATLAS passes only multiplicity to CTP)

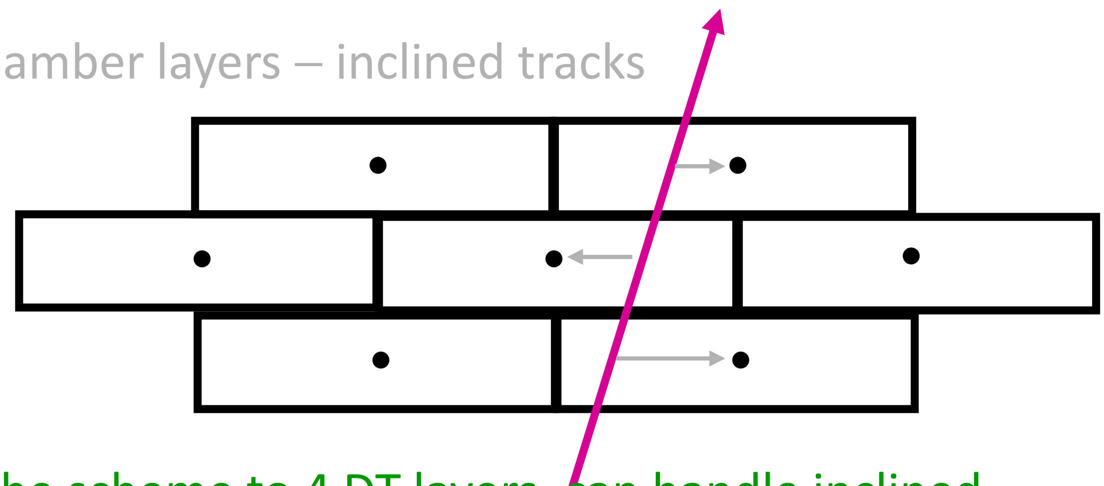
Illustration - principle of DT trigger



2 chamber layers



3 chamber layers – inclined tracks



"mean timer"

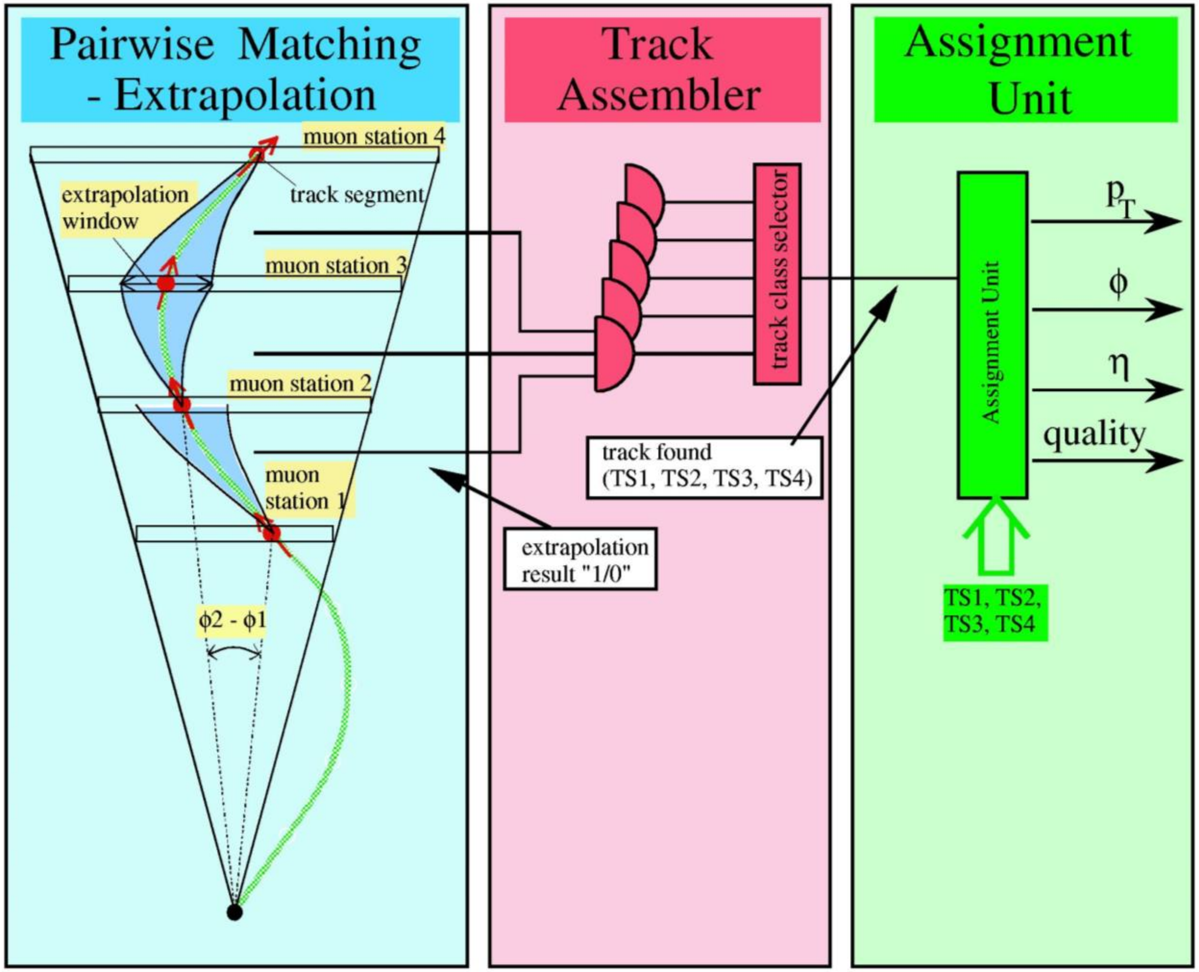
$T_1+T_2 = T_{max}$

$(T_1-T_2)/2v_d = x$

Extending the scheme to 4 DT layers, can handle inclined tracks even if 1 hit lost due to inefficiency or dead region

- provides identified BC, position, angle with high efficiency

CMS DT track finder



Track-finder electronics is mounted off detector and is implemented using FPGAs

- LUTs in FPGAs contain limits of extrapolation windows
- Track segments are combined to find the “best” two tracks within a sector
- The track parameters are then determined from the measurements in different stations