Preparing the LHCb data acquisition for LHC Run3

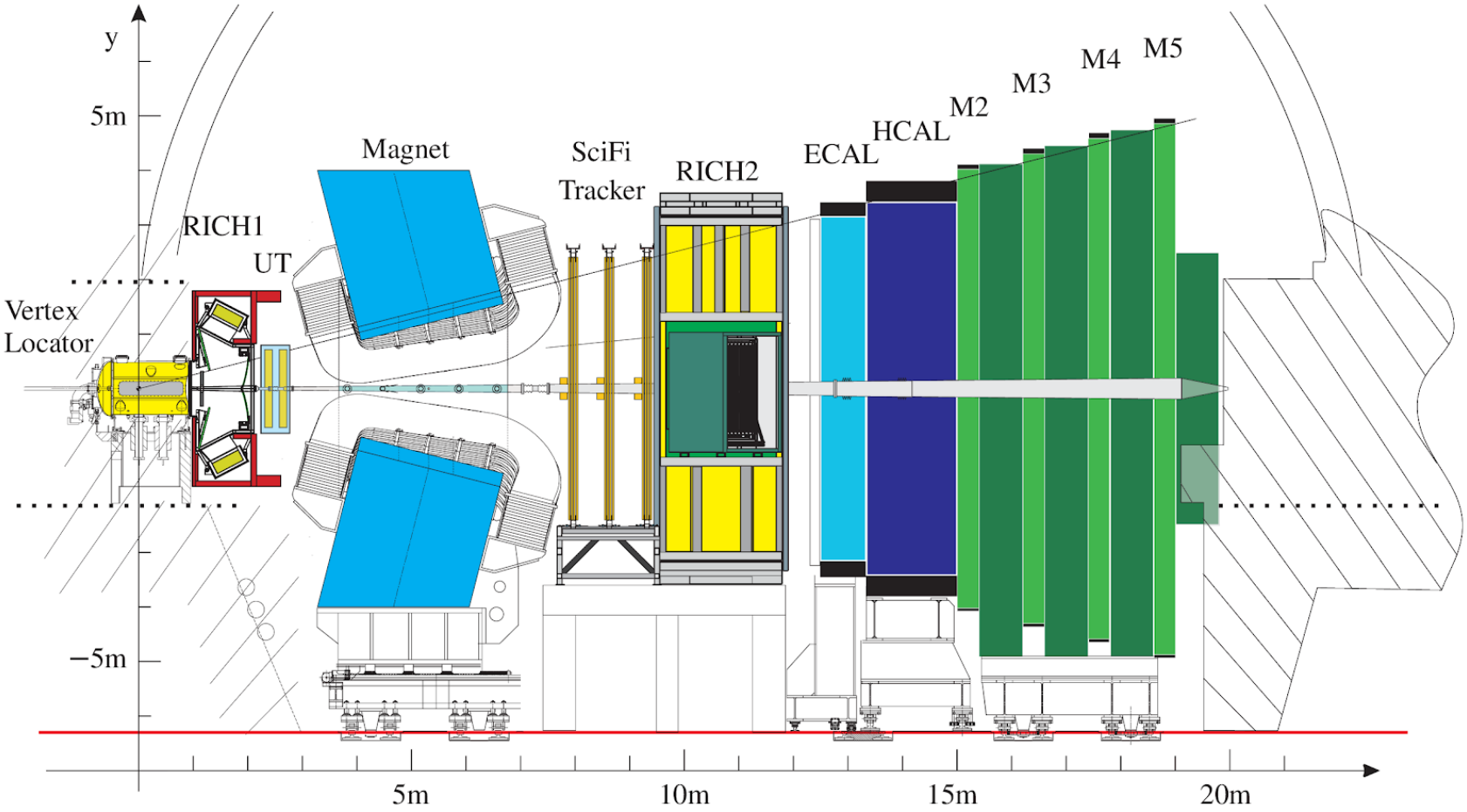
22nd Virtual IEEE Real Time Conference

Flavio Pisani for the LHCb Online team

CERN

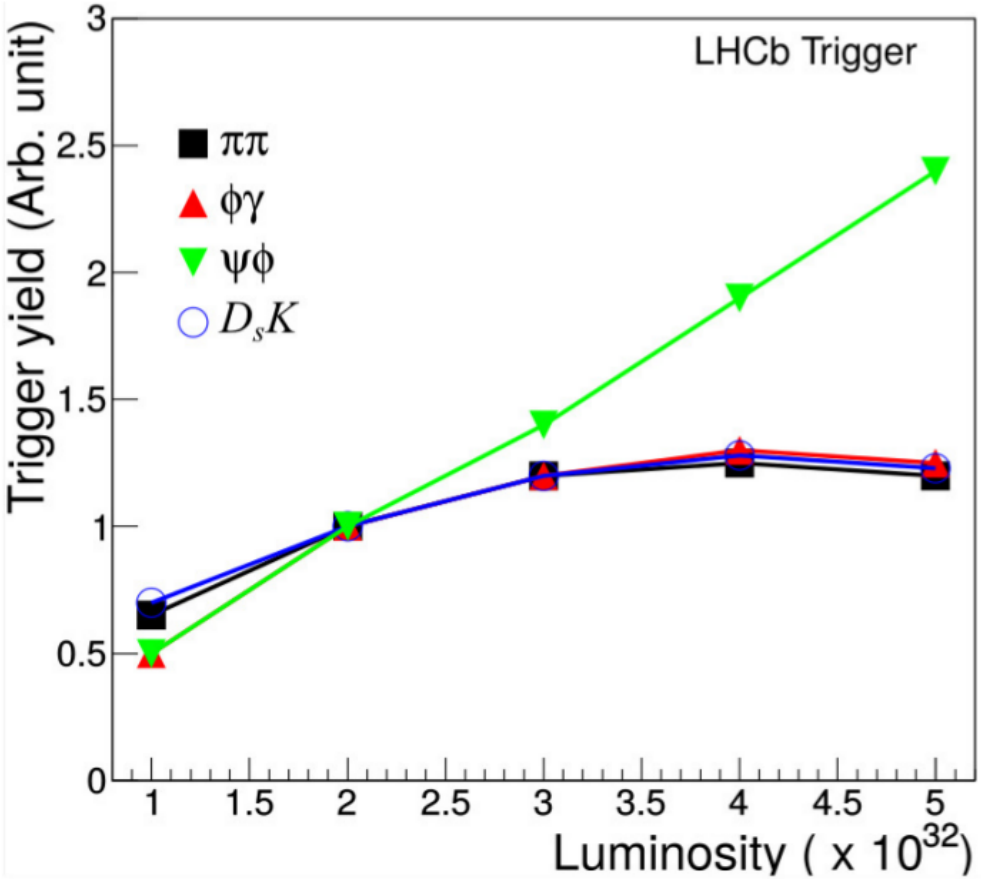
Oct 12 - 23, 2020

The LHCb experiment



Why do we want to read out every collision?

Low level trigger yield vs Luminosity ($cm^{-2}s^{-1}$)

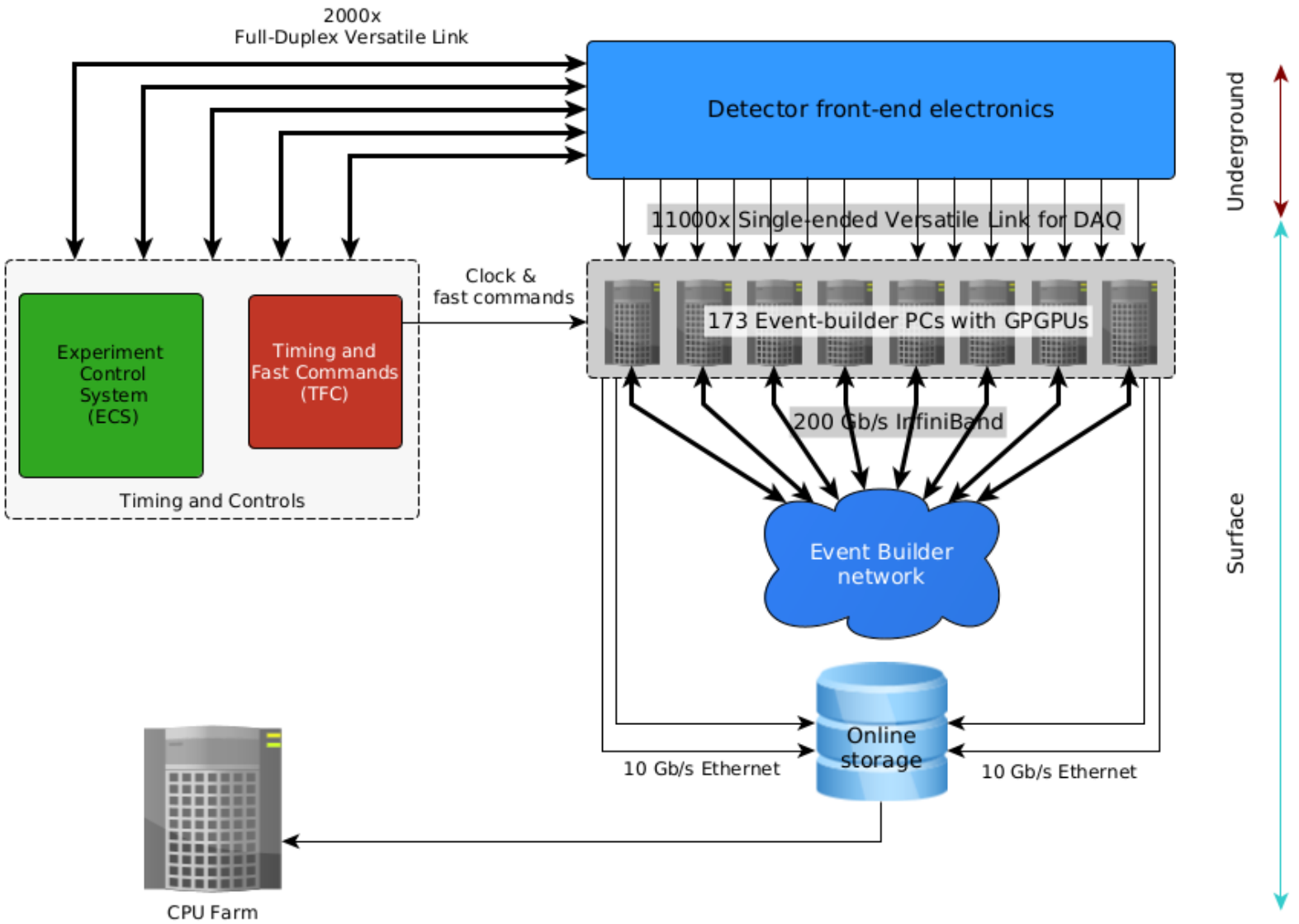


The low level trigger is not efficient at high luminosity

Why can we read out every collision?

- Spectrometer geometry (fibres/cables are not "in the way")
- "Zero-suppression" on the detectors
- Relatively low radiation levels permit to relax the constraint on the FPGAs used for "middle" layer processing
- Total event-size comparatively small ($\sim$ 100 kB)
- Software trigger can do online selection with offline-like reconstruction

LHCb "Online" system



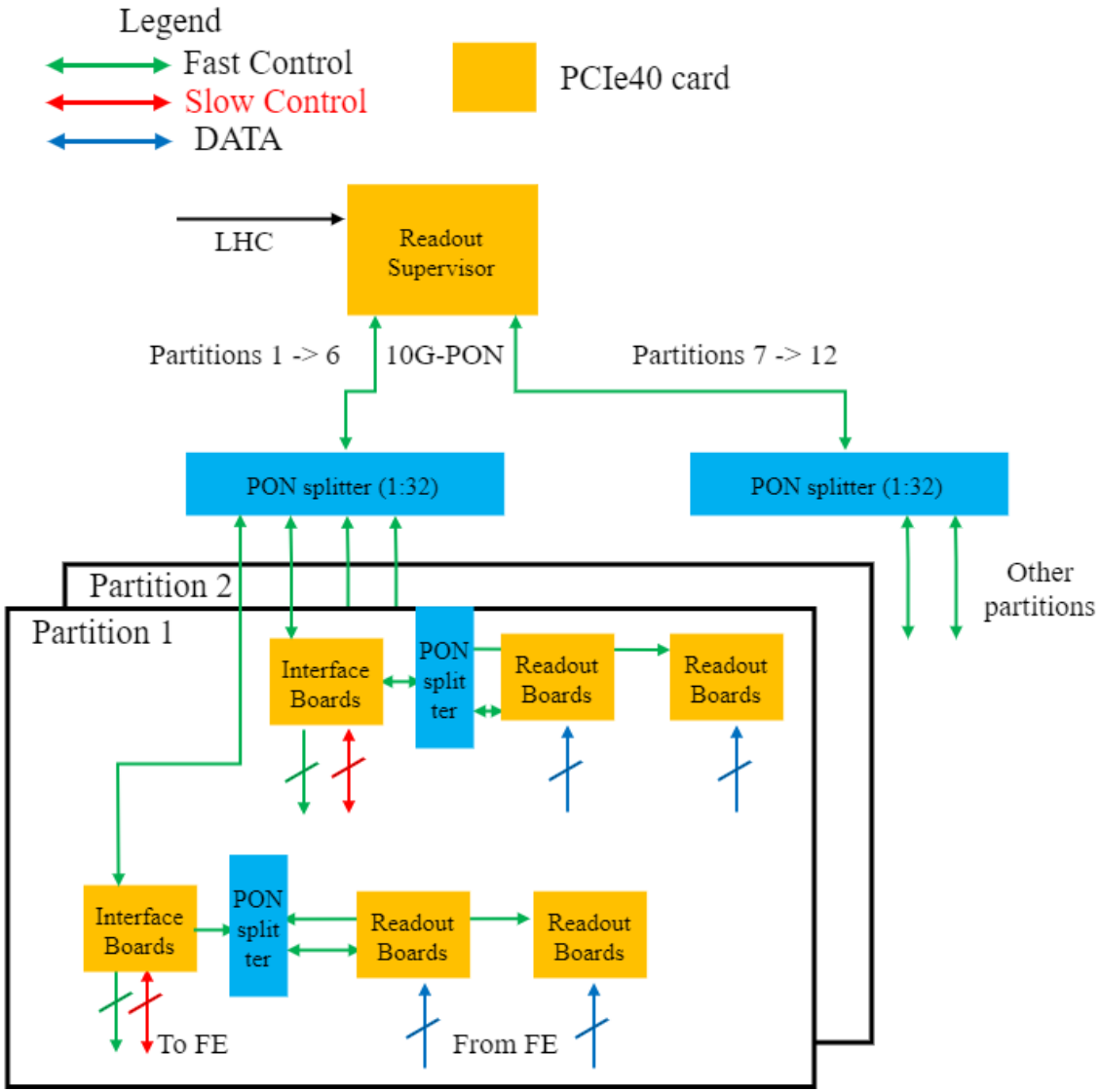
The PCIe40: a single custom-made FPGA board for DAQ and Control

- Based on Intel Arria10
- 48x10G capable transceiver on 8xMPO for up to 48 full-duplex Versatile Links
- 2 dedicated 10G SFP+ for timing distribution
- 2x8 Gen3 PCIe

One board / many firmware personalities

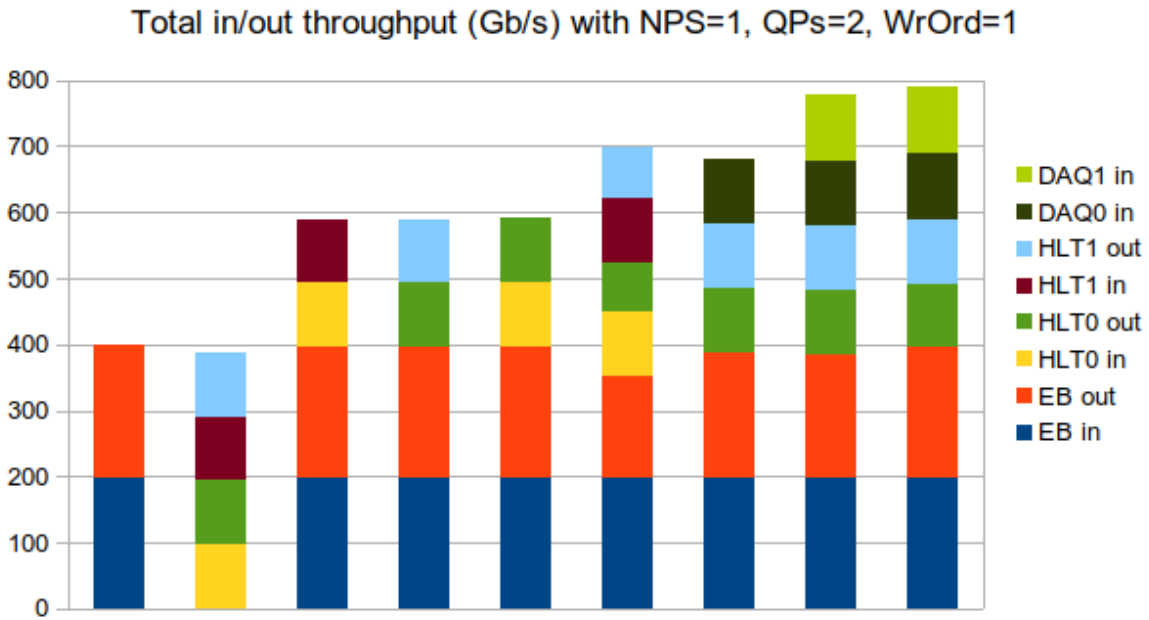
- **Readout Supervisor (SODIN):**
  - reception and distribution of global timing
  - generation and distribution of synchronous and asynchronous command
  - generation of events veto, triggers and calibration events
- **Interface Board (SOL40):**
  - distribution of the global timing to the Front-End (FE)
  - Interface bridge between the ECS and the FE
- **TELL40:**
  - acquisition and first pre-processing of the data

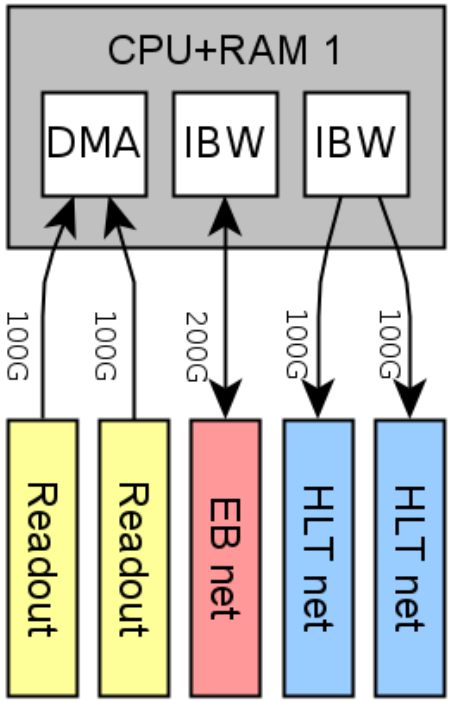
Timing, Synchronous Control & Partitioning



- Synchronously driving the Front-End electronics over the GBT
- 10G-PON for efficient Back-End signal distribution and fixed phase clock recovery
- Partitioning for debugging and commissioning

Challenges for the EB servers



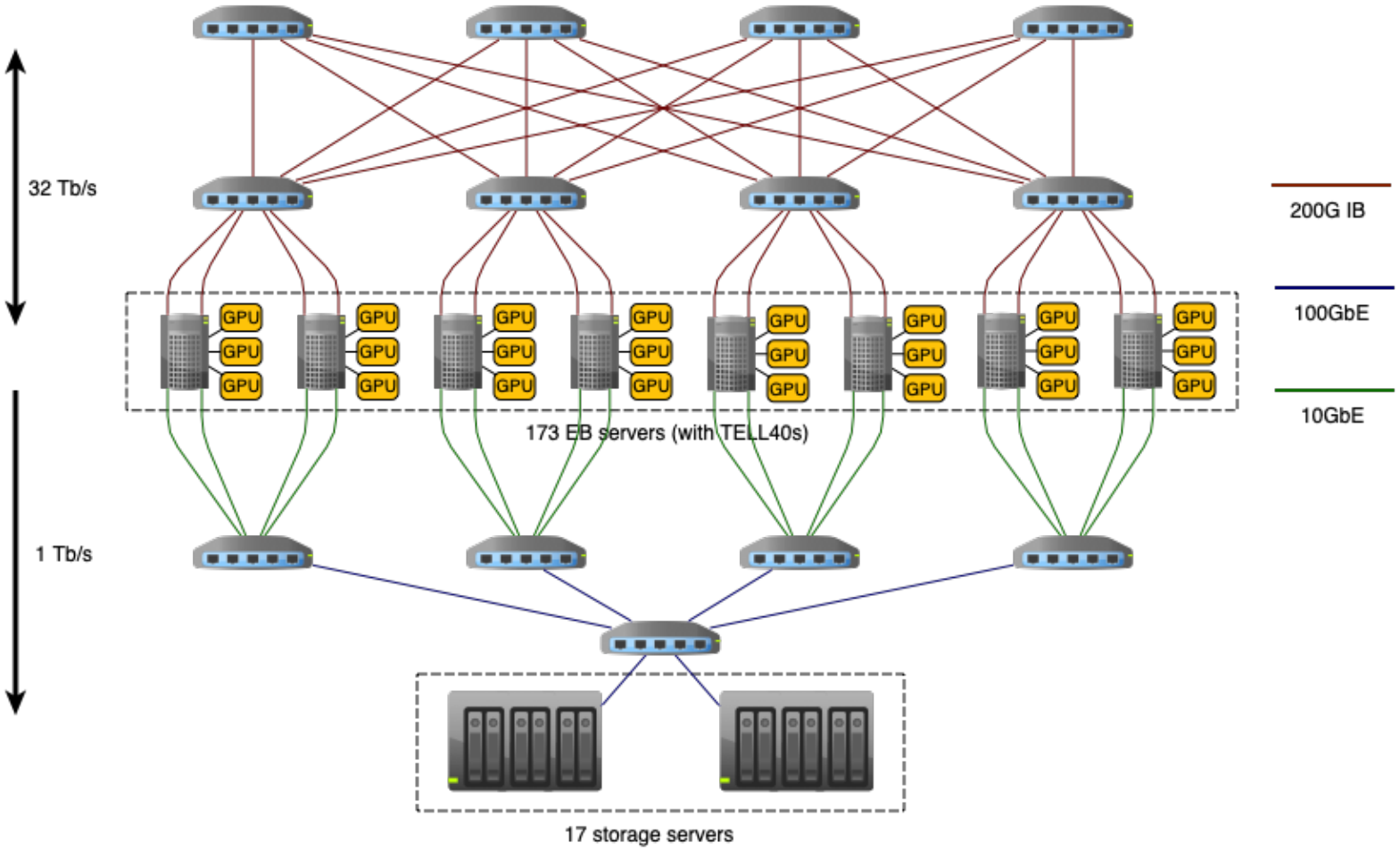


Selected server: dual socket AMD Rome-based solution with 8 PCIe Gen4 slots.

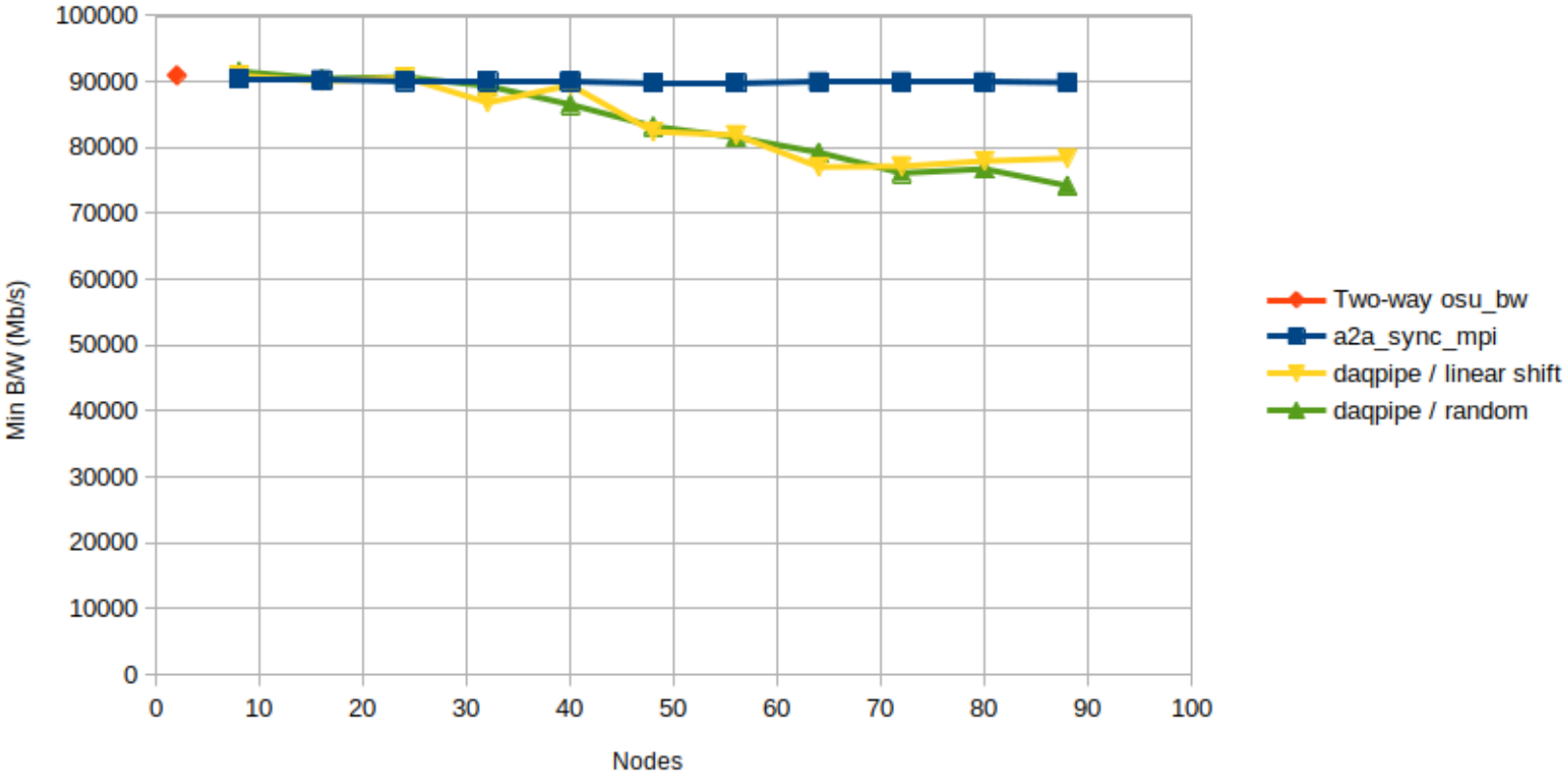
Challenges for the EB network

- Needs to collect data from 478 TELL40 FPGA boards into a single "location"
- And hand them over to compute units for further processing
- Traffic is inherently congestion inducing
- Want high link-load (cost)
- Want to use some kind of remote DMA to reduce server-load

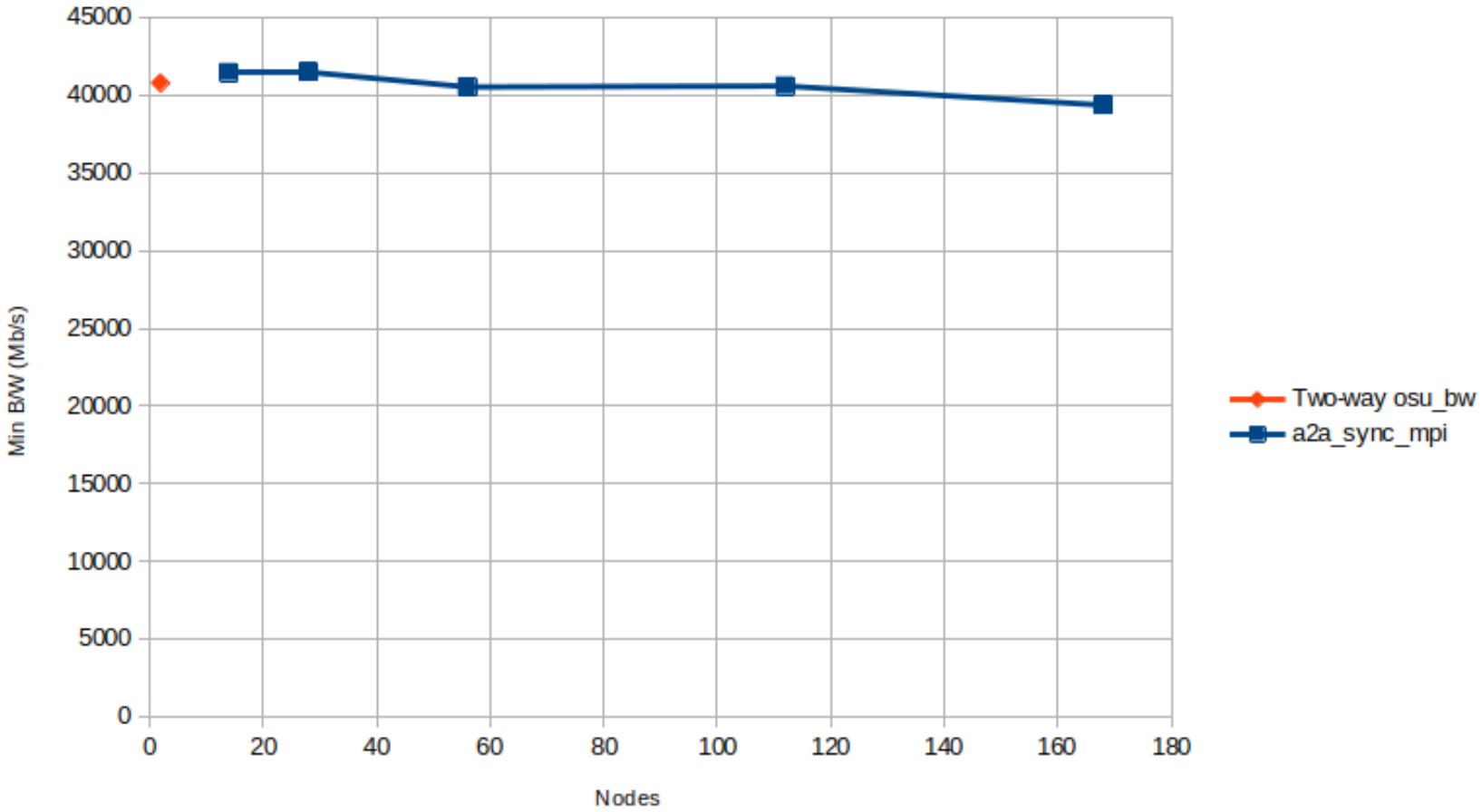
Event-builder (network-view)



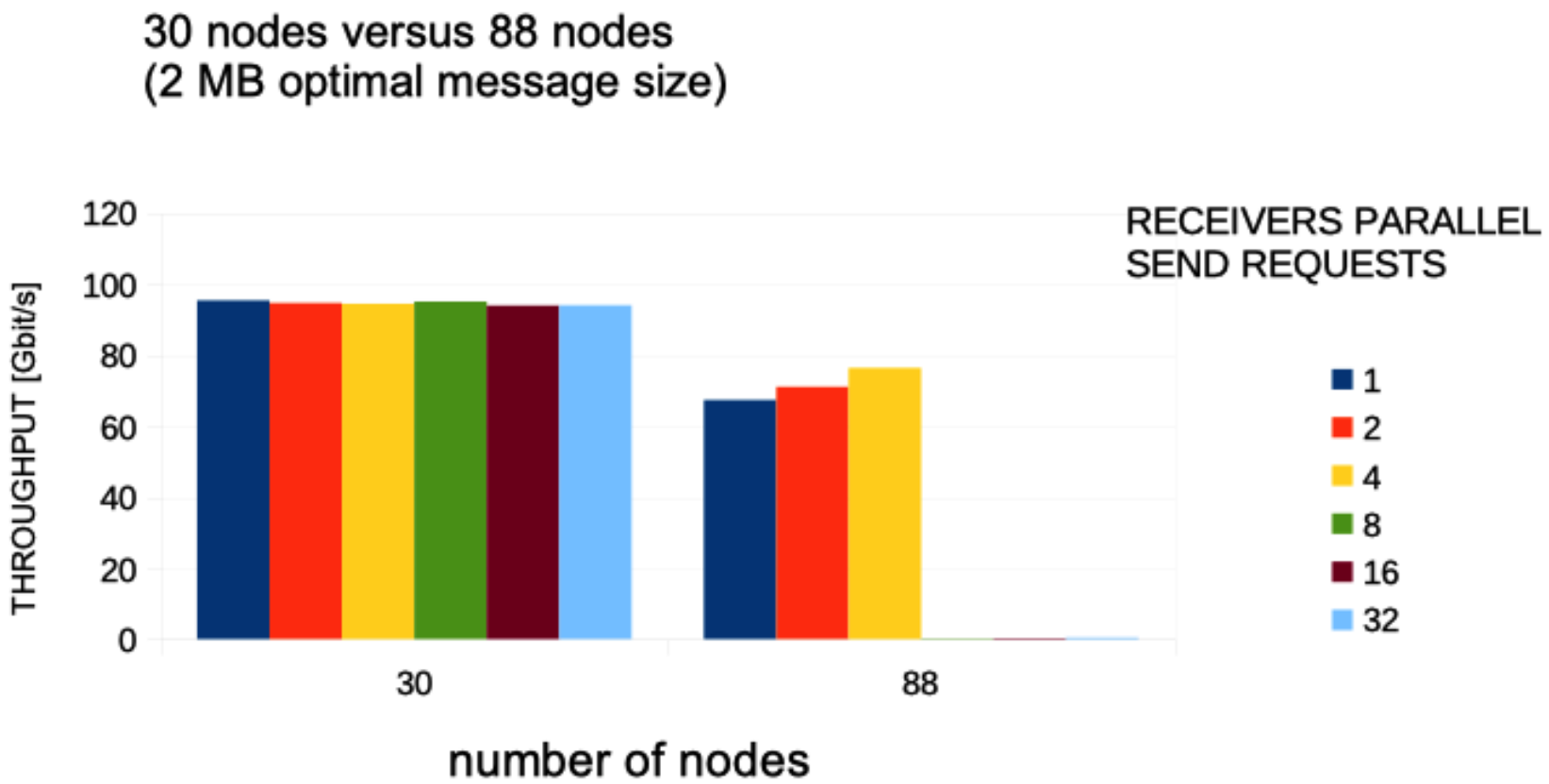
Scalability InfiniBand



Scalability InfiniBand



Scalability Ethernet (deep buffers)



Reasons for going with InfiniBand

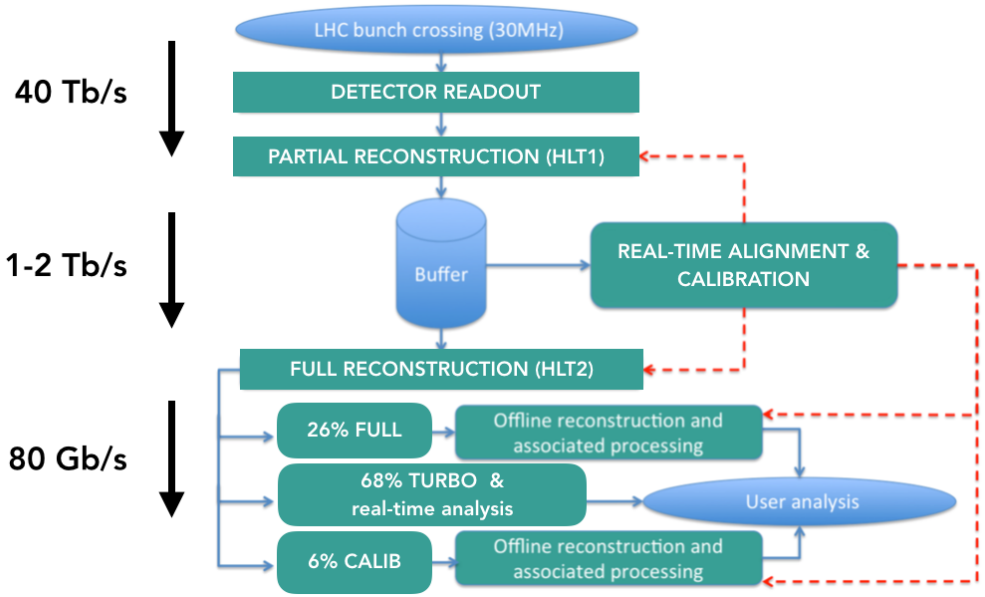
- PCIe Gen4 allows using 200 Gbit/s connections which save cost and help with scalability. However 200 Gbit/s so far only effectively exists for InfiniBand!
- Ethernet flow-control could not be made to work properly on available reference platforms
- Ethernet remains - for us - affected by worrying / irritating scaling issues
- Probably most important: could never get access to a really big Ethernet test-system: need the full event-builder for testing. For InfiniBand we have used super-computer sites

...ergo

Lowest risk solution - within our budget - is the InfiniBand solution

Data-processing and event selection

- Two stages of software filtering:
  1. "HLT1" on GPGPUs
  2. "HLT2" on a CPU-farm
- Large storage buffer to decouple the two HTL stages
- Calibration and alignment are performed "semi-live", while the data are buffered



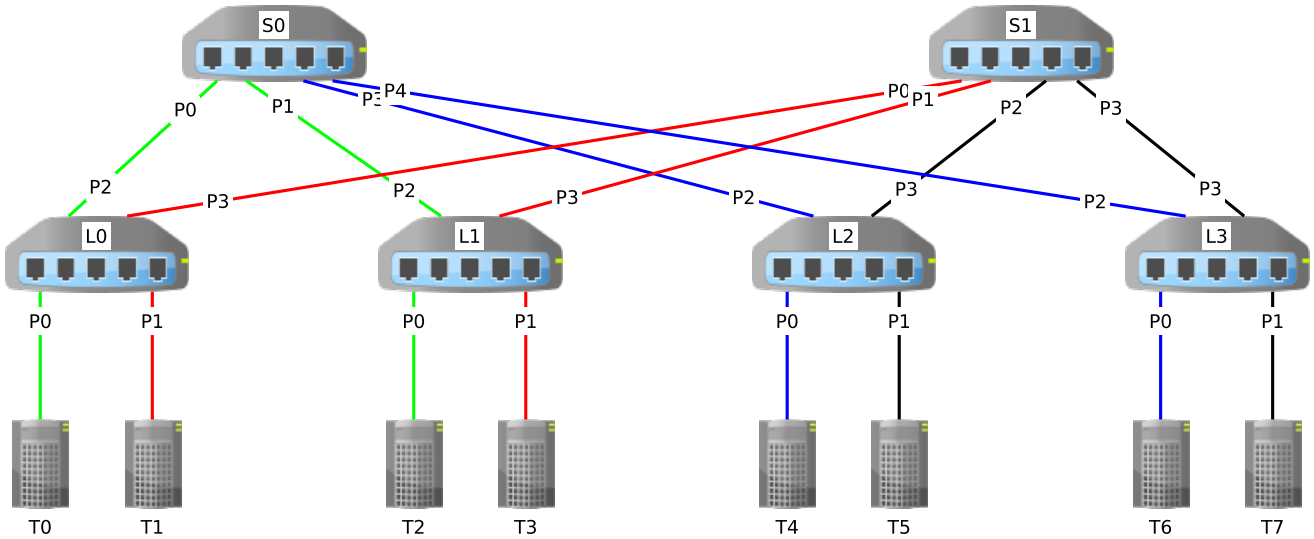
Summary

- LHCb can do and afford a full read-out at bunch-crossing rate
- Single stage synchronous readout built around GBT, PON and a single flexible FPGA board
- Detector control uses the same FPGA boards as the timing distribution system
- AMD Rome (PCIe Gen4) based servers make compact, very high-I/O event-builder, connected with 200 Gb/s InfiniBand
- Event-selection is entirely in software to maximize physics yield, increase the amount of data collected, flexibility and minimize cost
- The system is very well scalable, by up to 3 a factor without any substantial changes

THANK YOU FOR YOUR ATTENTION

More material

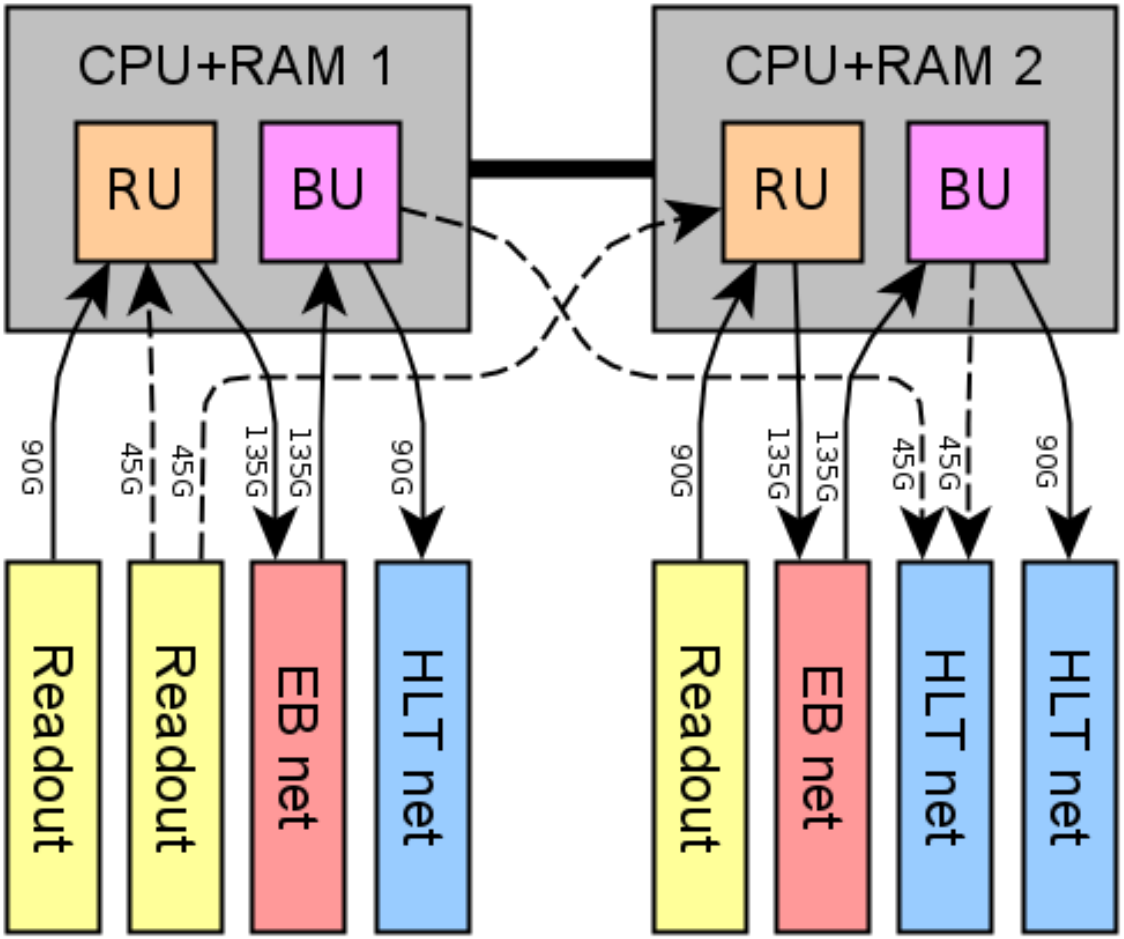
Linear shifting scheduling



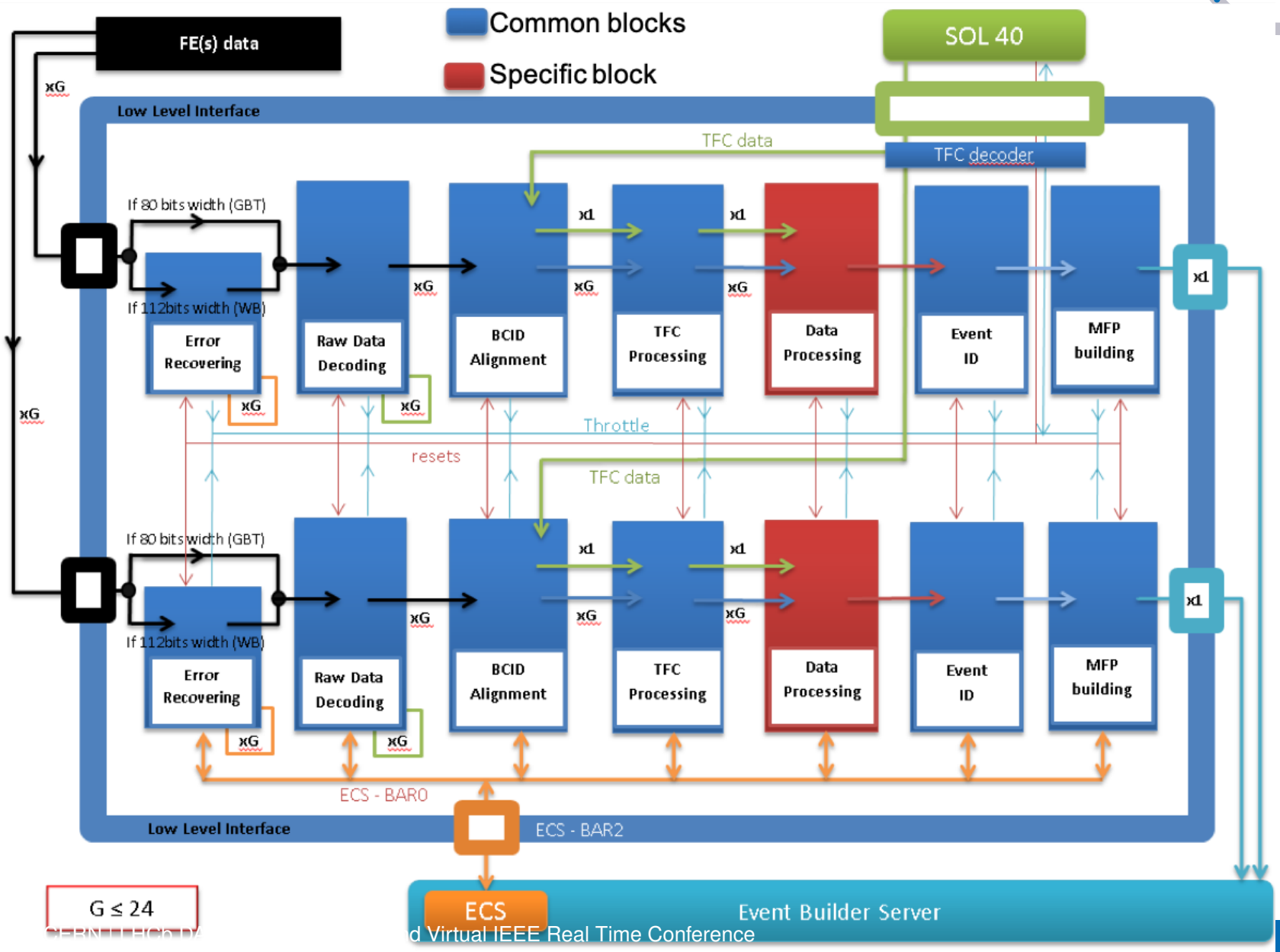
- The processing of N events is divided into N phases
- In every phase one EB-server sends data to only one EB-server, and it receives data from only one
- During phase n the EB-server x sends data to the EB-server $(x + n)\%N$
- All the hosts switch synchronously from phase n to phase $n + 1$

Congestion-free traffic on “selected networks” (i.e. non blocking networks)

Event-builder server architecture



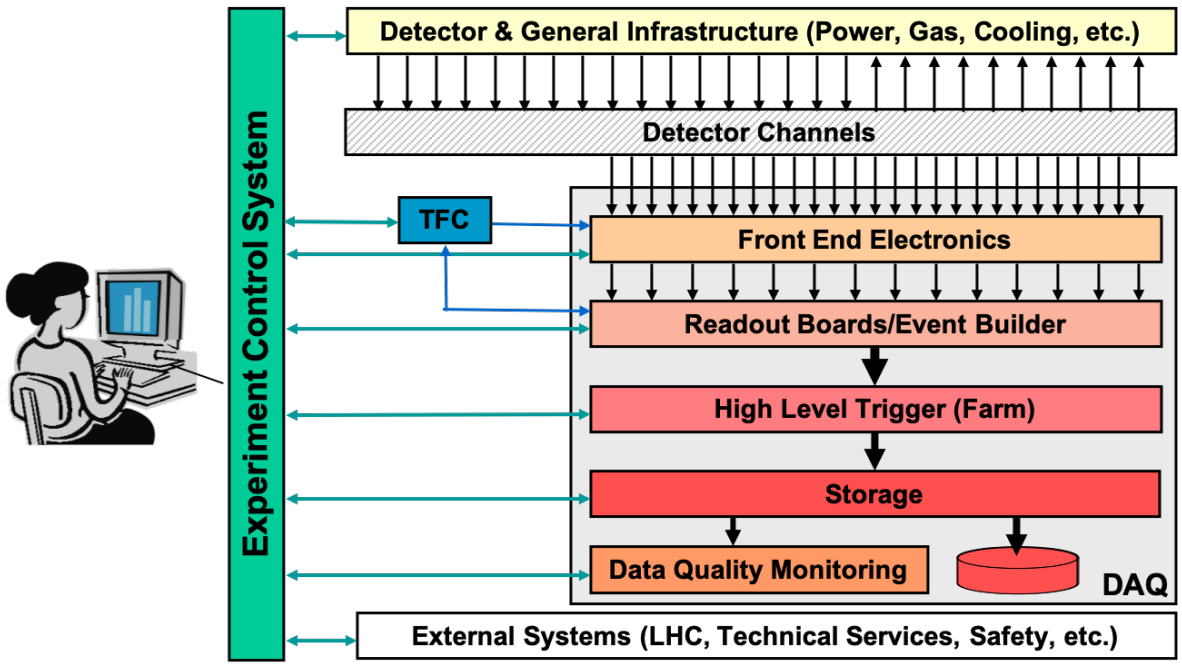
TELL40 architecture



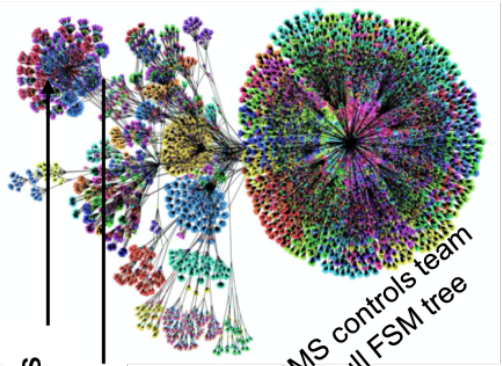
The Experiment Control System

Last but not least

- Operational efficiency of the system is crucial
- LHCb's ECS provides a uniform way to control the **entire** experiment and automate its operation



Uniform, hierarchical control based on FSM



Thanks to the CMS controls team for visualizing our full FSM tree

Status & Alarms

Commands