LHCb Trigger & DAQ System Overview

Guoming Liu

South China Normal University

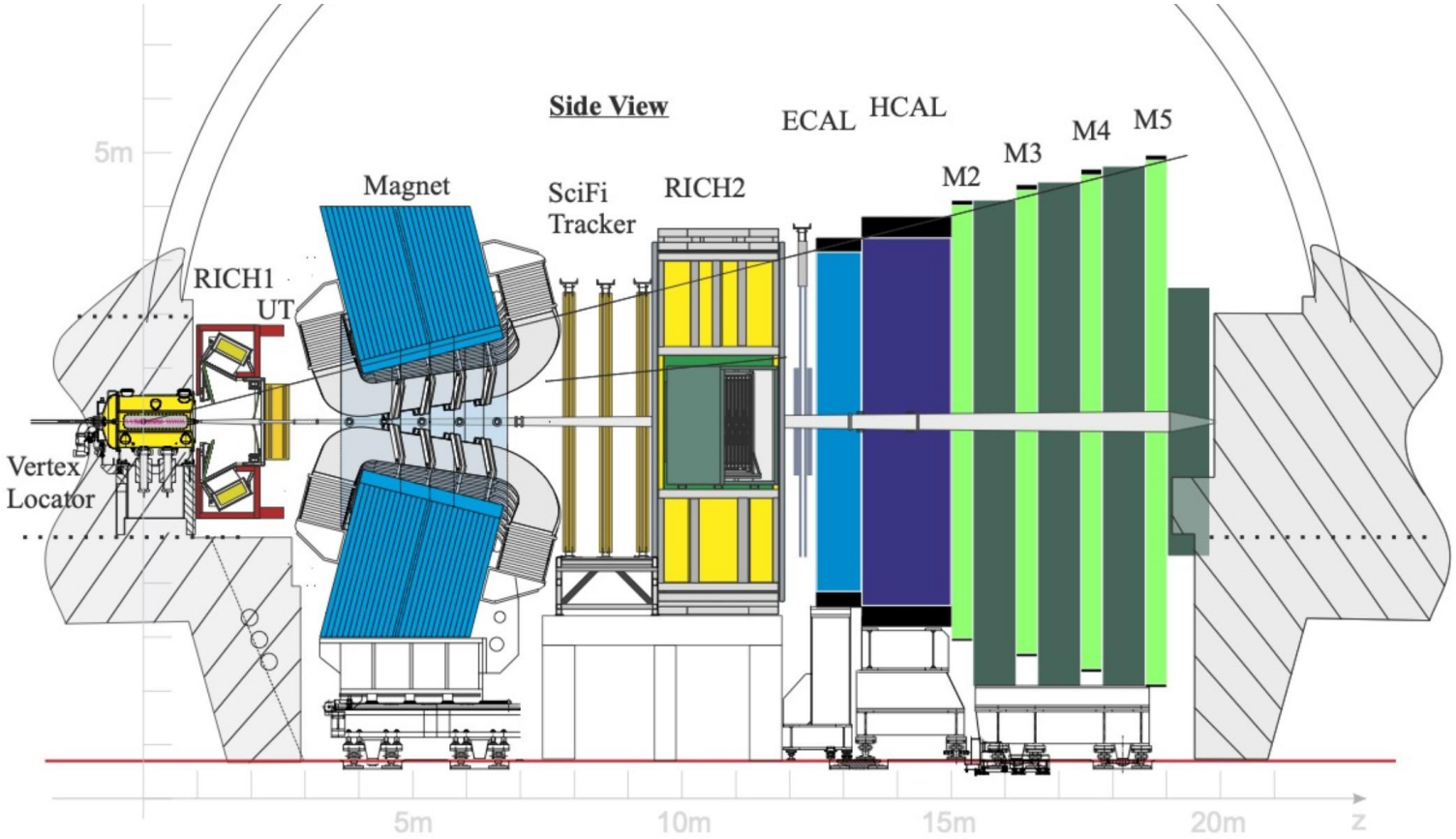
CEPC Workshop 2025, Guangzhou, China

Many slides from: F. Pisani,
T. Colombo, N. Neufeld, et al.

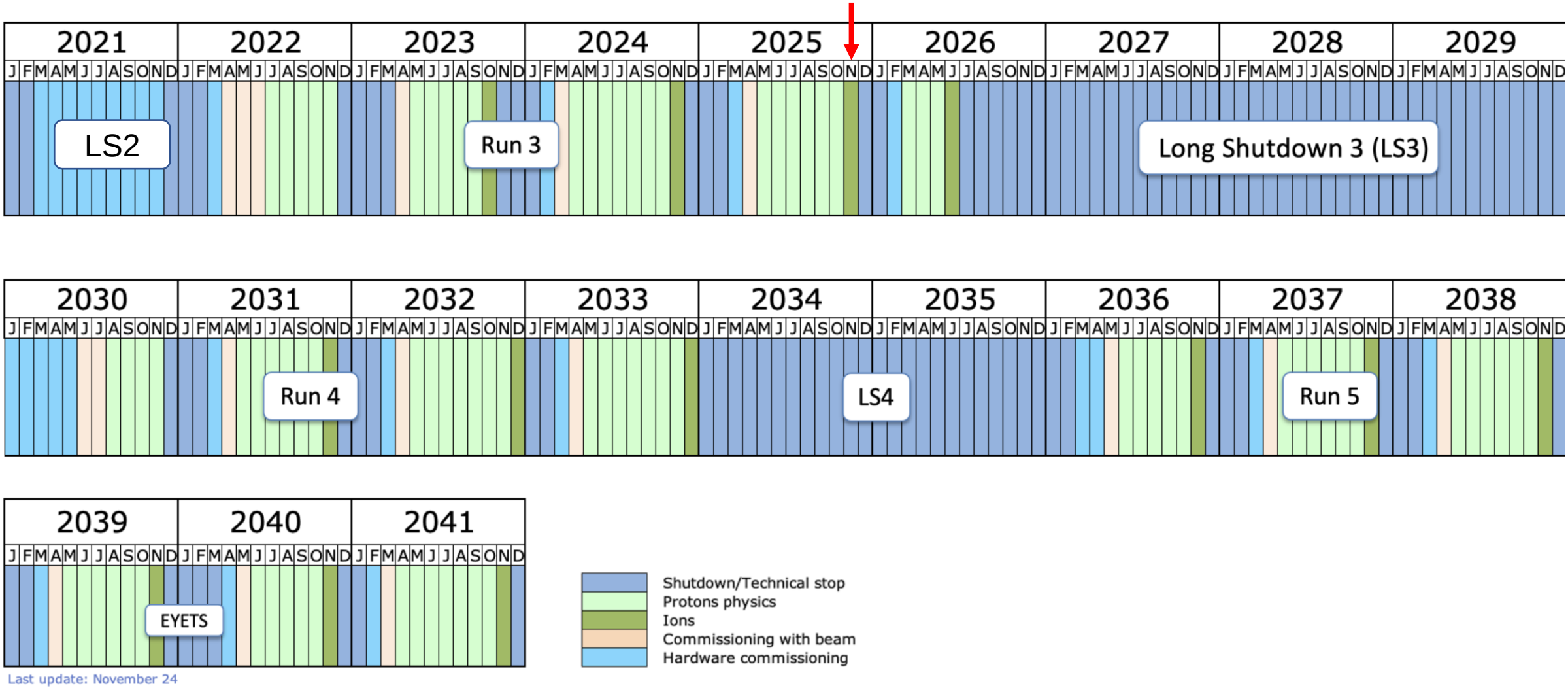
LHCb Detector in Run3

- Single-arm forward spectrometer at the LHC
- p-p bunch crossing rate: 30 MHz
- Luminosity:

  $2 \times 10^{33}\ cm^{-2}s^{-1}$



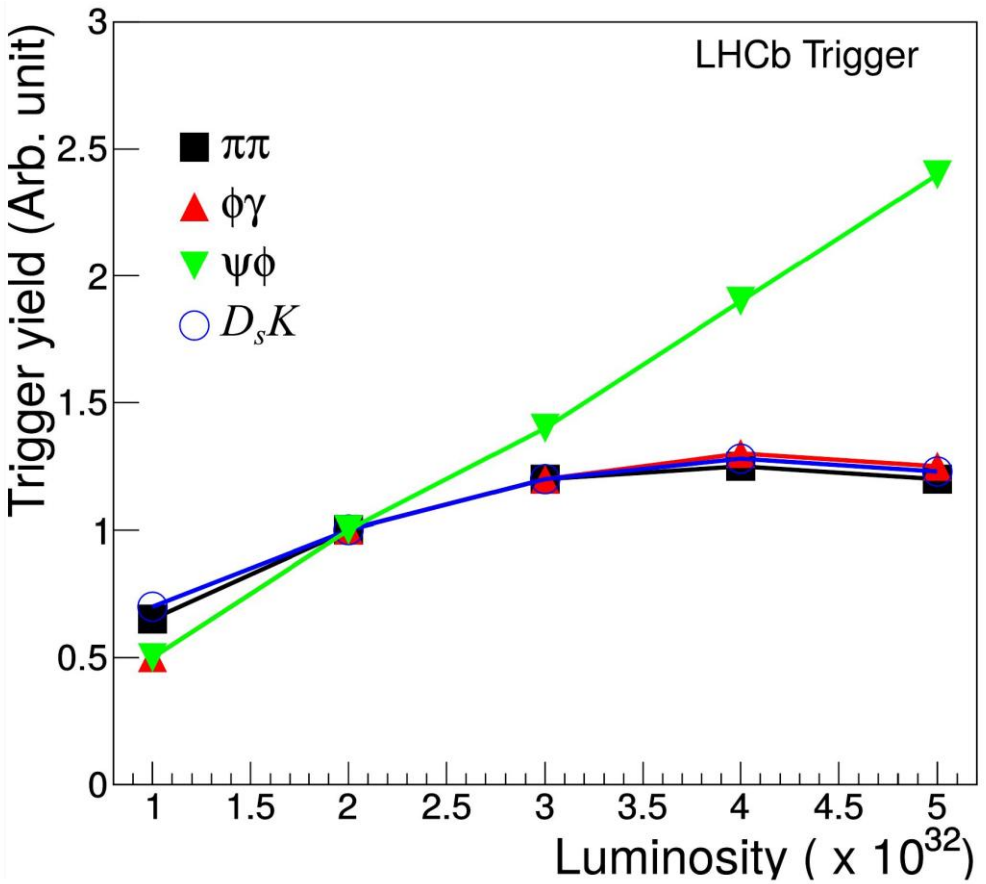
LHC Schedule



LHCb Upgrade I: trigger-less readout

- Why trigger-less readout?
- Run 1 & 2：
  - ➢ Instantaneous luminosity:

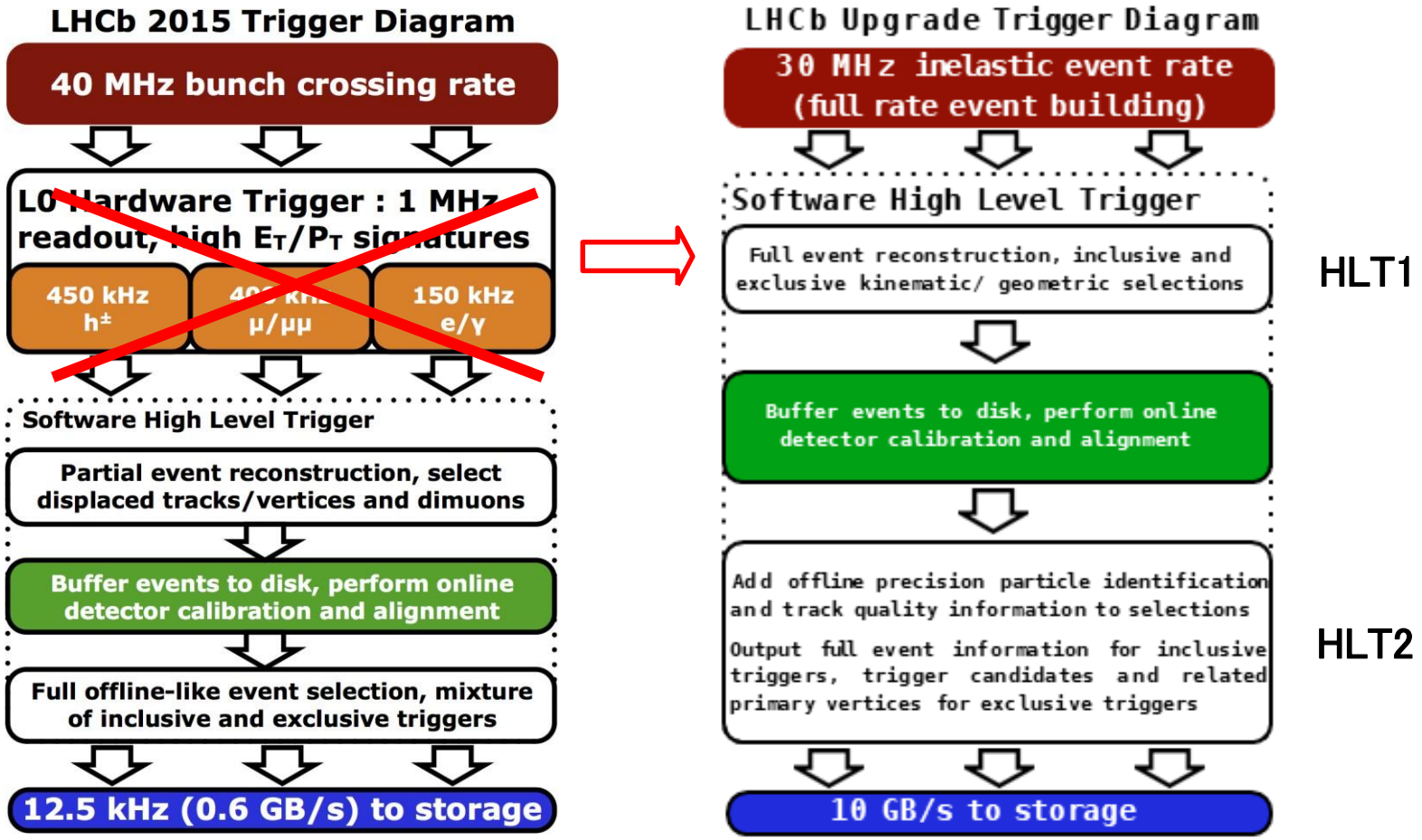
    $4 \times 10^{32}\ \mathrm{cm}^{-2}\ \mathrm{s}^{-1}$
  - ➢ L0 trigger: hardware (40 →1 MHz)
    - using high Pt / Et signatures
    - 1 MHz limit saturates hadronic modes
- An upgrade is needed to take advantage of the increased luminosity
- Solution: read full event at bunch-crossing rate
  - ➢ high event rate but small event size



LHCb Upgrade I: trigger-less readout

- Hardware trigger (L0) removed
- Trigger implemented purely in software (HLT1 + HLT2)

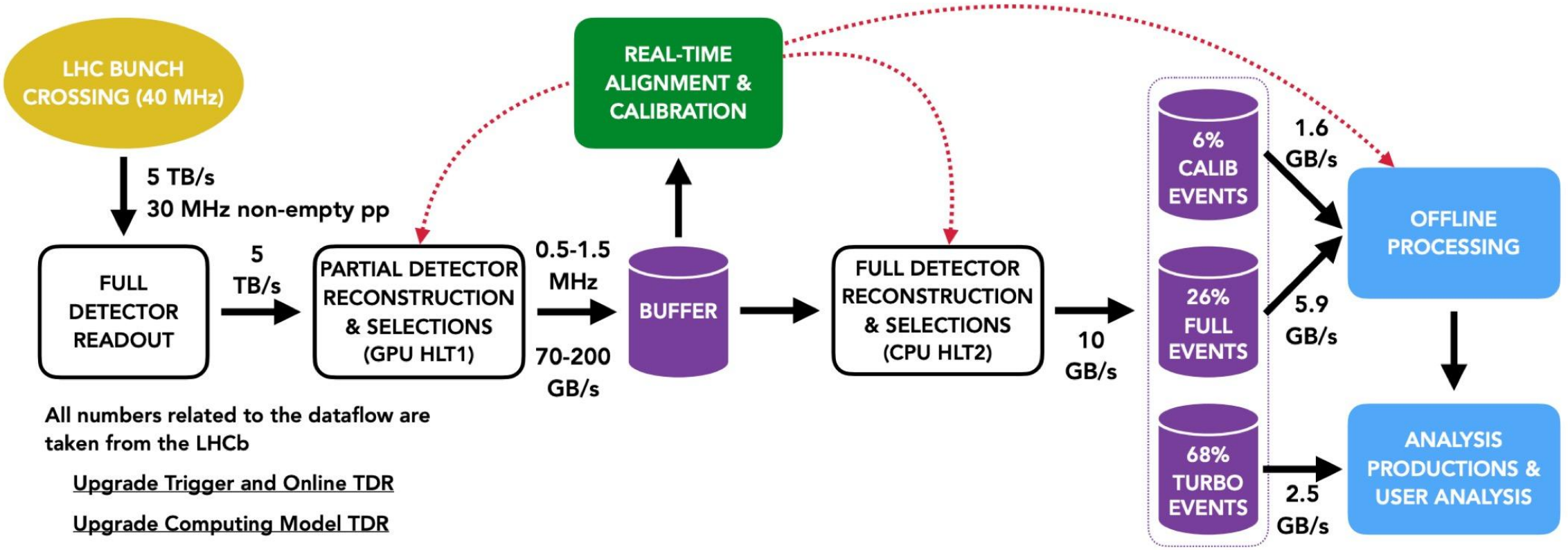
*HLT = High Level Trigger



Dataflow in Run3

- **Two stages of software filtering**
  - **HLT1 on GPGPUs**
  - **HLT2 on a CPU farm**

[LHCB-FIGURE-2020-016]



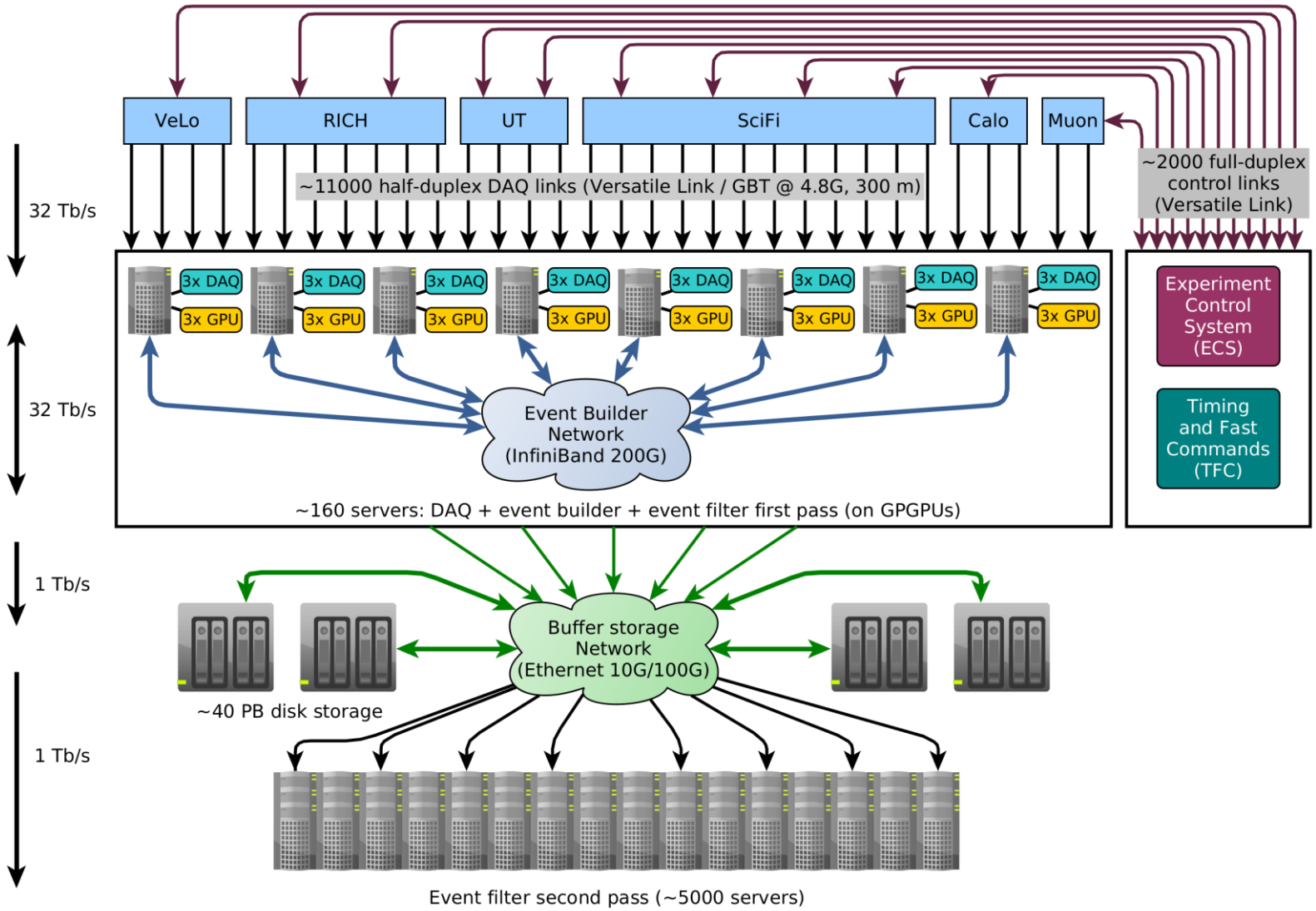
All numbers related to the dataflow are taken from the LHCb

Upgrade Trigger and Online TDR

Upgrade Computing Model TDR

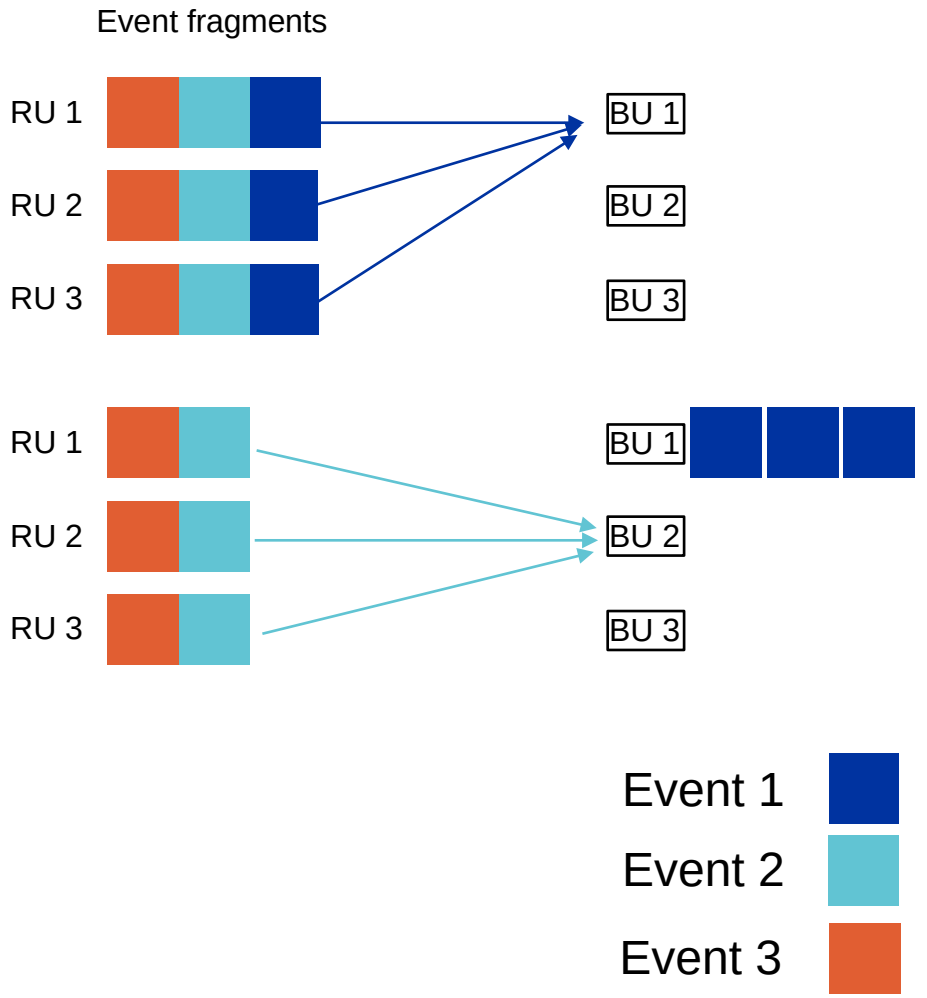
DAQ Architecture in Run3

- Event rate: ~ 30 MHz non-empty bunch crossing
- Data from the cavern (~ 100m underground) to the surface via versatile links
- Event size: ~ 100 kB
- Event Building (EB) **bandwidth: ~ 32 Tbit/s**



Event Building in a nutshell

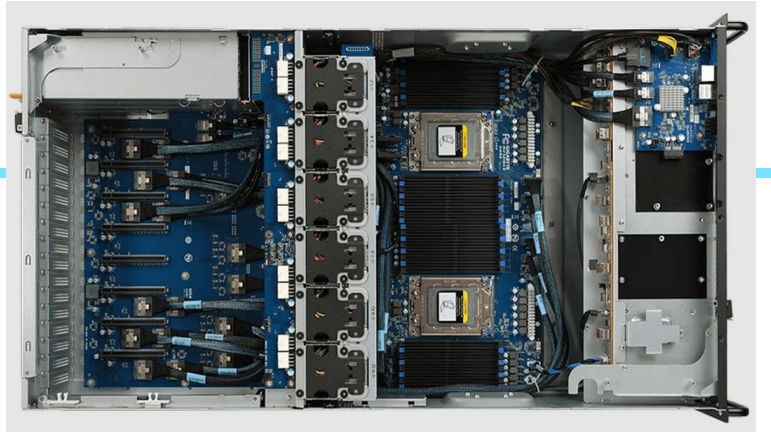
- Every event is divided into multiple fragments
- Every Readout Unit (RU) receives a fragment of the event
- Every Builder Unit (BU) needs to have all the fragments of the event
- The RU is responsible of pushing the fragment to the corresponding BU



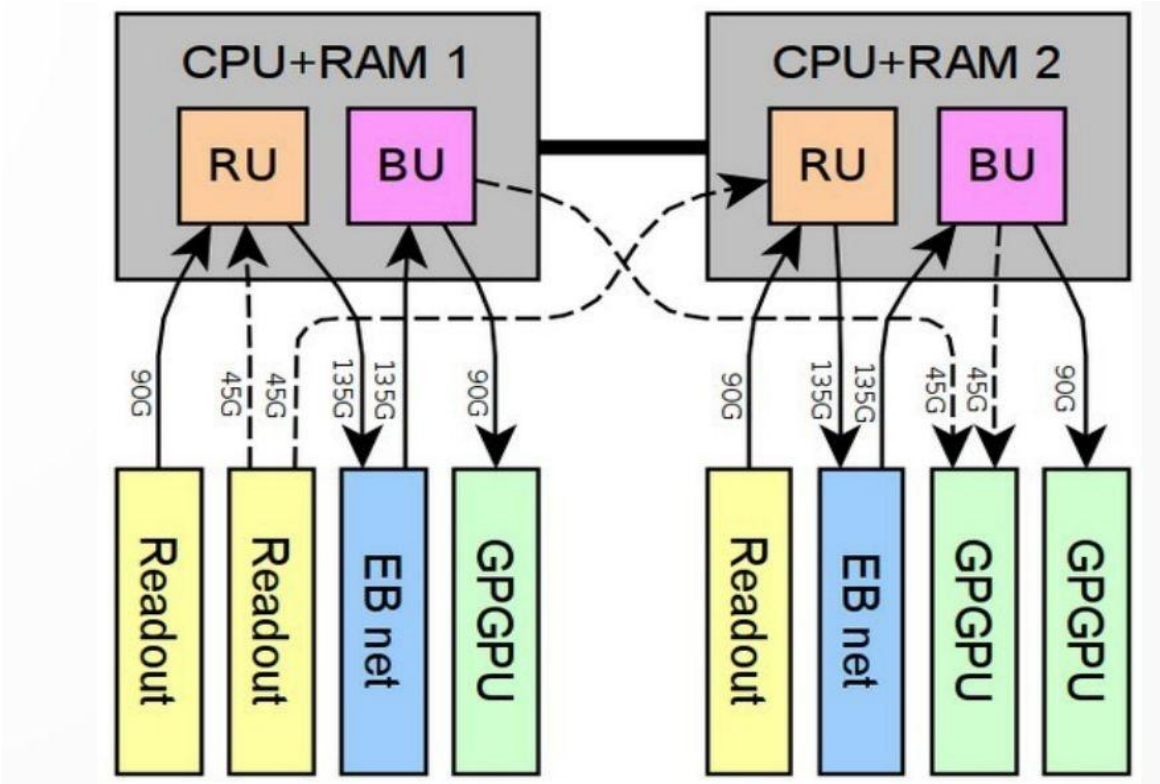
Challenges for the Event Building

- Needs to collect data from ~480 readout boards into a single “destination”
- Then hand over to GPGPUs + CPUs for further processing
- RU + BU + HLT1 running on the same server
- Want high link-load (keeping costs low)
- Two key components: EB server and EB network

Event Builder Server

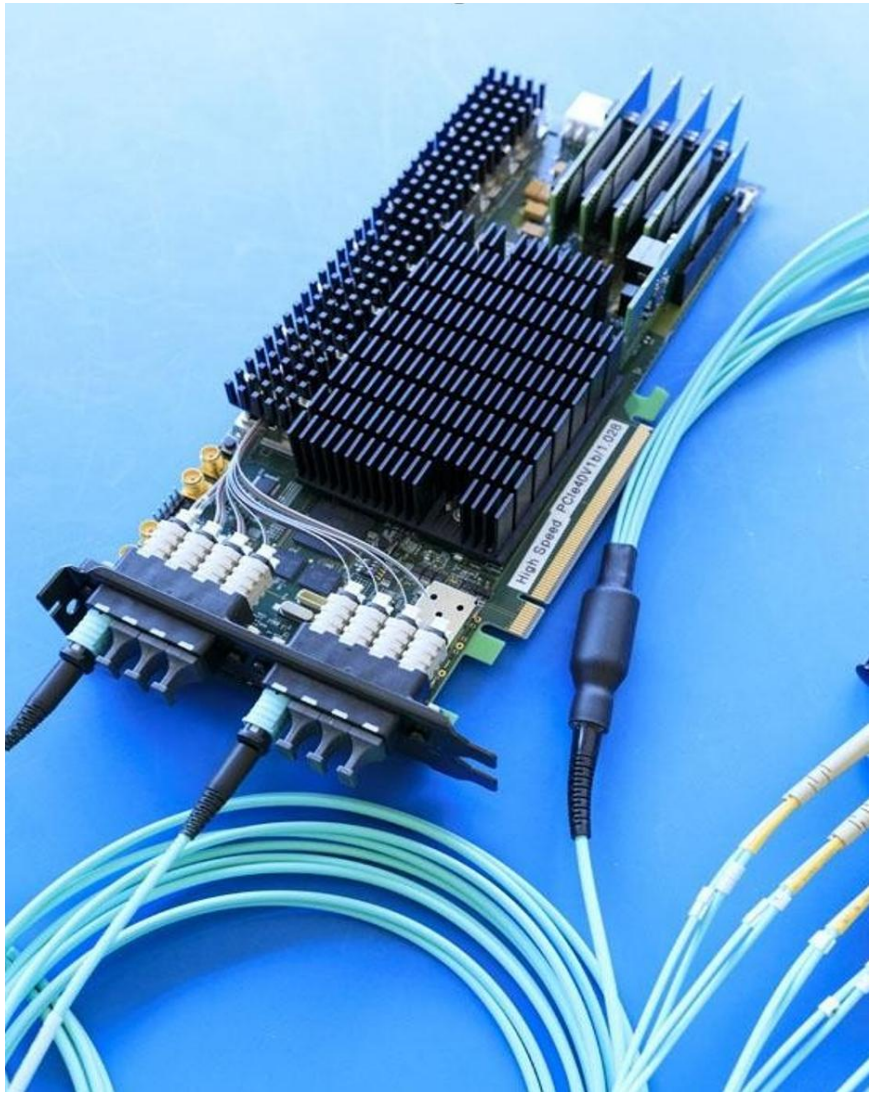


- **CPU:** 2x AMD EPYC 7502, 32 cores
- **RAM:** 512 GB DDR4
- 8x PCIe Gen4 x16 slots :
  - 1-3 GPGPU (RTX A5000)
  - 3 Readout Boards (TELL 40)
  - 2 InfiniBand HDR NICs (200 Gb/s)
- COTS (Commercial Off-The-Shelf) devices widely used
  - Make good use of the PCIe bus bandwidth
  - Remote DMA to reduce CPU-load

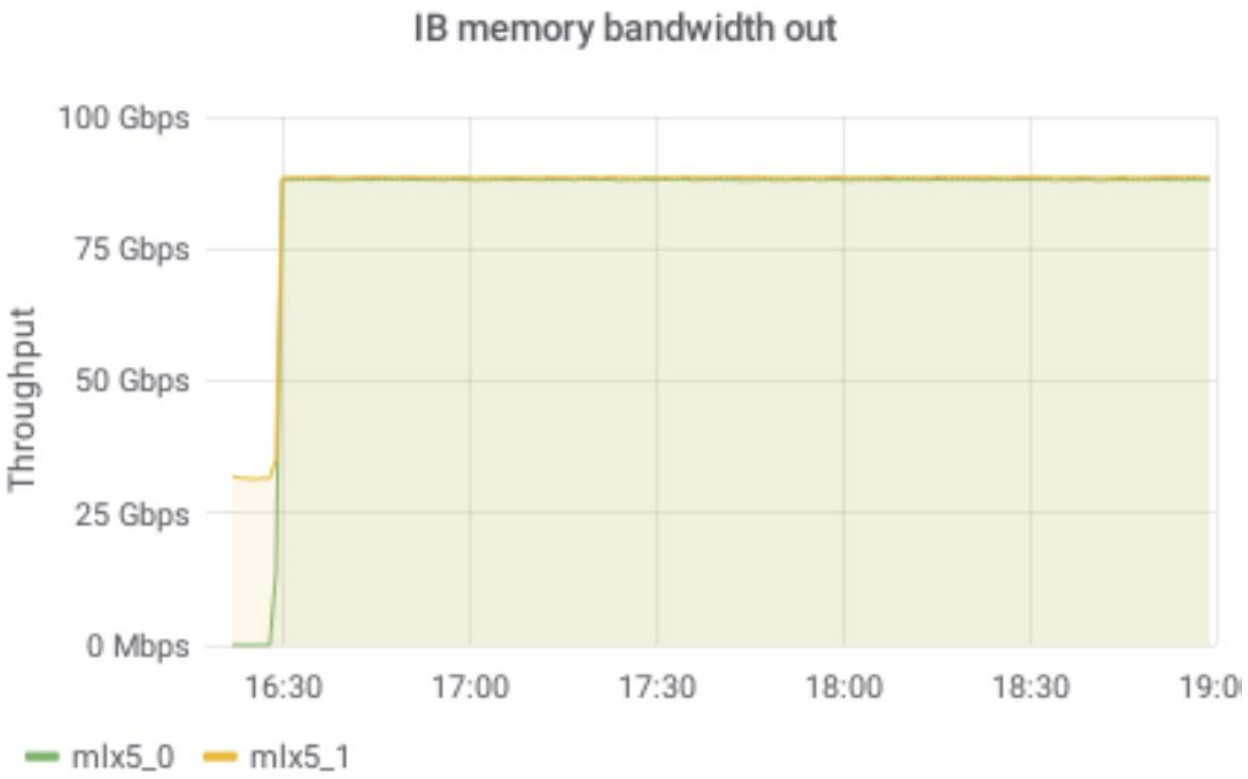


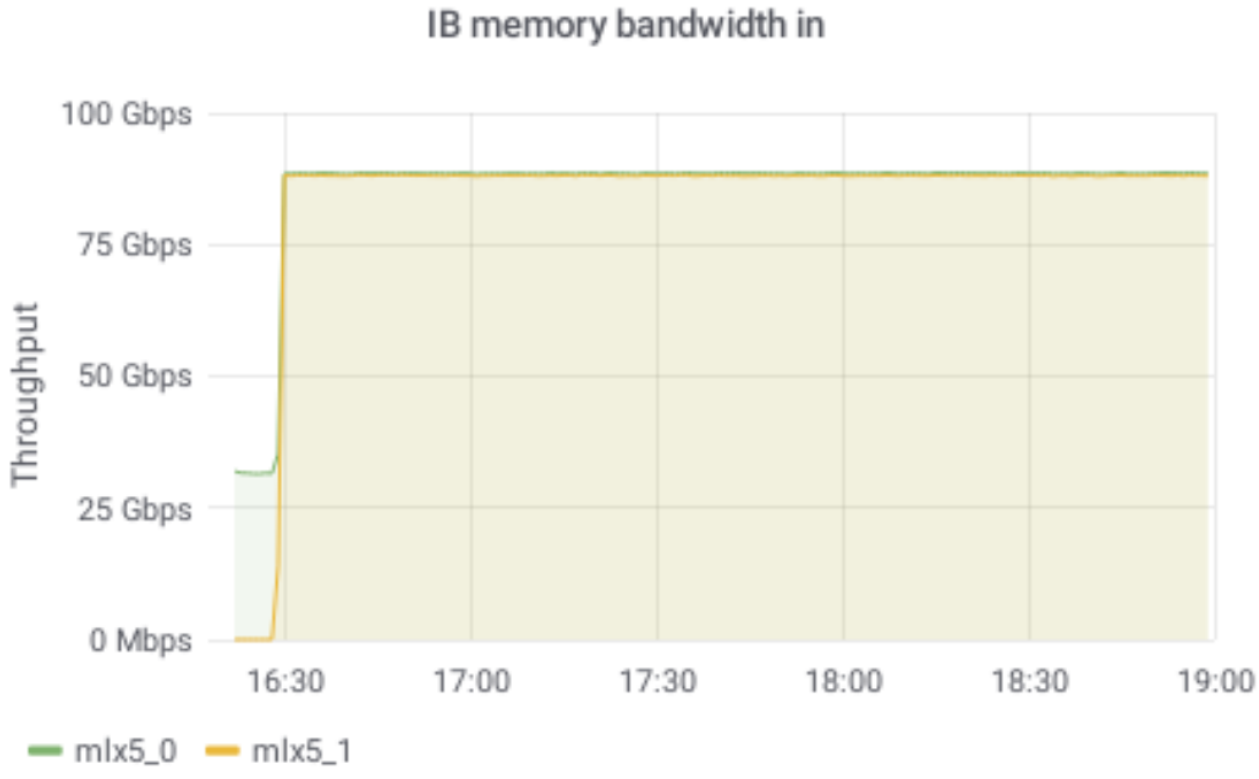
PCIe40 Readout Board

- Moderate FPGA: Intel Arria10
- PCIe Gen3 x16
- 48x10G capable transceiver on 8xMPO for up to 48 full-duplex Versatile Links
- First pre-processing of the data on board



Server Performance

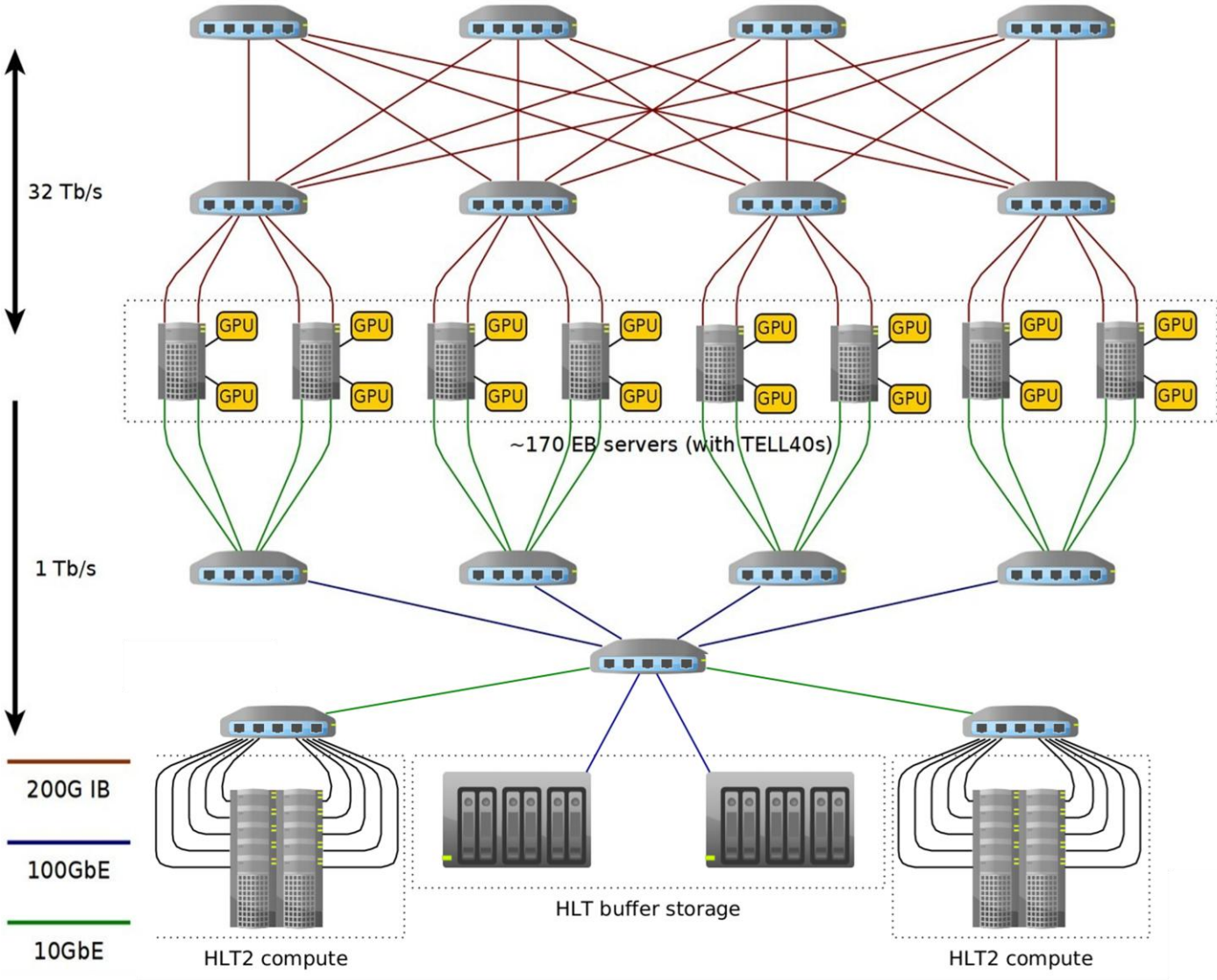




Server： 2X GPUs, 2X InfiniBand EDR 100G cards and 2X TELL40 cards.
(*Test done during R&D with InfiniBand EDR network)

Event builder networks

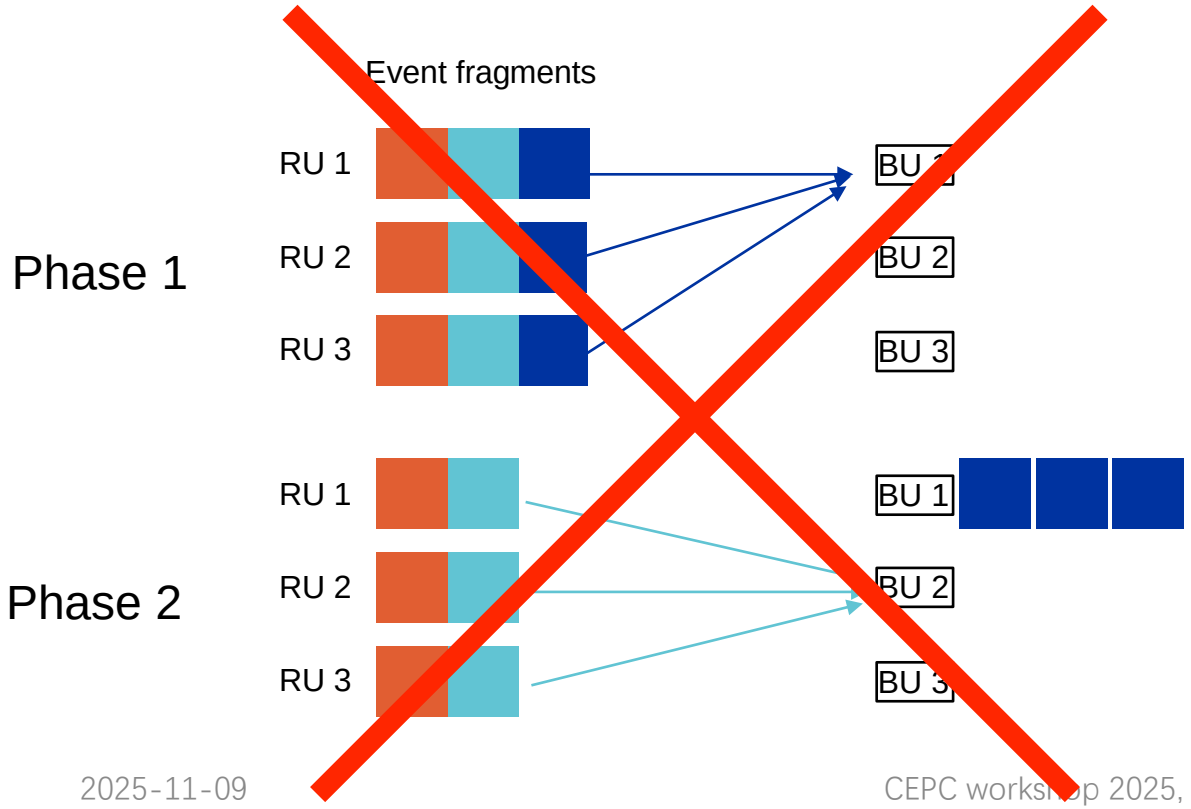
- Dedicated EB network:
  - InfiniBand 200G Fat tree
  - 28 switches 360 ports
  - Bi-directional traffic
- Storage / filter network:
  - Ethernet 10/100G

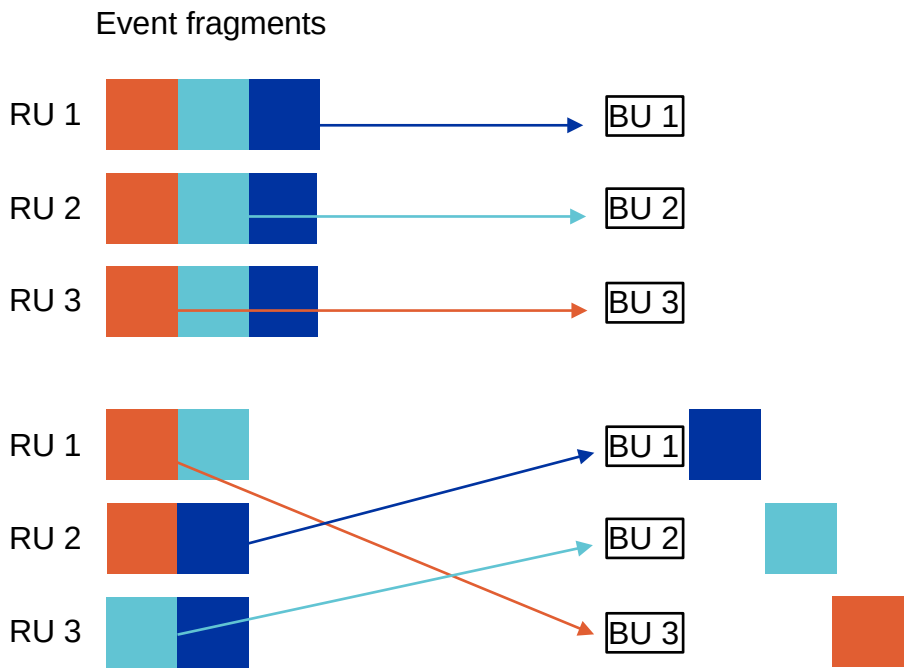


Event builder networks

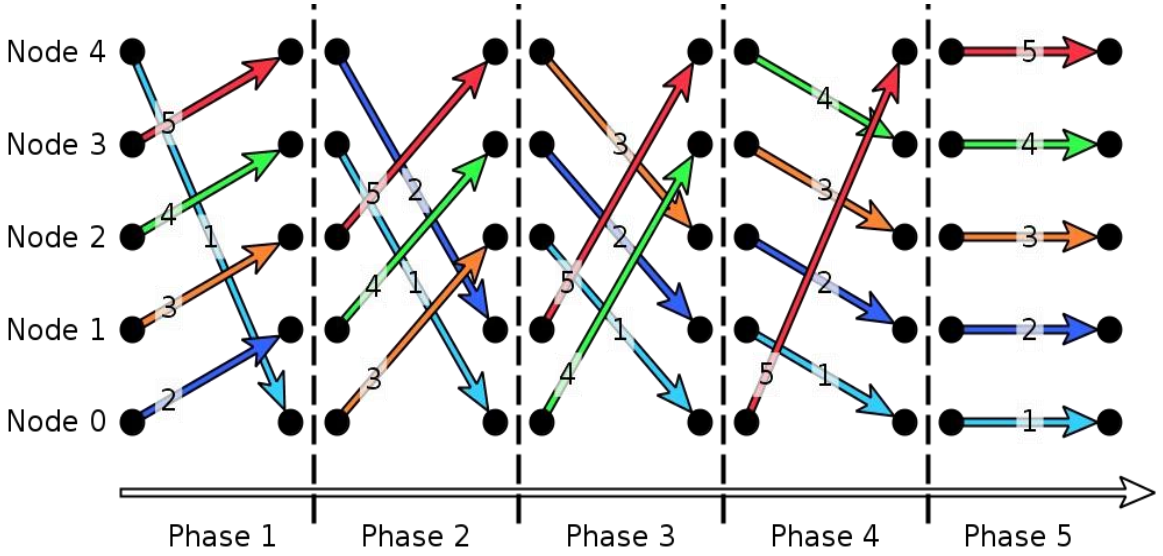
How to avoid traffic congestion? Two possible options:

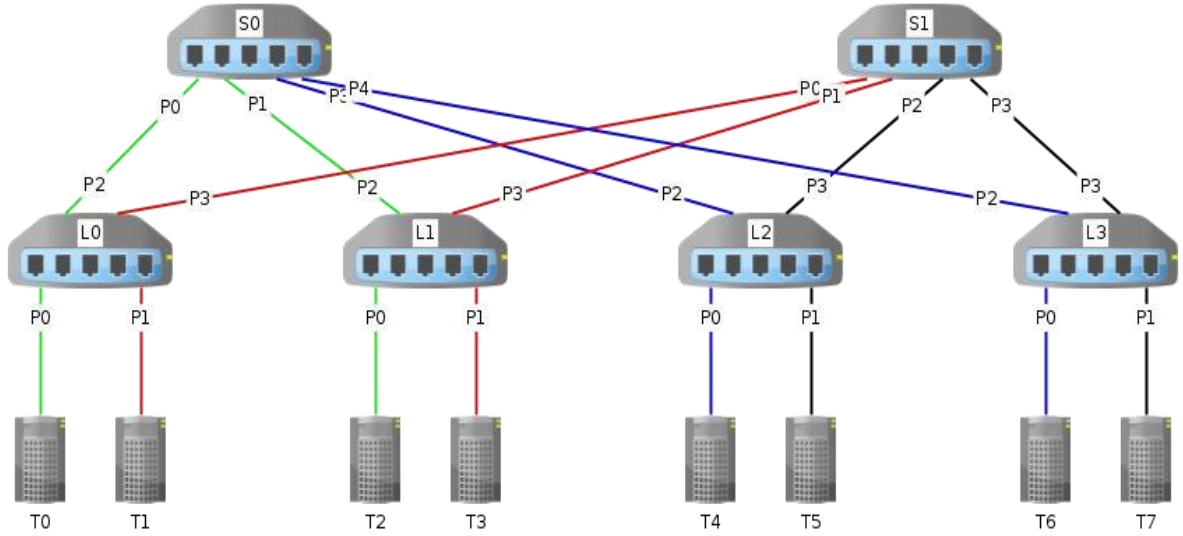
- Deep buffer in network switches
- Traffic shaping





Event Builder: Traffic scheduling



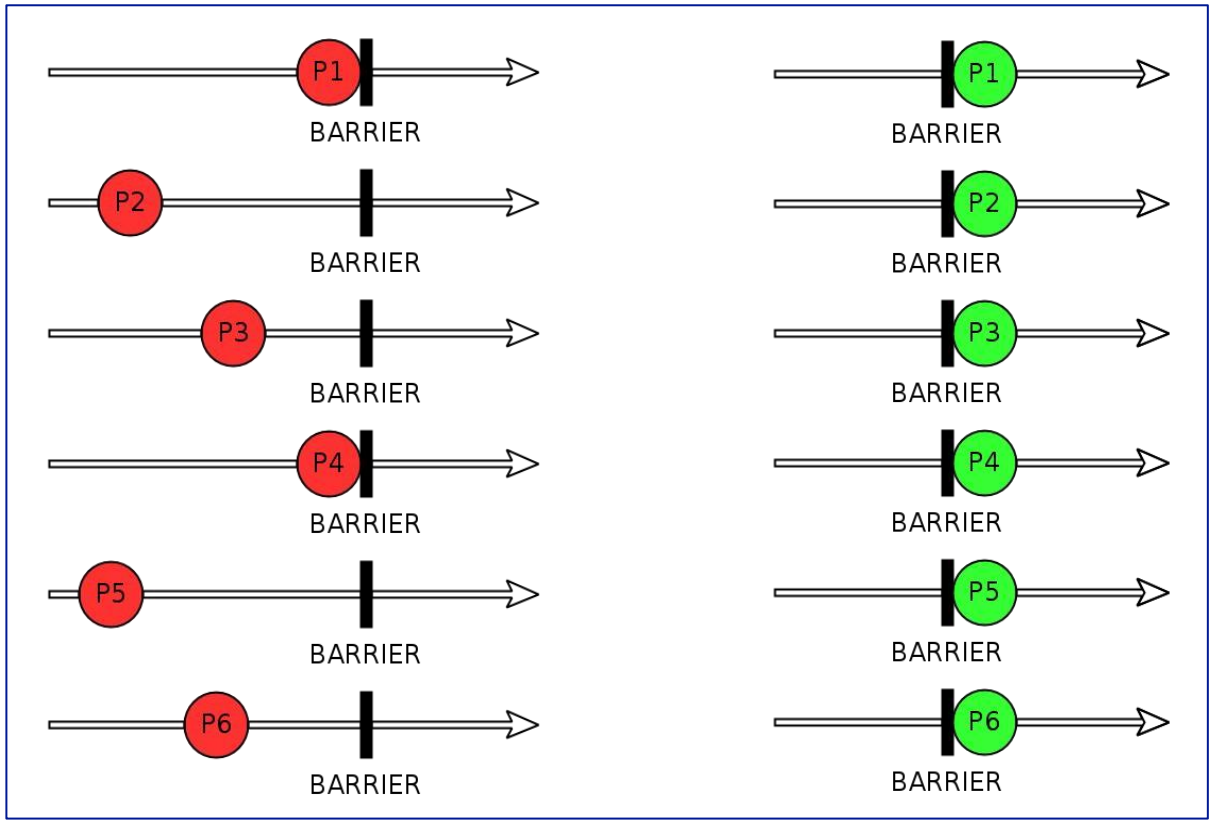


- The processing of N events is divided into N phases (N is the number of EB nodes)
- In every phase one RU sends data to one BU, and every BU receives data from one RU
- During phase n RU x sends data to BU (x + n)%N

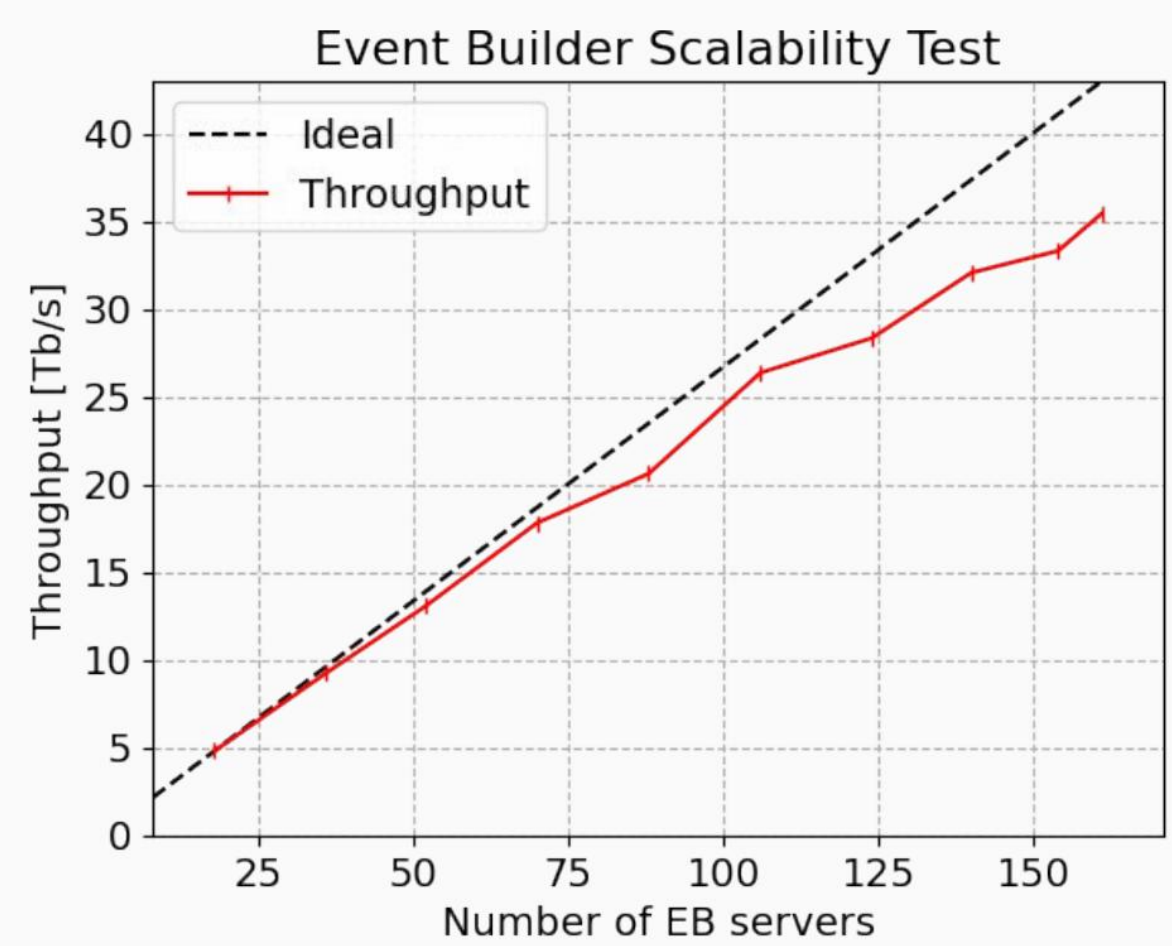
Congestion-free traffic on “selected networks” (e.g. fat-tree networks)

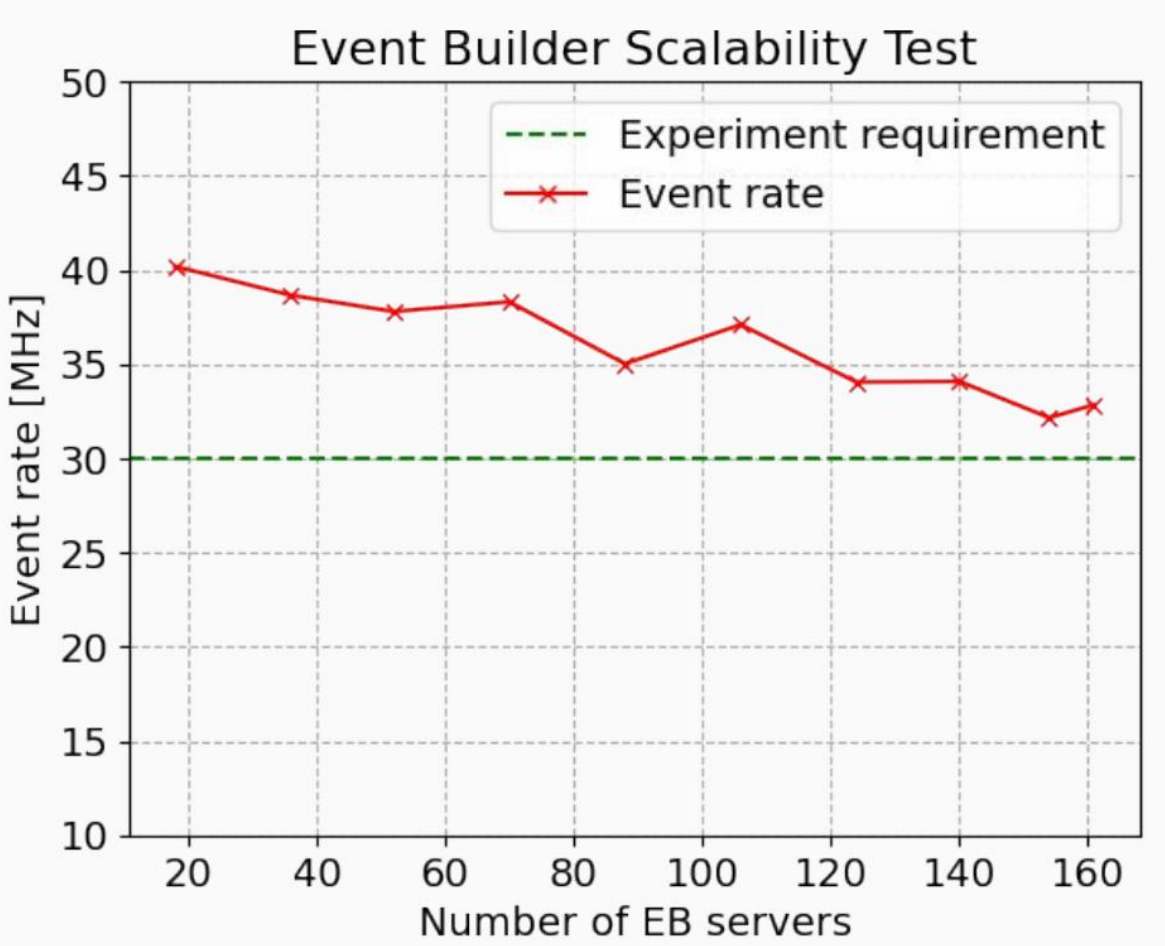
Synchronization

- Strong synchronization between all nodes is done at every step
- Synchronization Barrier:
  - Processes report they reached the barrier and wait for release
  - When all processes have reached the barrier, they are released.
- Multiple barrier algorithms are implemented in the communication library.



System scalability

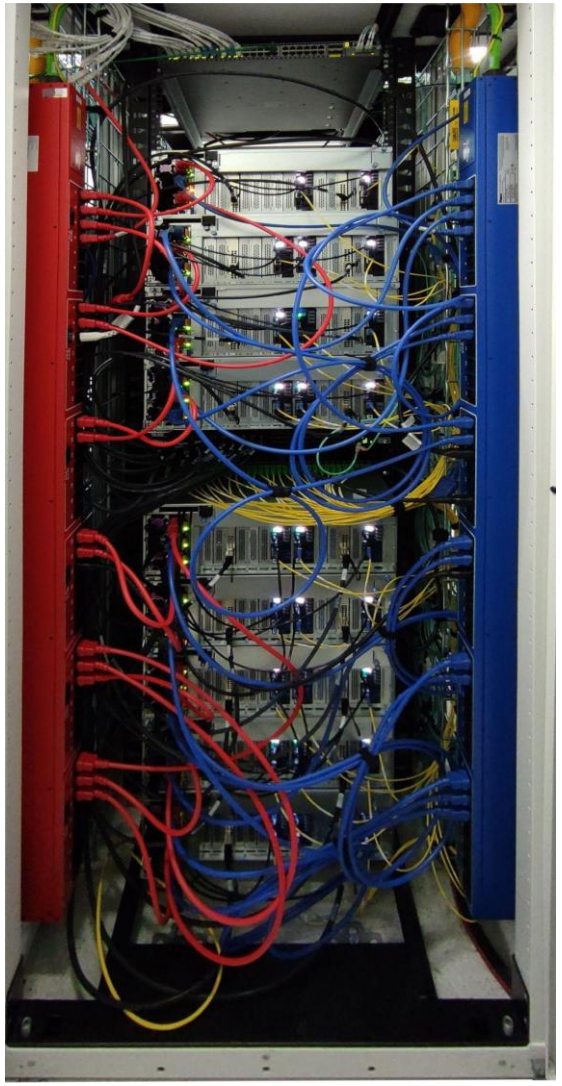


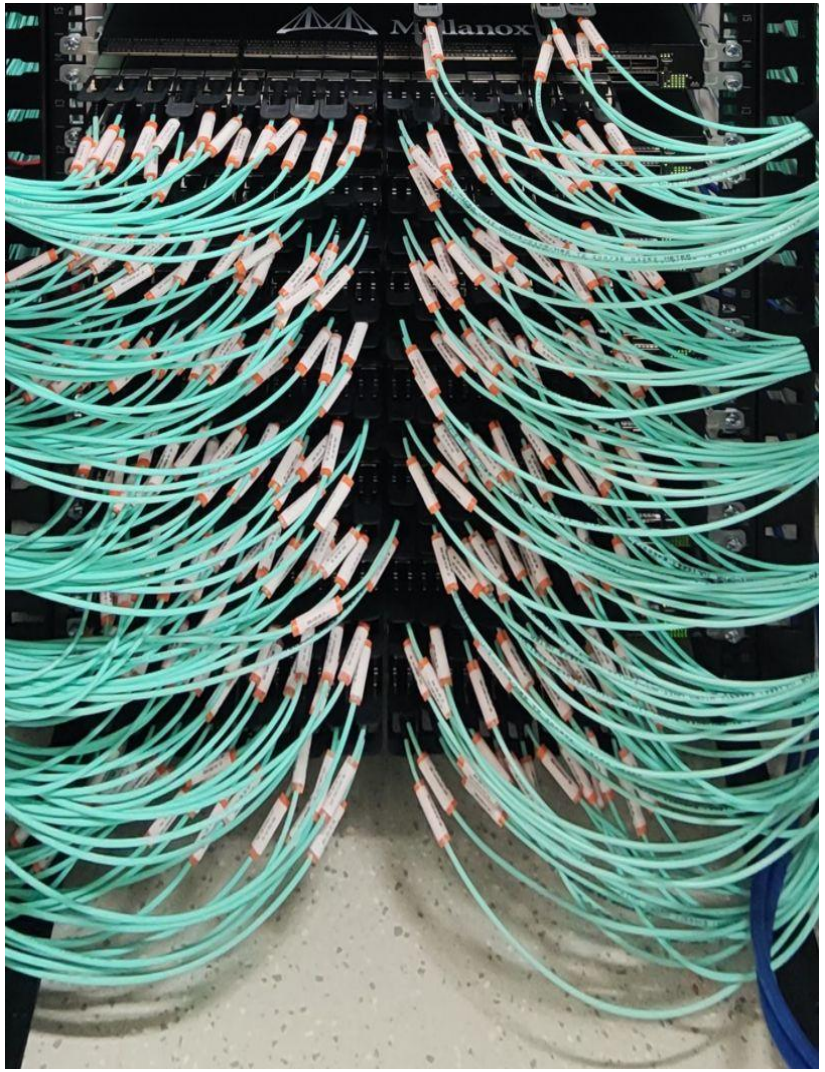


[One year of LHCb triggerless DAQ]

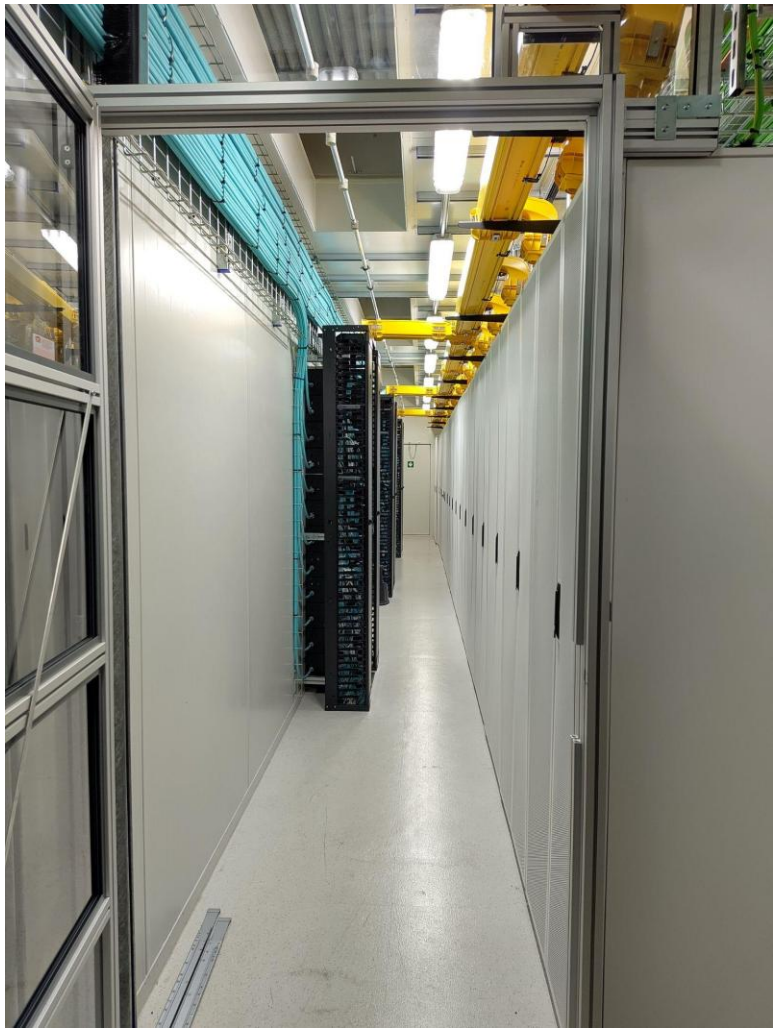
Even Builder - Rack

- 164 EB servers
- 24x racks: 18 EB, 2 control, 4 storage
- 28x 40-port IB HDR switches: 18 leaf and 10 spine





Even Builder Datacenter



Allen project: HLT1 on GPU

- Framework for GPU-based execution of an algorithm sequence

https://gitlab.cern.ch/lhcb/Allen

- Cross-architecture compatibility:

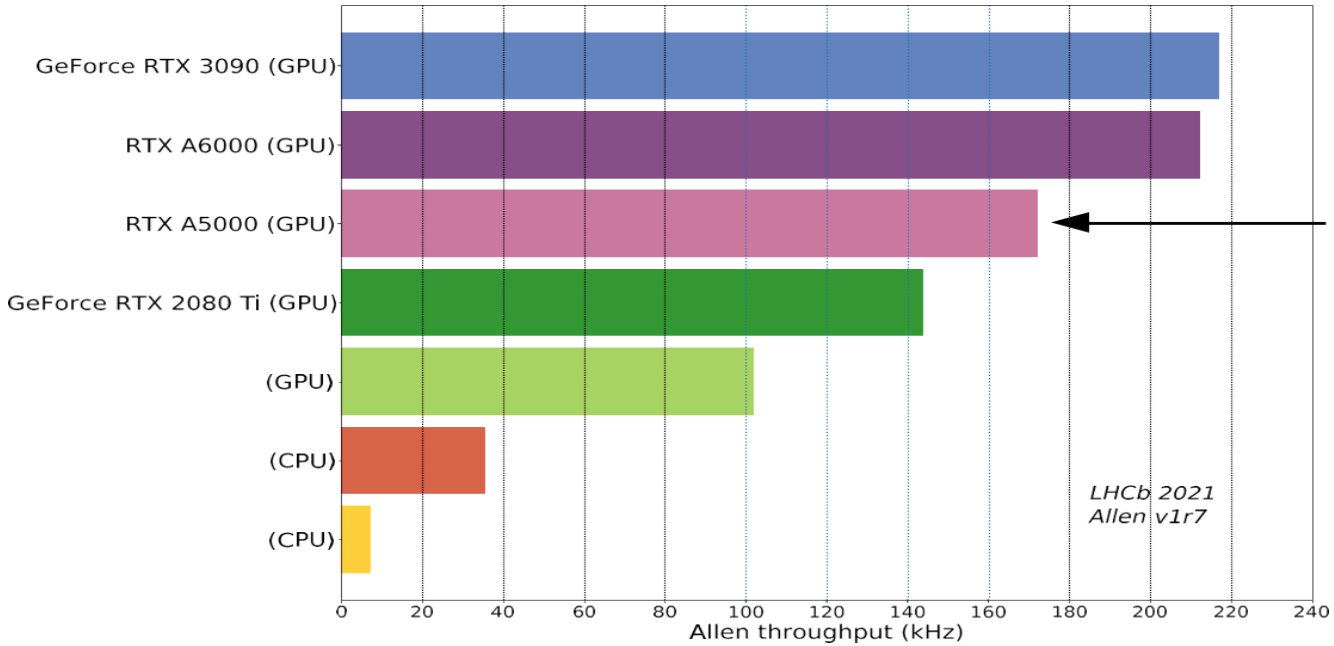
  CPU, NVidia GPU (CUDA), AMD GPU (HIP)

- Thousands of events are processed in parallel

[See Miro's talk later for more details]

HLT1 computing throughput

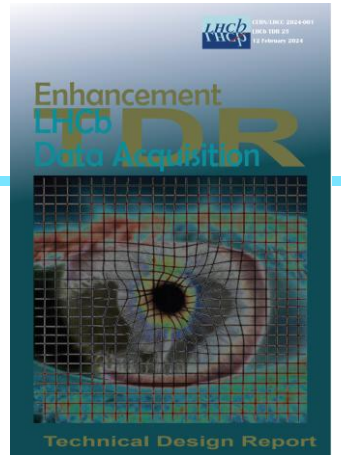
- Putting GPUs in EB free PCIe slots to reduce cost
- 30 MHz goal can be achieved with ~340 GPUs (maximum the Event Builder server can host is 500)
- Throughput scales well with theoretical TFLOPS of GPU card



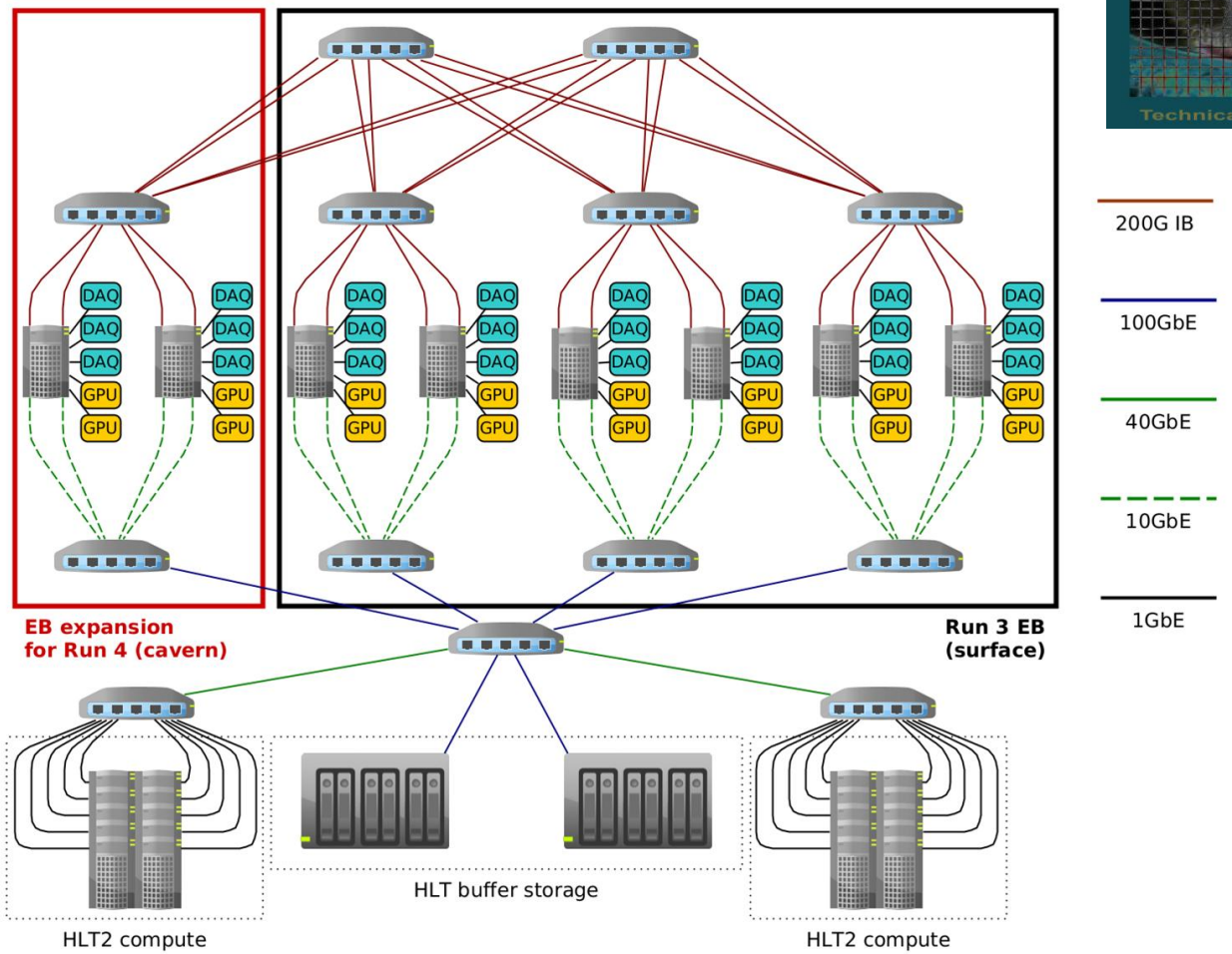
Chose RTX A5000 for the beginning of Run3

[LHCb-FIGURE-2020-014]

DAQ in Run4



- Basic architecture unchanged but more (lpGBT) links
- Upgraded RICH will use lpGBT
- lpGBT cannot reach the surface data center
  - ➢ EB expansion in cavern

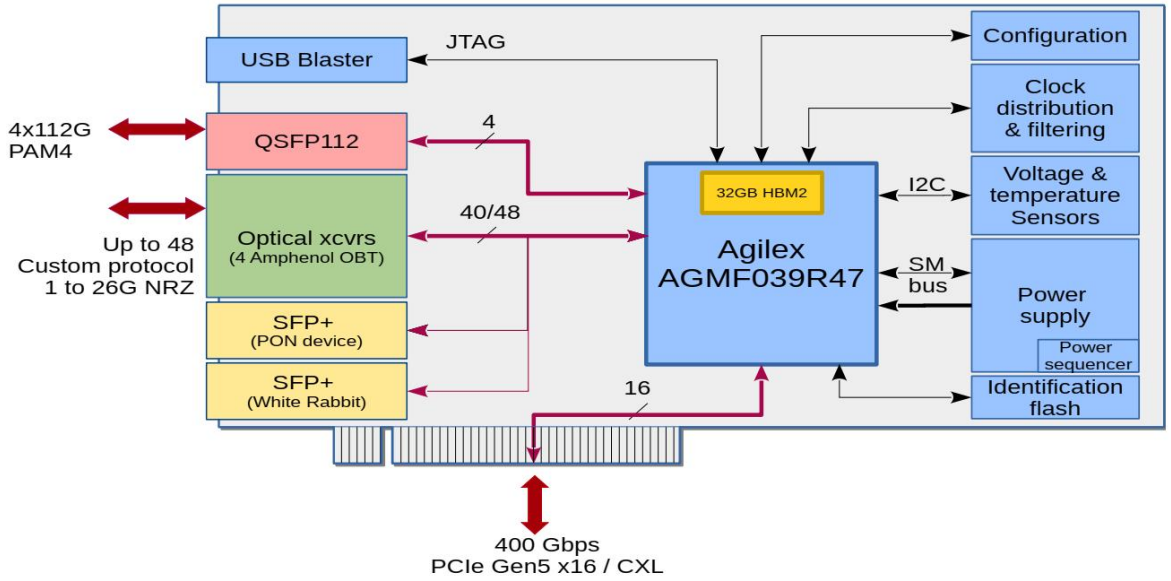


DAQ in Run4

New backend readout board PCIe400

- Agilex 7 M-series with 32 GB HBM and 4-core ARM
- PCIe Gen5 (×4 bandwidth w.r.t. PCIe40)
- up to 48 lpGBT links
- SFPs for timing distribution,

  400 GbE QSPF112



J. Langouët, Oct 2025

Summary

- The LHCb TDAQ system has been running reliably with the load of 32 Tb/s in Run 3
  - Relies heavily on COTS hardware
  - Flexibility and scalability
  - Very cost-efficient solution：~ 2 MCHF for the EB servers, ~1.5 MCHF for the network + NICs, ~1 MCHF for the GPGPUs, ~1 MCHF for the buffer storage
- The overall architecture keeps unchanged for Run 4

Thanks for your attention!