Commissioning LHCb's GPU high level trigger

Christina Agapopoulou on behalf of the LHCb collaboration

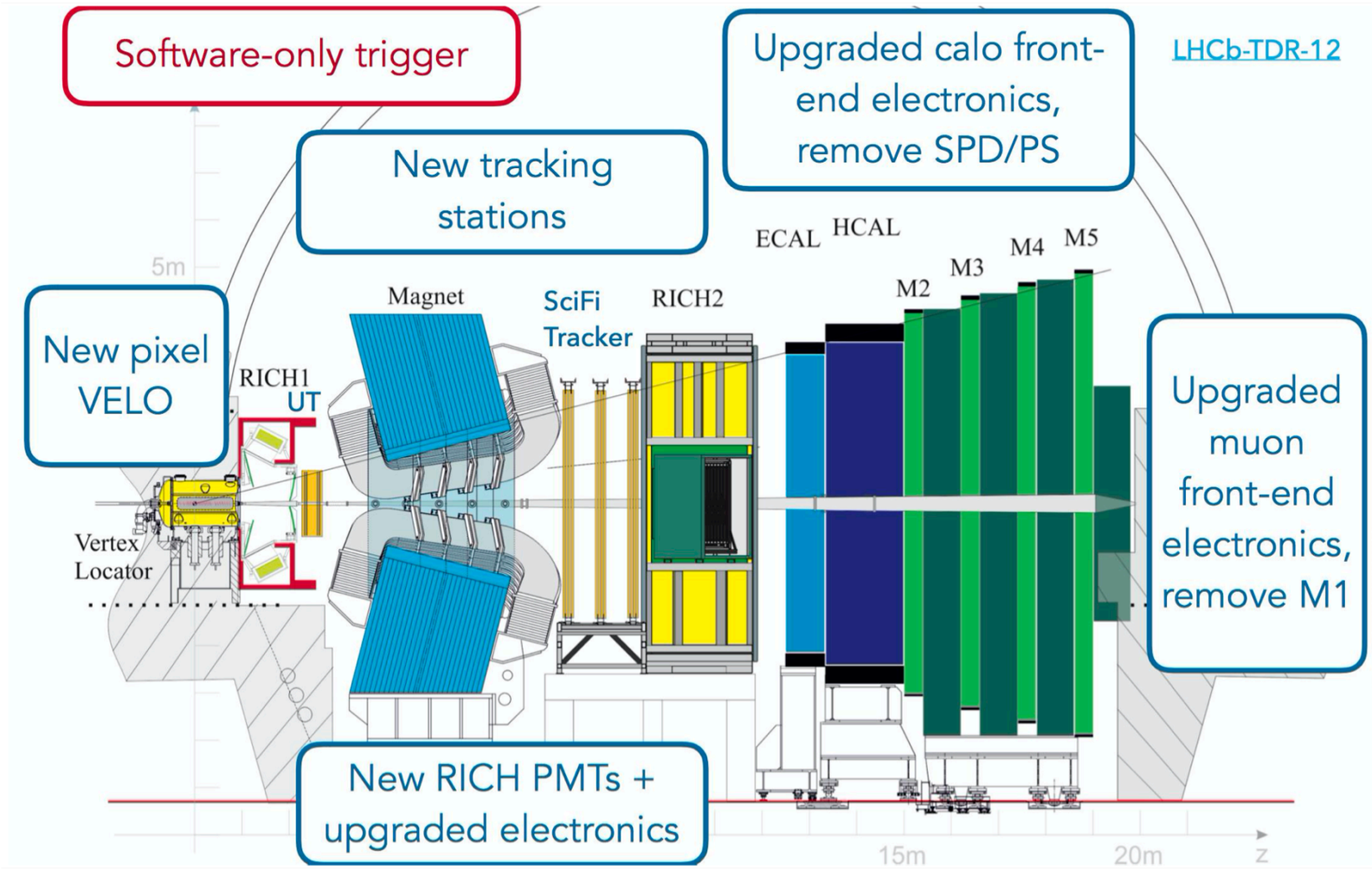
20th International Workshop on Advanced Computing and Analysis Techniques in Physics Research

30/11/2021

European Research Council

Established by the European Commission

The LHCb U1 upgrade



Will start in 2022!

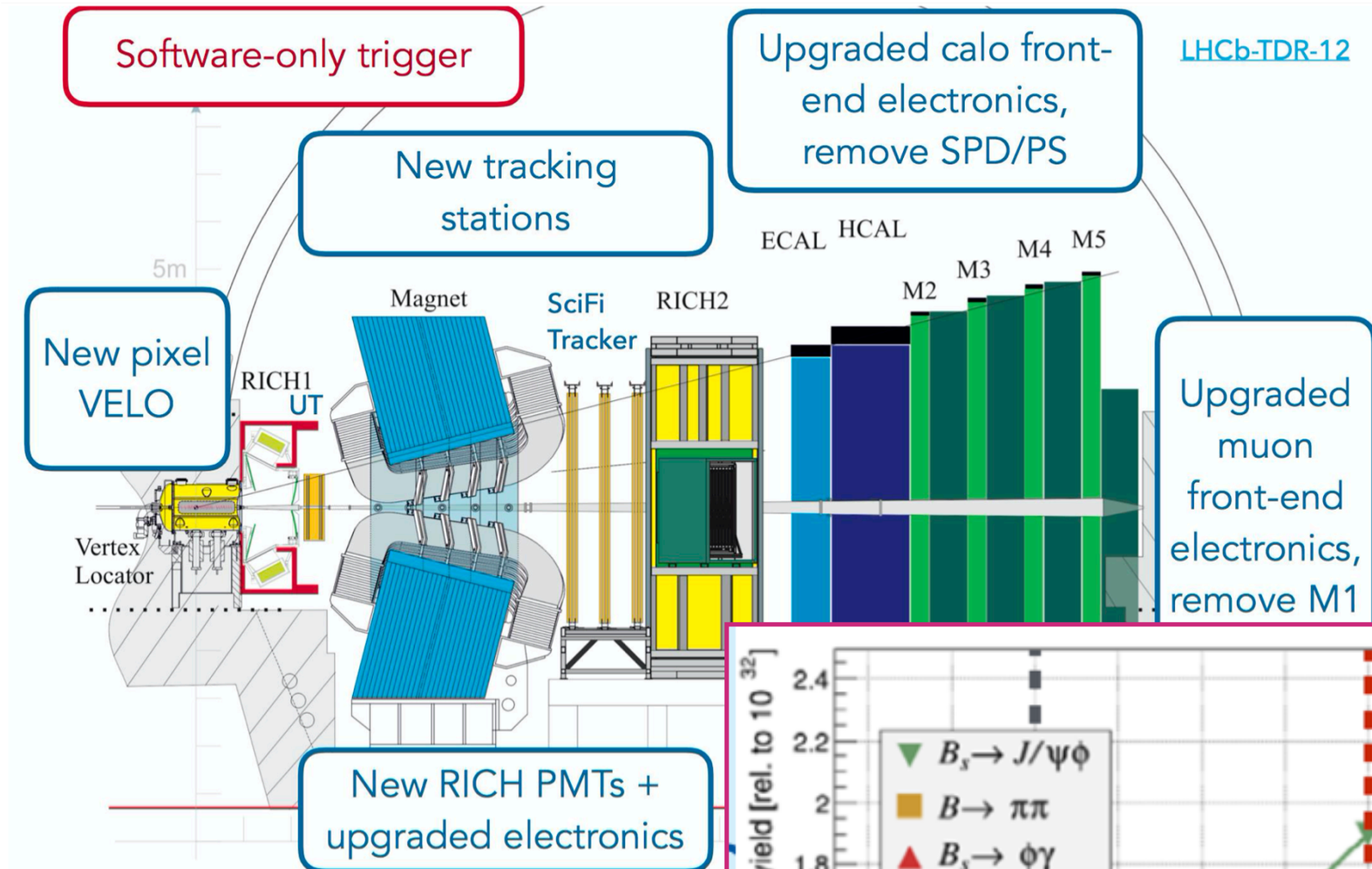
The LHCb detector at CERN:

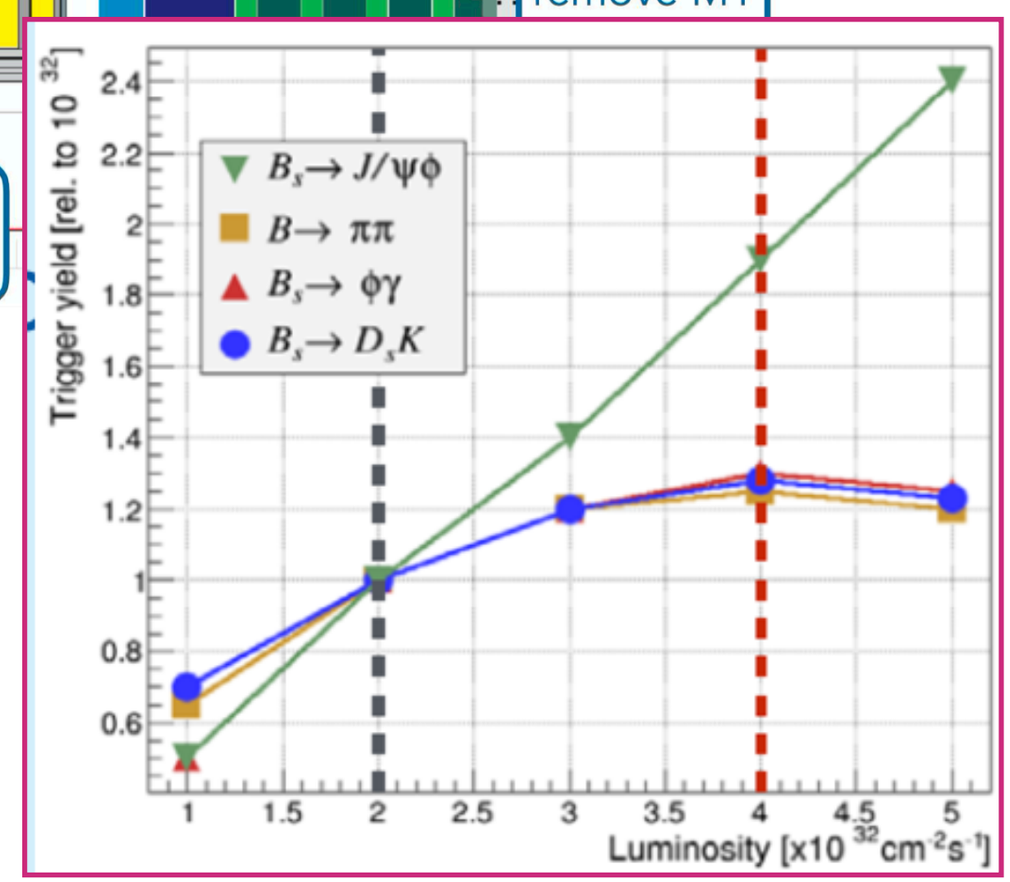
- Single-arm forward spectrometer for high-precision flavour physics
- High precision tracking and vertexing
- Complemented with excellent PID

The U1 upgrade

- Instantaneous luminosity will increase by x5
- Major upgrade in all sub-detectors to handle increased rates
- Software-only trigger!

The LHCb U1 upgrade

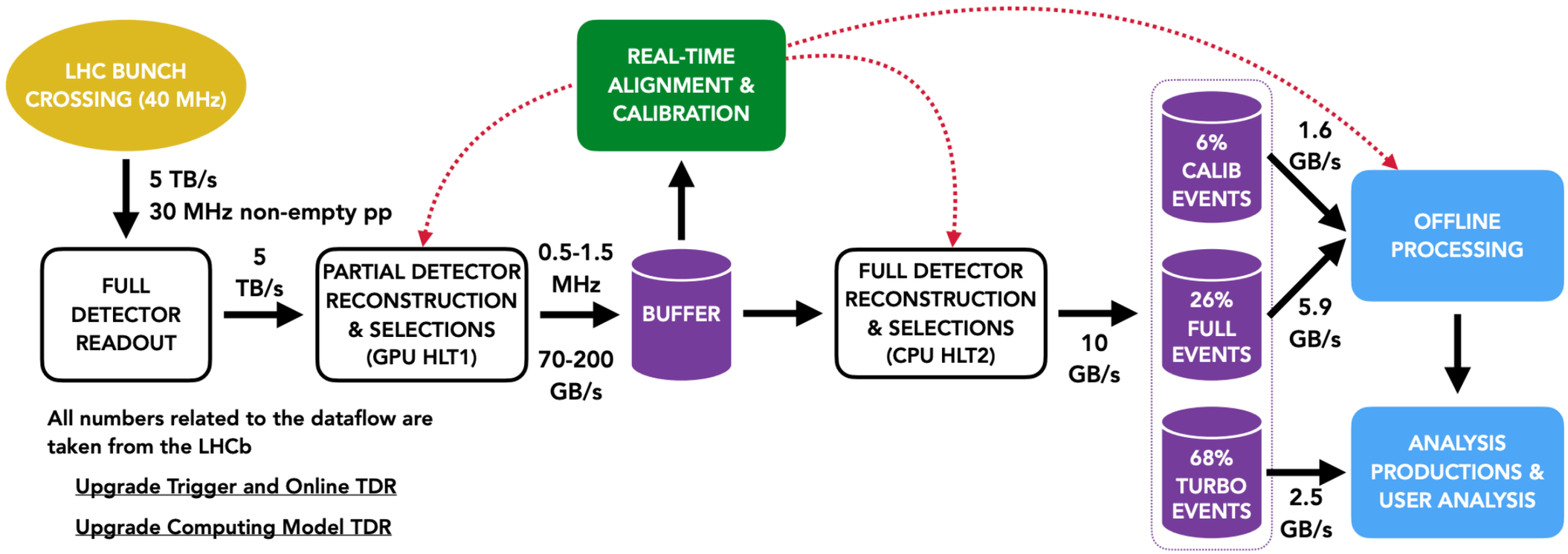




In Run 3 LHCb will move to triggerless read-out!

- Why: L0 rate limit would saturate fully hadronic modes
- Solution: full detector read-out at 30 MHz
- Fully software trigger based on track and vertex reconstruction !

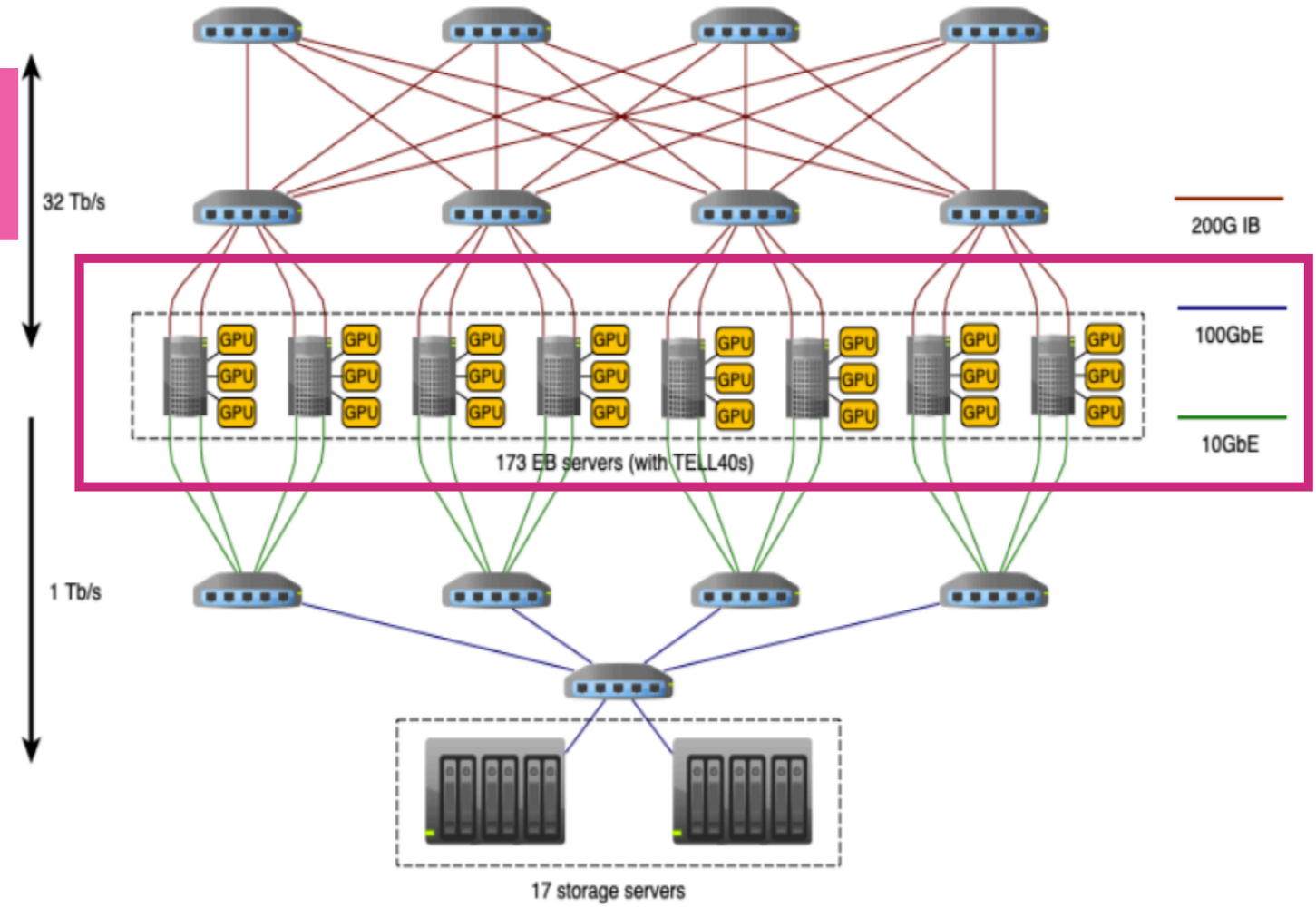
LHCb data-flow in Run 3



- Detector data received by O(500) FPGAs and built into events in the event building (EB) farm severs
- 2-stage software trigger:
  - HLT1: partial event reconstruction and coarse selection
  - HLT2: full event reconstruction and O(1000) selection lines
  - Buffering between HLT1 & HLT2 → real-time alignment & calibration
- After HLT2, 10 GB/s of data for offline processing

Are GPUs a good fit?

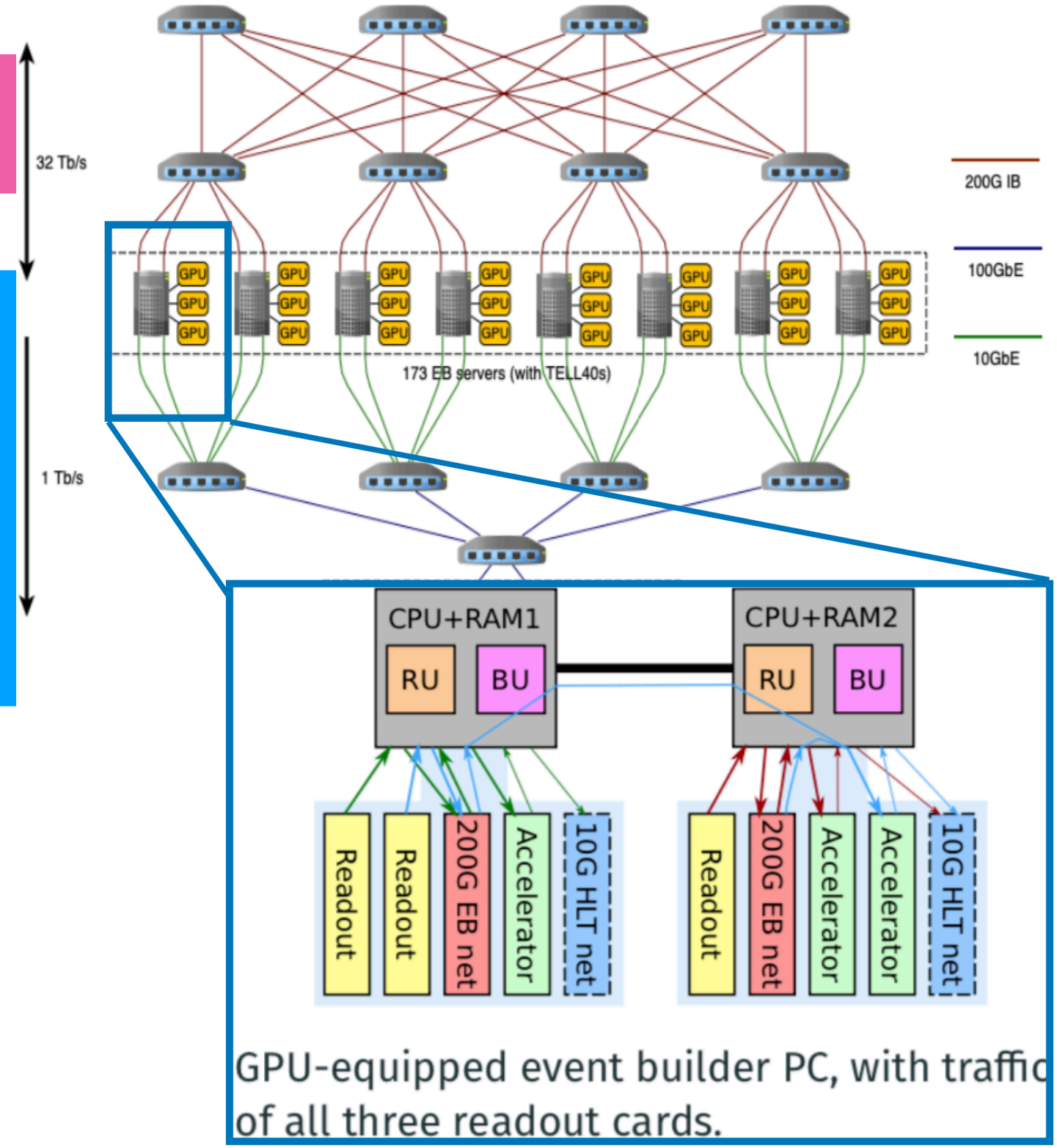
- Event builder farm equipped with 173 servers



Are GPUs a good fit?

- Event builder farm equipped with 173 servers

- Each server has 3 free PCIe slots
  - Can be used to host GPUs
  - Sufficient cooling & power
  - Advantageous to have GPUs as self-contained processors
  - Sending data to GPU is like sending data to network card



Are GPUs a good fit?

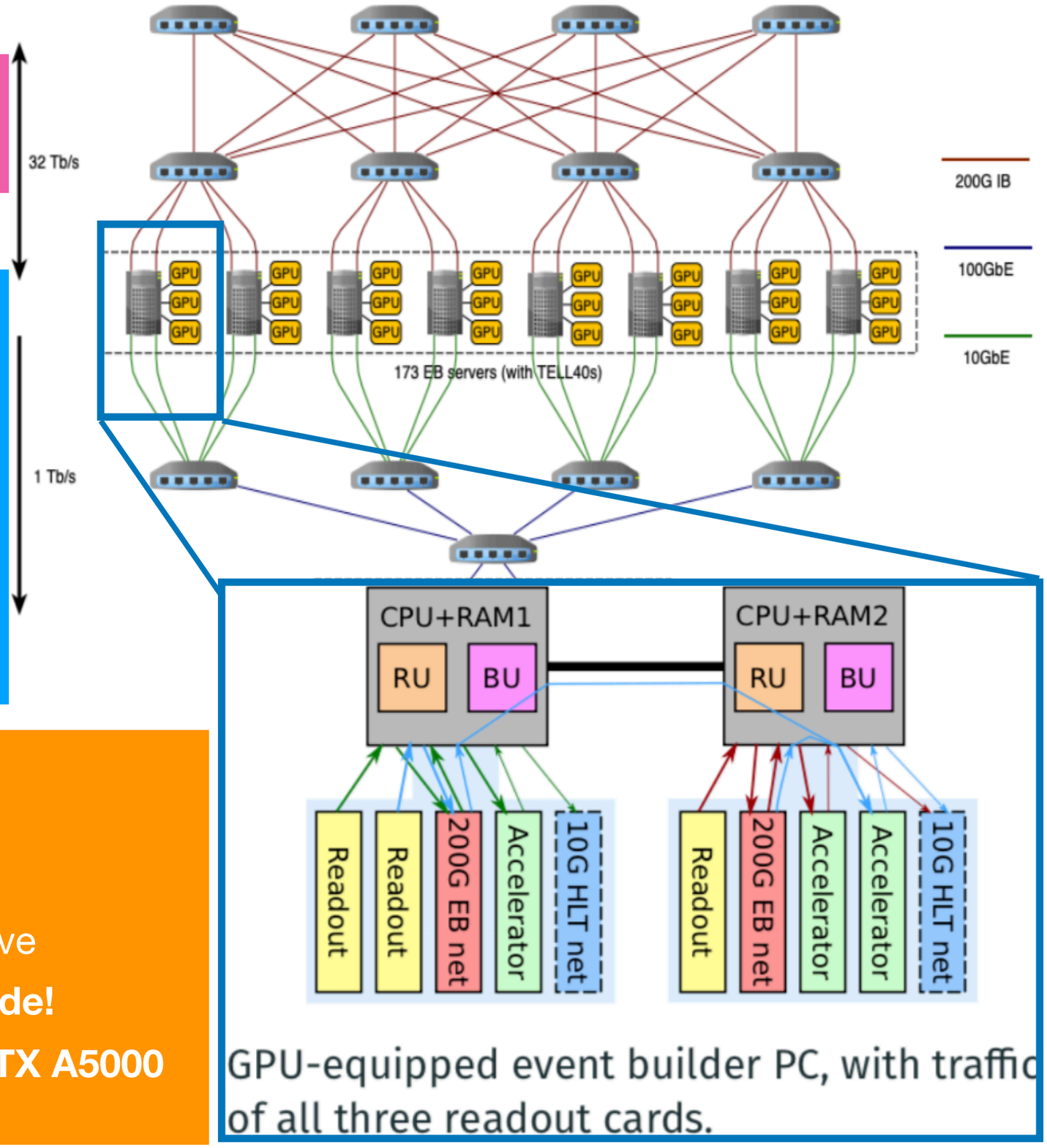
- Event builder farm equipped with 173 servers

- Each server has 3 free PCIe slots
  - Can be used to host GPUs
  - Sufficient cooling & power
  - Advantageous to have GPUs as self-contained processors
  - Sending data to GPU is like sending data to network card

- GPUs map well into LHCb DAQ architecture
- HLT1 tasks inherently parallelizable
- Smaller network between EB & CPU HLT
- Cheaper & more scalable than CPU alternative

➡**Was chosen as the baseline for the upgrade!**

➡**Will be implemented with O(200) Nvidia RTX A5000 GPUs**



GPU-equipped event builder PC, with traffic of all three readout cards.

Allen: a GPU HLT1 trigger platform

- Public software project: [gitlab repo]
- Supports three modes:
  - Standalone
  - Compiling within the LHCb framework for data acquisition
  - Compiling within the LHCb framework for simulation and offline studies
- Runs on CPU, Nvidia GPU (CUDA, CUDACLANG), AMD GPUs (HIP)
- GPU code written in CUDA
- Cross-architecture compatibility (HIP, CPU) via macros

Documentation in progress

Links to more readmes

The following readmes explain various aspects of Allen:

- This readme explains how to add a new algorithm to Allen.
- This readme explains how to add a new HLT1 line to Allen.
- This readme explains how to configure the algorithms in an HLT1 sequence.
- This readme explains how to call Allen from Moore.
- Building and running inside Docker.
- Documentation on how to create contracts.
- Produce MDF input files for standalone Allen

readme.md

Allen

pipeline passed

Welcome to Allen, a project providing a full HLT1 realization on GPU.

Requisites

The following packages are required in order to be able to compile Allen. Package names listed here are CentOS 7 names, package names for other distributions may slightly change:

- cmake version 3.18 or newer
- boost-devel version 1.69 or newer
- clang version 9 or newer
- json-devel
- zeromq-devel
- zlib-devel
- python3

The following python3 packages are also needed, which can be installed with pip, conda, or the package manager:

- wrapt
- cachetools
- pydot
- sympy

Further requirements depend on the device chosen as target. For each target, we show a proposed development setup with CVMFS and CentOS 7:

- CPU target: Any modern compiler can be used:

```
source /cvmfs/sft.cern.ch/lcg/views/setupViews.sh LCG_101 x86_64-centos7-clang12-opt
```

- CUDA target: The latest supported compilers are gcc-10 and clang-10. CUDA is available in cvmfs as well:

```
source /cvmfs/sft.cern.ch/lcg/views/setupViews.sh LCG_101 x86_64-centos7-clang12-opt
source /cvmfs/sft.cern.ch/lcg/contrib/cuda/11.4/x86_64-centos7/setup.sh
```

- HIP target: Either a local installation of ROCm or CVMFS are required:

```
source /cvmfs/sft.cern.ch/lcg/views/setupViews.sh LCG_101 x86_64-centos7-clang12-opt
source /cvmfs/lhcbdev.cern.ch/tools/rocm-4.2.0/setenv.sh
```

Optionally the project can be compiled with ROOT. Histograms of reconstructible and reconstructed tracks are then filled in the track checker. For more details on how to use them to produce plots of efficiencies, momentum resolution etc. see this readme.

HLT1 sequence

Reconstruction

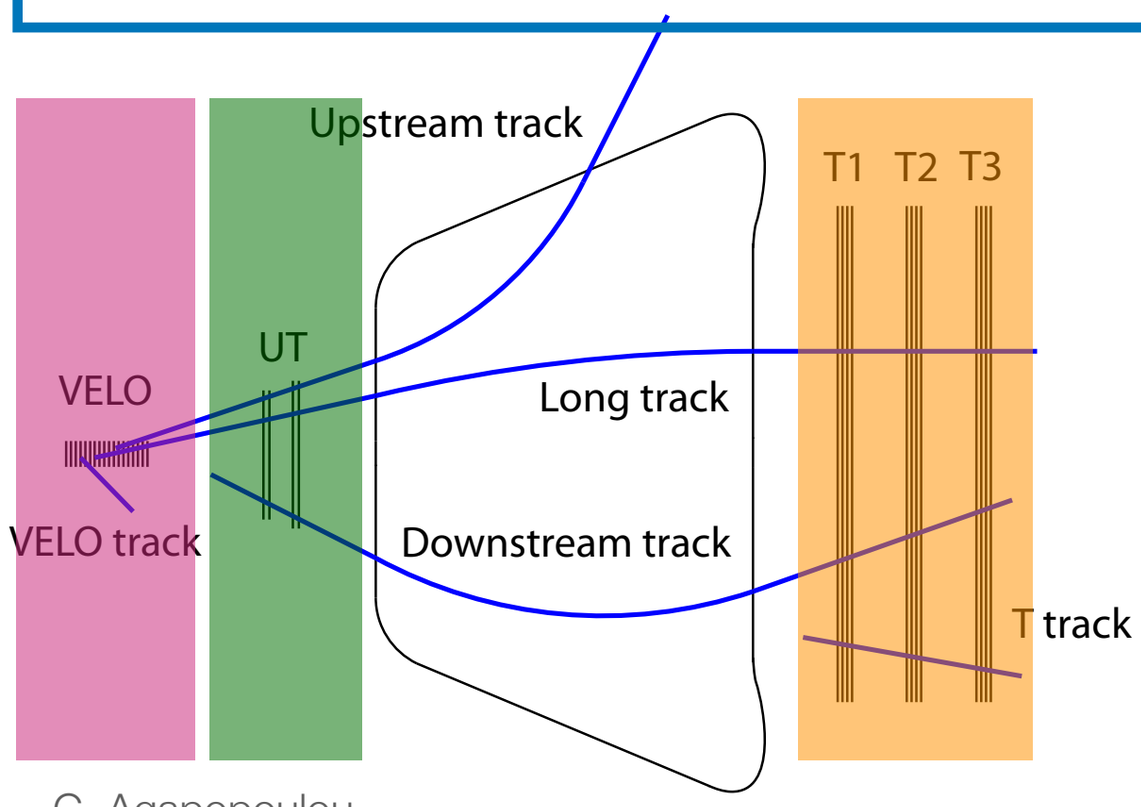
Tracking at the core of the HLT1 reconstruction

Relies on 3 sub-detector systems:

- VELO: clustering, tracking, vertex reconstruction
- UT: track reconstruction, momentum resolution, fake rejection
- SciFi: track reconstruction, momentum measurement

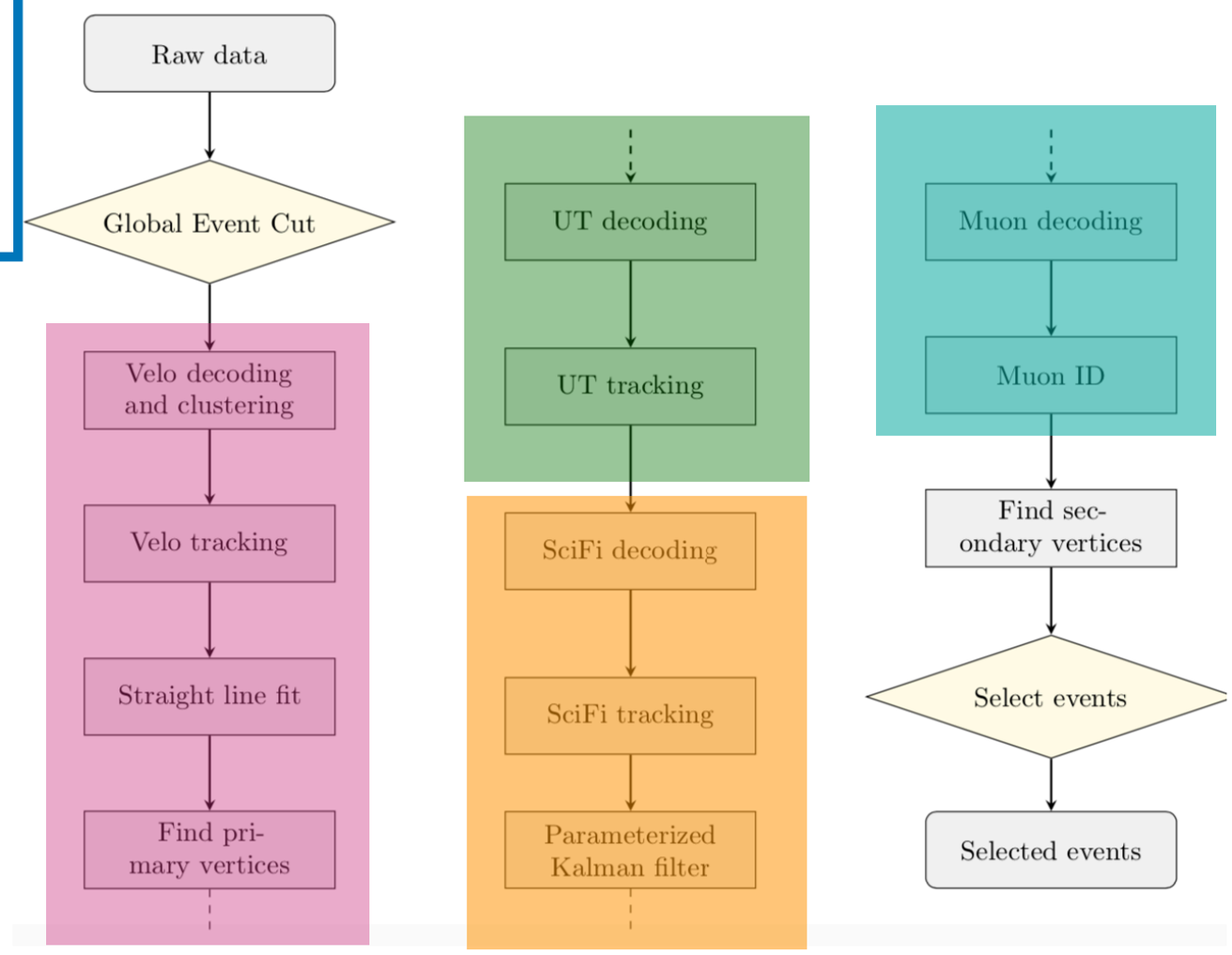
+

Muon PID from muon stations



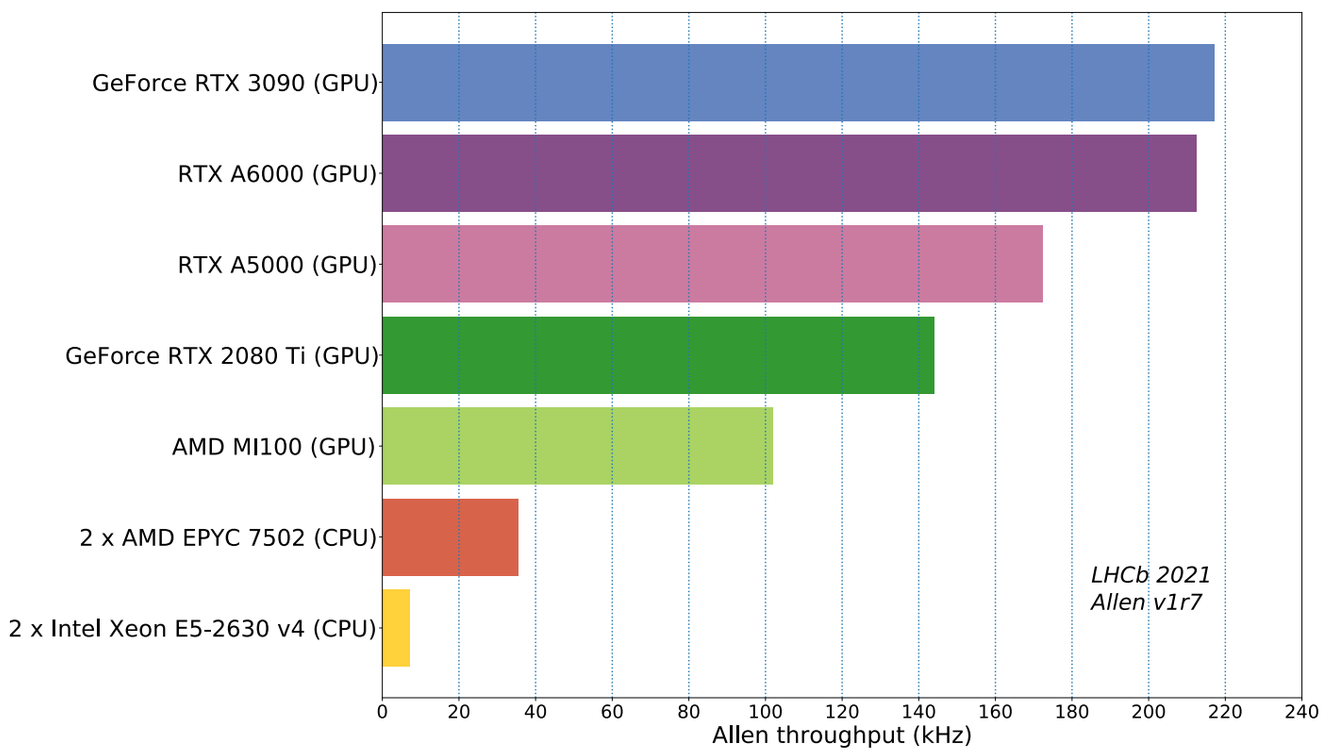
Trigger lines

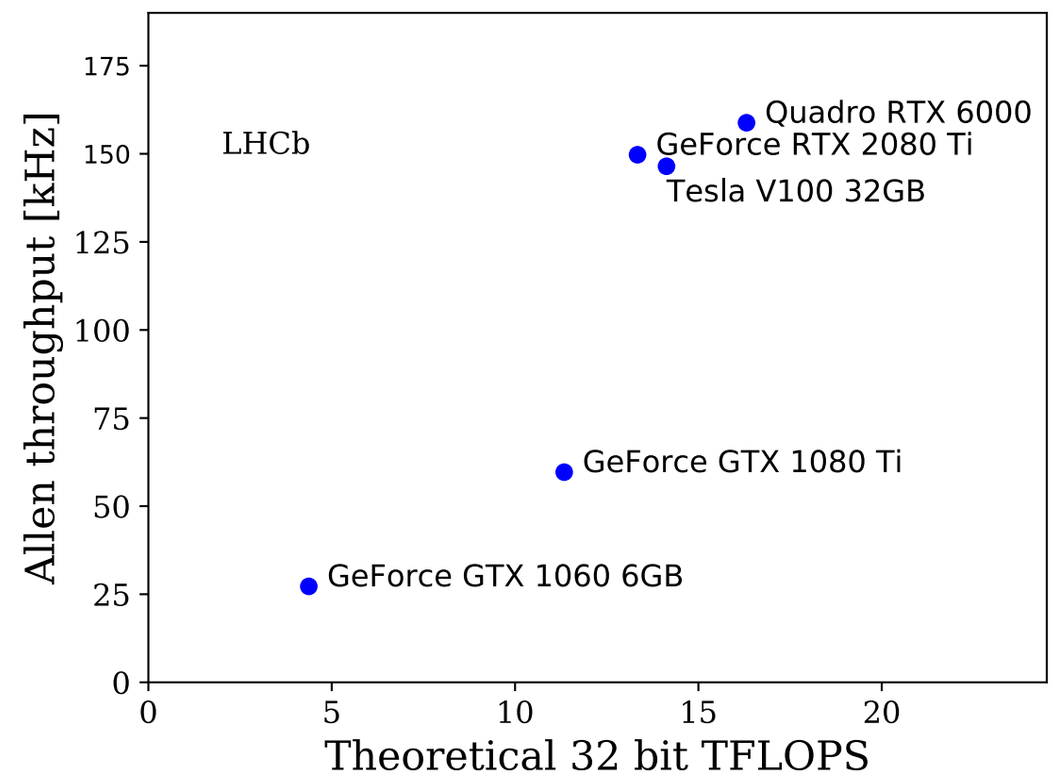
Event selection



Throughput performance

- 30 MHz goal can be achieved with O(200) GPUs (maximum the EB server can host is 500)
- Throughput scales well with theoretical TFLOPS of GPU card

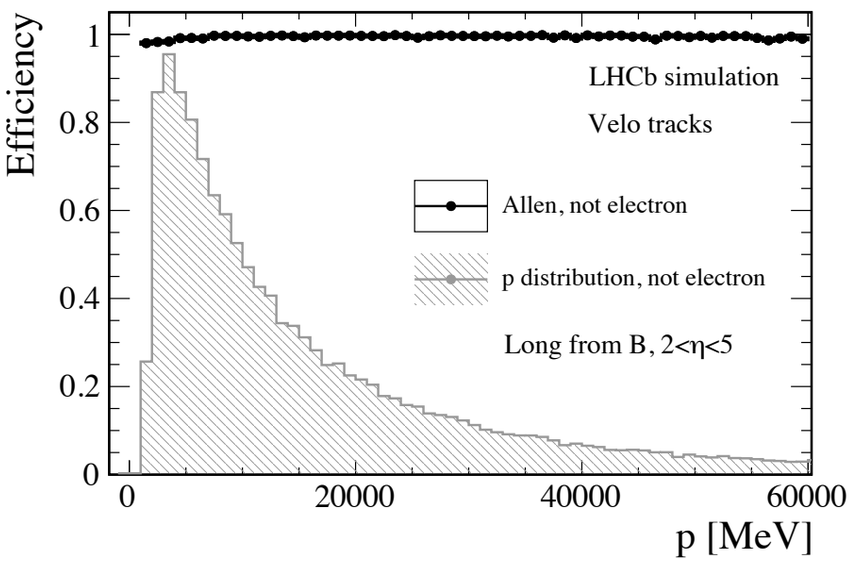


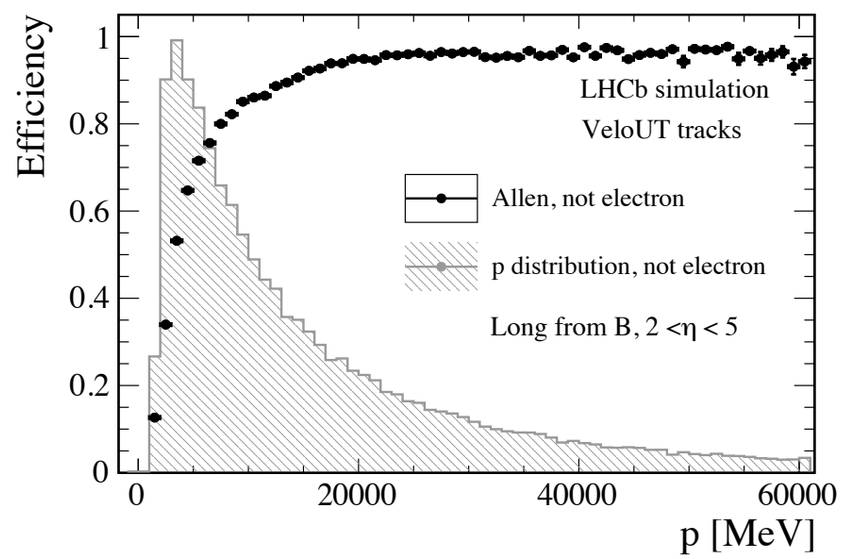


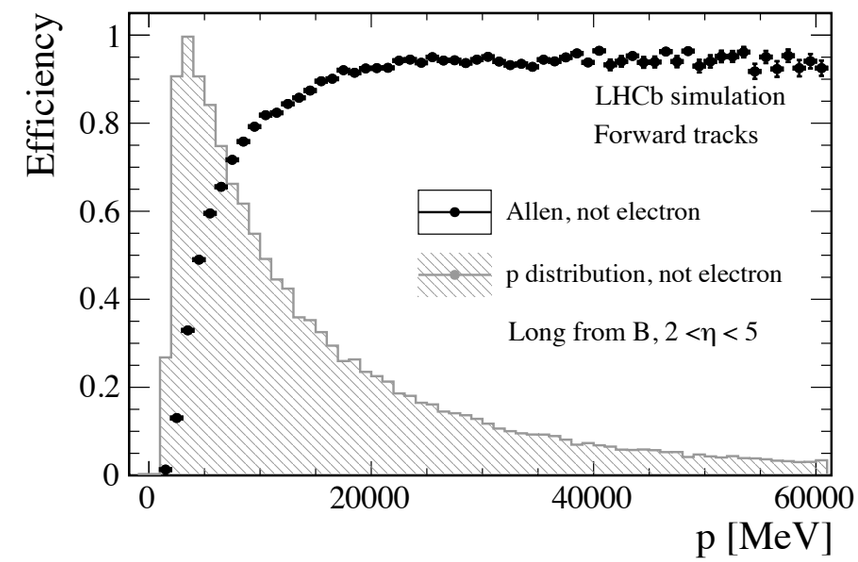
LHCb-FIGURE-2020-014

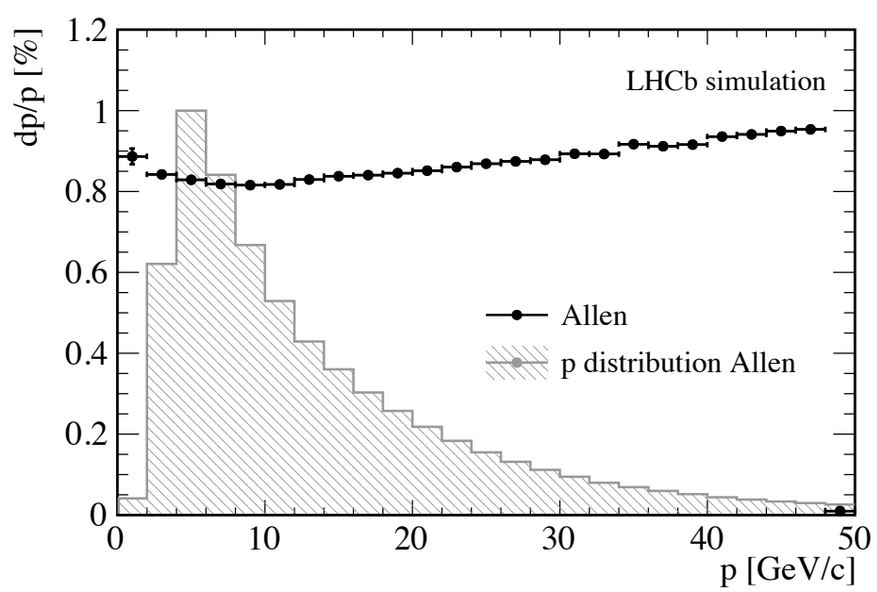
Reconstruction performance

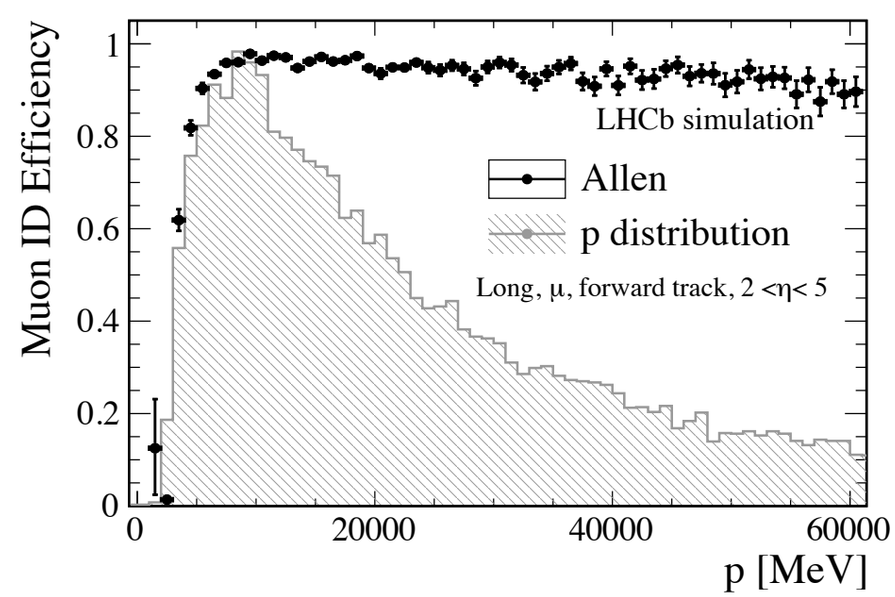
- Excellent track reconstruction efficiency (> 99% for VELO, 95% for high-p forward tracks)
- Good momentum resolution and excellent muon-ID
- Addition of further algorithms for PID (calo) or downstream tracking (seeding) is being explored
- Compatible performance between CPU and GPU [2105.04031]









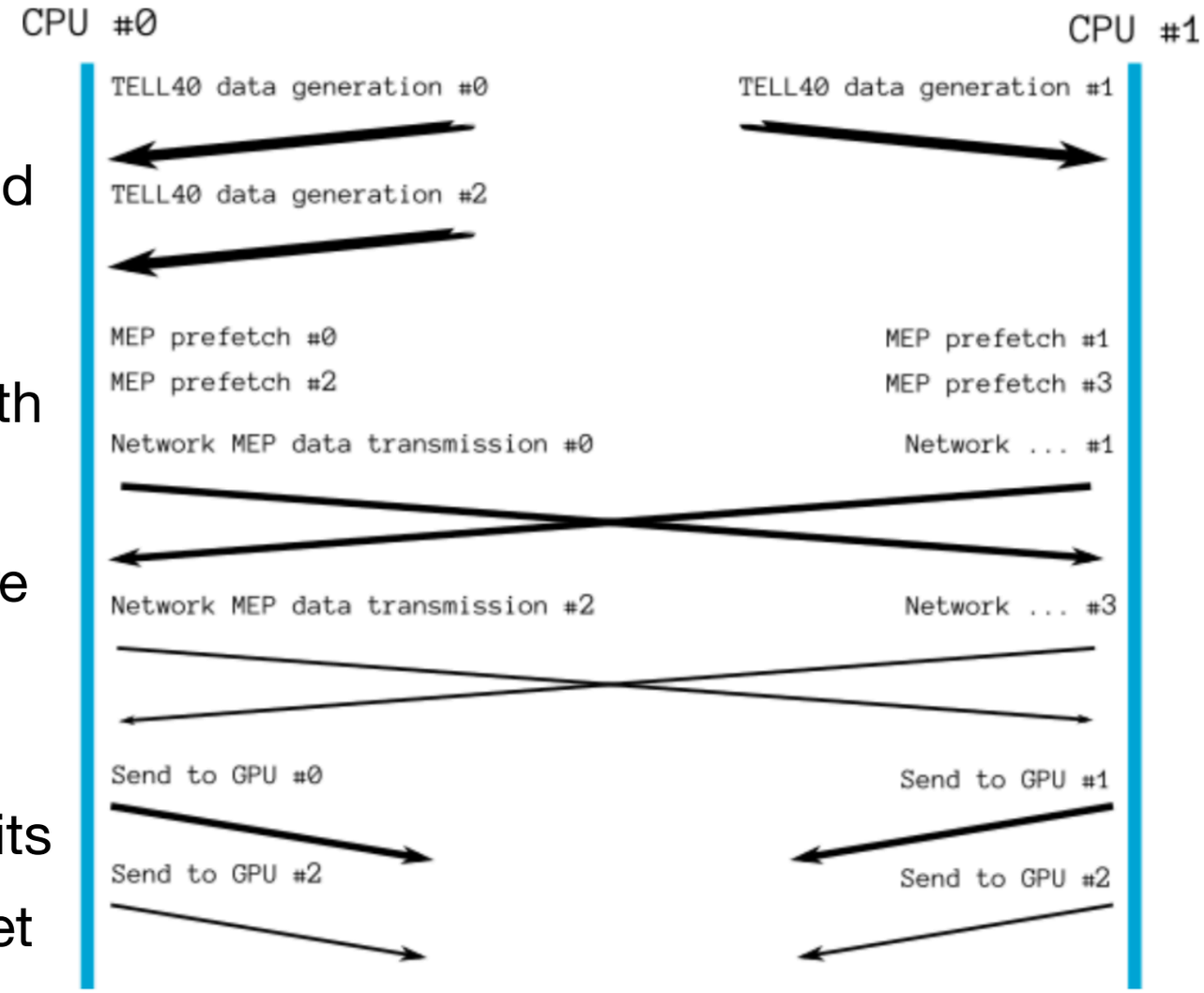


Towards the integration of Allen in the online system

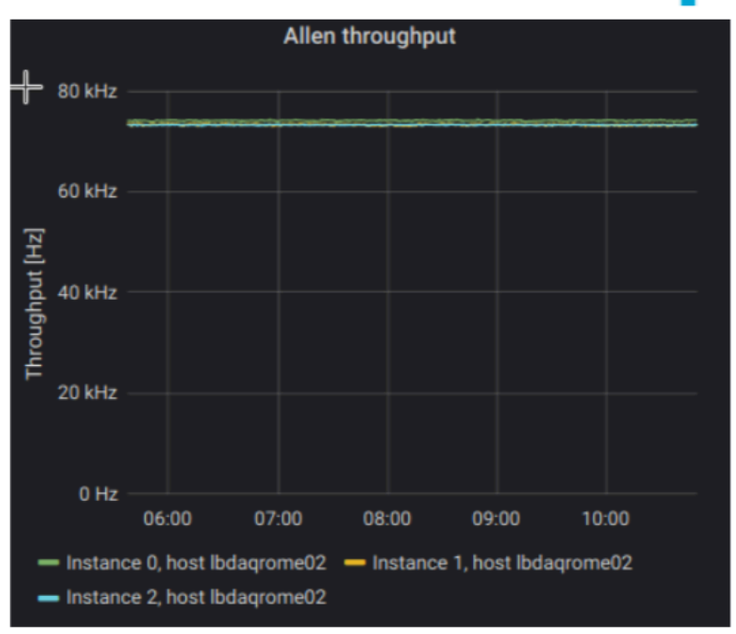
Challenge of fully commissioning Allen: we need the real detectors and EB server first!

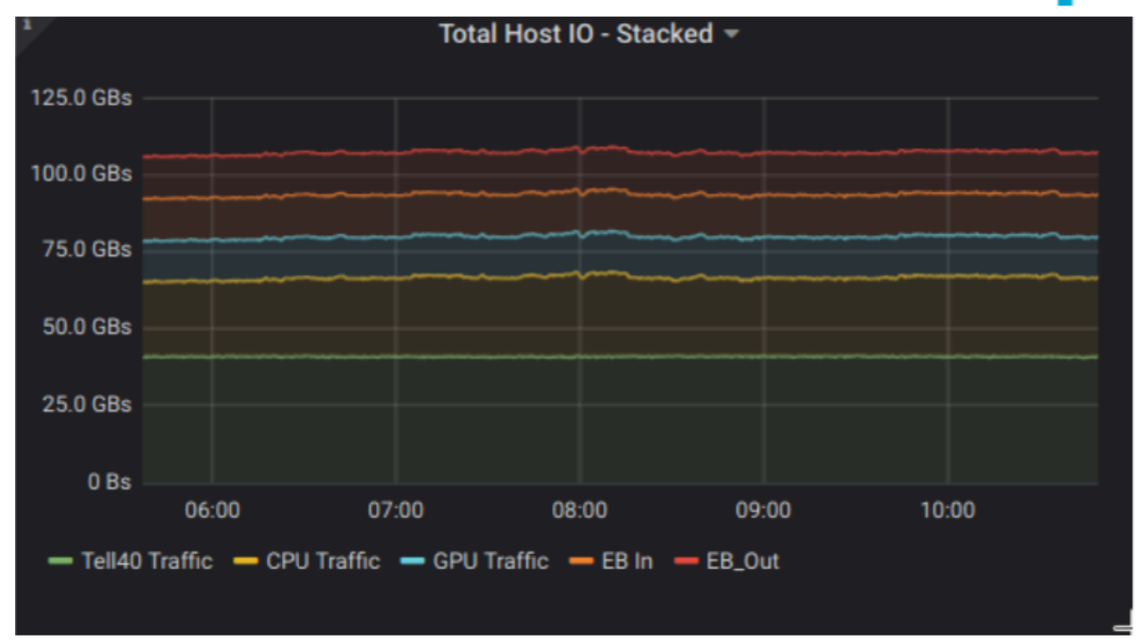
First integration tests in smaller-size servers with pre-loaded simulation data

- Emulate network traffic and memory pressure with mock-up data from FPGAs
- Stable throughput at 70 kHz
- I/O memory bandwidth stable and within limits
- Cooling and memory usage requirements met
- Proof of principle!



Tests in 2019
A lot of performance improvements since!





LHCb-TDR-021

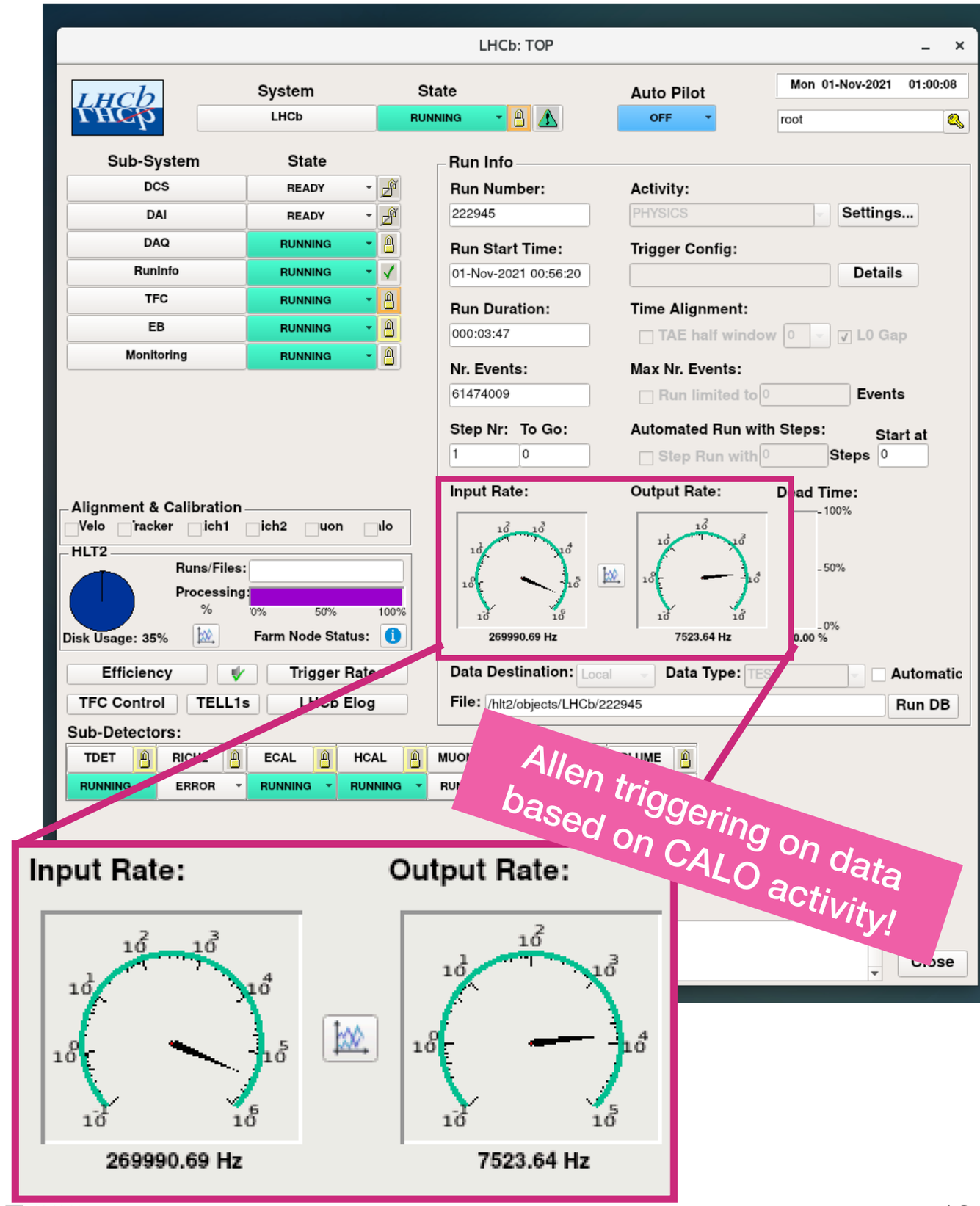
Towards the integration of Allen in the online system

Challenge of fully commissioning Allen: we need the real detectors and EB server first!

In October 2021, two-week LHC beam test

- LHCb ran with upgraded RICH, Calorimeter and muon stations for the first time

- AND, successfully tested for the first time Allen in the LHCb ECS!
  - Both on GPUs and CPUs
  - In pass-through mode
  - And with simple calo-activity trigger line!



Towards the integration of Allen in the online system

LHCb October beam-test was major milestone for the Allen commissioning

Steps for the future:

- As more sub-detectors get installed, commission more parts of the decoding, reconstruction and selection chain
- Commission the full chain (EB → HLT1 → HLT2 → storage & offline)
- Monitoring
- Continue the installation of GPU cards in the LHCb Data center
- Throughput, memory, cooling and stability tests with larger-scale system
- Data taking with stable beams expected to start in spring 2022!

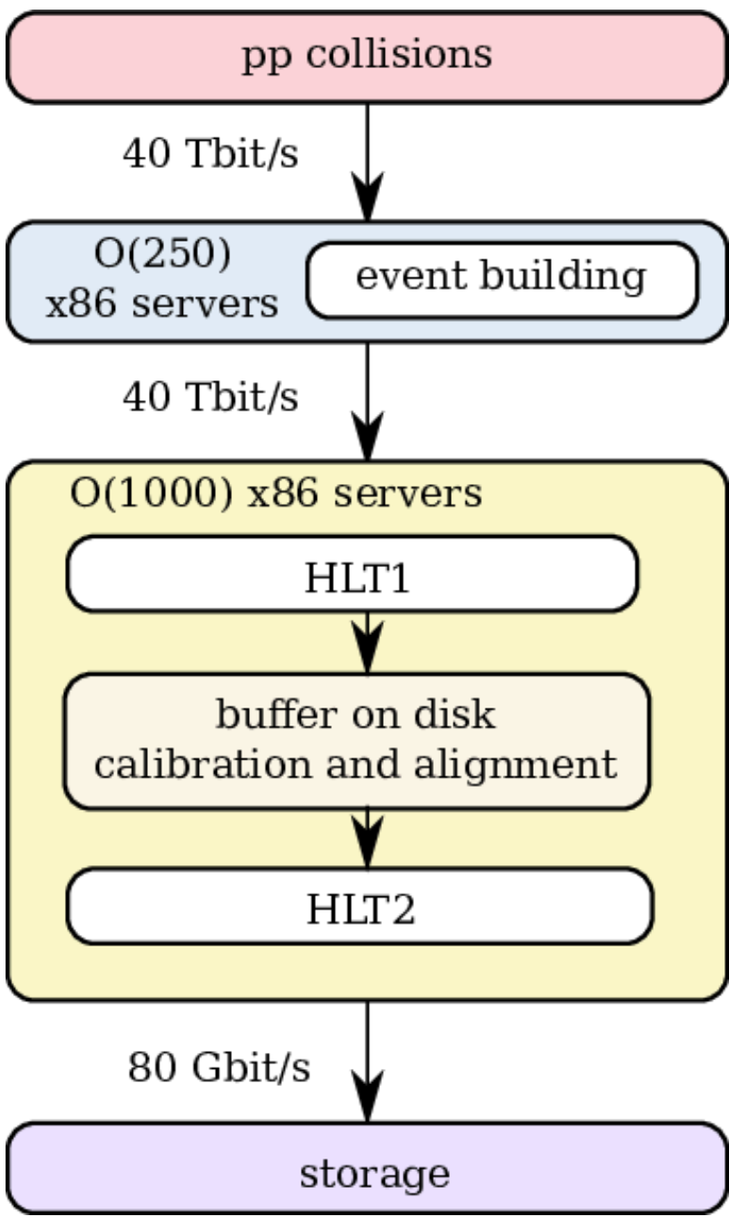
Conclusions

- LHCb is currently undergoing its first major upgrade in order to increase its instantaneous luminosity by x5
- Major changes on the trigger strategy:
  - Remove L0 hardware trigger, read-out full detector at 30 MHz
  - New, software-only first level trigger based on GPUs
- Partial event reconstruction and trigger selection lines implemented, excellent physics performance
- Throughput O(150) kHz → system can be realised with around 200 GPUs
- A lot of work ongoing for the commissioning:
  - Integration to the online tested in external servers
  - And for the first time in the LHCb ECS in pass-through mode and with triggering on real calo data!
- Installation of GPUs in the EB server and commissioning are ongoing - first collisions expected in spring 2022

Stay tuned for more updates!

Backup

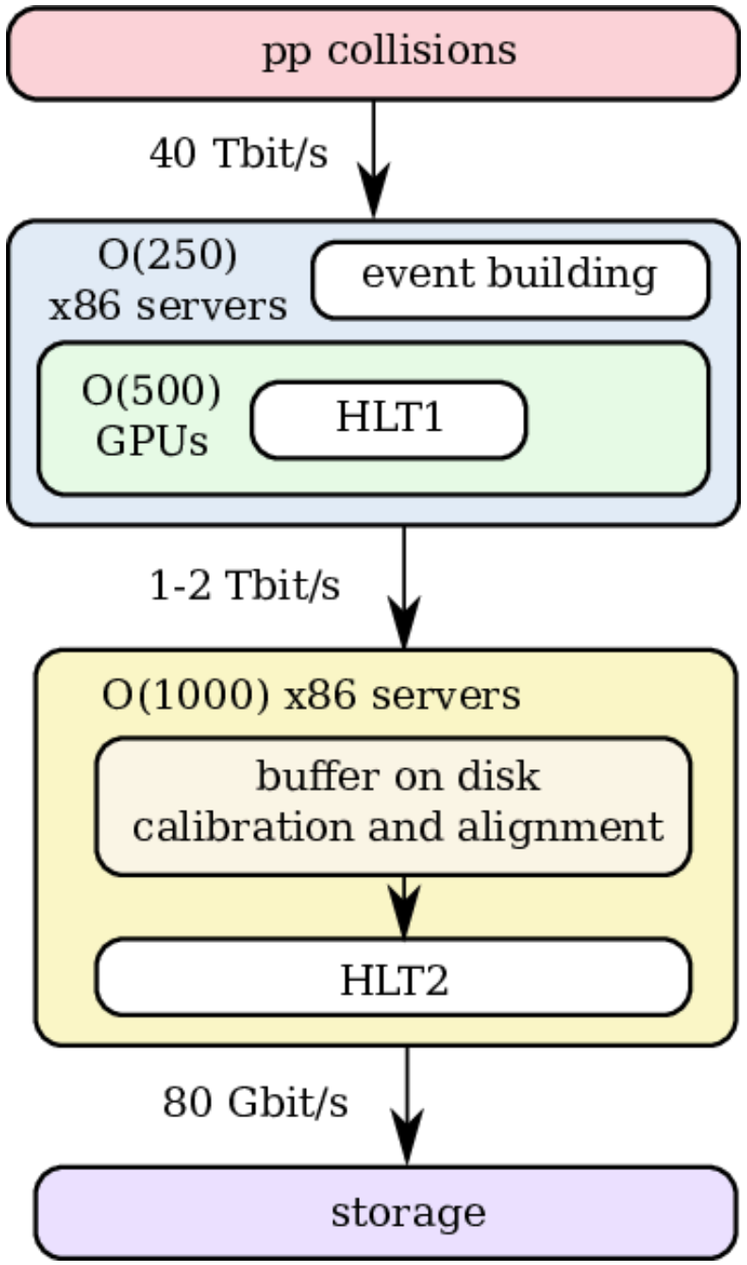
Architecture upgrade options



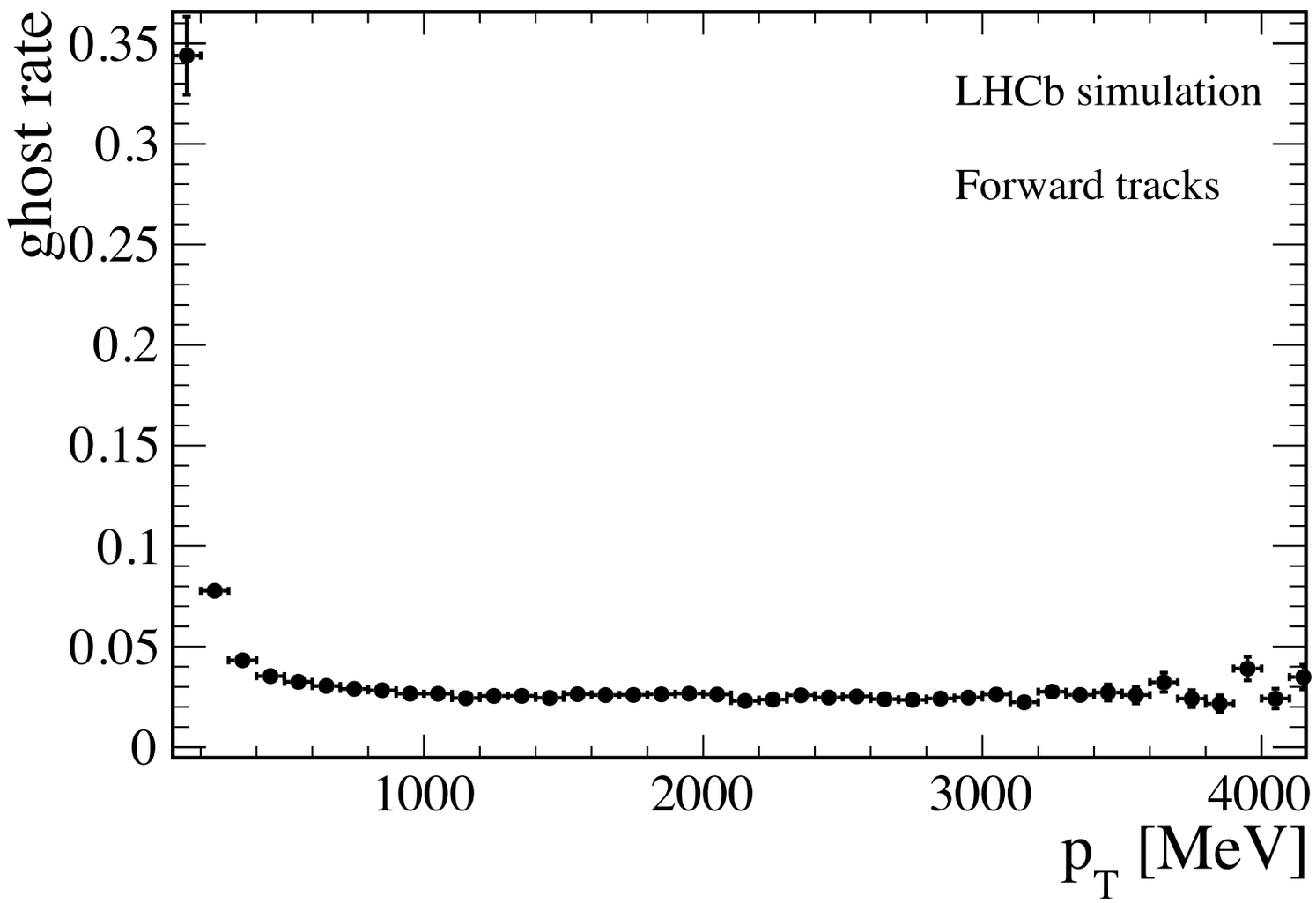
Detector data received by O(500) FPGAs and built into events in the EB severs

Two options:

1. Send full 40 Tb/s to a CPU processing server → extra network needed
2. Fill extra EB slots with GPUs → reduce rate locally to 1 Tb/s before full processing



Ghost rate



LHCb-FIGURE-2020-014