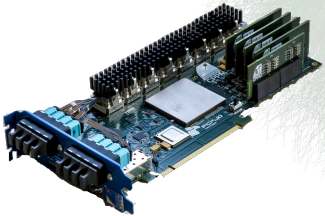
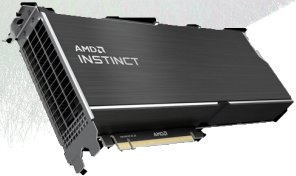
Use of hardware acceleration for online event reconstruction for Run 3 and later

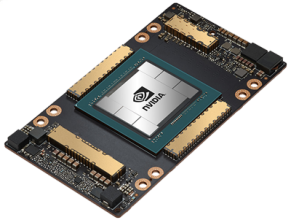
Ole Schmidt on behalf of the ALICE, ATLAS, CMS and LHCb collaborations

CERN and Physikalisches Institut, Heidelberg University

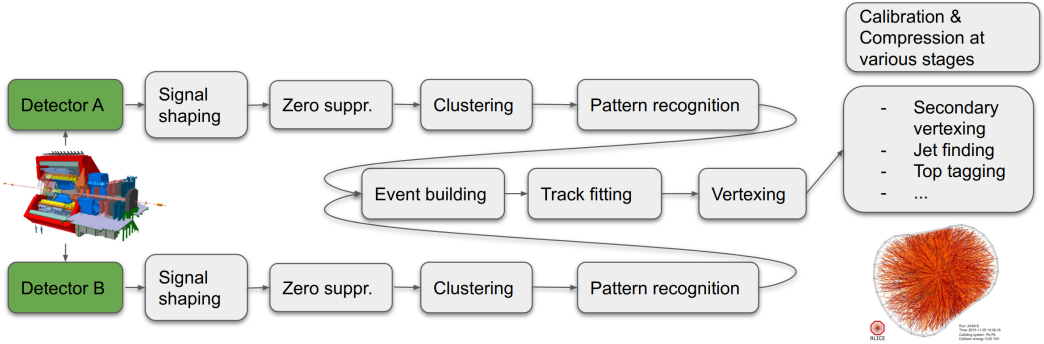
ole.schmidt@cern.ch







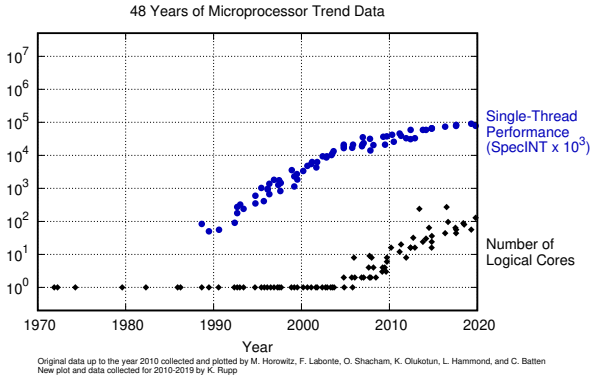
Online event reconstruction - a schematic workflow

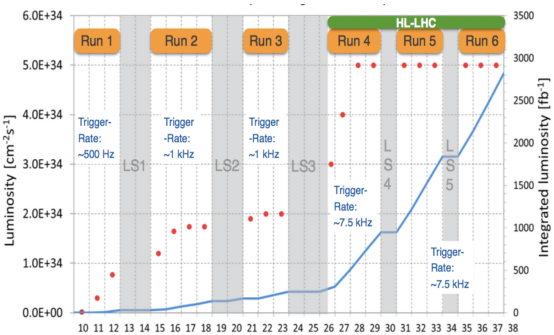


- typically moving from simple and local to complex and global tasks
- triggers (hard- and software-based) can filter events at different stages
- all selected event data must be written to disk

Why is hardware acceleration needed?

- more complex global tasks are usually performed on CPUs, since they are
  - flexible
  - easily programmable with high-level languages
  - available where data from different detectors are merged



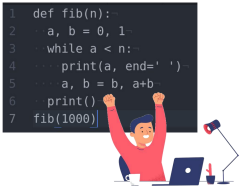


- single-thread performance levels off
- price/performance evolution not compensating

- increased trigger rate during HL-LHC period
- significantly increased readout rate for ALICE

Different options for hardware acceleration

- hardware accelerators are for example FPGAs, ASICs or GPUs
  - FPGAs used for a long time especially in readout cards
  - ASICs are optimized for specific tasks, but have limited flexibility; not cost-effective for small experiments
  - all large LHC experiments investigate usage of GPUs for data processing in the future or are already employing them
- considerations apart from processing power: software complexity, work scheduling, code portability, code maintainability, ...

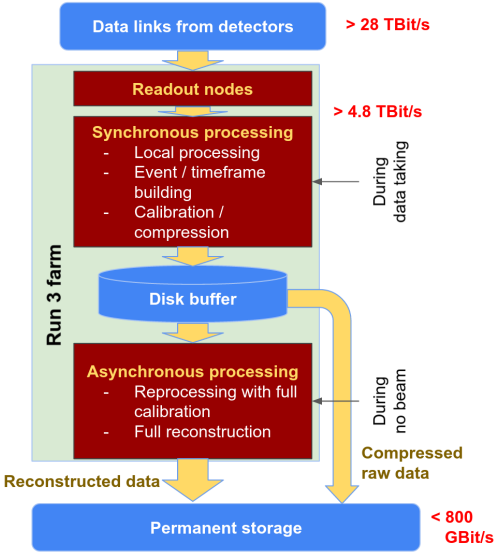


Fibonacci with python...

... and with CUDA

ALICE - online reconstruction

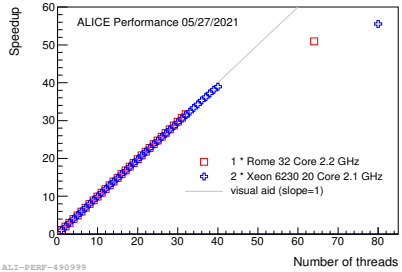
- new compute farm at Point 2 for Run 3 with 250 dual socket servers, each equipped with 8 AMD MI50 GPUs to process Pb–Pb collisions at 50 kHz online
- already during Run 2 the TPC reconstruction was performed on GPUs in the HLT
- single source code base supports different GPU backends
  - yields results identical to CPU processing
- GPUs will also be used in asynchronous reconstruction
  - studies ongoing for usage in simulations
- FPGAs on CRUs perform low-level data processing on detector raw data
  - e.g. zero suppression for TPC



data flow in Run 3

ALICE - computing time details

- synchronous processing dominated by TPC
  ⇝ **will run fully on GPUs**
- asynchronous reconstruction also dominated by TPC, but other detectors relevant too
  ⇝ **85% of async. processing offloaded to GPUs**
- additional steps can be ported to GPUs in the future



Synchronous processing

Processing step	% of time
TPC Processing	99.37 %
EMCAL Processing	0.20 %
ITS Processing	0.10 %
TPC Entropy Coder	0.10 %
ITS-TPC Matching	0.09 %
MFT Processing	0.02 %
TOF Processing	0.01 %
TOF Global Matching	0.01 %
PHOS / CPV Entropy Coder	0.01 %
ITS Entropy Coder	0.01 %
FIT Entropy Coder	0.01 %
TOF Entropy Coder	0.01 %
MFT Entropy Coder	0.01 %
TPC Calibration residual extraction	0.01 %
TOF Processing	0.01 %

Asynchronous processing

Processing step	% of time
TPC Processing	72.01 %
TRD Tracking	12.69 %
TOF-TPC Matching	9.94 %
MFT Tracking	1.69 %
ITS Tracking	0.78 %
TPC Entropy Decoder	0.73 %
Secondary Vertexing	0.69 %
ITS-TPC Matching	0.56 %
Primary Vertexing	0.14 %
TOF Global Matching	0.11 %
PHOS / CPV Entropy Decoder	0.10 %
FIT Entropy Decoder	0.10 %
ITS Entropy Decoder	0.06 %
TOF Entropy Decoder	0.05 %
MFT Entropy Decoder	0.05 %

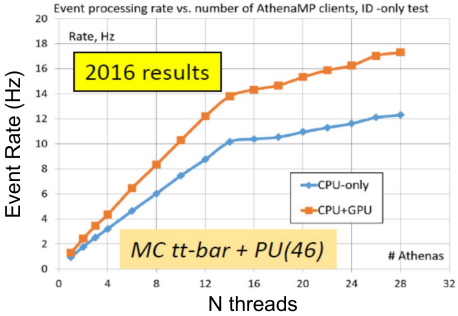
Running on GPU in baseline scenario **Running on GPU in optimistic scenario**

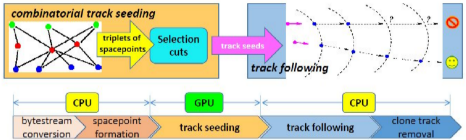
Preliminary numbers: some algorithms not yet complete or not optimized!

D. Rohr, vCHEP 2021

ATLAS - GPU studies for HLT

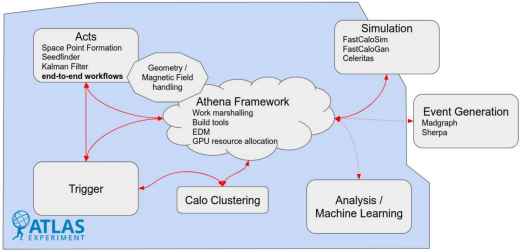
- prototype for Inner Detector (ID) trigger on GPUs 2012 (ATL-DAQ-PROC-2012-006)
  - GPU-oriented algorithm integrated into existing trigger software Athena
  - Tesla C2050 GPU gave x12 speedup with respect to a single CPU core
- GPU trigger demonstrator 2015 (ATL-DAQ-PROC-2016-035)
  - pixel + strip clustering + ID seed making yielded x28 speedup on GTX1080 GPU
  - overall trigger server throughput increased by 40% with GPU c.f. CPU-only
- **conclusion of these studies:**
  - GPUs not considered cost-effective for ATLAS Run 3 trigger
  - studies are being re-evaluated for Run 4 and beyond

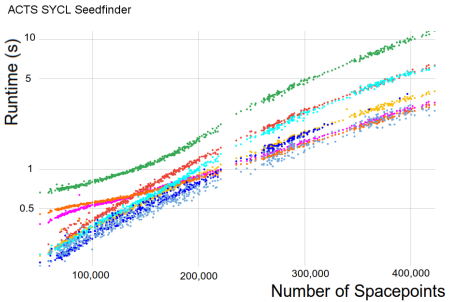




ATLAS - R&D activities and code portability

- heterogeneous computing forum to coordinate effort for use of accelerators in online and offline computing
  - mainly targeting Phase-II
  - primary development areas:
    - GPU resource management
    - geometry and B-field handling
    - event data model
- various ongoing projects:
  - Acts: end-to-end tracking workflow on GPU
  - FCS: parametrized calorimeter simulation
  - many ML projects
- studies of different portability layers ongoing (Kokkos, SYCL, Alpaka, OpenMC/ACC)

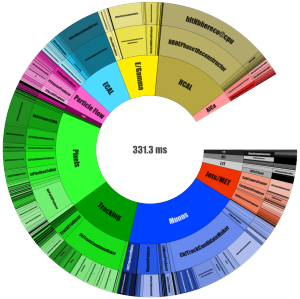




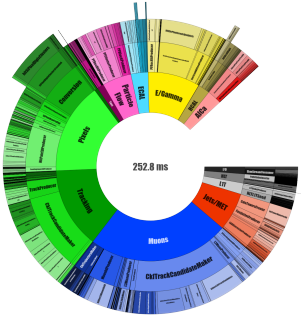
CMS - processing benchmarks

- pie charts show CPU time distribution in different CMSSW modules
  - timing measured for pileup 50 events, running 4 jobs with 32 threads each on a full node[1]
  - innermost ring indicates physics objects or detectors

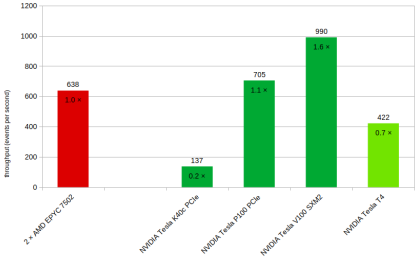
- GPUs are used to accelerate: a) Pixel Tracker reco, b) HCAL local reco and c) ECAL unpacking and local reco
  ⇝ 24% reduction of CPU usage, 22% increased throughput



CPU-only reconstruction



with GPU-offloading

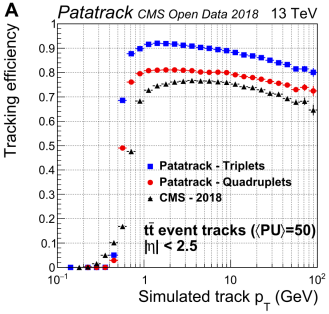


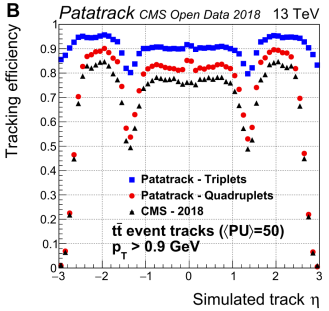
throughput for offloaded HLT reco tasks

[1] Dual socket AMD "Rome" 7502, SMT enabled; equipped with an NVIDIA T4 GPU

CMS - selected ongoing projects

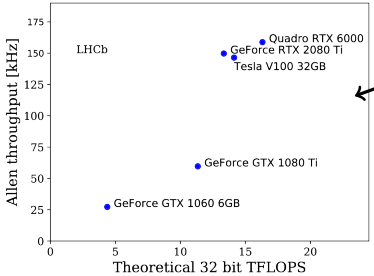
- Patatrack reconstruction of the pixel tracker Front. Big Data 2020
  - processing fully offloaded to GPU
  - SoA data structure, can be converted to legacy data structure
  - CPU and GPU version of Patatrack reco deliver identical physics results and improved *physics performance* compared to CMS-2018 reconstruction
- CLUstering of Energy (CLUE) for parallel density-based clustering (significant speedup on GPUs)
  probably to be used for HGCAL reconstruction in Phase-II
  Front. Big Data 2020
- investigation of performance portability frameworks:
  Alpaka, Kokkos, HIP, SYCL/oneAPI
  M. Kortelainen, vCHEP 2021



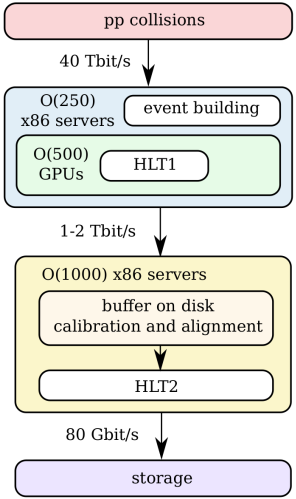


LHCb - Allen project

- High Level Trigger 1 runs entirely on GPUs, see also talk tomorrow by T. Boettcher (link)
- event building server host about 500 GPUs reducing network cost significantly
- full processing of all events at 30 MHz input rate
  $\rightsquigarrow$ rate reduction by factor 30
- event rate reduction as opposed to pure compression in ALICE



throughput scales with reported peak 32-bit FLOPS performance of various GPU models

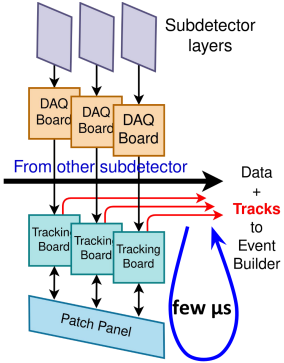


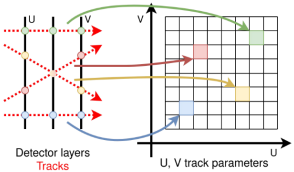
Comput Softw Big Sci **4**, 7 (2020) / LHCb Upgrade GPU HLT TDR (2020)

LHCb - Accelerator testbed for Run 3

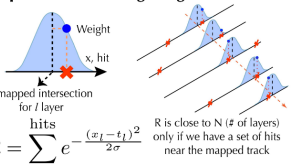
- connect parasitically to part of the data flow to study accelerators
- Artificial Retina picked here as example: real-time tracking on FPGAs
- size limitation bypassed by dedicated distribution network utilizing high-bandwidth transceivers
- prototype implemented on Intel Stratix-V FPGAs:
  - input data injected at 30 MHz
  - maximum signal delay of single input of 10 µs tested
  - system split in two FPGAs: one for the network, the other with tracking cells

  ⇒ throughput stable with a latency of 0.4 µs



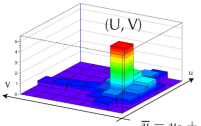


Step 2: Accumulating weights (each cell)



$$R = \sum^{\text{hits}} e^{-\frac{(x_l - t_l)^2}{2\sigma}}$$

Step 3: Find the local maxima and compute centroid



$$\overline{u} = u_0 + \frac{\sum_{ij} iR_{i,j}}{\sum_{ij} R_{i,j}} \quad \begin{matrix} i=U-1,U,U+1 \\ j=V-1,V,V+1 \end{matrix}$$

$$\overline{v} = v_0 + \frac{\sum_{ij} jR_{i,j}}{\sum_{ij} R_{i,j}}$$

More details in CTD 2020 proceedings

Summary

- increasing data taking rates at the LHC continue to challenge the available computing infrastructure
- all LHC experiments use or are evaluating the feasibility of using GPUs for online event reconstruction
  - different approaches towards how GPUs are used efficiently (accelerator vs. standalone general purpose processor)
  - ALICE has offloaded 99% of the online event reconstruction to GPUs
  - LHCb runs the entire HLT1 on GPUs
- applicability of FPGAs for complex tasks including full online tracking is being investigated
- hardware acceleration is not available "for free", but different portability layers facilitate the development and maintenance of the software

Many thanks to the ALICE, ATLAS, CMS and LHCb collaborations for providing valuable input for this presentation!