TWEPP 2018 – Antwerp – 17-21 September, 2018

Towards a heterogeneous High Level Trigger farm for CMS

Dr. Andrea Bocci <andrea.bocci@cern.ch> - CERN

The CMS online reconstruction

The Compact Muon Solenoid experiment has been designed with a two-level trigger system: the Level 1 Trigger (L1T), implemented with custom-designed electronics, and the High Level Trigger (HLT), running a streamlined version of the offline reconstruction software and algorithms (CMSSW) on a traditional computer farm.

The HLT farm has been dimensioned to sustain the nominal L1T rate of 100 kHz, and a peak LHC instantaneous luminosity in excess of 2×10^{34} $cm^{-2}s^{-1}$. It comprises over 1000 dual-Xeon nodes, with over 30'000 CPU cores and 60'000 threads.

Figure 1 shows the fraction of time spent reconstructing different detector and physics objects, for 2018 data with 50 pileup events; the ECAL, HCAL and Pixel local reconstructions, due to their highly parallel nature, are prime candidates for implementation on accelerators.

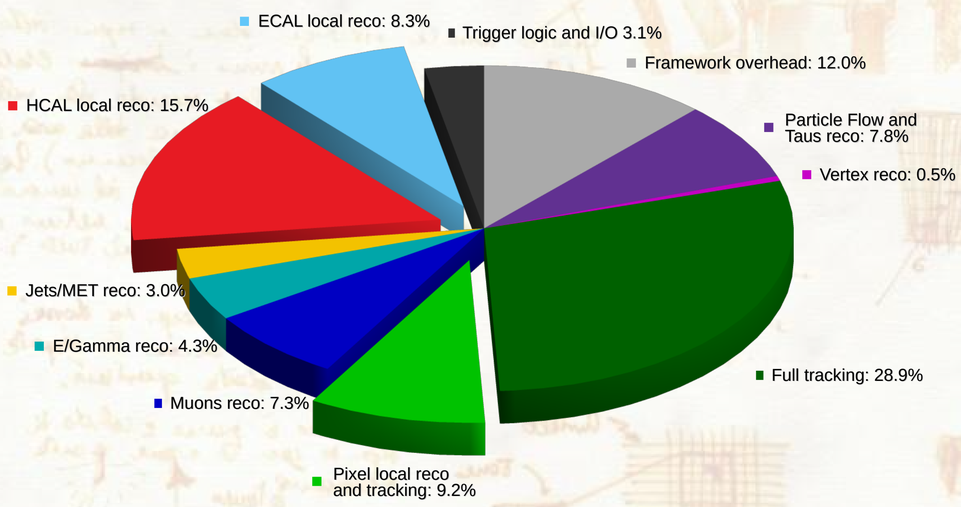

Figure 1: fraction of the online reconstruction time spent, on average, reconstructing different detector and physics objects, in collisions with 50 pileup events

Parallelism at HLT

The CMS online reconstruction shows excellent scaling characteristics: multiple events can be reconstructed concurrently running separate jobs or within a single job, taking full advantage of multicore CPUs. Figure 2 shows the linear performance scaling for a simplified reconstruction job, performing the Pixel local reconstruction and Pixel-only tracking over 2018 data.

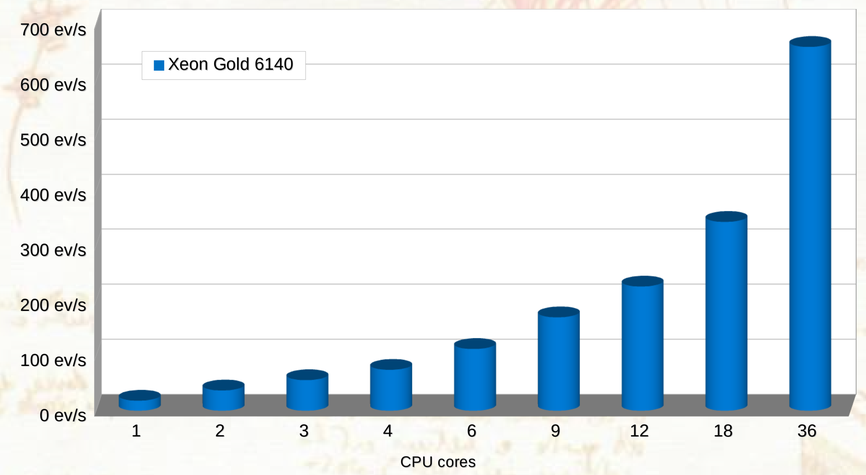

Figure 2: throughput of a CMSSW job performing local reconstruction and tracking with the CMS Pixel detector, as a function of the number of CPU cores utilised.

Looking ahead, the processing power of conventional CPUs is growing by 10~15% per year. By 2026 we can expect it to be 2-3 times larger than today.

However, even assuming an optimistic linear extrapolation of the processing power requirement to the Phase 2 conditions, with a L1T rate of 750 kHz and an instantaneous luminosity of 7×10^{34} $cm^{-2}s^{-1}$, CMS would need an HLT farm 25 times more powerful than today – an order of magnitude more than what traditional CPUs may provide.

Towards an heterogeneous HLT farm

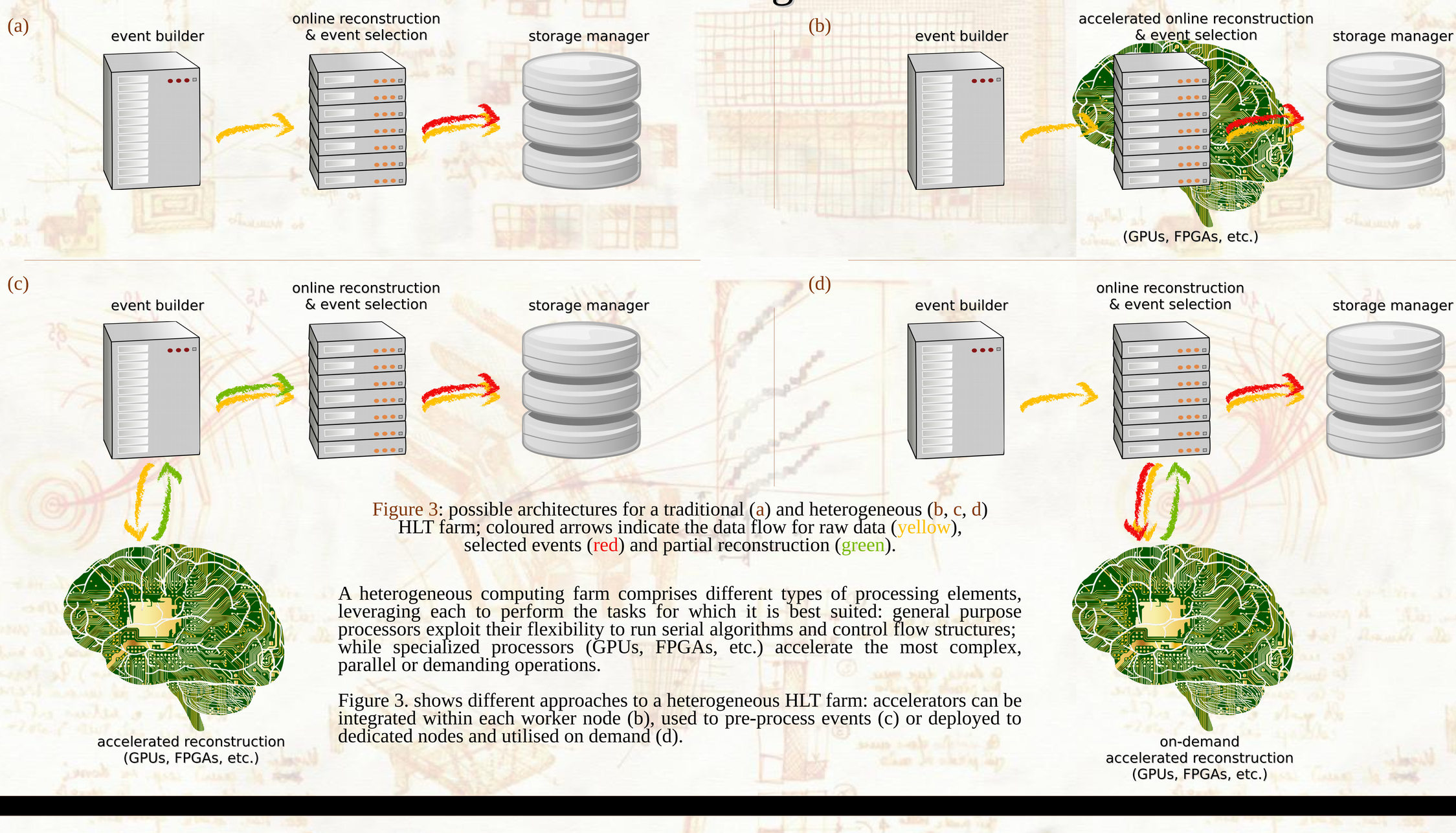

Figure 3: possible architectures for a traditional (a) and heterogeneous (b, c, d) HLT farm; coloured arrows indicate the data flow for raw data (yellow), selected events (red) and partial reconstruction (green).

A heterogeneous computing farm comprises different types of processing elements, leveraging each to perform the tasks for which it is best suited: general purpose processors exploit their flexibility to run serial algorithms and control flow structures; while specialized processors (GPUs, FPGAs, etc.) accelerate the most complex, parallel or demanding operations.

Figure 3. shows different approaches to a heterogeneous HLT farm: accelerators can be integrated within each worker node (b), used to pre-process events (c) or deployed to dedicated nodes and utilised on demand (d).

Pixel-based track reconstruction on GPUs

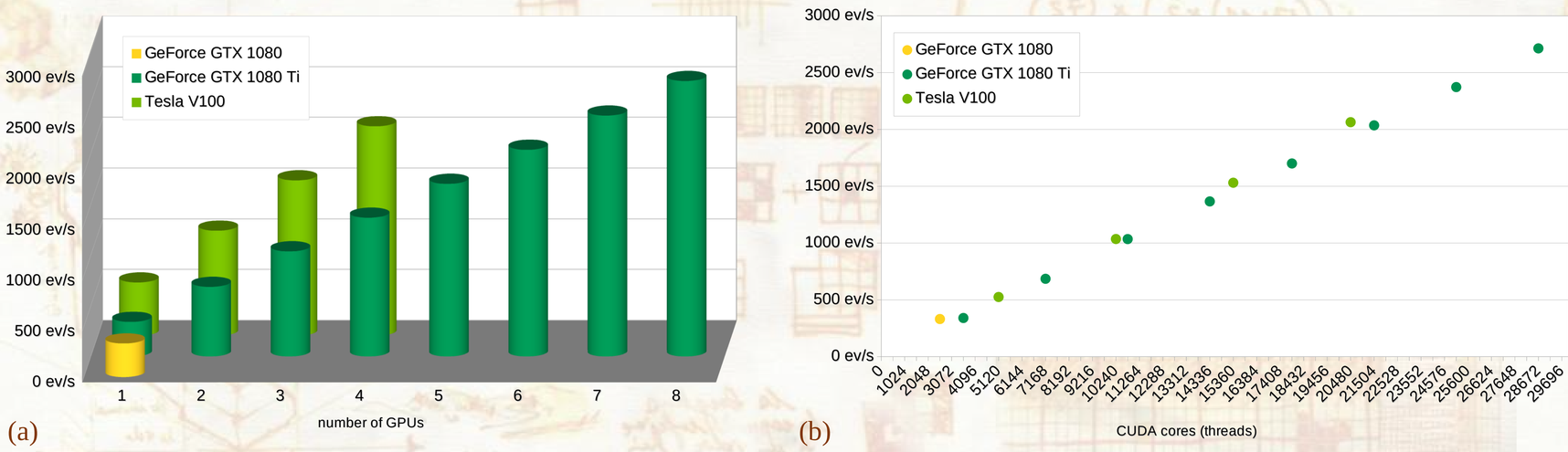

Figure 4: throughput of multiple jobs performing local reconstruction and tracking with the CMS Pixel detector, running over different types of NVIDIA GPUs, as a function of the number of GPUs (a) and CUDA cores (b); each GPU is utilised by a single job.

To demonstrate the feasibility of offloading part of the online reconstruction to GPUs, the Pixel local reconstruction and Pixel-only tracking has been ported to CUDA and integrated in the full CMSSW framework.

Figure 4 shows the performance when utilising a single GPU (in yellow), and the linear scaling when taking advantage of dedicated nodes with multiple GPUs (in green).

While different generations of GPUs have varying performance, to first approximation the scaling is linear with the number of available "CUDA cores", or processing threads.

To increase the flexibility of the system, work is ongoing to let multiple jobs share individual GPUs, and to let a single job exploit multiple GPUs.