CMS

HEP-CCE All Hands Meeting
April 20, 2022

Presented by Oliver Gutsche
Produced with the help of many people! Thanks!
All mistakes and omissions are mine!

HEP-CCE and HPC landscape

HEP-CCE

- Address the development and implementation of HEP scientific applications on next-generation computing, storage, and networking systems.
- Current focus of the HEP-CCE: develop common strategies to efficiently run HEP software applications on pre-exascale and exascale high-performance computing systems

HPC landscape

- Current: facilities are switching to accelerator-based architectures with the majority of the compute power coming from accelerators ➜ focus: GPUs
- Future: more exotic accelerated architectures and hybrid facilities with specialized hardware architecture components (one for AI, one for HTC, one for HPC, … speculation)

CMS Offline & Computing

Strategic Goals

General O&C Performance Goals with the aim of enabling physics research:

- Efficiently use all of the resources available to us
- Minimize computing resource needs
- Maximize throughput
- Minimize job failures
- Minimize manual operations (effort)
- Requests are completed in short, predictable amounts of time (*not completely under the control of O&C*)

A computing model in which we can utilize processing (including accelerators and heterogeneous architectures, HPCs), disk & tape storage, and network in a flexible manner, all while having a unified code base, best fulfills the above goals.

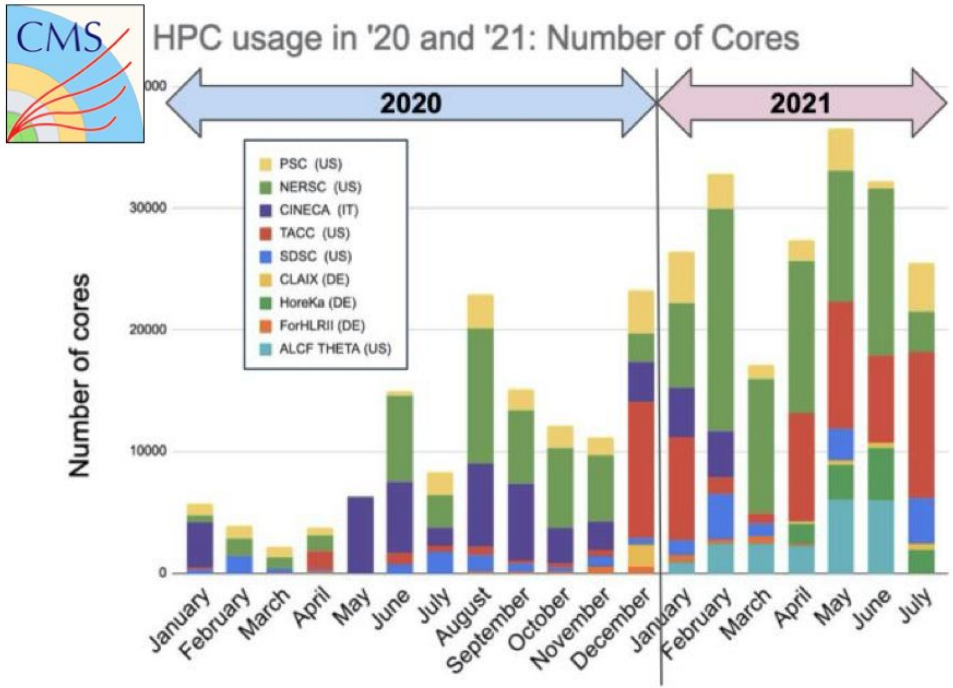
Summary: CMS usage of HPC

HPCs and Opportunistic Resources in 2021

- Opportunistic capacity: 46.5% of the total
- HLT farm: 17.9% biggest single contributor
- HPCs (aggregated): at the same level of HLT
  - Our “3rd site” by capacity

Evolution of average compute capacity used by CMS at HPCs:

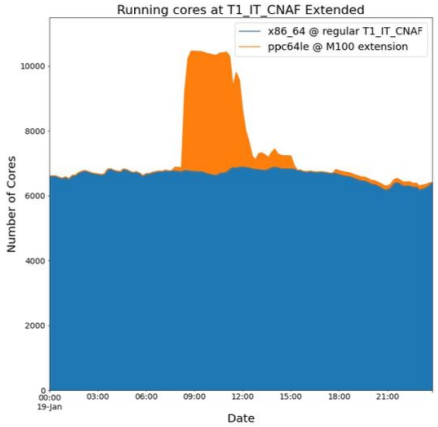
- 2019: 34 kHS06
- 2020: 108 kHS06
- 2021: 362 kHS06



2020/2021 snapshot

CMS and HPC: what is coming next

- Marconi: PowerPC physics validation (translates to Summit)
  - Simulation w/o pileup: OK
  - Data reconstruction: ongoing
  - Simulation w/ pileup: Summer (enough network: allows to mix events)
- High-statistics validation of GPU-enabled HLT reconstruction
  - We successfully managed to run a gpu-enabled workflow on the Grid
- A solid base to expand usage of GPUs into offline processing
  - Reconstruction and (planned) simulation and generation
  - Both ML and non-ML approaches to be explored
- Remote offloading to GPUs ➜ SONIC
  - First for ML inference, then maybe expand to C++ GPU code



Message

- Infrastructure is being put in place
- All depends on software availability for HPC architectures

CMS Physics Software

CMS software framework (CMSSW) schedules algorithm developed by domain experts (scientists, experts in detector components and/or reconstruction algorithms)

- Many small kernels
- Since 2015: multi-threaded framework in production (TBB)
- For Run-3: HLT will use GPUs
  - >25% of the HLT reconstruction algorithms will be offloaded to GPUs
- Next step: establish GPU-based offline workflows
  - Use at GPU clusters and HPC machines

I/O

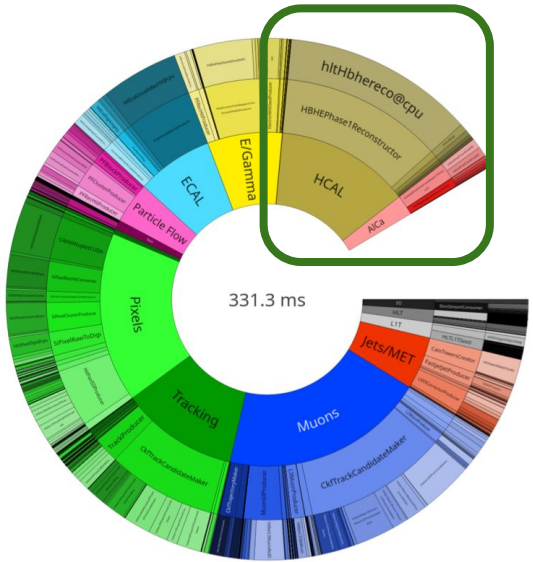
- All ROOT-based

GPU@HLT

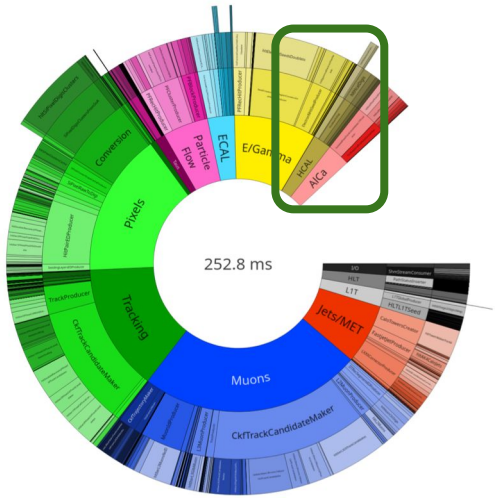
25% of HLT code offloaded

- Pixel local reconstruction , tracks and vertices; HCAL local reconstruction , ECAL unpacking and local reconstruction

CPU



CPU+GPU



From https://cds.cern.ch/record/2759072

HEP-CCE and CMS: portability

Matti is co-leading the CMS Core Software group, Martin is member of that group

- Portability is discussed in this group, with contributions from the HLT group
- All PPS results are discussed
- Adopted the PPS metrics and created an enlarged version specific for CMS

Needed a decision on portability for Run-3:

- Based on results from CCE and the Alpaka port ➜ all supported by the Core Software group
- Alpaka was chosen because currently it is the most “useable” and performant

Decision will be revised in time for HL-LHC (on the timescale of beginning of LS3)

- Now focus on portable Run-3 algorithms for HLT and offline workflows

HEP-CCE and CMS: I/O

Discussions in CMS Core Software Group, Chris is leading CMS' efforts

- General question related to HPC systems:
  - Single node: how do high core count machine I/O behave
  - Multi-node: how do thousands of processes interact with a shared file system
- Two thrusts of interest:
  - Scalability studies using Chris' I/O framework
  - I/O profiling with Darshan
- Observations so far
  - Waiting on large scale scaling studies with Darshan
  - Parallel file writing from many processes may be far into the future for CMS and would imply big changes in e.g. Workflow Management
  - Discovered and fixed performance bottlenecks in ROOT when doing I/O in multiple threads
  - Need to understand how I/O with our current data formats scale for high core count machines, and to thousands of jobs
  - Current understanding: A separate simple file format not necessary for production because ROOT to implement similar storage strategies than other solutions

Strategic thoughts

I/O

- Separate file format seems to be not necessary, ROOT is implementing similar storage strategy
  - If large scale studies reveal issues, we would like to resolve them
- Study optimization of data formats remains high priority
  - In context of HLT, work is ongoing in CMS to look into new SoA structures

Portability

- Need to continue portability solution investigations small
  - Clear need for domain experts to adapt algorithms for Run-3
  - Usefulness of computing experts without interest in becoming domain expert limited

Learned from portability studies: need for event batching concepts large!

Outlook

Potential R&D Theme: efficient usage of accelerators

- Event batching
- Data formats
- Other aspects of efficient usage

General theme: we want to enable as many physics algorithms as possible to use accelerators

Going beyond the current scope

Infrastructure questions are not solved (mostly at LCFs)

- How to get data in and out of the HPC systems efficiently
- Edge services