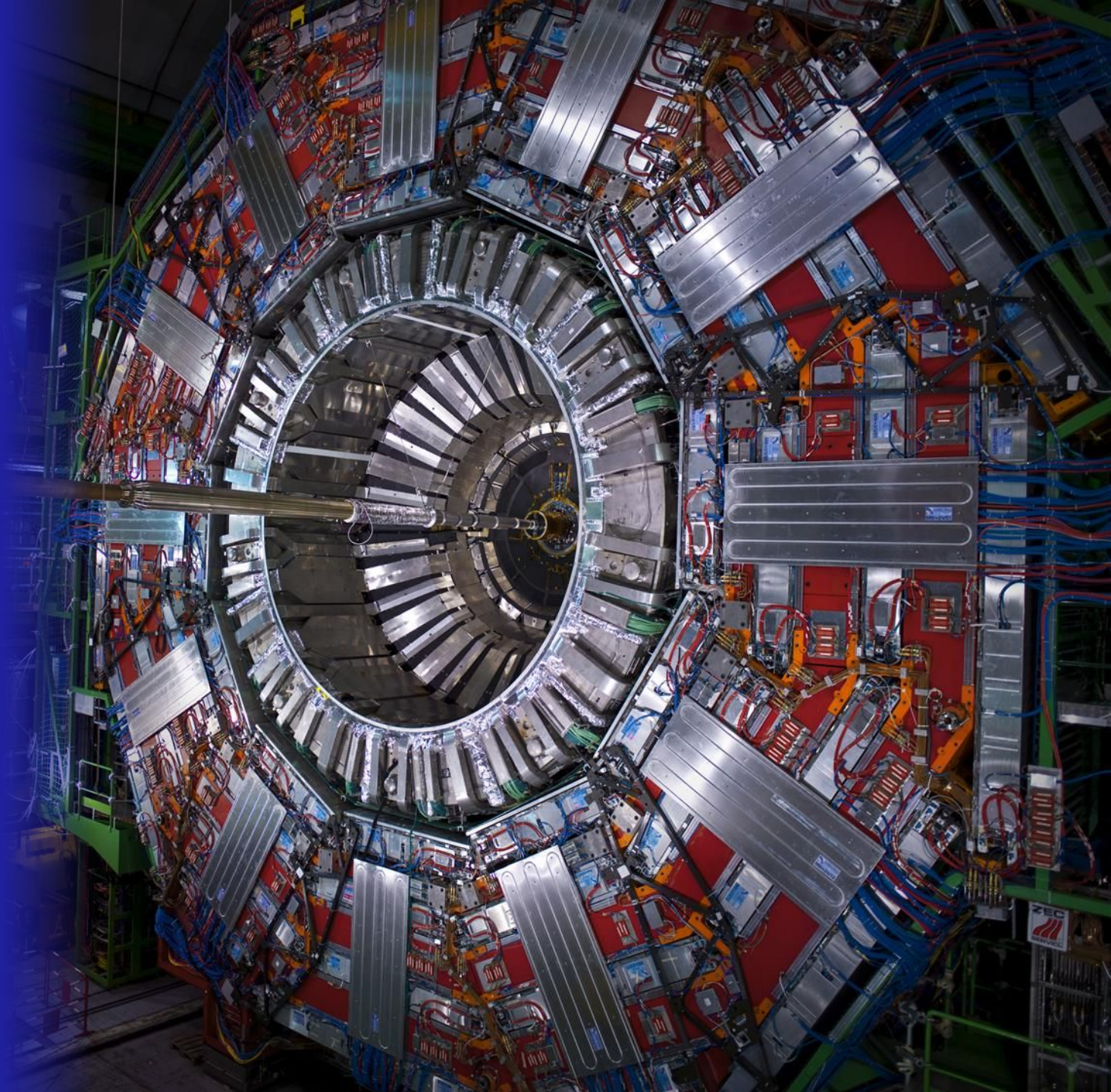


Leveraging Heterogeneous Computing Resources in CMS

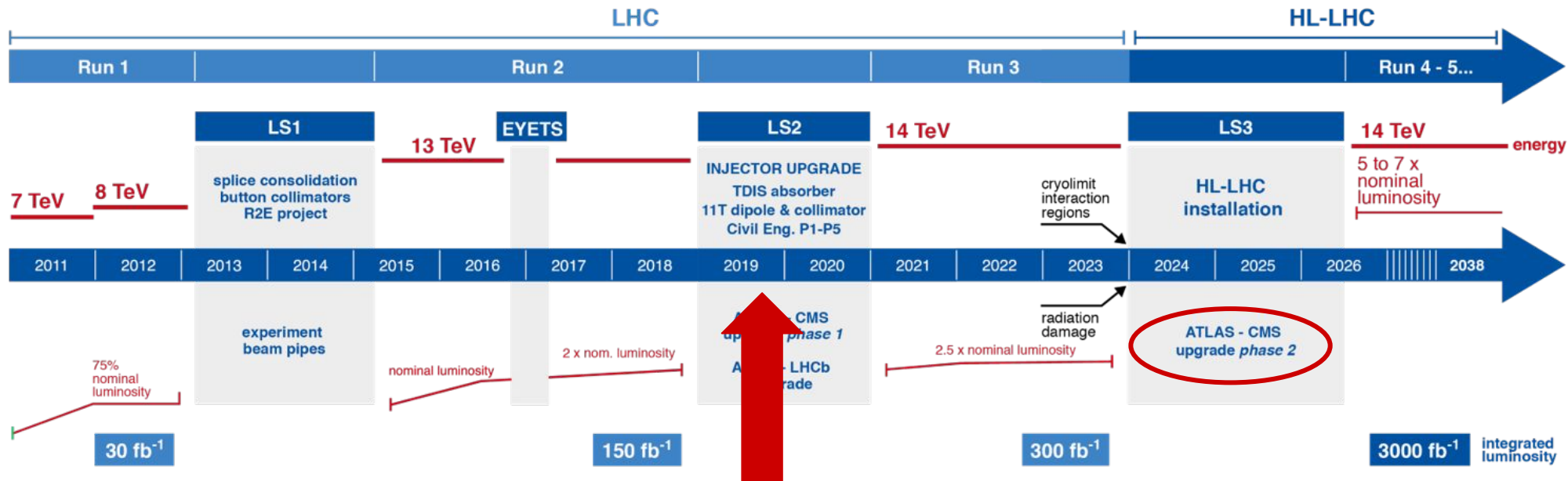
D. Piparo (CERN - EP) for CMS - Scientific Computing Forum Meeting, Oct 2nd, 2019



A Challenge Ahead



High Luminosity LHC (HL-LHC)

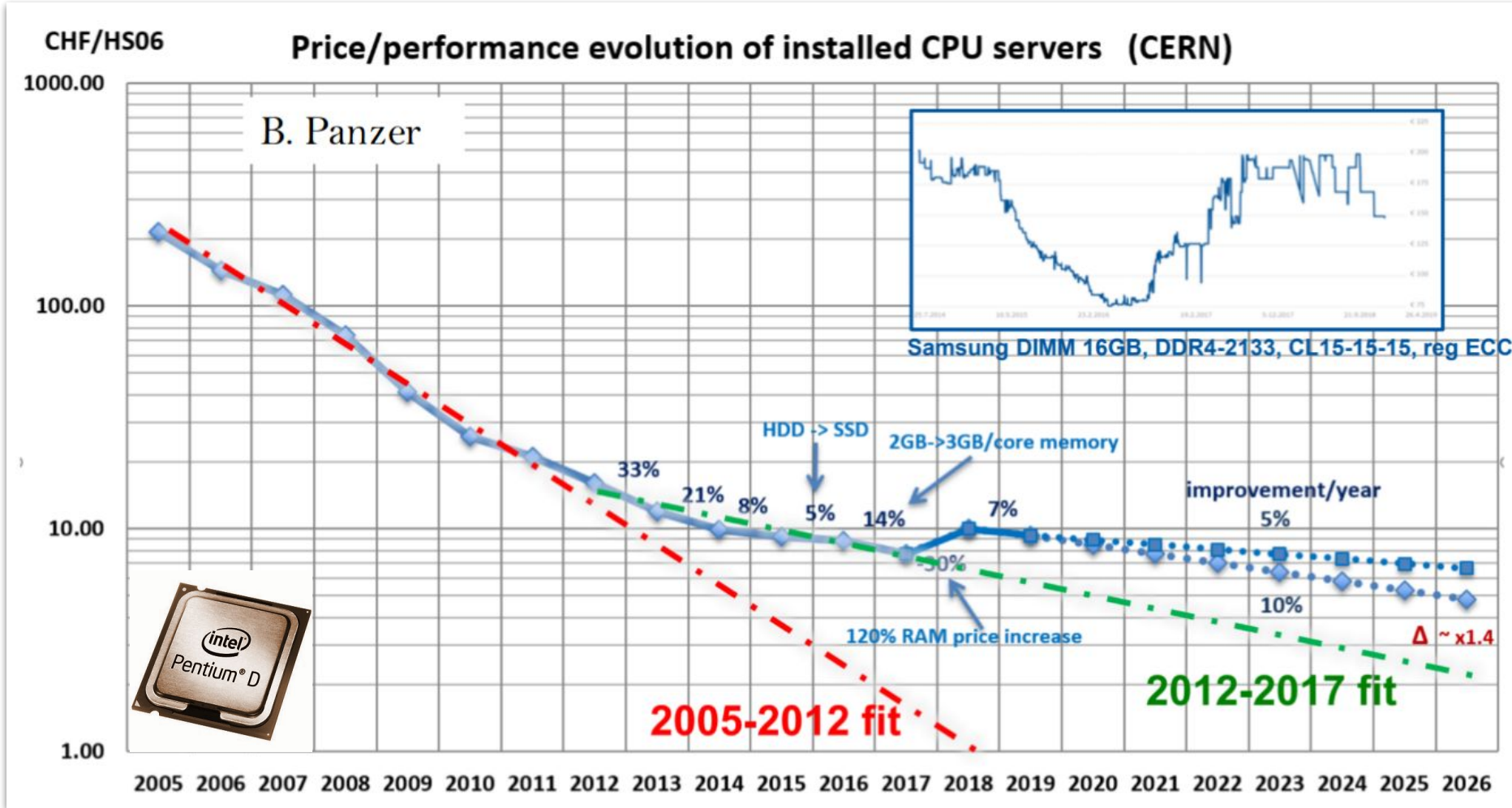


- ▶ Not only **an unprecedented accelerator upgrade...**
- ▶ **... but also an ambitious CMS detector upgrade**
- ▶ Much more data, much more complex!

We also call this scenario *Phase-2*



Market Trends of CPUs



Summit Supercomputer:
4608 nodes. **Per node, 2 IBM POWER9™ CPUs, 6 NVIDIA V100 GPUs.**

[https://en.wikipedia.org/wiki/Summit_\(supercomputer\)](https://en.wikipedia.org/wiki/Summit_(supercomputer))

- ▶ **The entire computing power for HL-LHC data processing might not come from CPUs alone**
- ▶ But this is a *fierce* competition (AMD, Intel, IBM, ARM, ...): we keep our eyes open



A Taste of Phase-2 Challenges for Software & Computing

The CPU computing power / price affects CMS (and other experiments). If one considers for CMS:

- ▶ The current Grid computing approach
- ▶ A **flat budget** to finance its computing resources
- ▶ **No change in its computing model and data processing software**

>10x factors for storage and CPU still missing to successfully trigger, process and analyse Phase-2 data

CMS is undertaking an intense R&D effort for an economically viable HL-LHC exploitation

- ▶ Smarter data handling and operations
 - e.g. deferred processing strategies, scouting (high rate but small event contents)
- ▶ Reduced data formats
- ▶ Faster event simulation
 - Invent new approaches, optimise the traditional ones
- ▶ Increase usage of HPCs
- ▶ Better (faster!) algorithms and data structures
- ▶ Effectively exploit heterogeneous architectures

Focus on these two very entangled aspects in this talk

Scientific software expertise can reduce the costs of computation

CPU + GPU Heterogeneity: Part of the Solution?



CMS and Heterogeneous Architectures

Work is ongoing to support the paradigm shift towards heterogeneous architectures.

Two main directions:

1. Support for heterogeneity in the CMSSW framework
2. Re-thinking of reconstruction algorithms and data structures

Framework: CMSSW

▶ Orchestrator of the data processing, schedules algorithms, handles collision data, non-event data, manages the writing of outputs.

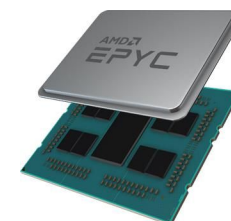
▶ **CMSSW, supports since 2014 parallel scheduling of algorithms on a thread pool**

- In production: used effectively at High Level Trigger and offline during Run2
- Concurrent processing of events, parallelism within events and within algorithms!



▶ Asynchronous system: **offload of algorithms on accelerators being tested**

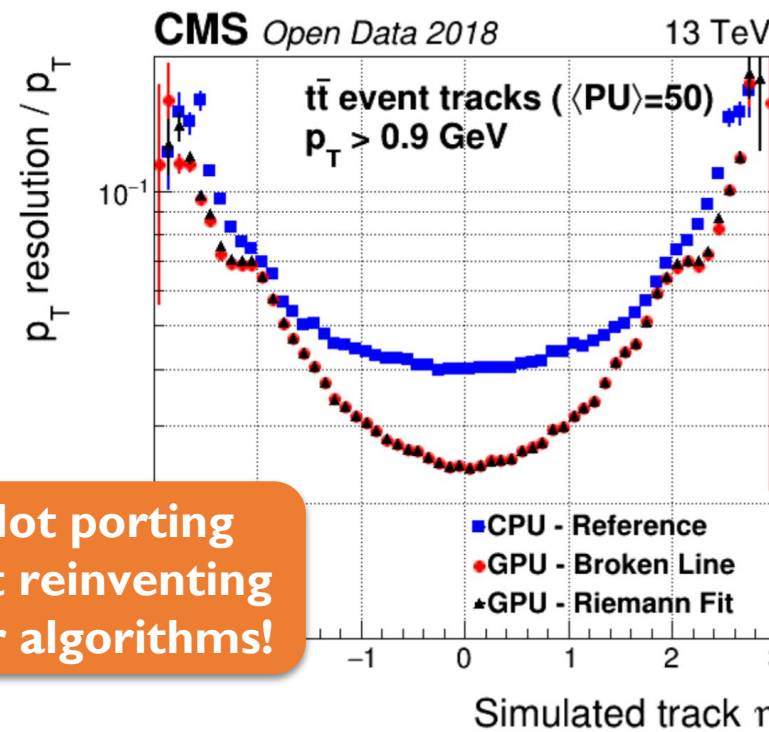
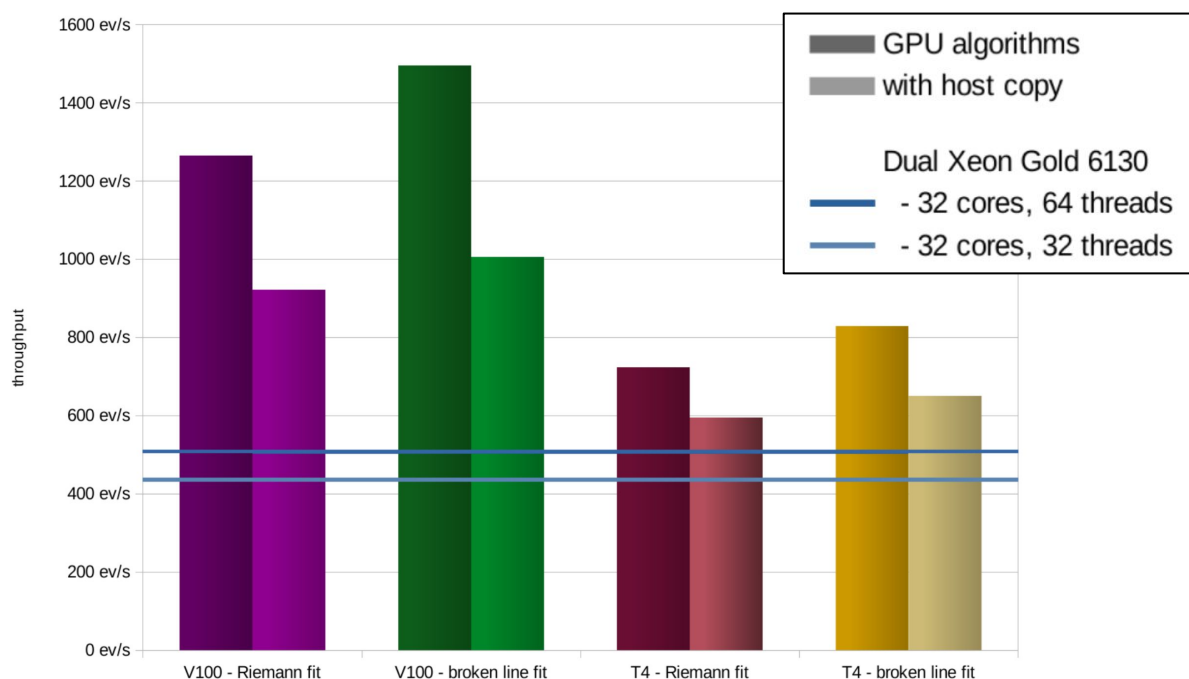
- Not a special case but a generic support, valid for any workflow on any machine
- Necessary to offload work and data on accelerator, keep CPU cores busy, retrieve results
- **Decide algorithm by algorithm whether to run calculation on CPU or accelerator**



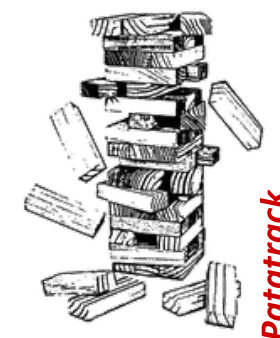
Asset of CMS: **same framework and most algorithms offline/online (different configurations)**

CMS and Heterogeneous Architectures: Algorithms

- ▶ Re-think algorithms and data structures for accelerators: main focus now NVIDIA GPUs
- ▶ **Expertise in physics and software development is needed**
 - Grow and acquire software expertise
- ▶ Work ongoing, e.g. [Patatrack incubator](#) at CERN, [NESAP](#) postdoc @ NERSC, [DEEP-EST](#) project
- ▶ Regular Patatrack Hackatons organised and other trainings and schools



**Not porting
but reinventing
our algorithms!**



Pixel track reconstruction, Pixel reconstruction consumers can either work directly on the GPU or ask for a copy of the tracks and vertices on the host
[Patatrack: accelerated Pixel Track reconstruction in CMS, CDT/WIT 2019, F. Pantaleo et al.](#)

How to Handle Two Implementations for Two platforms

- ▶ CMSSW works producing and consuming data products in a “container”: the EdmEvent
 - Example data products: tracks, jets, hits, muons...



How to Handle Two Implementations for Two platforms

- ▶ Enabling software technology: “Switch Producer”
- ▶ Can decide at run time how to produce data






[C. Jones, HOW2019](#)

A Switch Producer can decide whether to run version A or version B

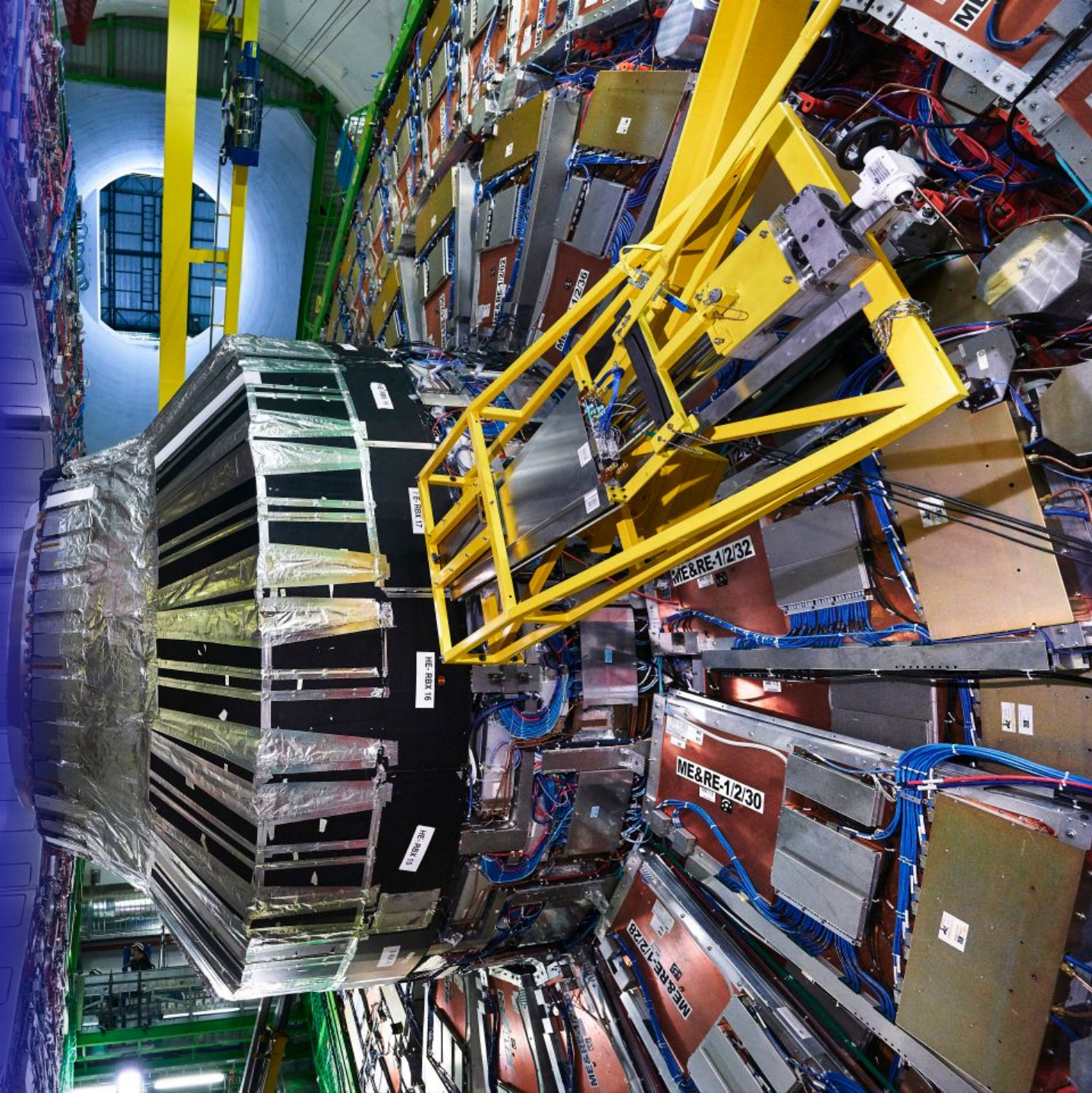
- Via explicit configuration (“use GPU here”)
- Via autodiscovery during CMSSW startup (“I see a GPU, I use it for this job”)
- Module per module, looking at what resource is free

Performance Portability

- ▶ Need one **single high-level code base** for all backends (CPU, NVIDIA-AMD GPU, ?, ...)
 - Keep **costs and sustainability** (validation, synchronisation, maintenance) under control
 - Cannot afford same algorithm in C++, NVIDIA Cuda, <language xyz>
- ▶ Investigate **performance portability libraries**: work ongoing
 - Allow to have one code base for many backends
 - Several products investigated: OpenACC, Raja, Kokkos, Alpaka, SYCL, OpenMP, ...
- ▶ At the moment three credible alternatives:
 - **Kokkos** - [Sandia Lab](#) 
 - **Alpaka** - [Helmholtz-Zentrum Dresden](#) 
 - **SYCL** - [Khronos Group](#) 
- ▶ Need to gain more experience to make a sensible choice
 - E.g. programming model, ease of use, support in the long term...



A Pragmatic Approach to Test Heterogeneous Solutions



Online/Offline Coherent Vision: Run3 Heterogeneous HLT

Heterogeneous HLT farm: a possible solution for CMS Phase-2 event filtering

**Aim to a CPU-GPU
Trigger Farm for Run3**

- ▶ Use **Run3** as “**Baptism by fire**”
- ▶ Completely under CMS control, e.g. type of hardware.
- ▶ **An evolution, not a revolution**

If all R&D about development and operation of GPUs goes well, the plan is:

- ▶ **Keep existing infrastructure at P5** (e.g. cooling, power, racks) and:
 - Replace old HLT computers at end of life with machines equipped with CPUs and entry level enterprise GPUs
 - Add same GPUs in the other computers not yet at the end of life
 - In total ~700 GPUs (one per node) foreseen in the trigger farm
- ▶ **Same cost as the projected full CPU solution, same or greater physics performance**
 - Requires to break even with resources used to buy GPUs wrt CPUs
 - **Break-even point: 20% of the payload runs on GPU** - that leaves space for more GPU algorithms, since the GPUs would be partially used at that point
 - To be confirmed at in September 2020

Consequences for Production Software and Processing

Success scenario: a heterogeneous HLT farm is deployed for Run3

- ▶ **CMS offline processing could also exploit accelerators...**
- ▶ ***... but CPUs will still need to be available on T0, T1s and T2 centres on the Grid for offline processing and cannot be replaced by GPUs (or other accelerators) during Run3!***
 - At the moment, it looks unlikely that a substantial portion of offline reconstruction is ready to be offloaded to GPUs for the start of Run3.
- ▶ **Make CMS software more suitable for new generation HPCs**
 - Other difficulties to be solved, **heterogeneity alone: not enough to use all HPCs for HEP!**
 - The “easy” ones become even more attractive, e.g. CSCS

**Heterogeneity:
opportunity to expand
the resource base**

Consequences for the Usage of HPCs

▶ HPCs are part of the computing infrastructure for science: there to stay!

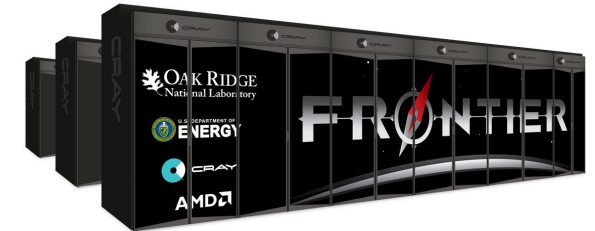
- Exascale machines will be well available by HL-LHC

▶ Being able to **use accelerators helps leveraging HPCs**

- **But is not sufficient**

▶ There are other hurdles to overcome to use HPCs for HEP

- HEP and HPC: **language spoken by experts can be different**
- **Data access** (access, bandwidth, caches ...): HEP has data processing applications (HTC)
- **Submission of tasks** (MPI vs Batch systems vs proprietary systems)
- **HPC: Handful computational kernels VS HEP: thousands of small kernels**
- Environment **less open than Grid** one (OS, access policies, ...)
- **Node configuration** (low RAM/Disk, ...)
- Primary **architecture** (x86_64, Power9, ARM, proprietary, ...)
- Relationships between providers and **CMS are decades long**



Solving those hurdles requires also investing in people



Consequences for the Software Development Cycle

Software development cycle and validation processes will need GPU systems:

- ▶ Would need to be provided by CMS institutes and supporting labs
- ▶ For **development**: “lxplus-like” nodes, GPU equipped, small tests and benchmarks
 - Now using private machines + nodes offered by some CMS institutes, e.g. Simons Foundation
 - ~30 “central” (virtual) machines with one *value* (e.g. NVIDIA T4) GPU will be enough
- ▶ For **continuous integration**: batch nodes, GPU equipped, automated building and testing
 - Now using V100 nodes in IT + 2 K20 nodes (shared with EP-SFT) @Techlab for small unit tests
 - ~10 “central” (virtual) machines with one *value* GPU will be enough
- ▶ For **validation** more substantial amount of resources will be needed
 - Few 100k events need to be reconstructed and output of algorithms checked
 - Few 10s (virtual) machines with one *value* GPU will be enough

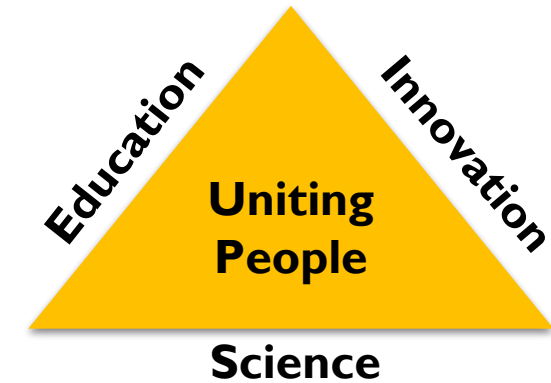
More Added Value



Not Only More FLOPs for Triggering

Growing programming expertise, educate, innovate

- ▶ Opportunity to **reinvent our algorithms**
- ▶ Leading edge programming skills
 - Useful in CMS, labs, industry and other research fields
- ▶ Learning and sharing knowledge
 - Help from existing institutions such as CSC



New interesting use cases for GPUs mounted @ HLT

- ▶ HLT used since years parasitically during beam-off periods for central offline, e.g. Sim, Reco
- ▶ Caveat for running years: **not much time at disposal** for parasitic usage of the farm!
- ▶ Potential for **machine learning training: centralised, streamlined workflows**

Summary

- ▶ **HL-LHC poses unprecedented challenges** to CMS Software and Computing
- ▶ **Intense R&D program** ongoing to cope with them
 - The usage of heterogeneous hardware is a necessity to **expand the resource base** (e.g. HPC))
- ▶ Ongoing work to **support usage of accelerators**: in CMSSW framework and algorithms
 - A paradigm shift!
- ▶ If everything goes well: **use in production GPU equipped HLT for Run3**
- ▶ Assess technology with HL-LHC in mind
- ▶ A **significant step forward for offline processing too** but
 - Not enough alone to leverage fully new HPCs
 - This does not mean that less CPU power is needed for Run3 offline processing
- ▶ **No additional cost, new leading edge use cases accessible, lots of expertise generated**