Software and Computing in High Energy Physics.

3 April 2023

Davide Costanzo

Outline

- The case for computing in particle physics
- Data selection and processing
- Simulating and designing particle detectors
- Final remarks on
    - Machine learning, career and training, sustainability and quantum computing

Disclaimer: I had to select topics for this talk, and the selection may reflect personal bias

A bit of history…

Journal of Scientific Instruments (Journal of Physics E) 1969 Series 2 Volume 2

The use of computers in high energy physics experiments

(link)

D Lord and G R Macleod
Data Handling Division, CERN, Geneva, Switzerland

1.3 *Areas of computer use*

Computers are used in the planning, data acquisition and data analysis phases of high energy physics experiments, as well as in control functions for accelerators (Howard 1967a, beam switchyards (Howry 1967) and bubble chambers (Simpson 1967). In planning an experiment simulation calculations can be made by using Monte-Carlo techniques (James 1968) for estimating event rates to be expected, counting rates due to background, optimum disposition of detectors and so on. Much beam optics design (Whiteside and Gardner 1963)

The role of Software and Computing

Detector data

Trigger
Real-time system

Experiment operation

Reduced data format

Initial design
Modeling of running exp

Offline processing

User analysis

Simulation (digital twin)

- Physics interaction
- Parton shower
- Particle interactions with material
- Detector response

Software and computing today

- From the European Strategy of Particle Physics (2020):
  Large-scale data-intensive software and computing infrastructures are an essential ingredient to particle physics research programmes [...] **The community must vigorously pursue common, coordinated R&D efforts in collaboration with other fields of science and industry, to develop software and computing infrastructures that exploit recent advances in information technology and data science.** (link)

- In the US, the Snowmass process included a "Computational frontier" (link)
  Along with others (energy, neutrino, rare processes, cosmic, theory, accelerators, instrumentation)

- The UK Particle Physics Technology Advisory Panel (PPTAP) (link)
  [34] Likewise within software and computing the UK has significant leadership in a significant number of important areas, including in exploitation of computing accelerators, exploitation of low power compute units, computing operations, enabling software and computing, reconstruction algorithms, software framework development, development of cross- experiment development tools, use of HPC and development of collision simulation/generation programmes.

- The HEP Software Foundation (HSF) established to meet the demand of a software "upgrade"
  Along with the **SWIFT-HEP** project in the UK

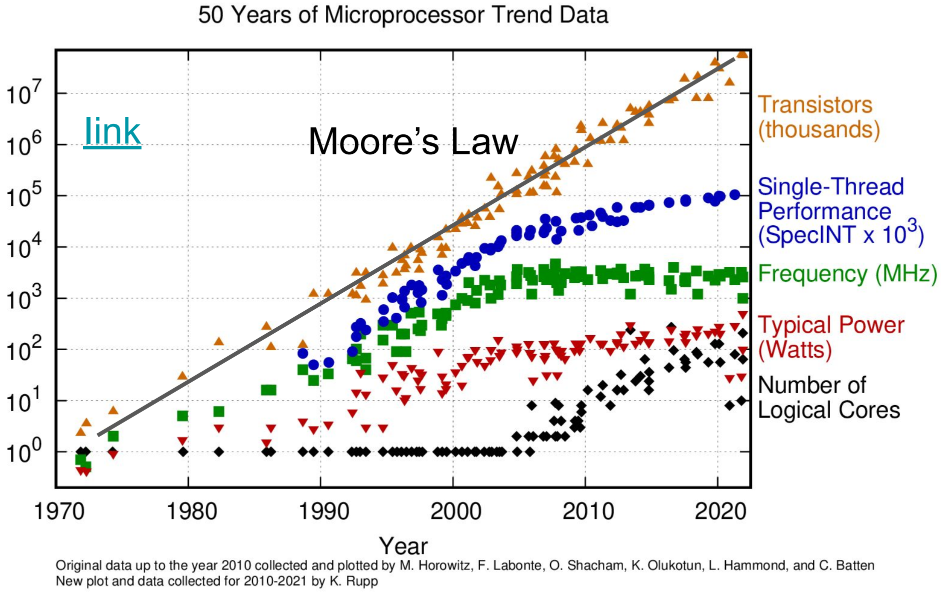
The evolution of computing technology

Moore's Law continues to deliver increases in transistor density

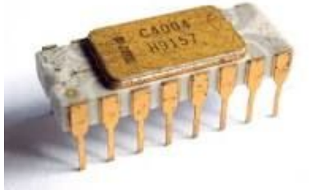
- But, doubling time is lengthening
- Clock speed scaling failed around 2006
- No longer possible to ramp the clock speed as process size shrinks
- Leak currents become important source of power consumption

So we are basically stuck at ~3GHz clocks from the underlying Wm^{-2} limit

- This is the Power Wall
- Limits the capabilities of serial processing
- **Parallelism** is the name of the game



1971: Intel 4004



From 10 um feature size

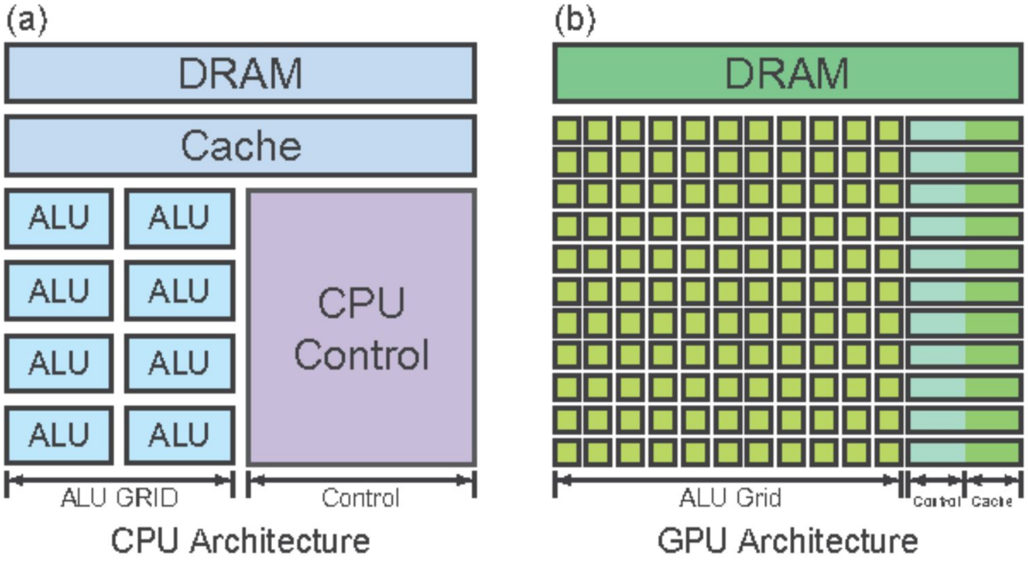
To 1.4 nm feature size (under development)

From CPUs to GPUs and beyond

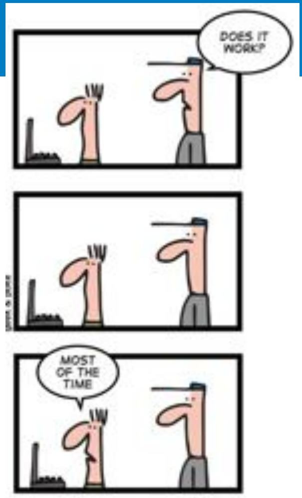
Parallelism is the name of the game
- GPUs focus on arithmetics in multiple threads
- Ideal for applications such as graphics rendering
- Matrix algebra (e.g. deep learning) is well suited

HEP code not so well suited for GPUs
- Potential gains in efficiency to be made
- Requires huge development effort
- Training to use them

Hardware evolution is faster than our time scales
- Heterogeneous landscape
- We can not develop code for one specific architecture and do it again
- Software portability is needed, but not quite there yet



Software and computing infrastructure in HEP



High Energy Physics has a vast investment in software

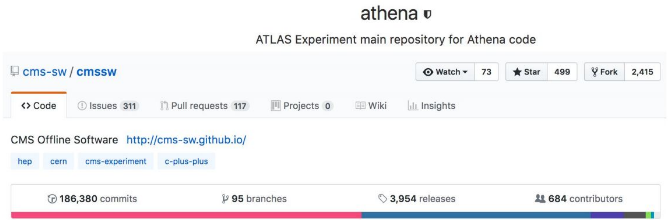
- Estimated to be around 50M lines of C++
- We can’t put a price tag to this, but it is a lot of effort over a long time

Computing is a critical part of our physics production pipeline

- LHC experiments use about 1M CPU cores every hour of every day (as part of the Worldwide LHC Computing grid - WLCG)
- We have around 1EB of data
- 100PB of data transfers per year (10-100Gb links)
- In the UK provided by GridPP and IRIS

This is a huge and ongoing cost in hardware and human effort

With significant challenges ahead of us to support our ongoing physics programme



Outline

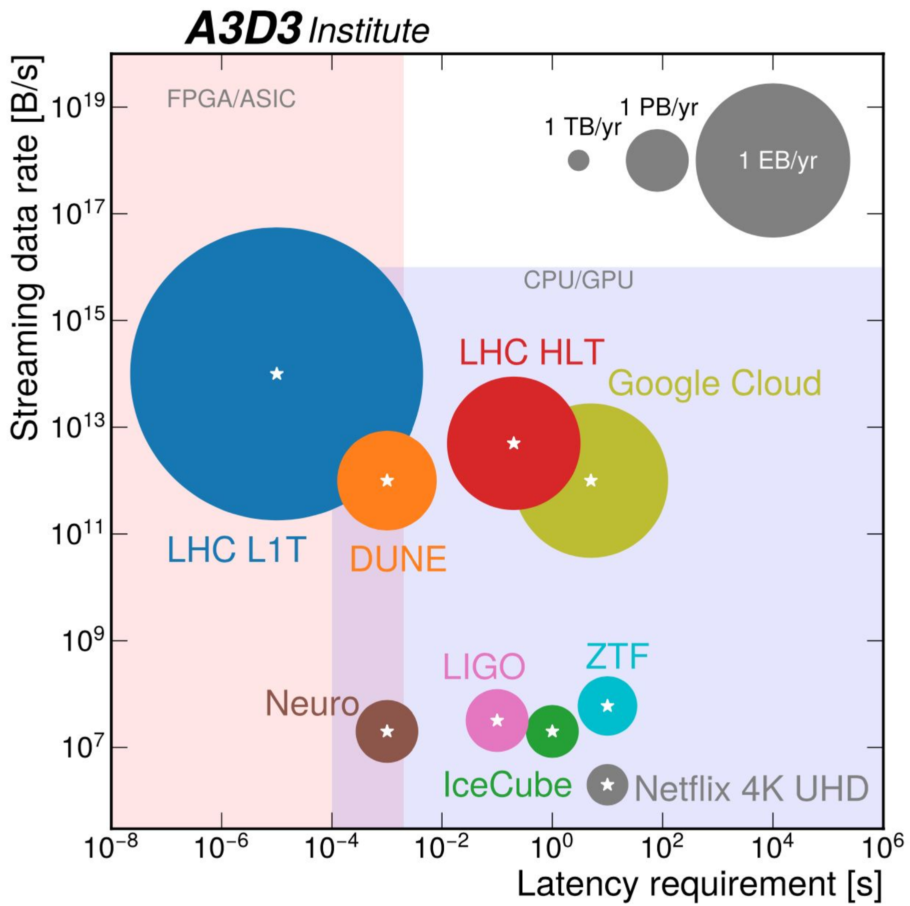
- The case for computing in particle physics
- **Data selection and processing**
- Simulating and designing particle detectors
- Final remarks on
  - Machine learning, career and training, sustainability and quantum computing

Data processing - online

- Data need to be selected, processed, and recorded

- Event rates of 40 MHz at the LHC
(e.g. one collision 1-2MByte every 25 ns)

- Rate of ~1 kHz from Supernova neutrinos
- Rapid processing of Supernova events at DUNE ([EPJC (2021) 81](#))
- Example: (arXiv [2004.090307](#))
  - 3000 charged current neutrinos
  - ~184 TBytes of data
  - 4hrs transfer time (over 100 GBs network)
  - 30000 CPU cores needed to process within the same time

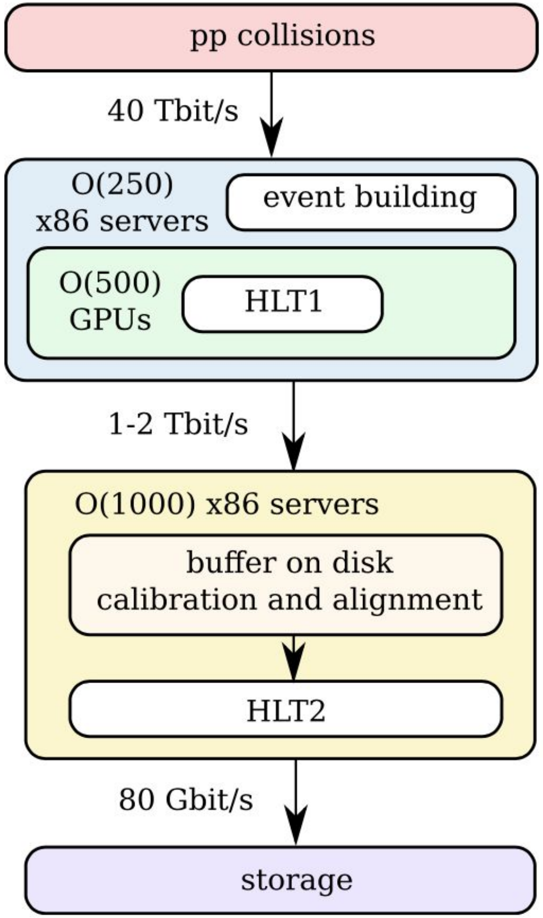


Allen: A High-Level Trigger for LHCb

Track reconstruction as a parallel problem
- Maps well on the GPU

500 GPUs hosted on the same server as event builders
- 40 Tbit/s data ingested and processed
- Full tracking sequence processed on GPUs

HLT2 performs event reconstruction
- Output reduced rate for most selections
- Uses almost real-time alignment data
- Reduces the need for offline processing



Comput Softw Big Sci 4, 7 (2020)

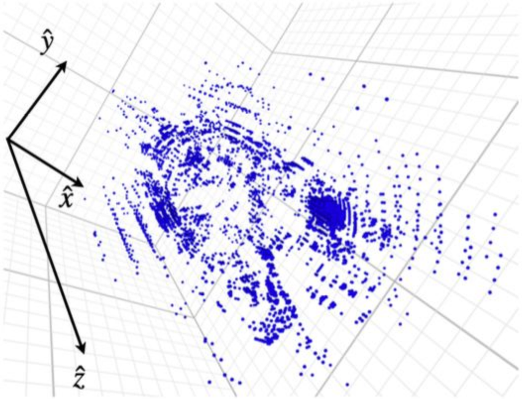
ATLAS: offline reconstruction of hadronic final states

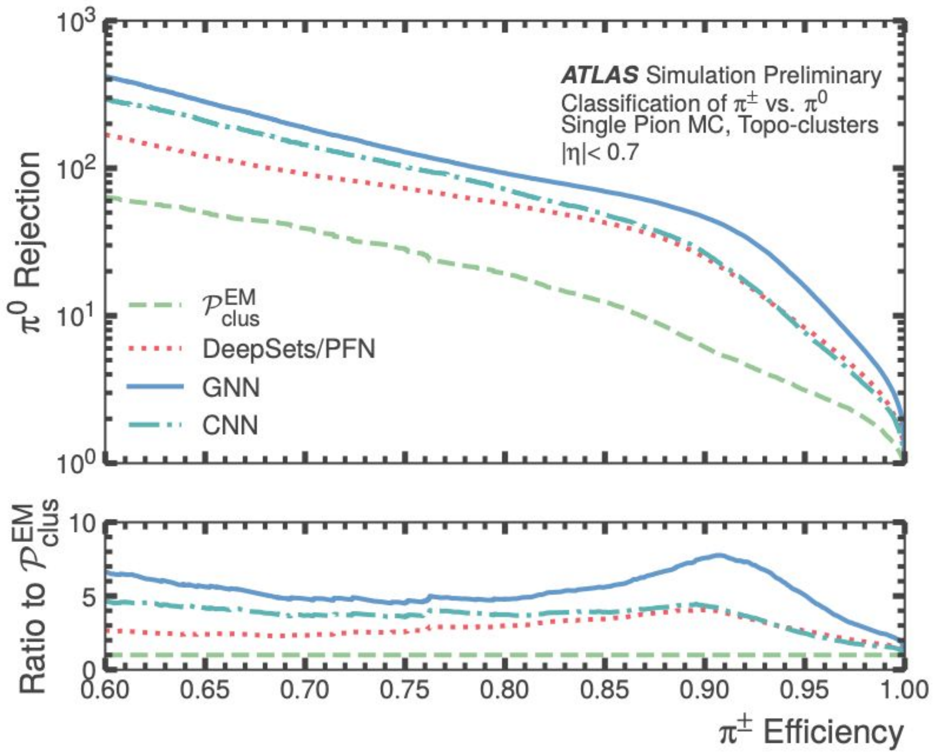
Hadronic final state reconstruction needs to consider multiple input parameters

- Most final state particles are $\pi^{\pm}$ and π^0
- 3D clusters of topologically connected cells (TopoClusters)
- Tracks pointing to the clusters

Multiple Machine Learning techniques can be used

- Comparison shown in plot
- All ML methods outperform the baseline $\mathcal{P}^{EM}$





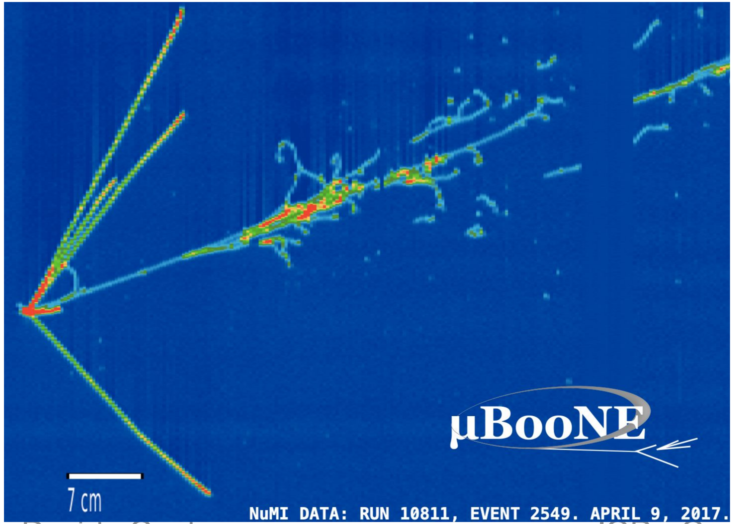
ATL-PHYS-PUB-2022-040

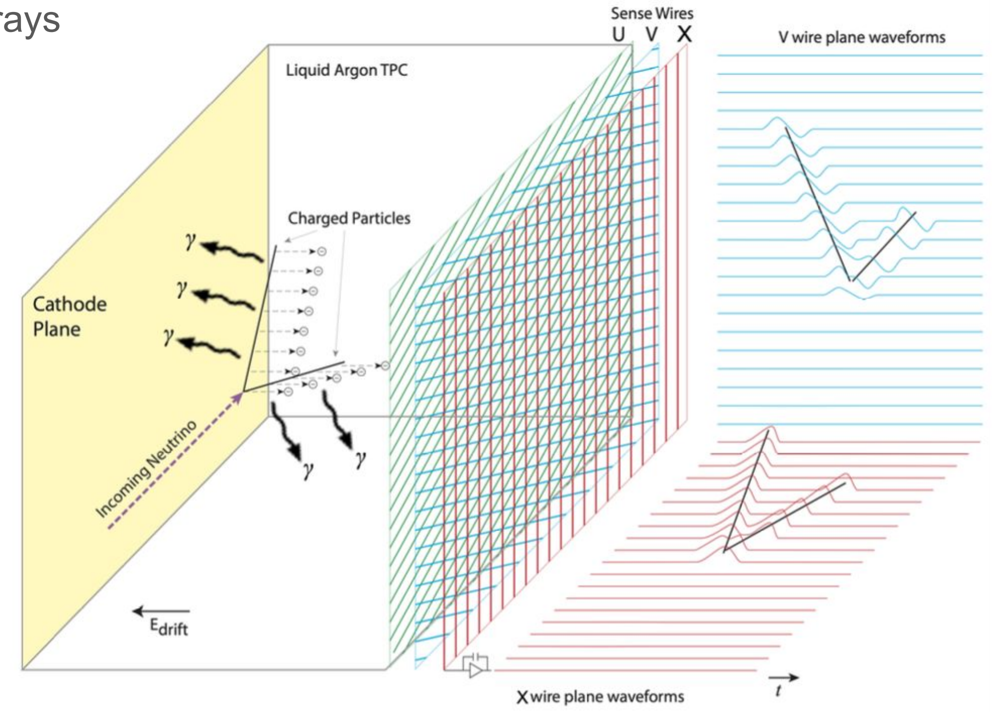
Pandora: Generic LArTPC Reconstruction

3D reconstruction of particle tracks

- Often complicated by the presence of cosmic rays

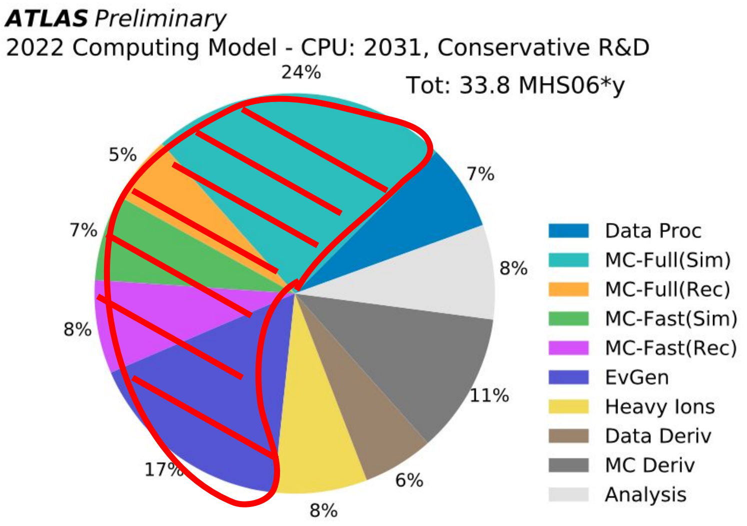
Deep Neural network techniques can be used to improve vertex reconstrution





Outline

- The case for computing in particle physics
- Data selection and processing
- **Simulating and designing particle detectors**
- Final remarks on
  - Machine learning, career and training, sustainability a



Around 60% of CPU time spent on simulating a detector! (link)

The role of Software and Computing

Detector data → Trigger Real-time system

Experiment operation

Reduced data format

Offline processing → User analysis

Initial design
Modeling of running exp

Simulation (digital twin)

- Physics interaction
- Parton shower
- Particle interactions with material
- Detector response

The role of Software and Computing

Detector data

Trigger
Real-ti

Experiment operation

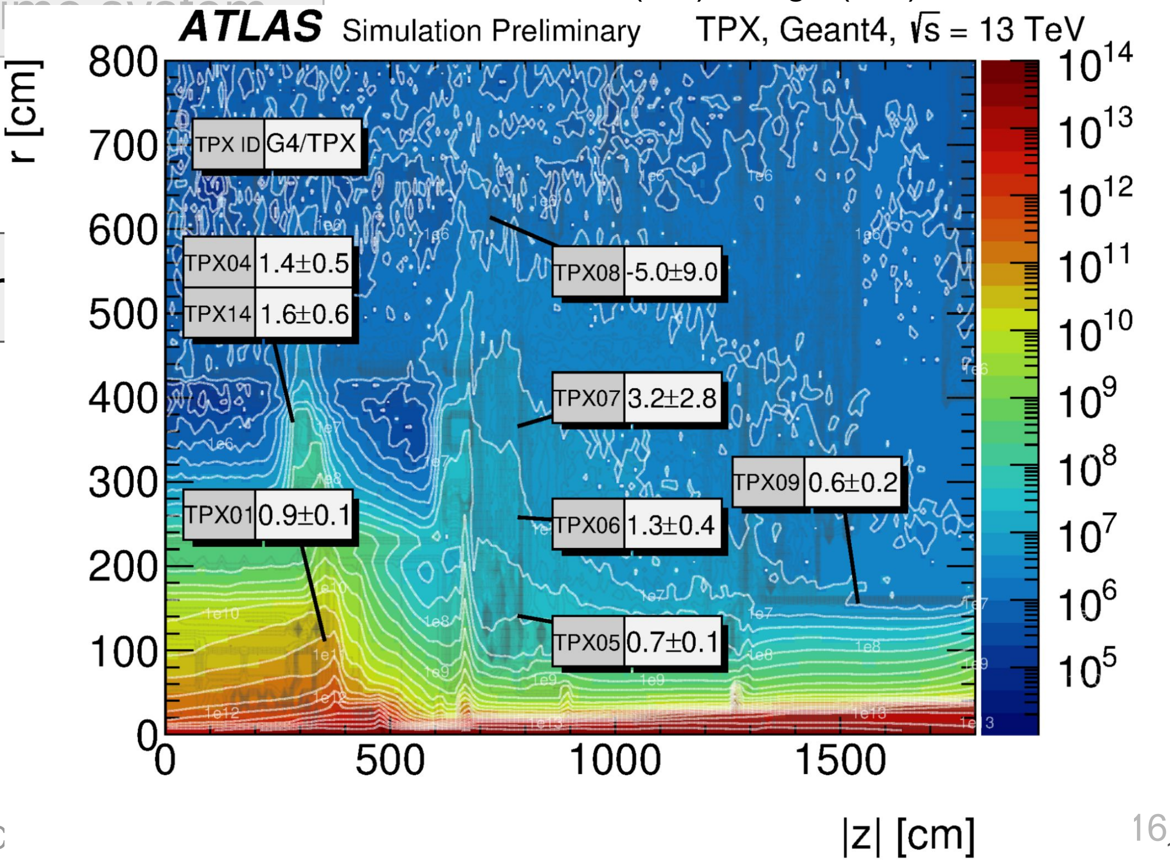
Initial design
Modeling of running exp

Simulation (digital twin)

Offlin

- Physics interaction
- Parton shower
- Particle interactions with material
- Detector response

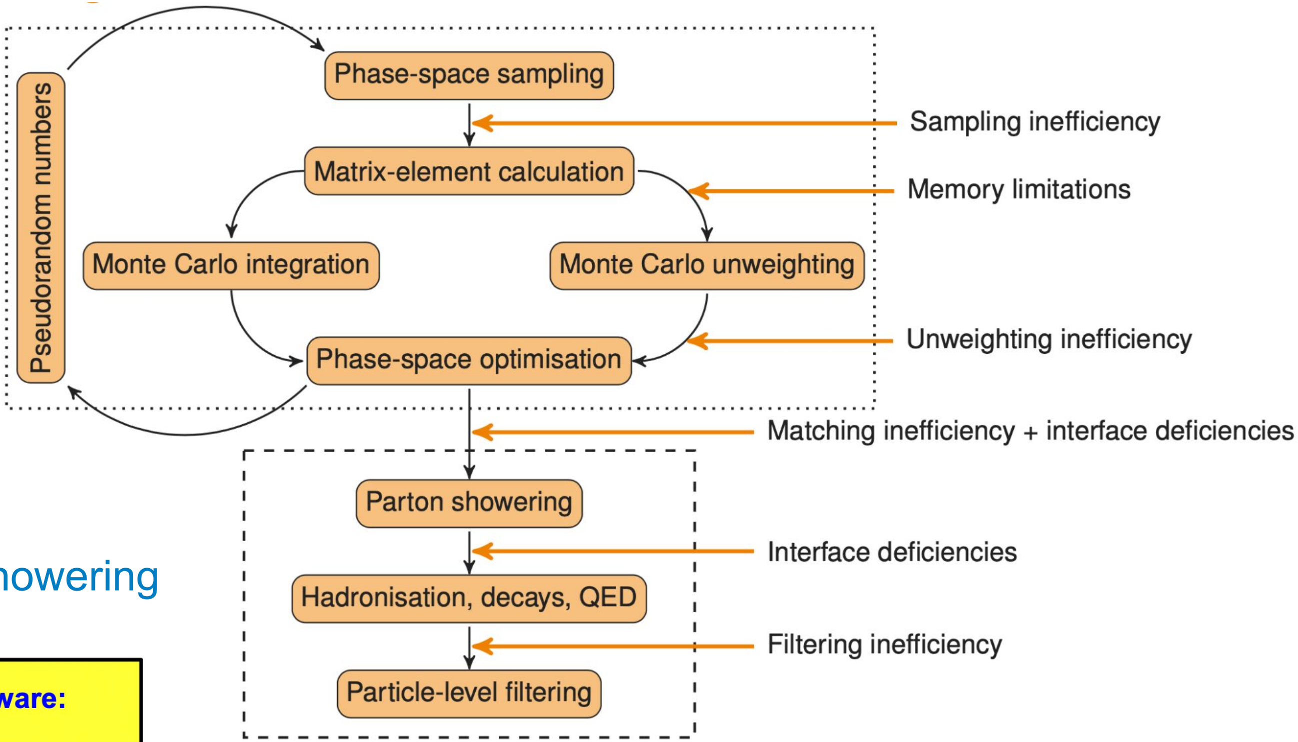
Example: Particle flow studies for the ATLAS inner tracker (ITk) design (link)



Event generators: reproducing the physics

Matrix element

Showering



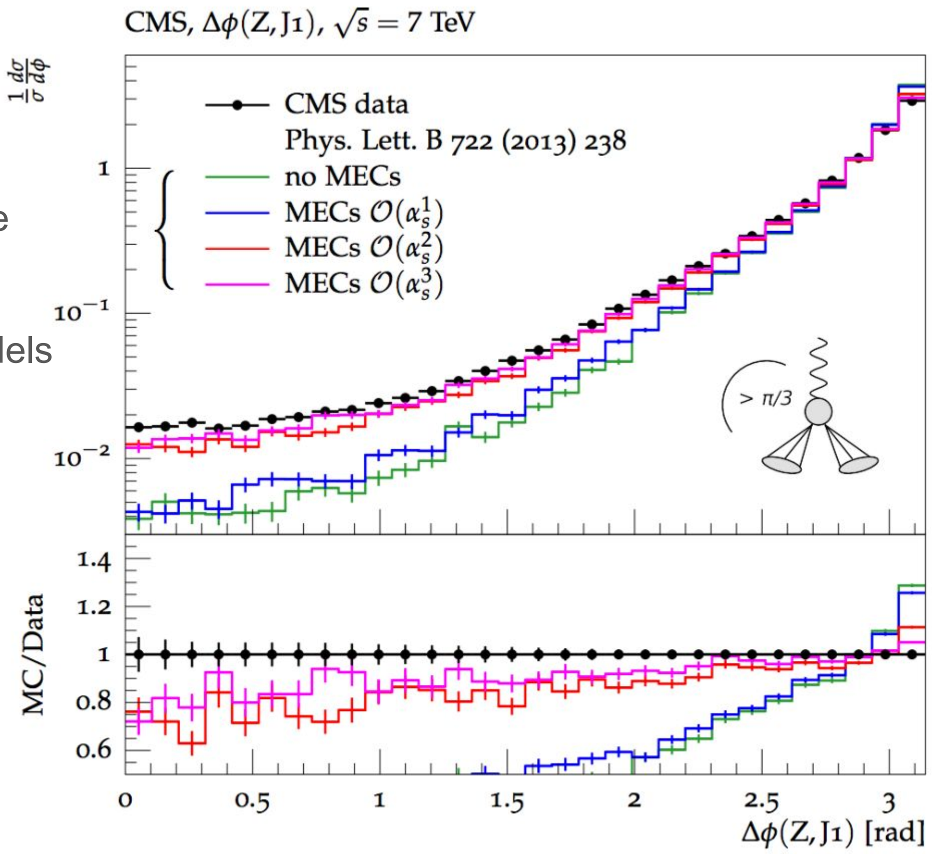
Improved calculations

Better physics requires improved calculations

- Development needed for high precision physics
- Which cost more CPU
- Better physics for fewer KWh is often possible

Maintenance of interfaces is often critical

- Often not priority for theorists developing models



GEANT4: Particles interaction with the detectors

High fidelity simulation of particle detectors

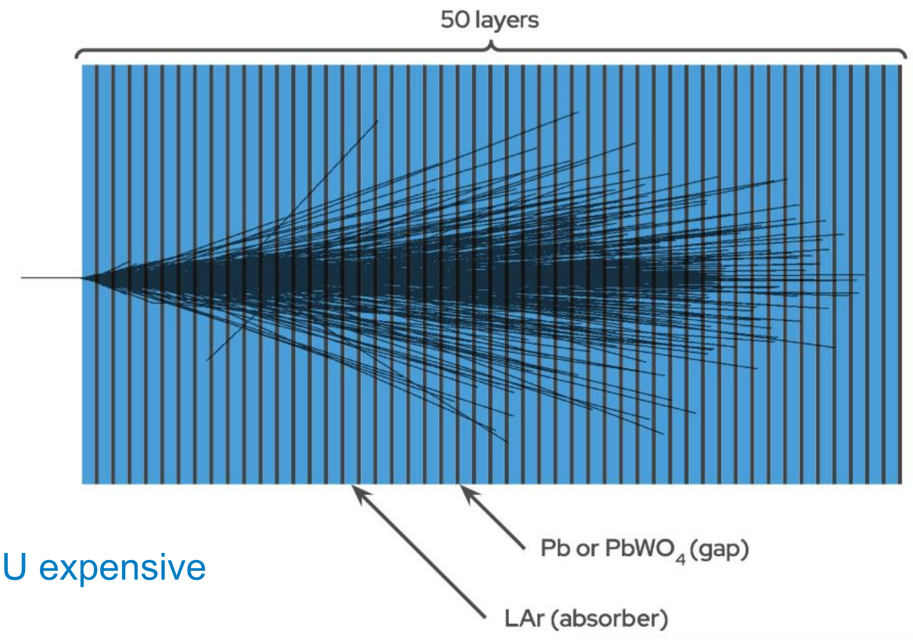
- Track particle interactions
- Reproduces a myriad of physics processes (EM, nuclear, …)

International collaboration

- First version of GEANT in 1984
- Developed in C++ from 1993 (GEANT4)
- UK formally a part of the collaboration

Used beyond particle physics

- E.g. nuclear medicine, space radiation, radiation dose calculations, etc



Full Geant4 simulation of experiments usually very CPU expensive

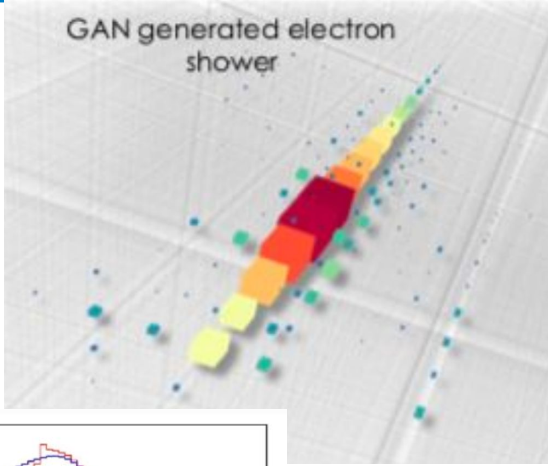
Fast simulation and generative models

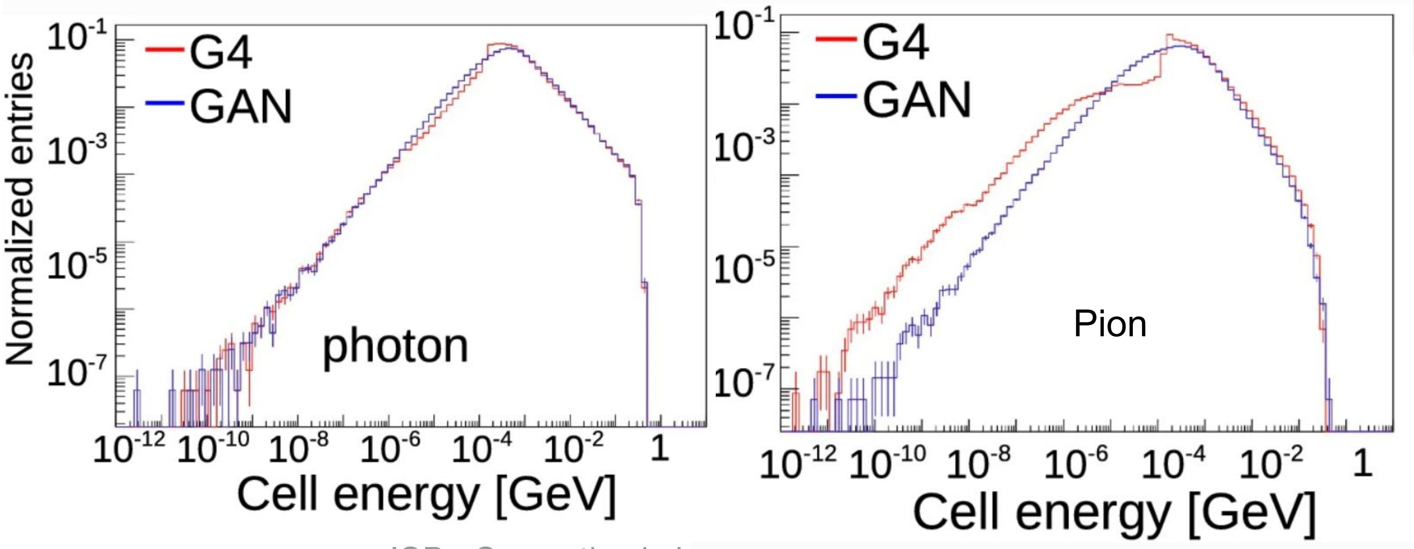
Development of code on GPU to accelerate EM showers

- AdePT, Celeritas

Development of fast simulation, for instance using generative models

- Learn shower shapes and produce alternatives (Eur. Phys. J. C 82, 386 (2022))



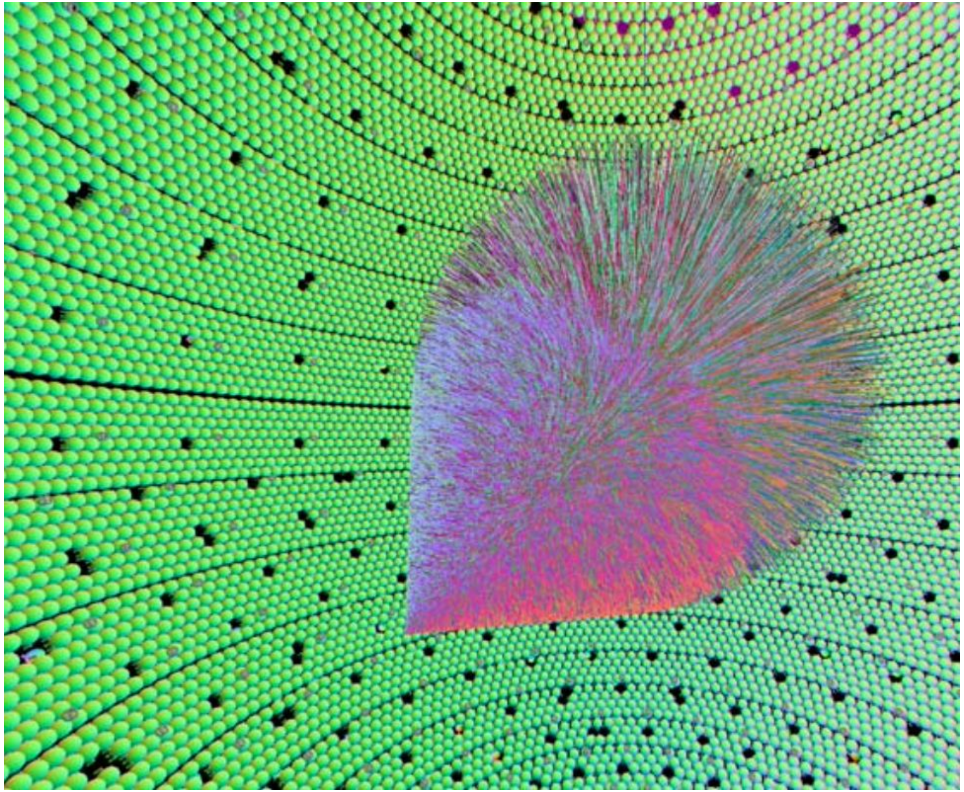


Using GPUs for ray tracing: Opticks

Lots of compute time is spent propagating photons across detectors

- Similar to rendering, which is what GPUs were designed for
- Opticks package interfaced to Geant4

Example: a muon emitting photons inside JUNO



Outline

- The case for computing in particle physics
- Data selection and processing
- Simulating and designing particle detectors
- **Final remarks on**
    - **Machine learning, career and training, sustainability and quantum computing**

Machine Learning and Artificial Intelligence

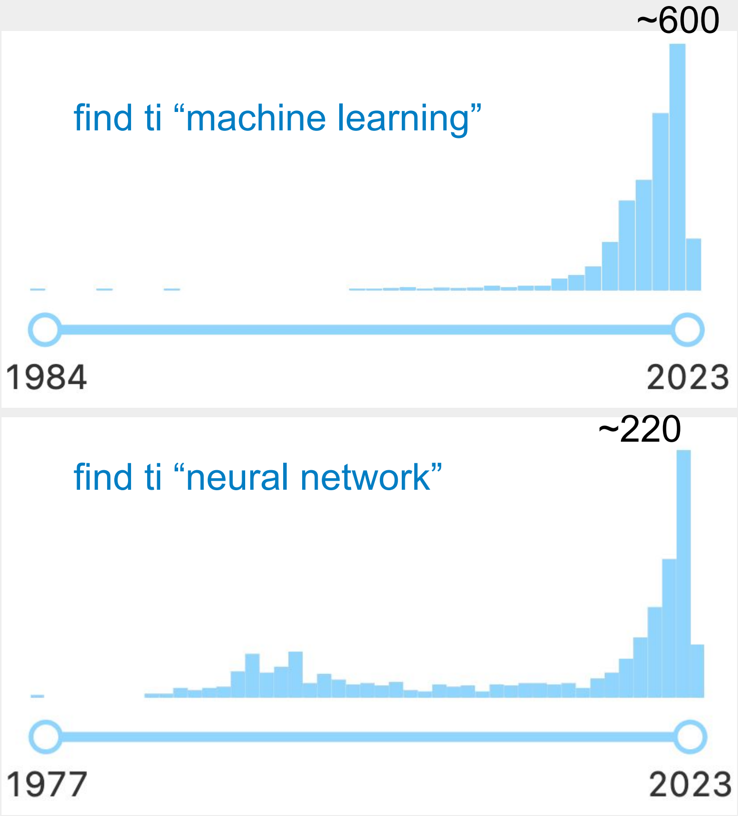
Ubiquitous in particle physics

- I showed a few examples for simulation and reconstruction
- You can find a lot of examples at this conference!
- Often a gateway to "more physics for your buck"

Future developments

- New techniques developed
- Artificial intelligence is driving technology development
- Important to stay on top of it

Inspire-HEP searches:



Sustainability and future compute requirements

Increase of demand for compute resources

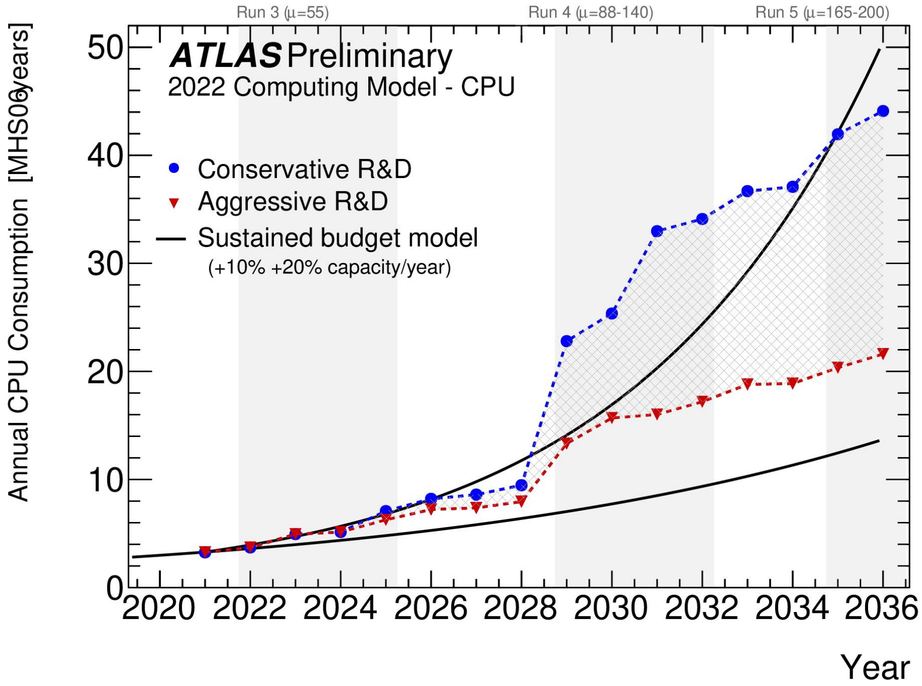
- Driven by advancement in detectors and data acquisition
- And more precision measurements
- This is NOT just the LHC

We clearly can not afford business as usual

- We can't waste resources and energy
- 10-15% per year expected due to an advances in compute techology (e.g. new chips, more efficient data centres,)

For further improvements we need to look at the software

- The case for software upgrades is being made



Quantum computing, the next frontier

Quantum technologies are probably the next big thing

- Quantum sensors are already developed for particle physics
- How can we benefit from a quantum computer?

We are currently in the Noisy Intermediate-Scale Quantum (NISQ) era

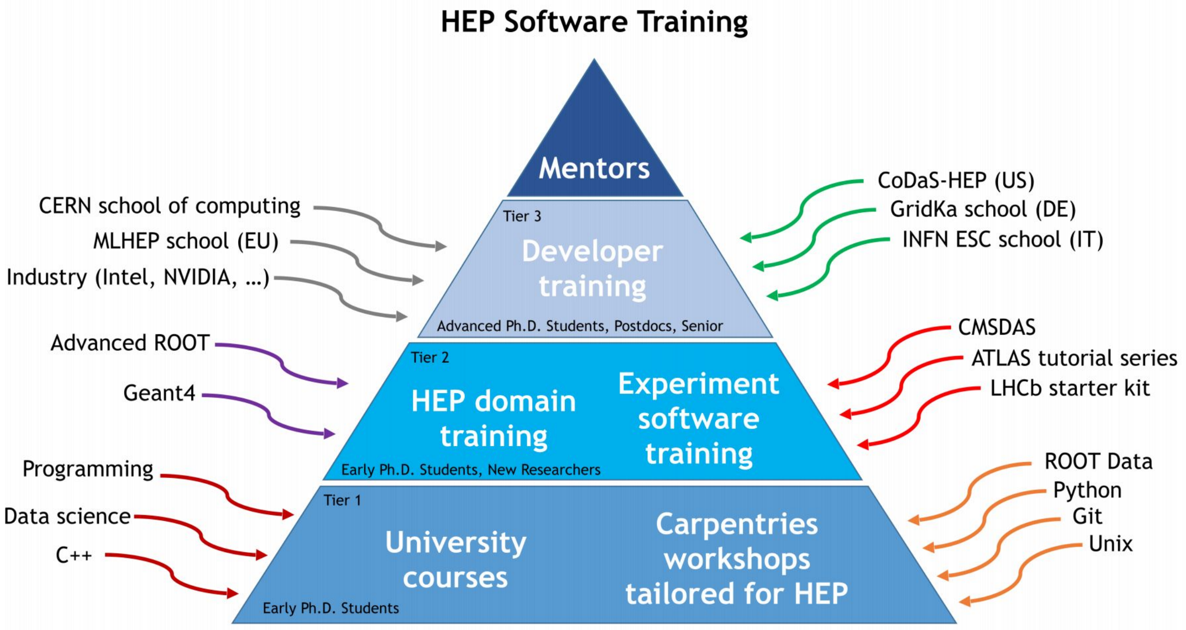
- 50-100 qubits with limited fault tolerance
- Not enough to do some “serious” stuff (yet)

Potential applications include:

- Development of QFT calculations in a “quantum” way
- Parton showers and quantum interference
- Tracking detector reconstruction (as a way to solve combinatorics)
- Quantum machine learning

People: Training and Career development

- The most important resource is people who operate and develop the systems we use
  - As we get older, we need to ensure the new generations are trained to become the new experts
  - It is important that good career prospects in HEP are available
  - Many people leave HEP for (more lucrative) jobs elsewhere. This good news! We can still attract great researchers in our field!
  - But we need to retain a few…



Conclusions

- I gave an overview of the use of software and computing in particle physics
  - Development, maintenance and operation is crucial for the future of our field
  - Computing is not just a bunch of boxes consuming electricity, but includes all the software and the years of development that led us where we are

- Needs and requirements are much experiment dependent
  - But a lot of the basic tools are common, and a joint approach needs to be developed
  - International collaboration are key, as in most of our experiments
  - Emerging new techniques can be the crucial for our future successes

- People are the most important resource we have
  - We can always buy more computer (perhaps...)
  - The experience of the people who make it happen is much harder to replace
  - Training and career developments are essential to form the next generation and new ideas