Software framework options for Run 5

M. Clemencic

6th Workshop on LHCb upgrade II – March 30, 2023

CERN - LHCb

Table of contents

1. A bit of history

2. Lessons to be learned

3. Can we plan for Run 5?

4. How to proceed

A bit of history

What happened since Gaudi inception

2000 Gaudi

- still in the golden era of Moore's law
- every year the application was faster (just change the CPU)
- state of the art practices meant compute on demand and use pointers
- hidden in a niche, somebody introduced programmable shaders

What happened since Gaudi inception

2000 Gaudi

- still in the golden era of Moore's law
- every year the application was faster (just change the CPU)
- state of the art practices meant compute on demand and use pointers
- hidden in a niche, somebody introduced programmable shaders

2006 Intel Core CPU (dual-core)

- Moore's law still valid if applied to the die
- single core performance not improving as fast as before

What happened since Gaudi inception

2000 Gaudi
- still in the golden era of Moore's law
- every year the application was faster (just change the CPU)
- state of the art practices meant compute on demand and use pointers
- hidden in a niche, somebody introduced programmable shaders

2006 Intel Core CPU (dual-core)
- Moore's law still valid if applied to the die
- single core performance not improving as fast as before

2009 GPUs are used in TOP500 HPCs

2011 ARMv8 (64bits)
- ARM chips are not anymore just toys for mobile devices

2020 Fugaku, the fastest HPC, uses ARM processors

What happened since Gaudi inception

2000 Gaudi
- still in the golden era of Moore's law
- every year the application was faster (just change the CPU)
- state of the art practices meant compute on demand and use pointers
- hidden in a niche, somebody introduced programmable shaders

2006 Intel Core CPU (dual-core)
- Moore's law still valid if applied to the die
- single core performance not improving as fast as before

2009 GPUs are used in TOP500 HPCs

2011 ARMv8 (64bits)
- ARM chips are not anymore just toys for mobile devices

2020 Fugaku, the fastest HPC, uses ARM processors

2022 Allen, GPU based LHCb Hlt1

Current landscape

- The HPC hype is still high
  - money goes into HPCs and not into old-style computing centers
  - HPCs are the new off-the-shelf resources

Current landscape

- The HPC hype is still high
  - money goes into HPCs and not into old-style computing centers
  - HPCs are the new off-the-shelf resources
- We transitioned from multi-core to many-core
  - latest CPUs offer 30-90 cores
  - GPUs have thousands of computing cores

Current landscape

- The HPC hype is still high
  - money goes into HPCs and not into old-style computing centers
  - HPCs are the new off-the-shelf resources
- We transitioned from multi-core to many-core
  - latest CPUs offer 30-90 cores
  - GPUs have thousands of computing cores
- More specialized hardware is introduced every day
  - tensor-cores added to GPUs
  - graphic cards with ray-tracing dedicated hardware
  - FPGA companies were acquired by CPU companies

Current landscape

- The HPC hype is still high
  - money goes into HPCs and not into old-style computing centers
  - HPCs are the new off-the-shelf resources
- We transitioned from multi-core to many-core
  - latest CPUs offer 30-90 cores
  - GPUs have thousands of computing cores
- More specialized hardware is introduced every day
  - tensor-cores added to GPUs
  - graphic cards with ray-tracing dedicated hardware
  - FPGA companies were acquired by CPU companies
- AI hype is off the roof
  - we can expect more money to go in optimizing AI workloads

Lessons to be learned

Where are we going?

- Like it or not, the world thinks the way forward is *parallelism*
  ... and specialized hardware
- We have to consider hardware limitations too, for example:
  - sometime is faster to compute than access main memory
  - cache coherency between cores might limit parallelism

Was Gaudi design wrong?

- Gaudi design was not wrong
  - SMP was already used, but nobody expected the change of paradigm
  - nobody could imagine that making games look cooler could lead to GPUs
- Could we have done better?

Was Gaudi design wrong?

- Gaudi design was not wrong
  - SMP was already used, but nobody expected the change of paradigm
  - nobody could imagine that making games look cooler could lead to GPUs
- Could we have done better? probably not
  - nothing was hinting to what we have seen
- Can we still use Gaudi?

Was Gaudi design wrong?

- Gaudi design was not wrong
  - SMP was already used, but nobody expected the change of paradigm
  - nobody could imagine that making games look cooler could lead to GPUs
- Could we have done better? probably not
  - nothing was hinting to what we have seen
- Can we still use Gaudi? probably not
  - the world will continue on this road or find something new
  - either way, Gaudi design cannot scale

Can we plan for Run 5?

Staring hard at a crystal ball...

- We will continue on the current path for a while, e.g.
  - more powerful GPUs
  - more CPU cores with larger vector units
  - dedicated hardware (e.g. using embedded FPGAs)
  - variety of architectures
- Something new and unexpected will come
  - sooner or later, but sure it will come
  - can it be commodity quantum processors?
  - positronic brains?

What will we need?

We can try to imagine what a framework will have to address in 10 years time

- High scalability
  - the work must be split and distributed as much as possible
  - it's not enough to process events one by one
- Arbitrary architectures
  - support for x86_64, armv8, different flavours of GPUs, ...
- Flexibility
  - easy for users (like with functors)
  - one code to run everywhere
  - allow specific optimizations

Use cases to support

However we are going to do it, a few points must be taken into account

use cases

- Online computation / RTA
  - run Hlt1 and Hlt2 applications
  - monitoring and alignment tasks
- Offline analysis / DPA
  - Sprucing
  - analysis productions
- Simulation
  - full and fast simulations

resources

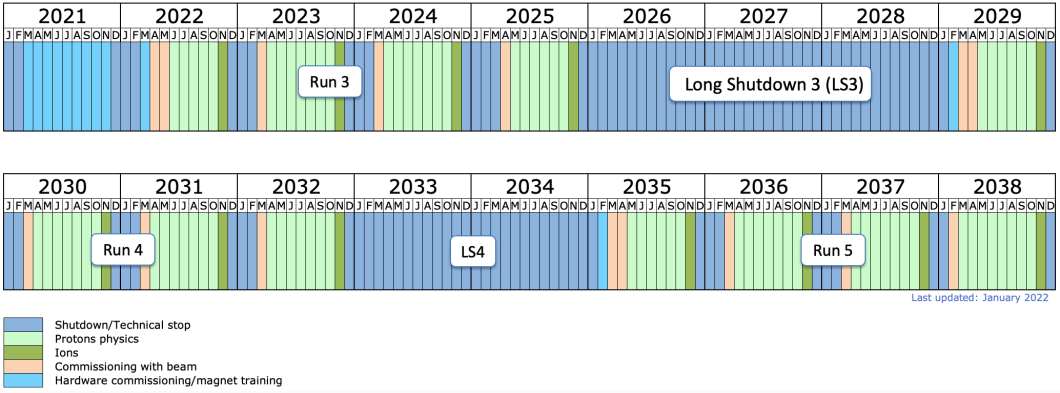
- Online farm
  - Event Builder farm
  - Hlt2 farm
- Offline resources
  - Grid Computing Elements (?)
  - HPCs
  - clouds

How to proceed

A new project

- Gaudi and Allen proved themselves valuable tools
- We will probably need something more to meet Run 5 needs
- Shift focus
  - from loop over *events* to (e.g.) stream processing
- We should join forces
  - combine Gaudi and Allen experiences
  - other communities are interested
  - LHCb can be the driving force

Time scale



In LS4 we will have to install the new detector.

We may use LS3 to concentrate on the commissioning of a new framework.

Technologies

There's no need to decide anything yet, but looking around does not harm

- Intel oneAPI
  a common developer experience across accelerator architectures
- SYCL
  abstractions to enable heterogeneous device programming

We can think about an HEP set of abstractions over existing technologies

- *C++/DPC++/SYCL "algorithms"*
- a functor based Domain Specific Language
- Gaudi-like services

Summary

Summary

- It's hard to imagine what's going to happen in 10 years
- I'm not sure we have the right tools yet
- We can definitely profit from a larger collaboration
  - we have to make sure LHCb has enough weight
- A workshop towards the end of the year will be good
  - bootstrap the project
  - define the terms of the collaboration