

# GPU Computing and ALICE

**Vikas Singhal**

Variable Energy Cyclotron Centre

*vikas@vecc.gov.in*

July 4, 2025



- 1 First Level Event Selection algorithm for MuCh CBM via GPU Computing
- 2 GPU programming frameworks
- 3 ALICE GPU Computing
- 4 Non-Deterministic and Deterministic
- 5 Discussions and Outlook

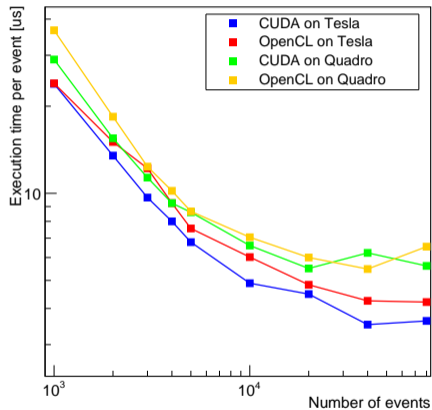
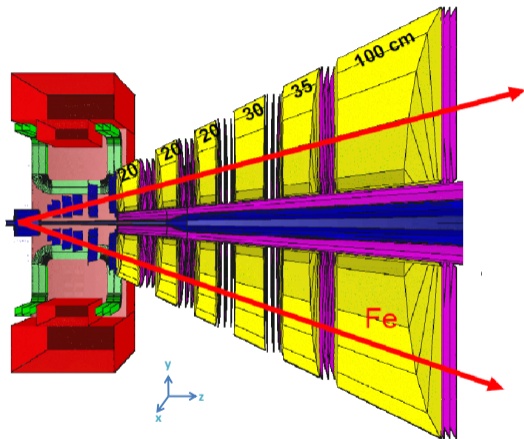


Figure: FLES for MuCh CBM and GPU Computing Paradigms

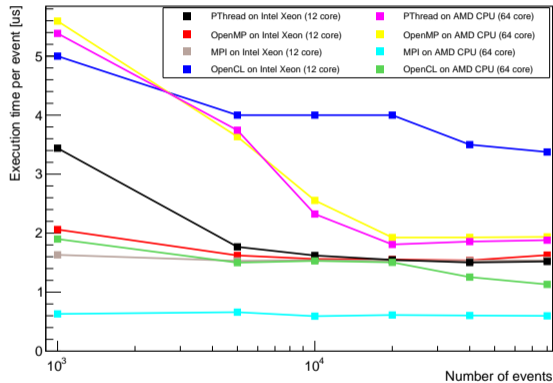
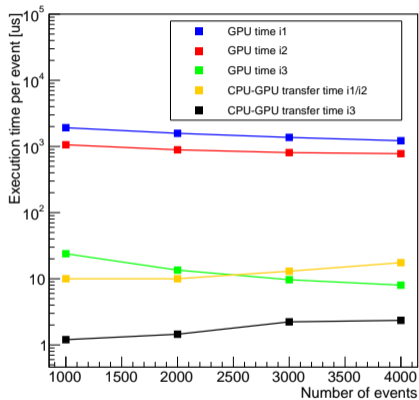


Figure: GPU Execution Time Optimization and Pure Parallel Paradigms



**A modern computer architecture: multiple dimensions to parallelism for high performance computing.**

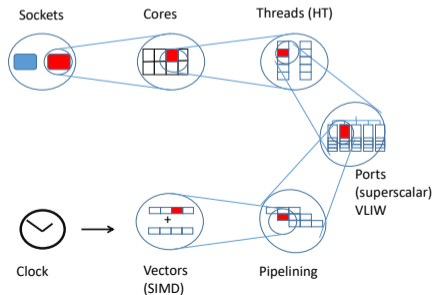




Figure: Architectural Difference between CPU & GPU

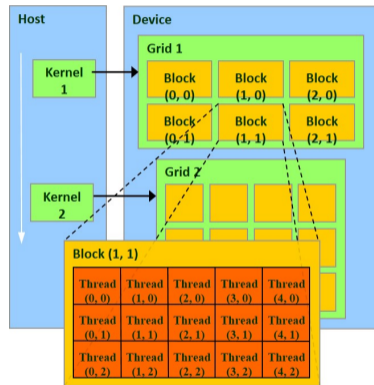


Figure: GPU's Grid, Block, Thread logic



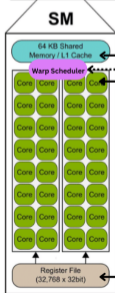
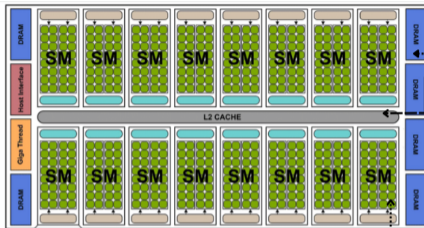
Feature	CPU	GPU
Core Count	Few powerful cores	Thousands of simpler cores
Control Unit	Advanced	Lightweight
ALUs	Limited per core	Massive number per chip
Cache Size	Large (L1, L2, L3)	Smaller
Thread Concurrency	Dozens	Tens of thousands
Latency vs Throughput	Low latency	High throughput
Designed For	Logic-heavy tasks	Data-parallel tasks

- GPUs use special memory structures like shared memory, registers, etc.
- CUDA: A proprietary framework by NVIDIA tailored for their GPUs.
- OpenCL: An open standard maintained by the Khronos Group for heterogeneous platforms including CPUs, GPUs, and FPGAs.
- HIP (Heterogeneous Interface for Portability): Developed by AMD to enable portable code across both AMD and NVIDIA hardware.

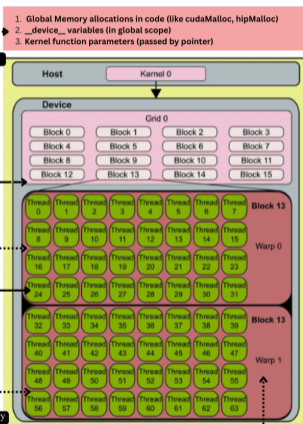
# Mapping of GPU Hardware Structure to Logical Programming Structure



PHYSICAL / HARDWARE MODEL OF GPU



MEMORY / LOGICAL MODEL OF GPU



- Global memory is allocated using `cudaMalloc/hipMalloc`
1. Global Memory allocations in code (like `cudaMalloc`, `hipMalloc`)
  2. `_device_` variables (in global scope)
  3. Kernel function parameters (passed by pointer)

L2 Cache accelerates global memory reads/writes by mapping into individual Grids

Thread blocks (or Blocks) are scheduled on individual SMs

Threads execute on individual CUDA cores (ALUs)

Thread blocks (or Blocks) are scheduled on individual SMs

L1 Cache or On-chip shared memory is used for block-level data sharing

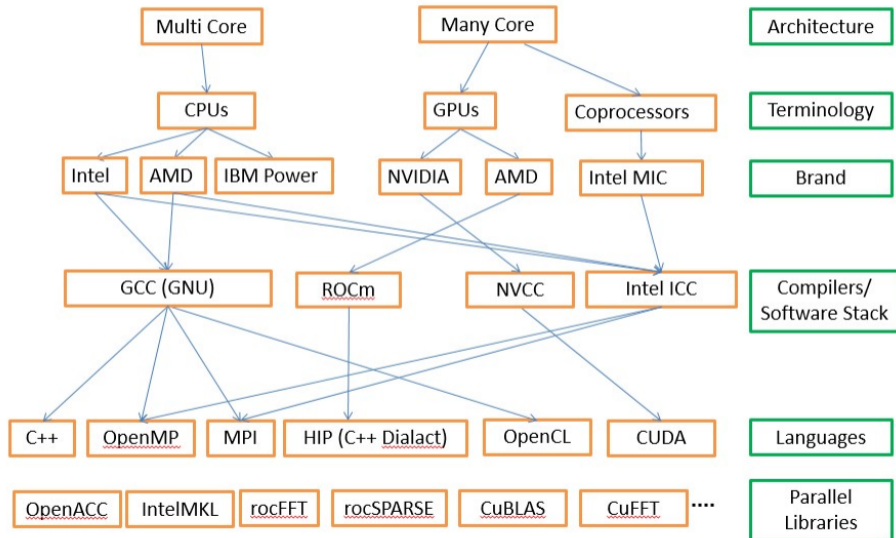
Warp scheduler issues instructions for 32-thread warps

Registers store per-thread local variables



Table

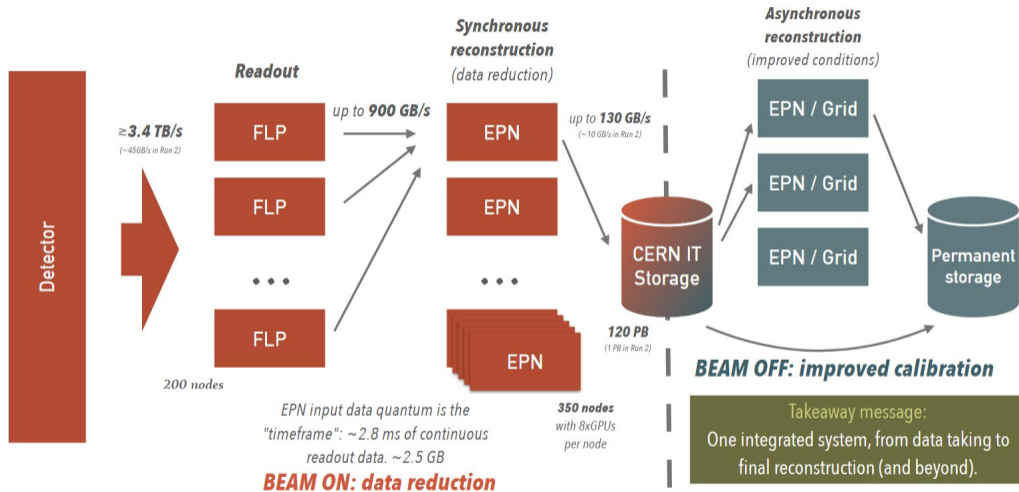
Feature	CUDA	HIP(NVIDIA)	HIP(AMD/ROCm)	OpenCL
Developer	NVIDIA	AMD	AMD	Khronos Group
Hardware Support	NVIDIA GPUs	NVIDIA GPUs (CUDA backend)	AMD GPUs (ROCm)	Multi-vendor
Language Basis	C/C++	CUDA-like C++	CUDA-like C++	C99-like
Portability	Limited to NVIDIA	Code-portable	Code-portable	High
Toolchain	nvcc, Nsight	hipcc (CUDA)	hipcc (ROCm)	clcc, profilers
Ease of Use	High	High	Moderate	Moderate to Low
Performance Tuning	High	High	ROCm tuning required	Medium





- Four ALICE Barrel Detectors
  - ITS (Inner Tracking System)
  - TPC (Time Projection Chamber)
  - TRD (Transition Radiation Reaction)
  - TOF (Time Of Flight)
- ITS and TPC are the most upgraded before Run-3
- High interaction rate
- Upto run 2, LHC provided 2kHz in p-p collisions, from Run 3 it is 50kHz for pb-pb collisions.
- This causes a huge data load.
- The High Level Trigger readout is changed to continuous time readout.
- An online-offline O2 process has been introduced that consists of synchronous and asynchronous phases.

# FLP (First Level Processor) and EPN (Event Processing Nodes) Data Flow



1

<sup>1</sup>David Rohr, Giulio Eulisse, CHEP 2023

# Alice O2 Data Processing Device Execution using ZMQ (Zero Message Queue)

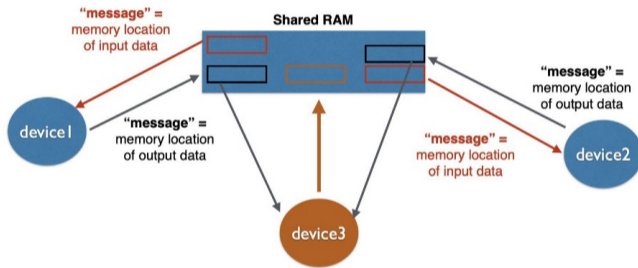


Figure: ALICE O2 Device



## Synchronous Processing

- During data taking – Detector calibration and data compression.
- The main workload is due to TPC
- The output – CTFs(Compressed Time Frames).
- Stored at an On-Site disk buffer.

## Asynchronous Processing

- Reprocesses and generates the final reconstruction output.
- In the O2 farm and GRID.
- During no beam or pp collision, only a part of the asynchronous process is performed on EPNs.

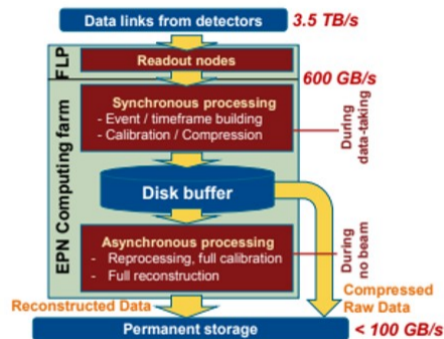
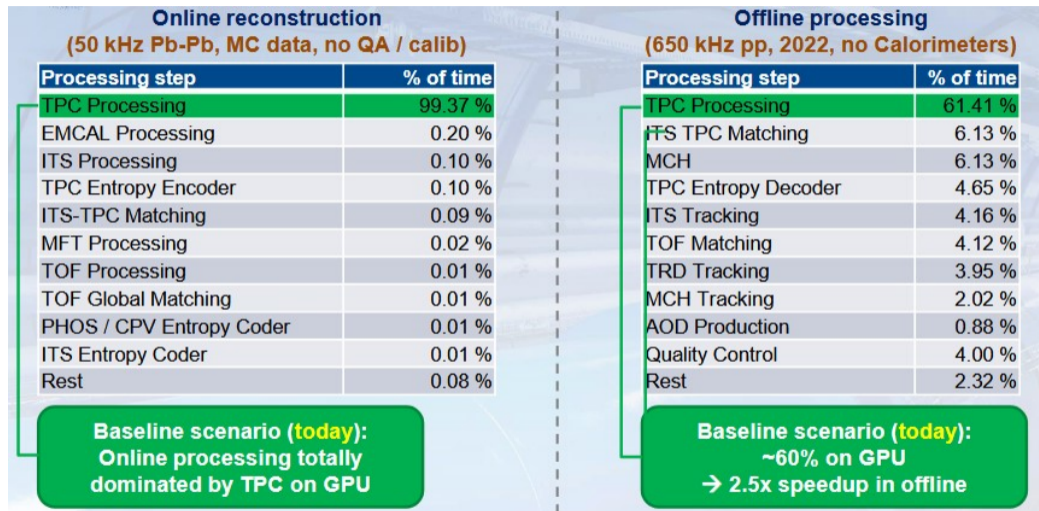


Figure: ALICE O2 Data Flow





- On GPU- The TPC tracking code is not fully deterministic, i.e. running multiple times on the same data set might yield a slightly different number of tracks on the O(per mille) level.
  - Due to concurrency.
  - Compile options and optimizations play a roll – ffast-math or fused-multiply-add might slightly change the rounding of floating point.
- ALICE O2 GPU Deterministic Mode by David Rohr
- It makes debugging, testing, and validation difficult.
- Deterministic mode is implemented, which should yield same reproducible results, on CPU and on GPU.
- To enable - use Compile Time Options
- and Run Time Options.
- For testing, learning and benchmarking a standalone benchmark mode is available.
- Deterministic mode is for correctness.
- For performance measurements use Non-deterministic mode.



- Online computing farm of 350 servers, 2800 GPUs.
- During LHC operation:
  - Online processing on GPUs.
  - Compressed raw data stored to disk buffer.
- When no beam in LHC
  - Offline processing data from disk buffer on online farm.
  - Offline processing always running in the GRID
- Offload more asynchronous reconstruction algorithms to GPU in offline.
- eg: ITS TPC Matching, ITS Tracking, TOF Matching, TRD Matching, etc
- Other subsystems offline codes.



## rANS Encoding

- Time frame  $\rightarrow$  Compressed Time Frame (CTF).
- ALICE Compressed Time Frame (CTF) passes through an rANS encoding.
- Presently it is performed on the CPU.
- Challenging academic problem.

**Thank You**

(Guidance from David Rohr, CERN and ALICE material (David Rohr, CHEP2024))