High Performance Computing in High Energy Physics

presentation hold by

Raphael Skuza

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

INTERNATIONAL MAX PLANCK RESEARCH SCHOOL PT FS FOR PRECISION TESTS OF FUNDAMENTAL SYMMETRIES

MAX-PLANCK-INSTITUT FÜR KERNPHYSIK HEIDELBERG

Agenda

HPC

Agenda

HPC

Motivation

Agenda

HPC

Motivation

What is HPC?

Agenda

HPC

Motivation

What is HPC?

Design

High Energy Physics: LHCb

High Energy Physics: LHCb

Evolution of Experiment

High Energy Physics: LHCb

Evolution of Experiment

Implications for Physics

Software Challenges

Software Challenges

Race Conditions

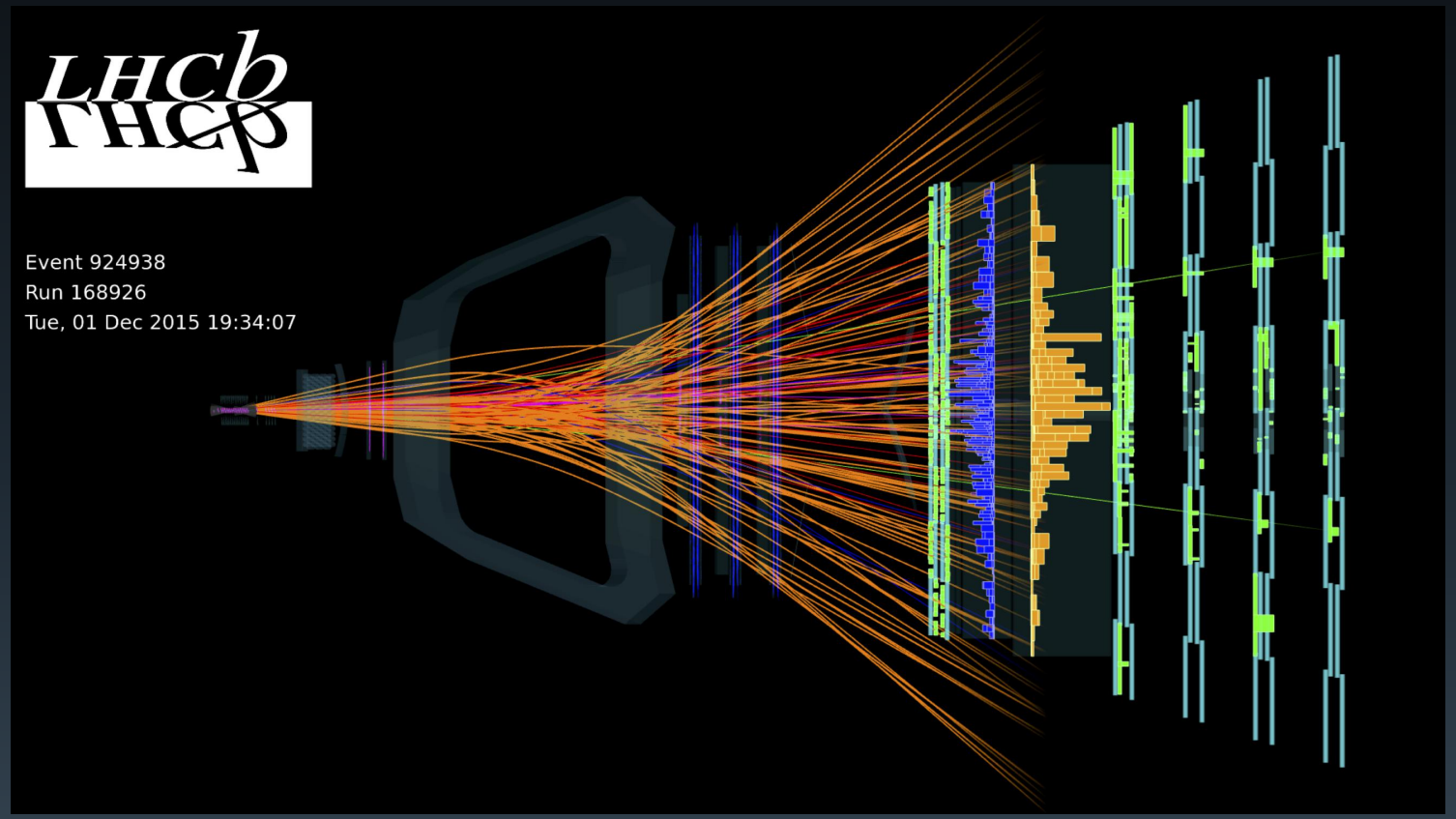
Software Challenges

Race Conditions

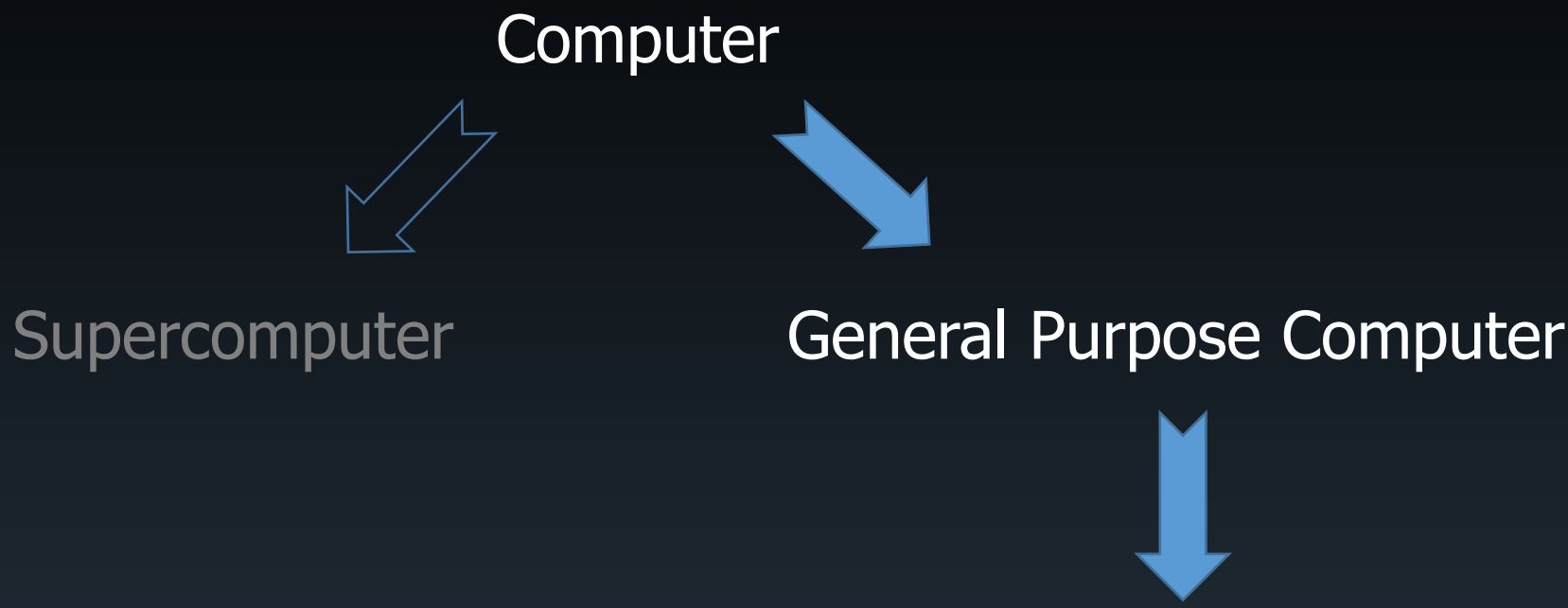
Deadlocks

Motivation: HEP

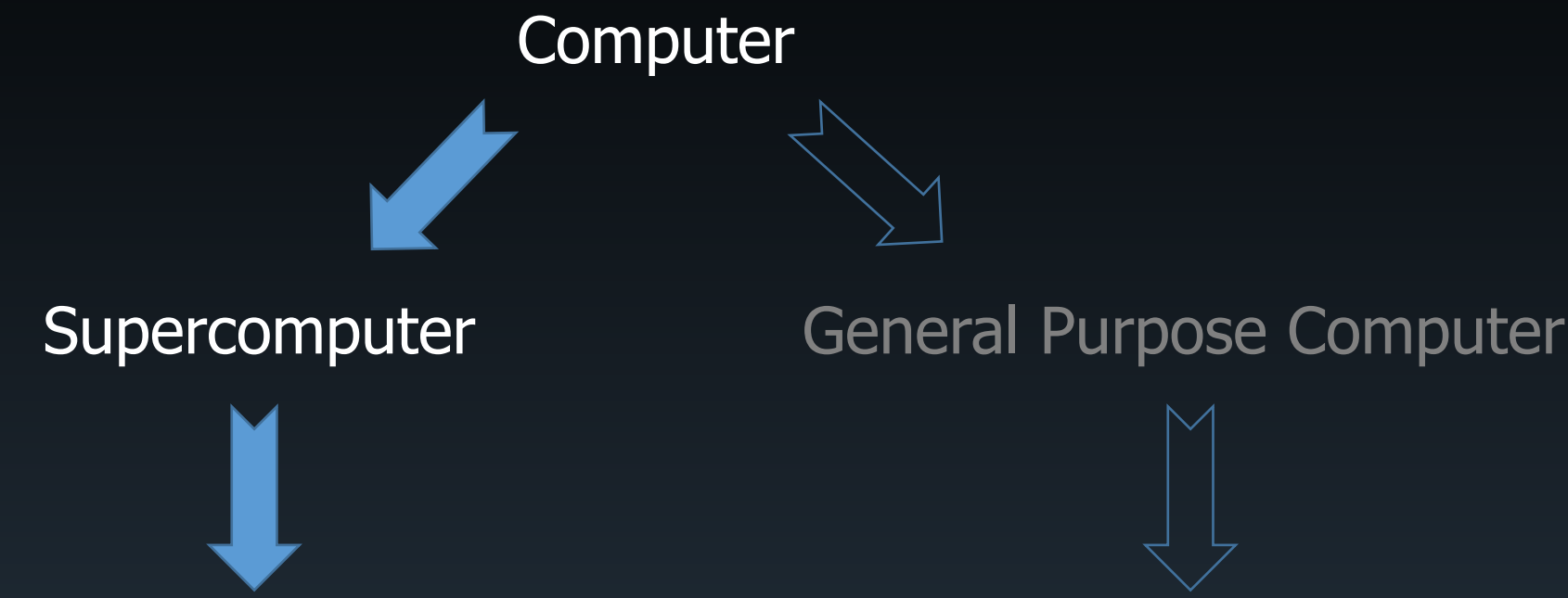
Pattern Recognition and
Parallel Processing for Triggering



Basic Classification



Basic Classification

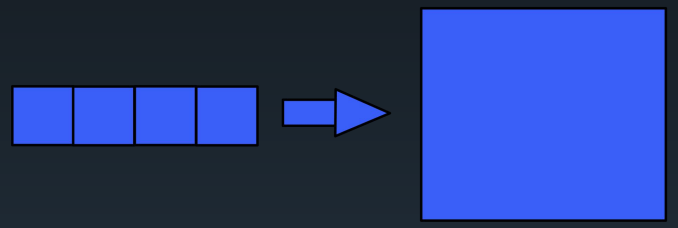


Design Principles

Goal: Maximize the maximum computing power given a fixed budget and specific application field

Past

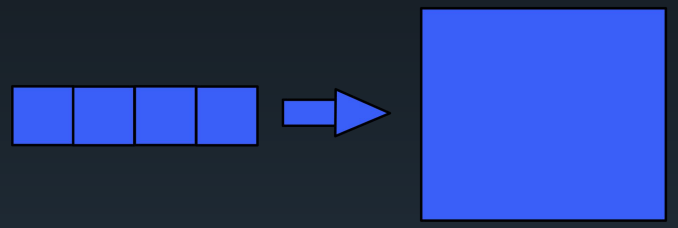
Present



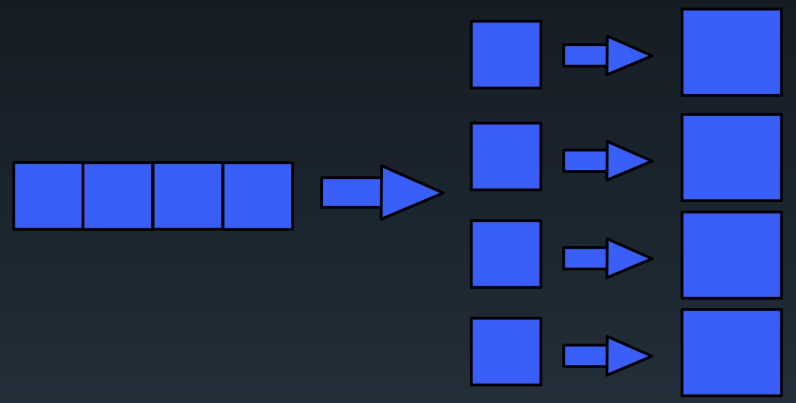
Design Principles

Goal: Maximize the maximum computing power given a fixed budget and specific application field

Past



Present



Massive Parallel Architectures

Designs for HPC

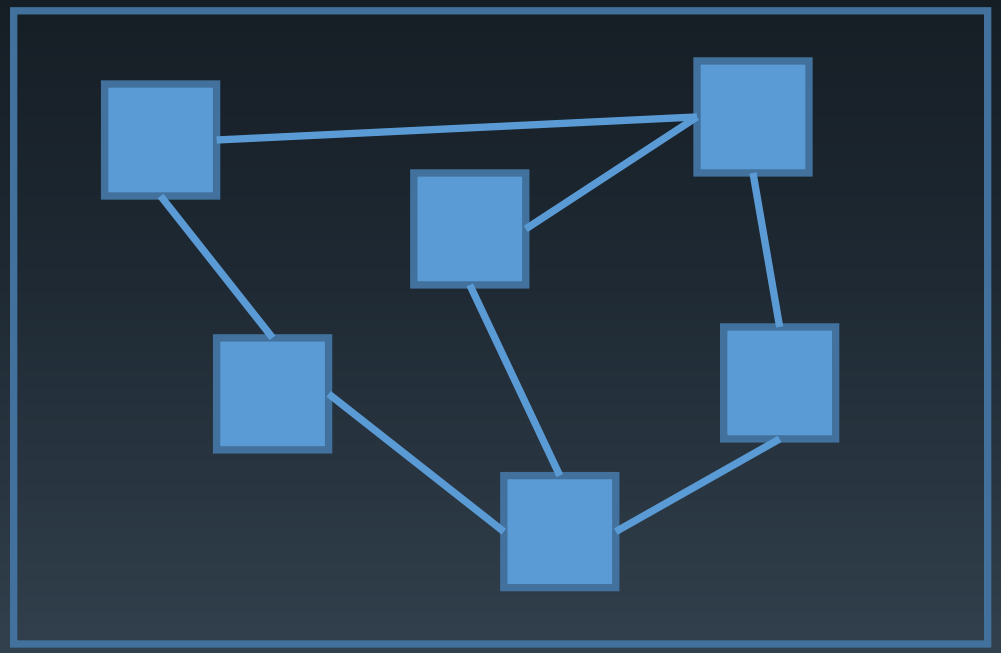
Massive Parallel Architectures

Designs for HPC

Massive Parallel Architectures

Grid Computing

Local/Dispersed

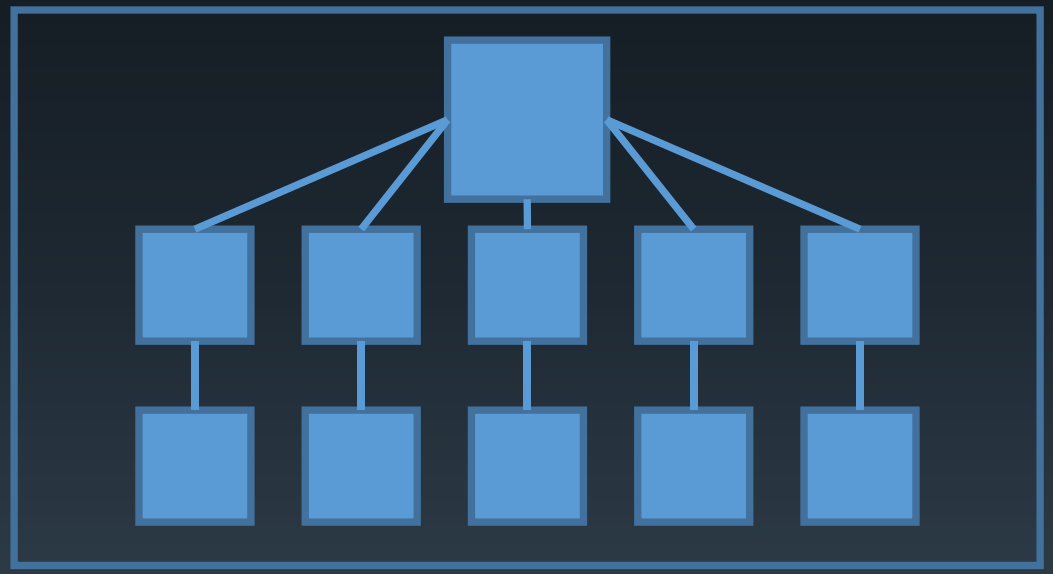


- heterogeneous hardware & OS
- connected via Internet or low speed communication system (bus)
- nodes are autonomous and can opt out
- every node manages its own resources

Designs for HPC

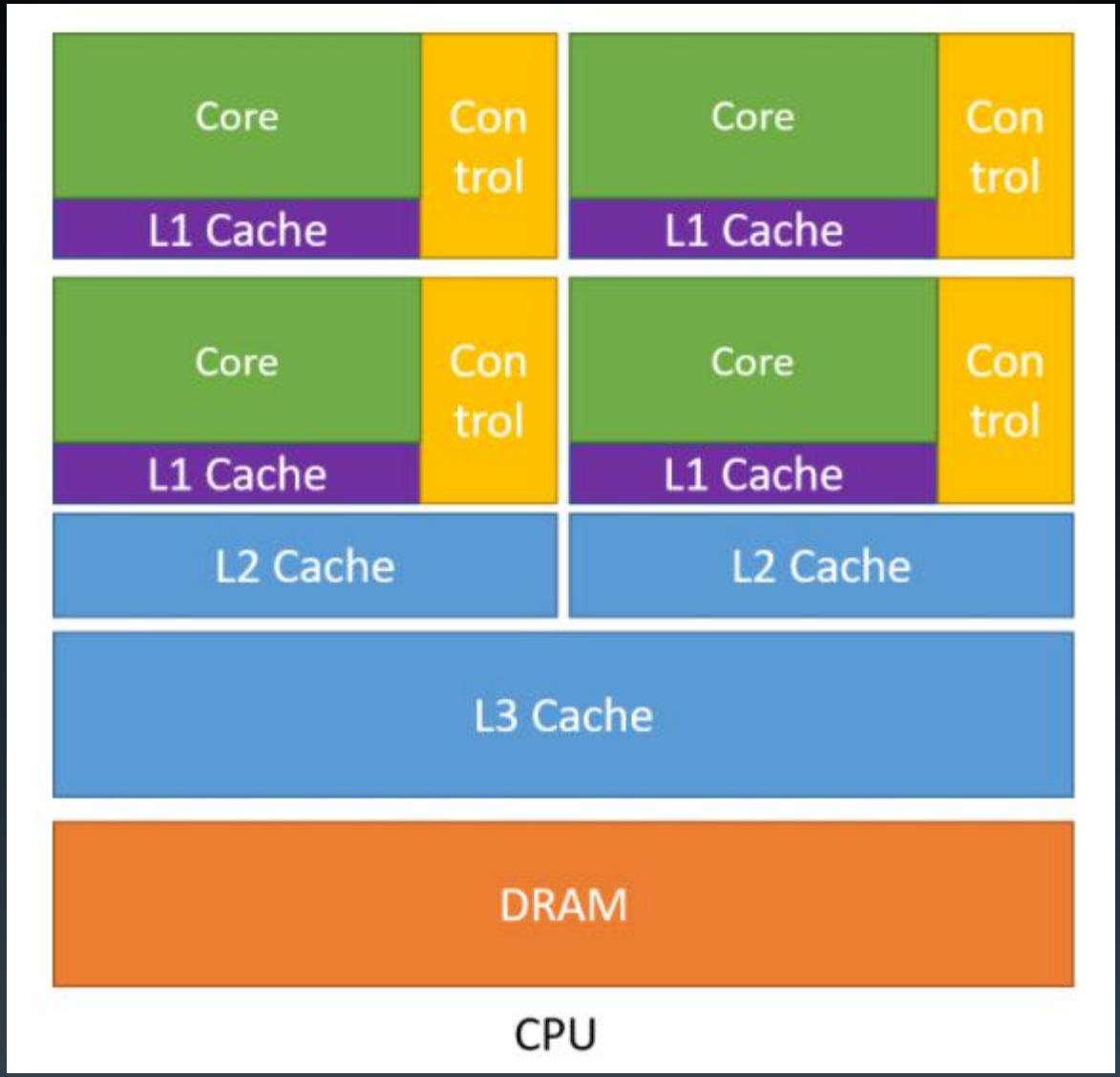
Massive Parallel Architectures

Cluster Computing



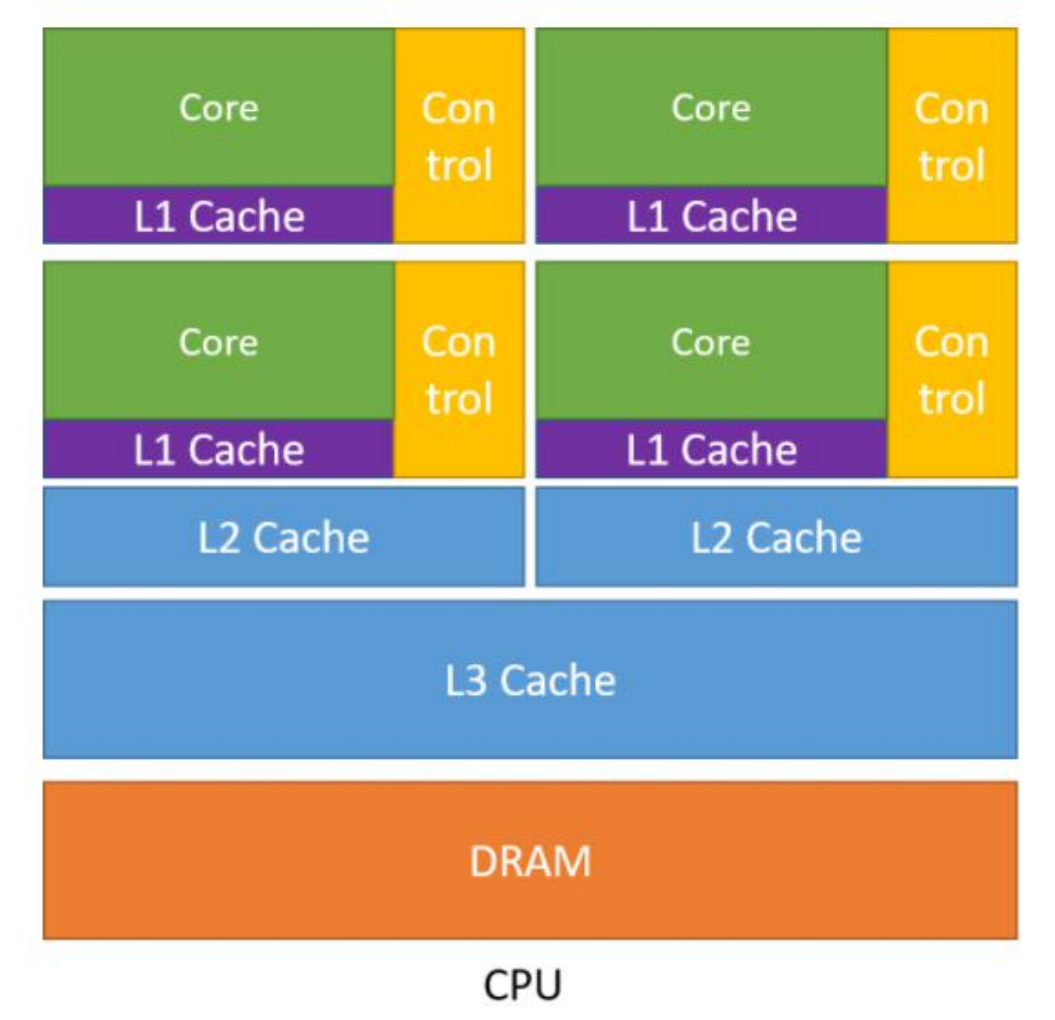
- homogenous, local
- perform same task
- connected via high-speed bus or LAN
- Scheduling done by central server
- Whole system behaves as if it were one system

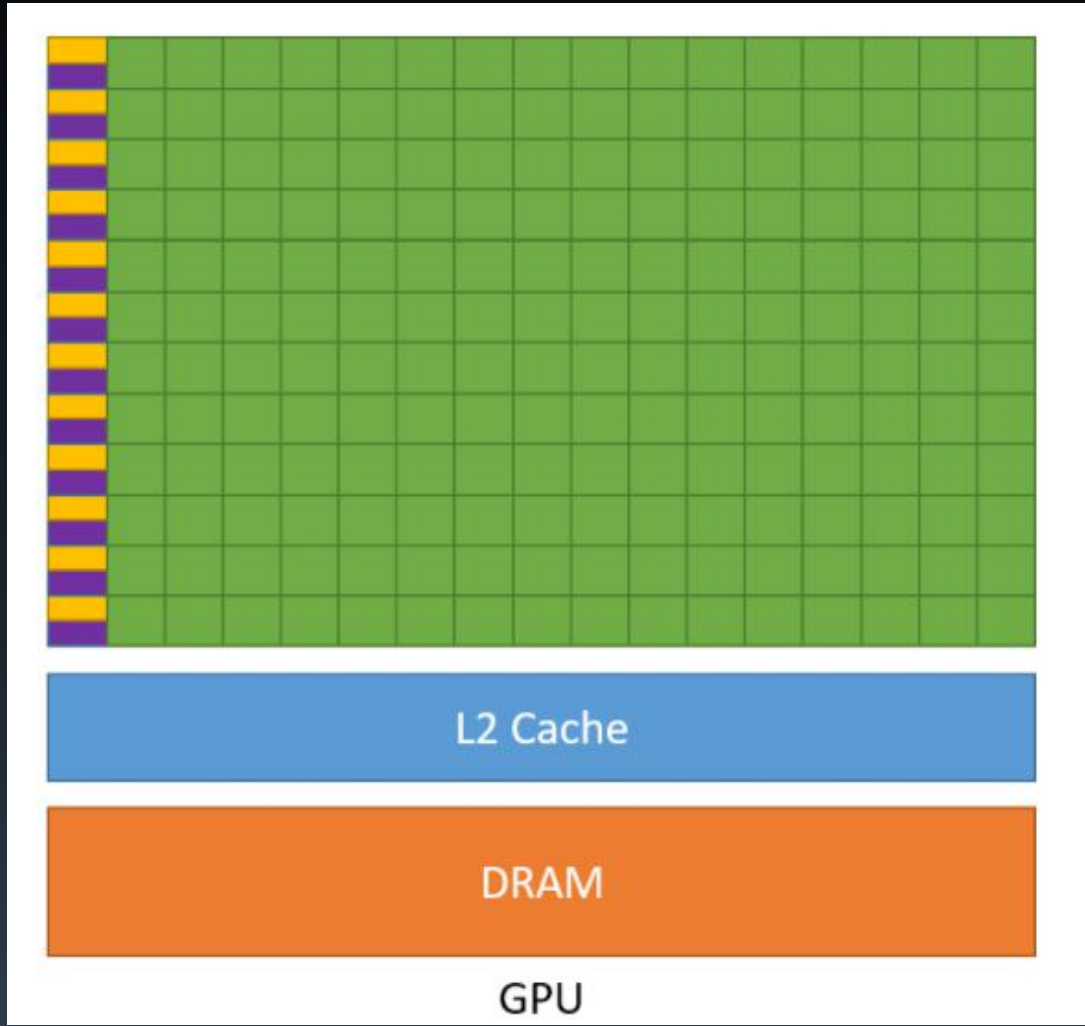
CPU & GPU


Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
L2 Cache
L2 Cache
L3 Cache
DRAM
CPU

[1]

CPU & GPU


Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
Core
Con trol
L1 Cache
L2 Cache
L2 Cache
L3 Cache
DRAM
CPU


L2 Cache
DRAM
GPU

[1]

CPU & GPU

CPU

- Serial Processing
- Few Cores
- Focuses on Low Latency

GPU

- Parallel Processing
- Lot of Cores
- Focuses on High Throughput

CPU & GPU

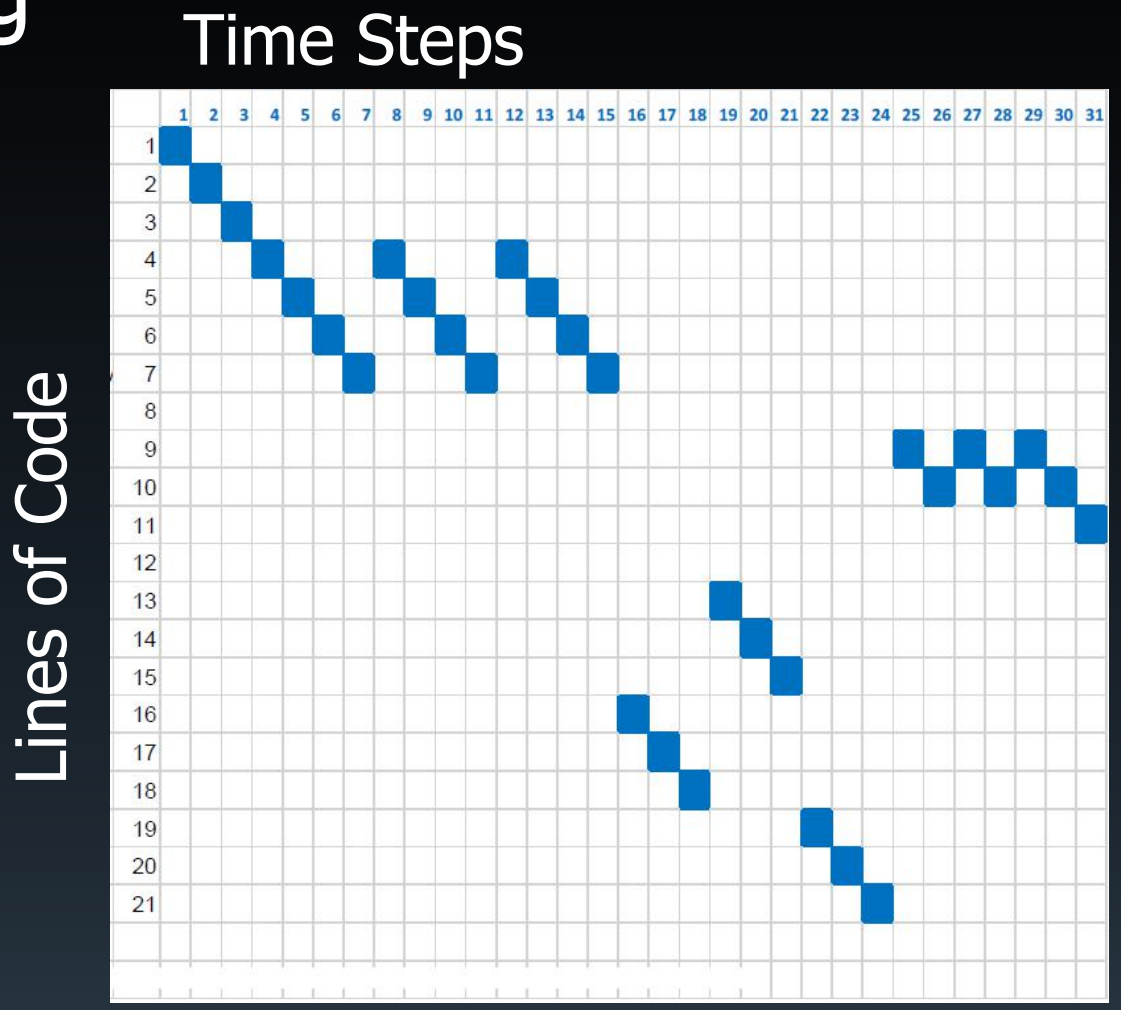
CPU

- Serial Processing
- Few Cores
- Focuses on Low Latency

GPU

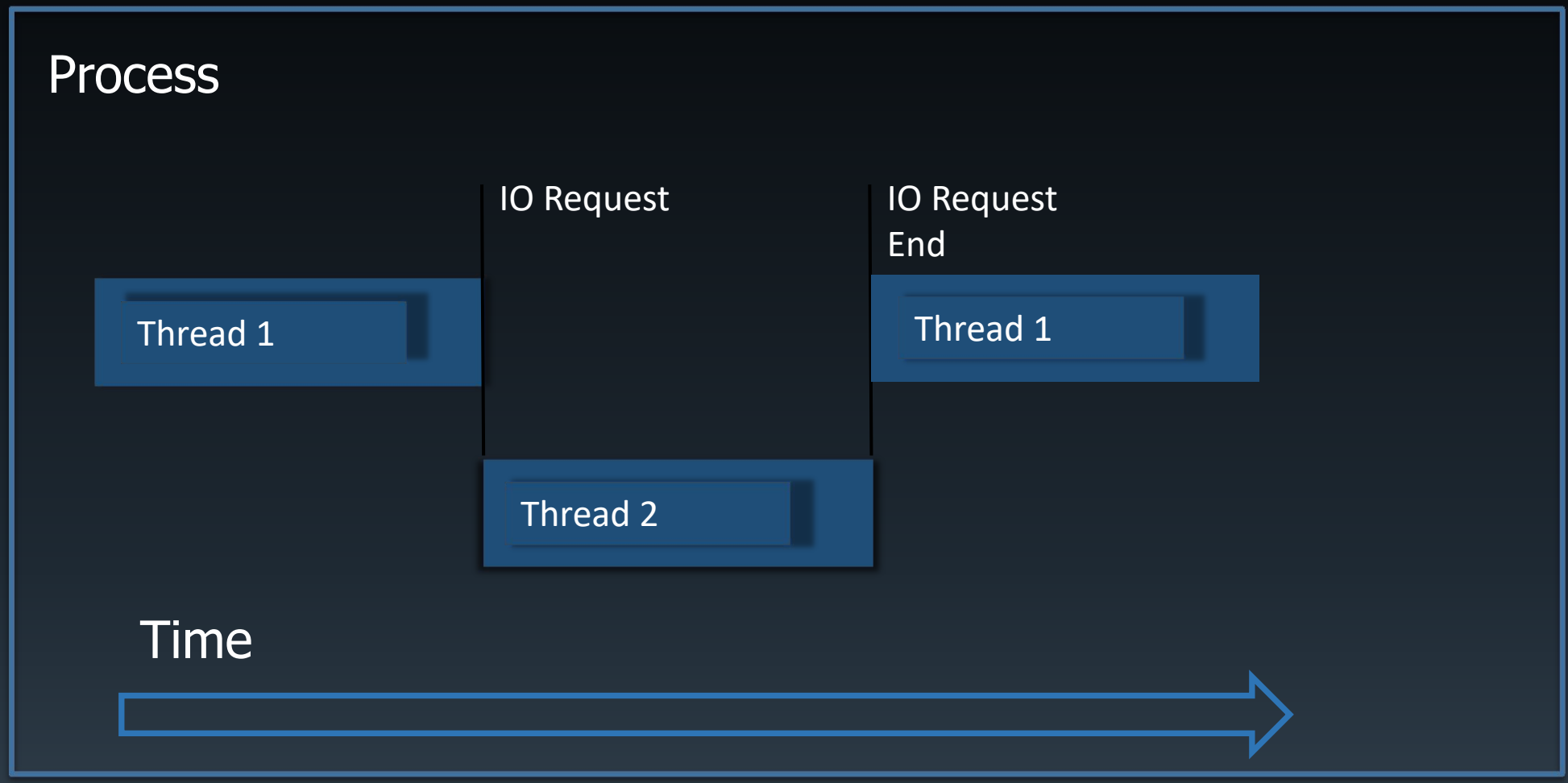
- Parallel Processing
- Lot of Cores
- Focuses on High Throughput

Multithreading



[2]

Multithreading



Multithreading

Sequential Programming

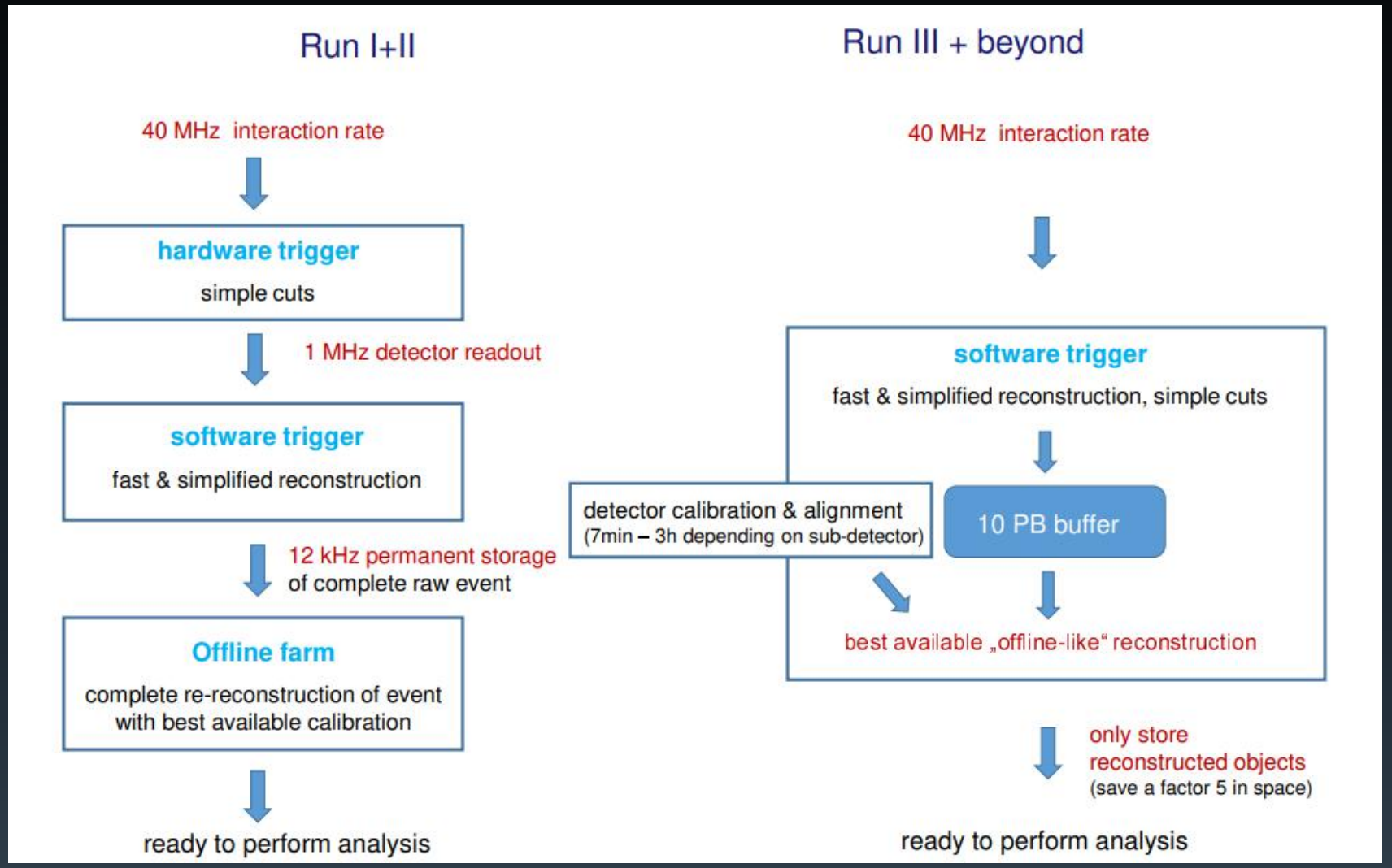
- Deterministic Process
- Easy - Medium
- Casual Software Bugs
- Highly Reproducible

Multithreading Programming

Multithreading

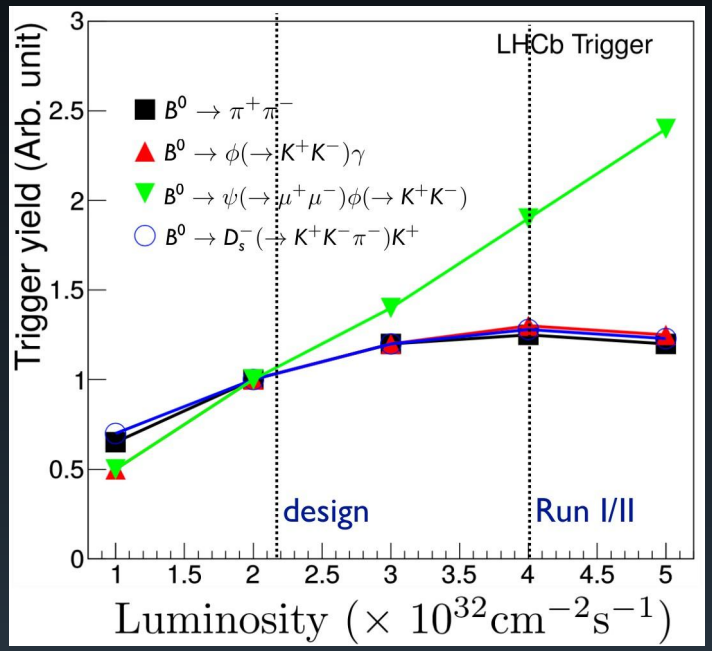
Sequential Programming	Multithreading Programming
➢ Deterministic Process	➢ Non-Deterministic Process
➢ Easy - Medium	➢ Difficult - very Difficult
➢ Casual Software Bugs	➢ Bugs ~ number of interactions
➢ Highly Reproducible	➢ Highly Non-Reproducible

Evolution of LHCb Experiment



Implications for Physics

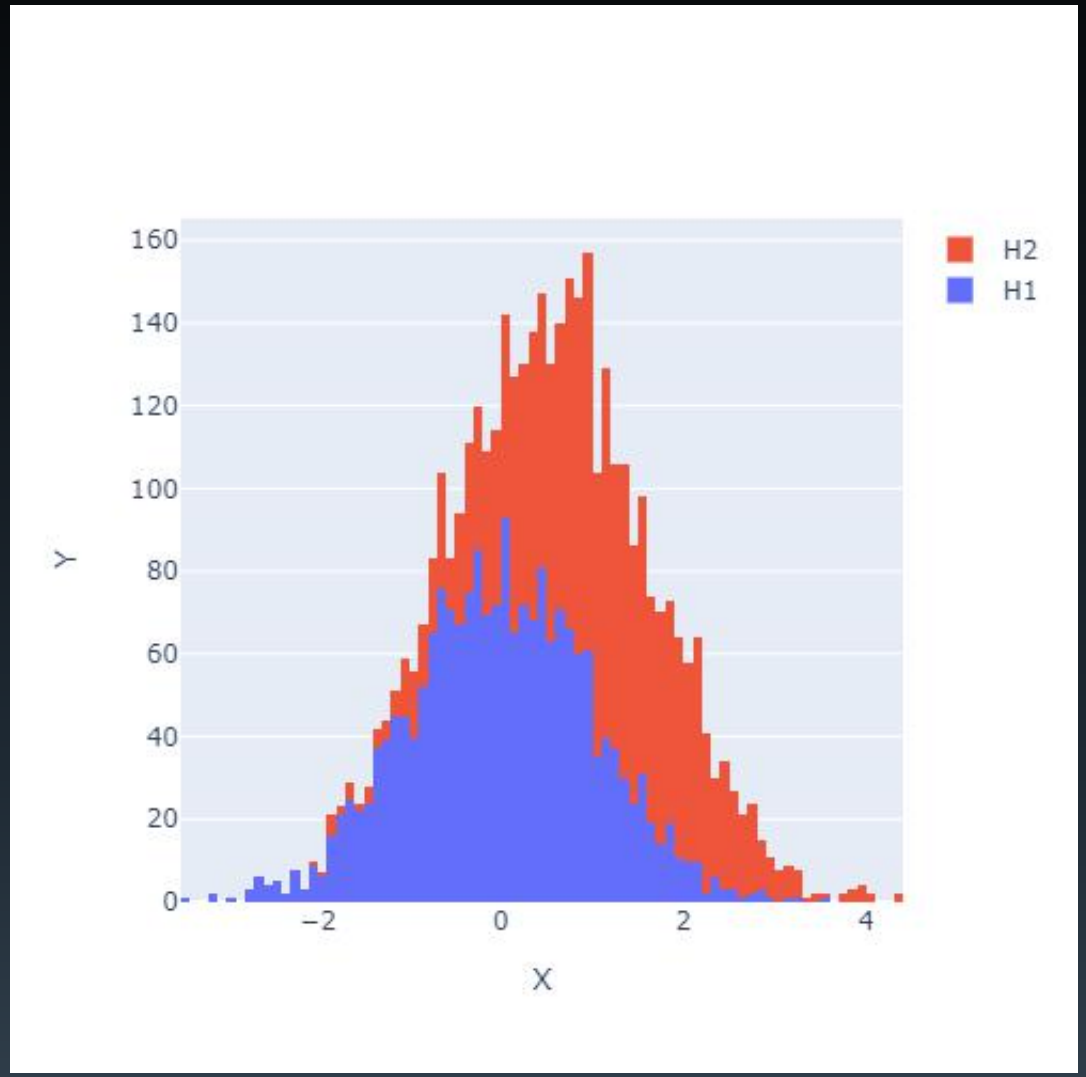
- Hardware Trigger was removed - removing tightness on the thresholds on p_T and E_T - inefficient for hadronic decay modes



- Higher amount of Data enables to push analysis forward - reducing uncertainties - just better statistics

Example for Multithreading

Computations



Software-Challenges

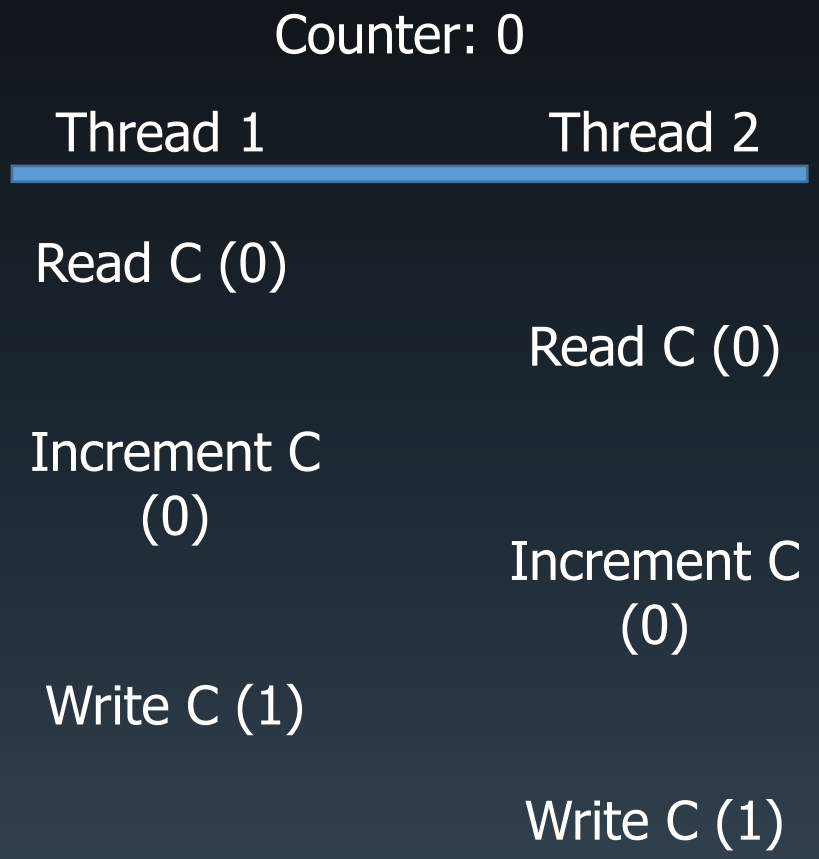
Race Conditions

Dead Locks

Software-Challenges

Race Conditions

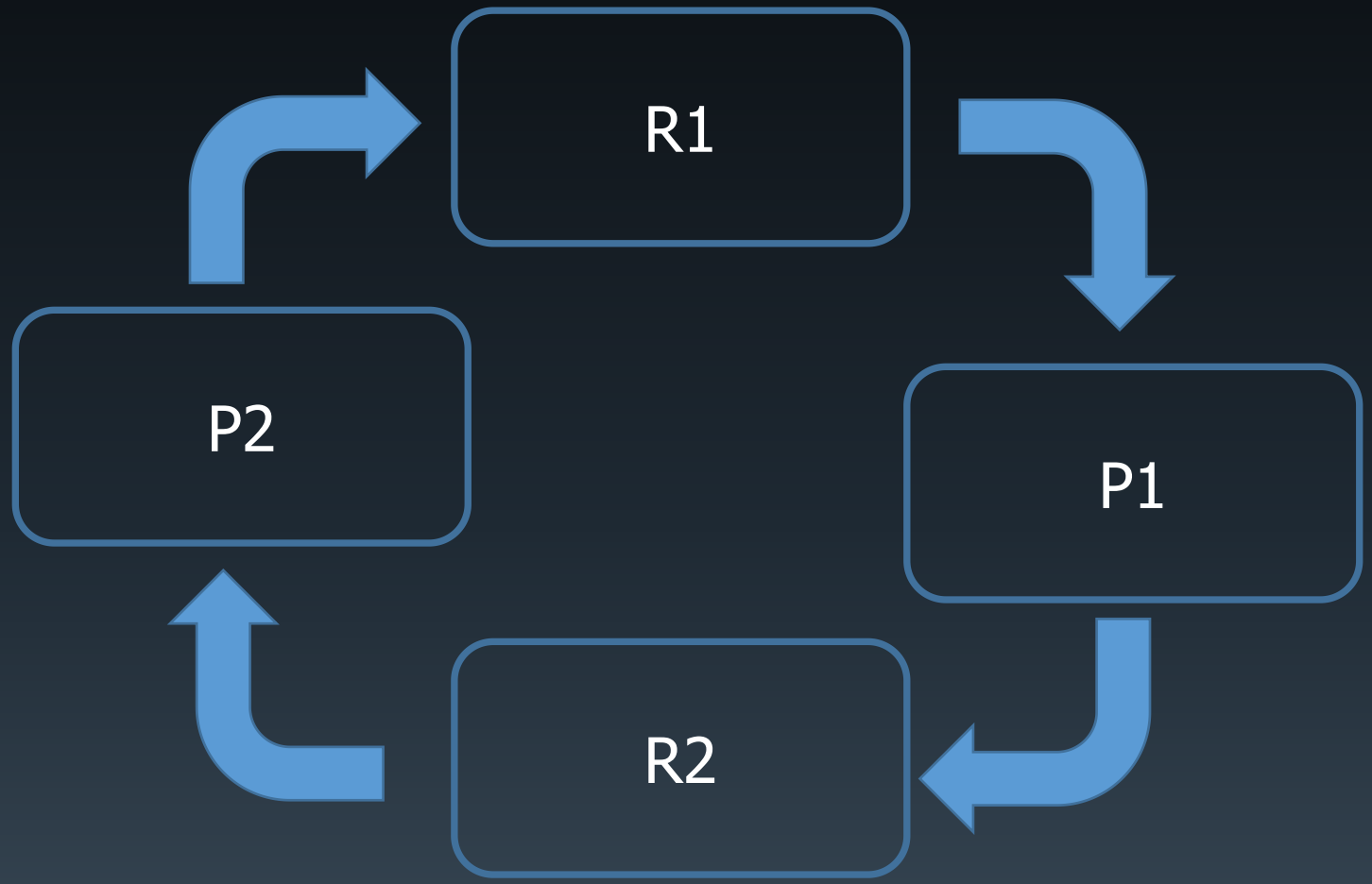
Dead Locks



Software-Challenges

Race Conditions

Dead Locks



Software-Challenges

Race Conditions

Dead Locks

Mutual Exclusion

Signaling and Waiting

No Preemption

Circular Dependency

Future of HPC

- HPC is needed in order to enable research institutes and companies to process huge amount of data or to build complex simulations
- HEP cannot evolve without HPC, HEP will heavily rely in the future on the state of HPC
- My Predictions for the 2020s: Cloud HPC, AI HPC (MLOps)

Referenzen

[1] CUDA Toolkit Dokumentation, Nvidia, Developer Zone, https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

[2] R. Arpaci-Dusseau, A. Arpaci-Dusseau, Operating Systems: Three Easy Pieces, https://pages.cs.wisc.edu/~remzi/OSTEP/

Supercomputers

Deep Blue

Riken
MDGRAPE-3

Folding
@home

Riken
Fugaku

Supercomputers

- Deep Blue
- Riken MDGRAPE-3
- Folding @home
- Riken Fugaku

IBM Supercomputer

1997:
After upgrade, it won against Garry Kasparov in 6-game match: 3 - 1 - 2

~ 11.38 GFLOPs

Supercomputers

Deep Blue

Riken MDGRAPE-3

Folding @home

Riken Fugaku

Institute of Physical and Chemical Research (RIKEN)

2006:
announced to be completed, molecular dynamics (esp. protein structure predictions)

~ 1 petaFLOPs

Supercomputers

Deep Blue

Riken MDGRAPE-3

Folding @home

Riken Fugaku

Washington University, led by Greg Bowman

2000 (initial release): protein dynamics

2021: COVID-19 Research Acceleration

~ 1.22 exaFLOPs

Supercomputers

Deep Blue

Riken
MDGRAPE-3

Folding
@home

Riken
Fugaku

Riken & Fujitsu

2014:
development started
2020:
debut, in June became the fastest supercomputer in the world (TOP500)

~ 442 petaFLOPs (HPL)
~ 1.42 exaFLOPs (HPL-AI)

Challenges of HPC: Multithreading

Advantages	Disadvantages
Thread A waits, Thread B goes	Software needs to be optimized for multithreading
Higher Performance if correctly used	Lower Performance if incorrectly used
Resource Sharing	Thread Scheduler!
Cost-Effective on Hardware Side	Cost-intensive on Software Side

Challenges of HPC: Multithreading

Advantages	Disadvantages
Thread A waits, Thread B goes	Software needs to be optimized for multithreading
Higher Performance if correctly used	Lower Performance if incorrectly used
Resource Sharing	Thread Scheduler!
Cost-Effective on Hardware Side	Cost-intensive on Software Side