Foundations of High Performance Computing

Lecture 2: INTRODUCTION TO HPC part 2

"Foundation of HPC" course

DATA SCIENCE &
SCIENTIFIC COMPUTING

2021-2022 Stefano Cozzini

Agenda

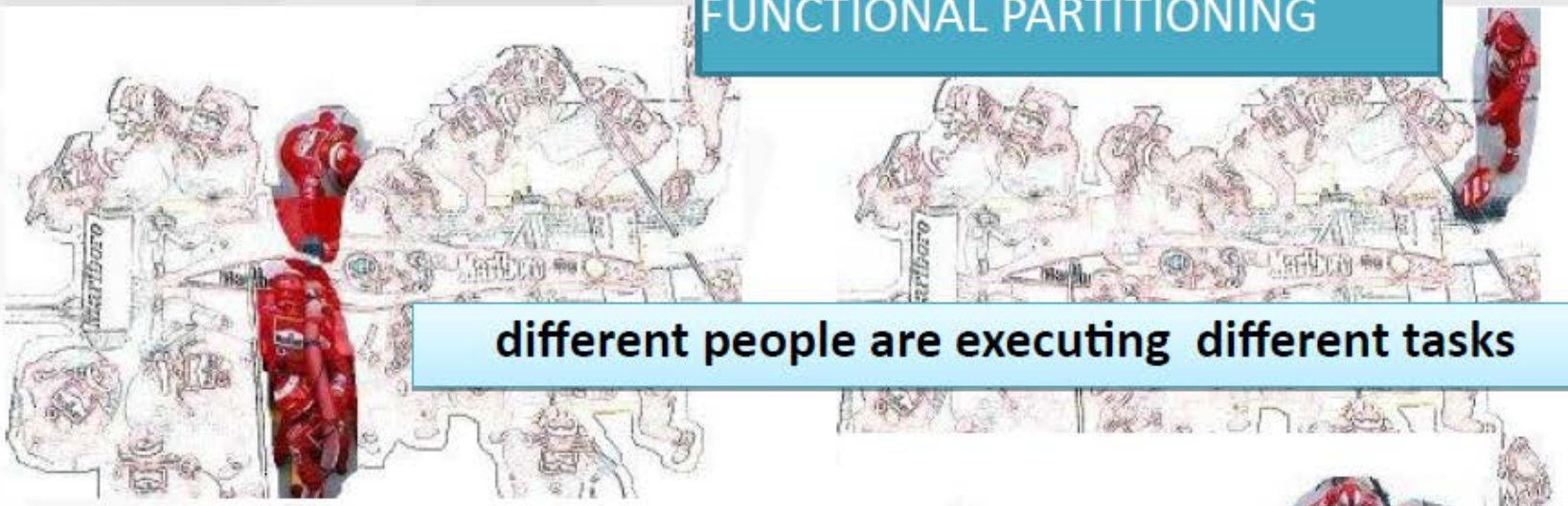
- Why HPC is parallel ?
- Serial Computers
- Parallel computers
- HPC infrastructure
- Parallel components in a cluster

Let us focus on High Performance problem

picture from http://www.f1nutter.co.uk/tech/pitstop.php

Analysis of the parallel solution

FUNCTIONAL PARTITIONING



different people are executing different tasks

DOMAIN DECOMPOSITION

different people are solving the same global task but on smaller subset

HPC

=

PARALLEL COMPUTING

HPC

=

PARALLEL

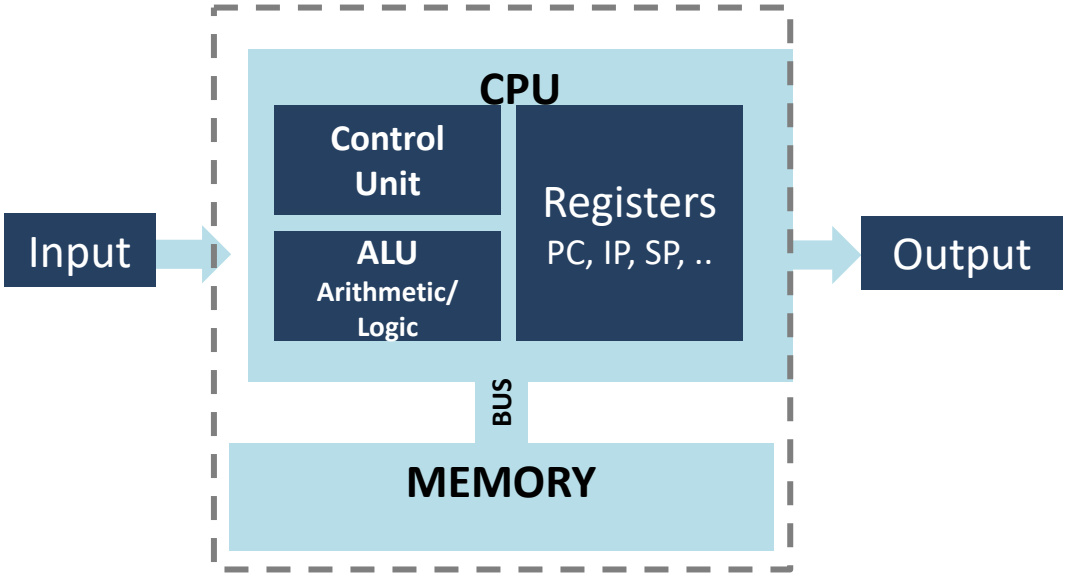
COMPUTERS

Agenda

- Why HPC is parallel ? ✓
- Serial Computers
- Parallel computers
- HPC infrastructure
- Parallel components in a cluster

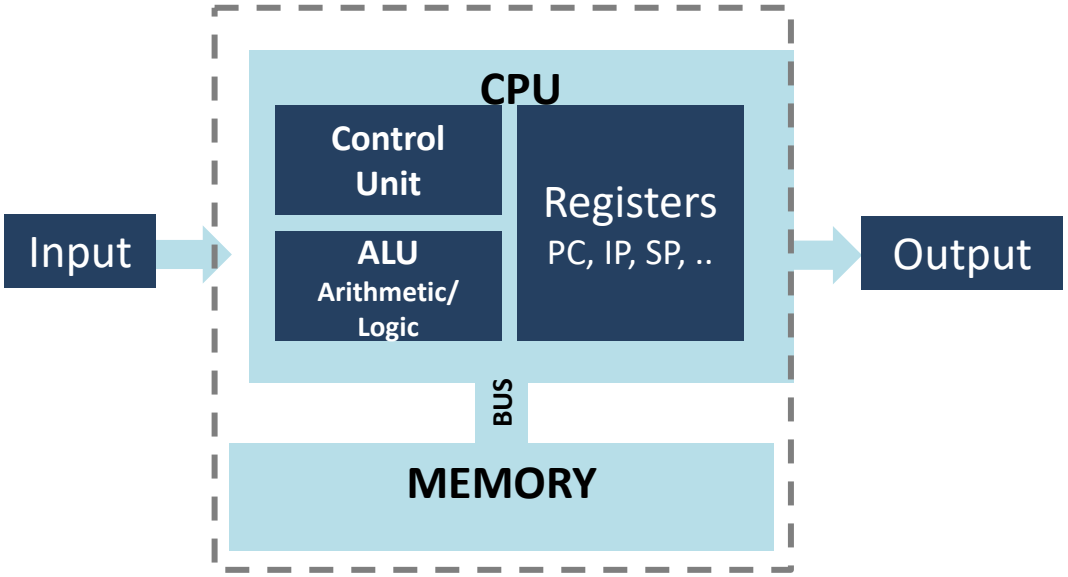
What is a serial computer ?

- Von Neumann architecture (the fundamental model)



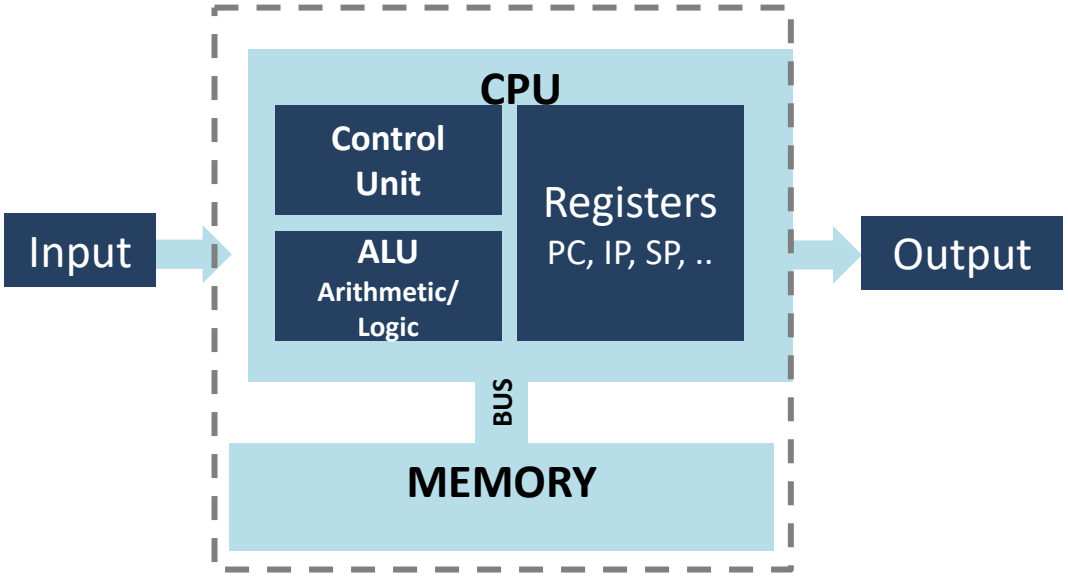
Von Neumann architecture:

- There is only one process unit (CPU)
  - Control Unit: processes instructions
  - ALU: math and logic operations
  - Register: store data



Von Nuemann architecture:

- 1 instructions is executed at a time
- memory is “flat”:
  - access on any location has always the same cost
  - access to memory has the same cost than op execution

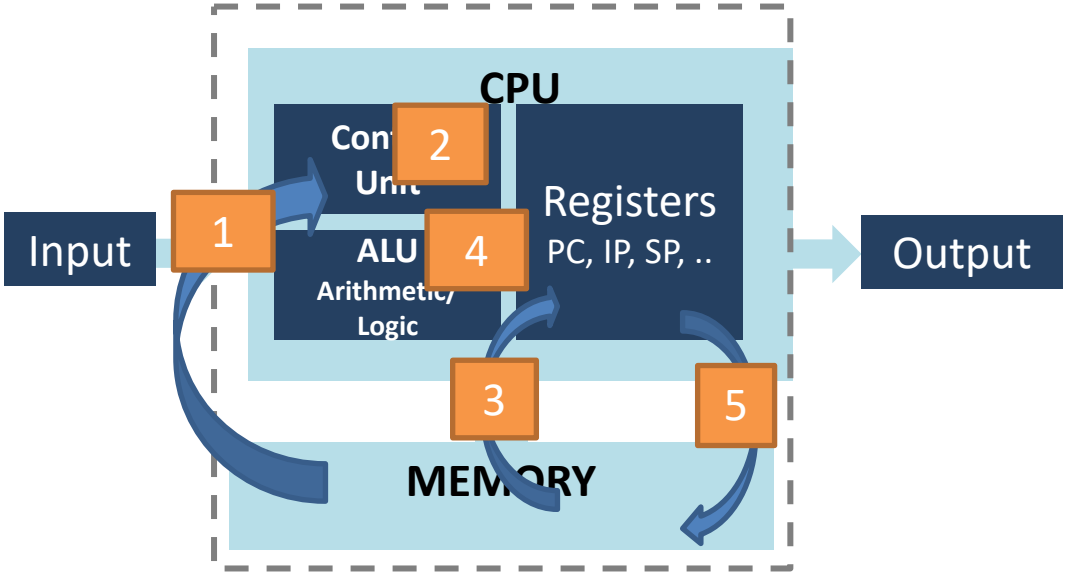


Von Neumann architecture:

5 step WORKFLOW:

1. instruction fetch
2. Instruction decode: determine operation and operands
3. Memory fetch: Get operands from memory
4. Perform operation
5. Write results back

Continue with next instruction



Instruction set architecture (ISA)

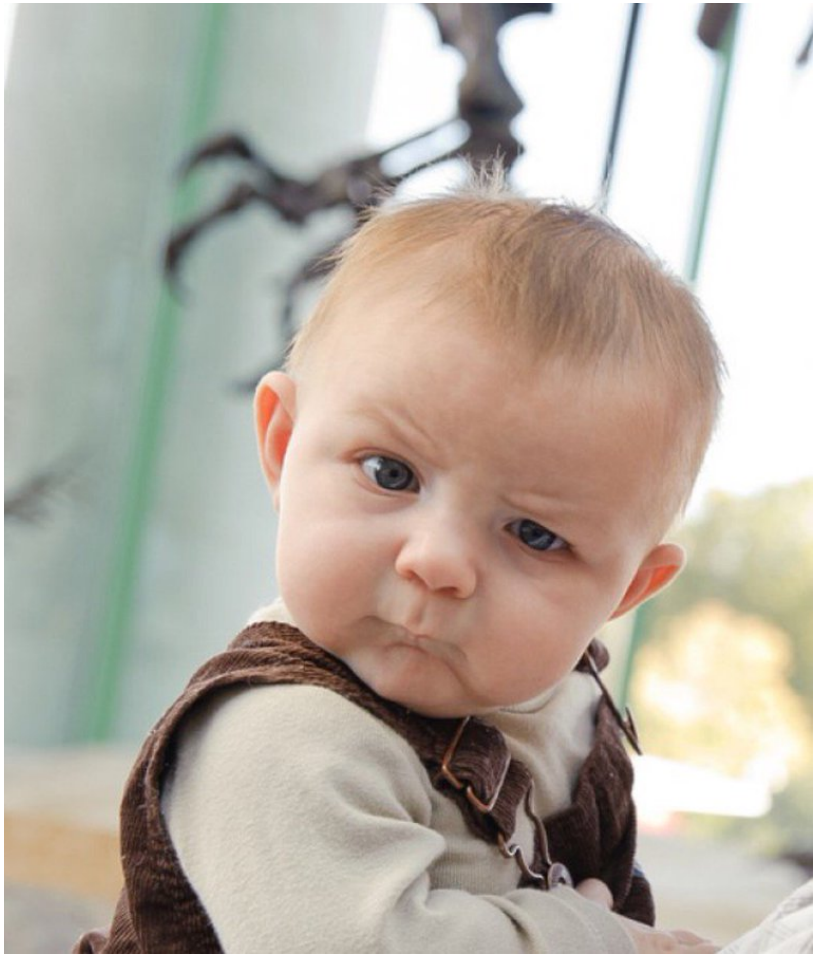
- The deeper level accessible to the programmers
- It is the boundary between SW and HW
- The interface between the programmer and the microarchitecture
- Different microarchitectures can have the same ISA (binary Compatible)
- Different generation of microarchitectures can be backward compatible
- For us: x86 instruction set

A very simple operation..

```
void store(double *a, double *b, double *c) {
 *c = *a + *b;
}
```

```
[exact@master ~]$ gcc -O2 -S -o - frammento.c
.file   "frammento.c"
.text
.p2align 4,,15
.globl store
.type        store, @function
store:
.LFB0:
.cfi_startproc
movsd      (%rdi), %xmm0   #load *a to mmx0
addsd      (%rsi), %xmm0   # load b and add to *a
movsd      %xmm0, (%rdx)  # store to C
ret
.cfi_endproc
.LFE0:
.size store, .-store
.ident      "GCC: (GNU) 4.4.7 20120313 (Red Hat 4.4.7-4)"
.section    .note.GNU-stack,"",@progbits
```

Does still exist serial computer ?



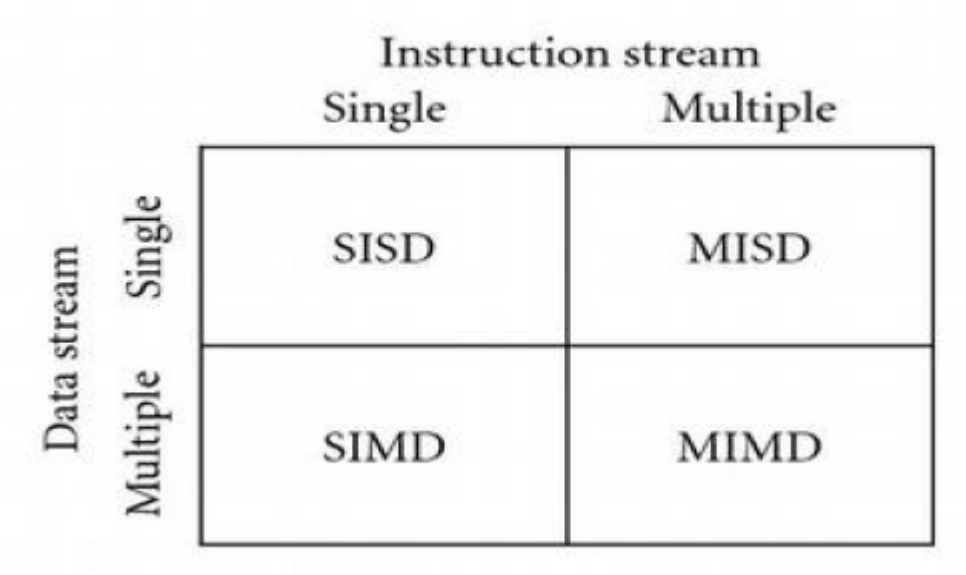
Agenda

- Why HPC is parallel ? ✓
- Serial Computers ✓
- Parallel computers
- HPC infrastructure
- Parallel components in a cluster

PARALLELISM IS EVERYHERE even in your laptop..

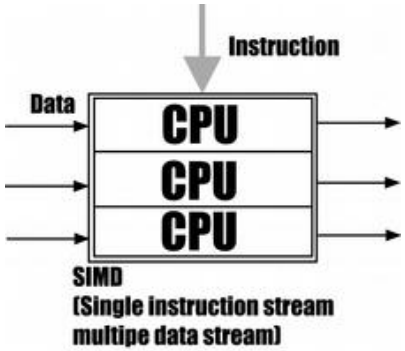
Parallel Computers

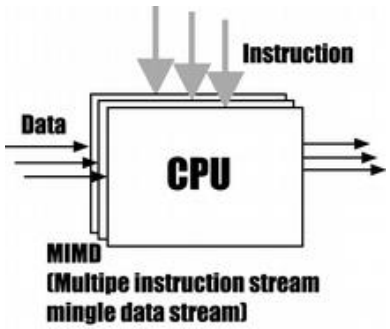
- Flynn Taxonomy (1966): may help us in classifying them:
  - Data Stream
  - Instruction Stream



Comments

- Flynn taxonomy does not help too much nowadays with modern HPC infrastructure
  - CPU and computers are changed too much in the last 50 years
- However, SIMD and MIMD concepts are still used HPC hardware

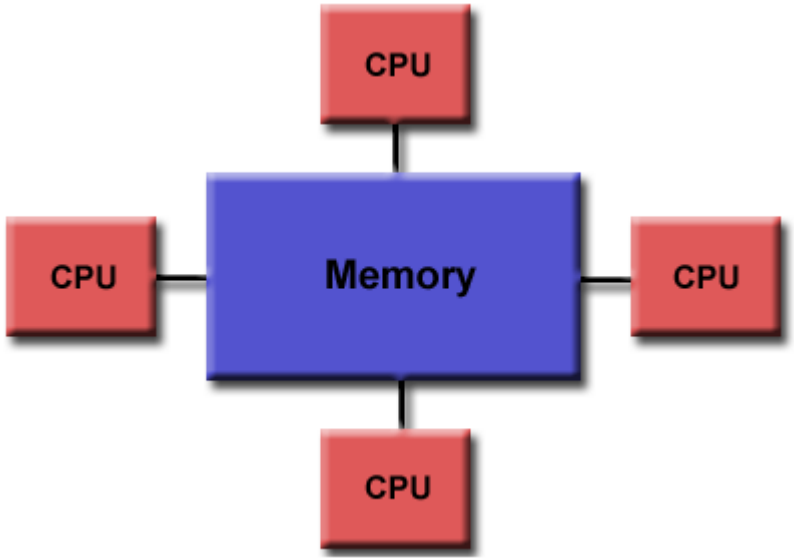




What about memory ?

- In the old time the simplest and most useful way to classify modern parallel computers was by their memory model:
  - SHARED MEMORY
  - DISTRIBUTED MEMORY

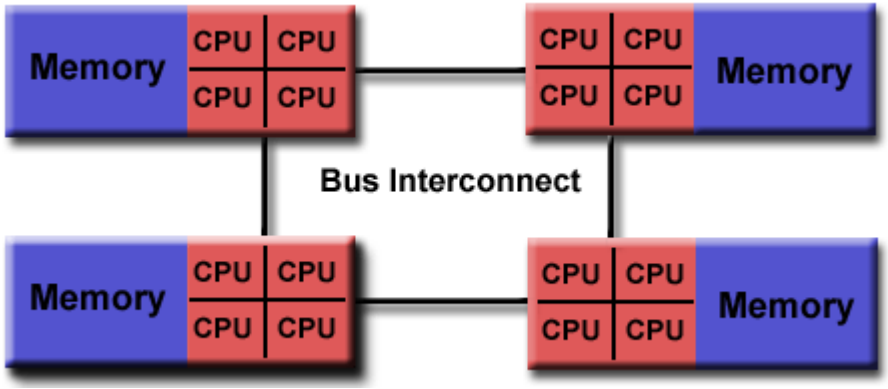
Shared memory: UMA



Uniform memory access (UMA): Each processor has uniform access to memory. Also known as symmetric multiprocessors (SMP)

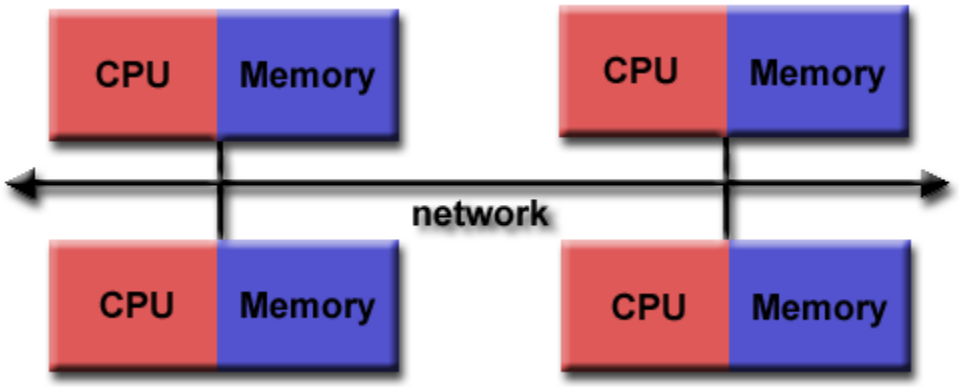
Shared memory: NUMA

Non-uniform memory access (NUMA): Time for memory access depends on location of data. Local access is faster than non-local access.



Distributed memory

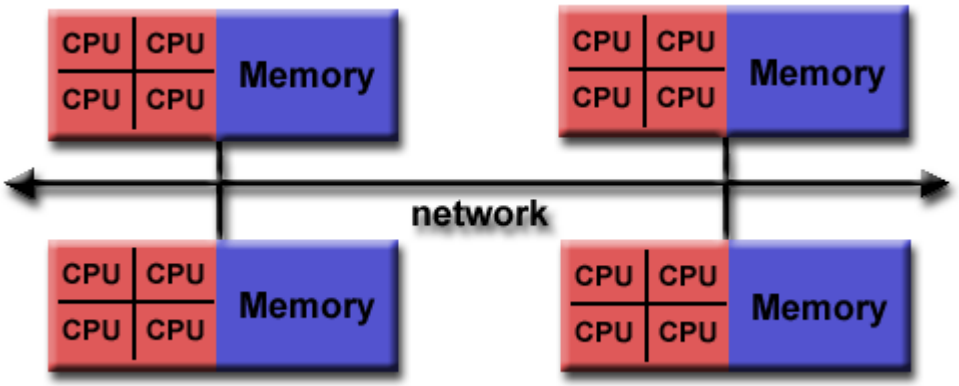
- Distributed memory
  - each processor has its own local memory. Must do message passing to exchange data between processors



ARE THESE MACHINES STILL AVAILABLE ?

Hybrid approach

The shared memory component is shared memory
The distributed memory component is the networking of multiple shared memory which know only about their own memory - not the memory on another machine.

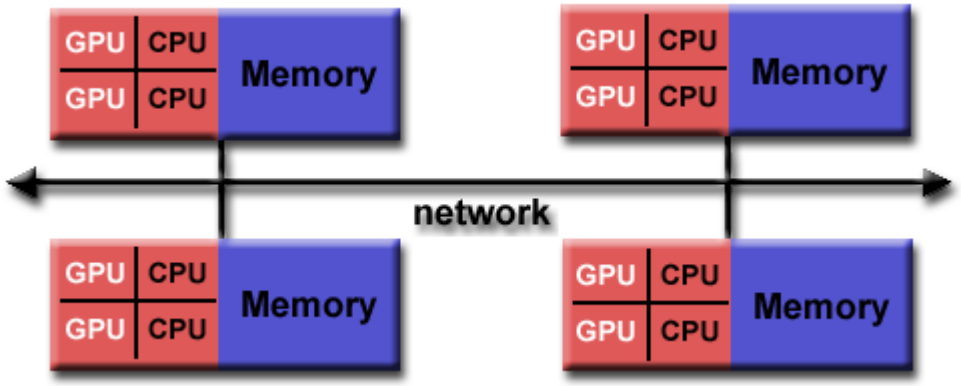


Agenda

- Why HPC is parallel ? ✓
- Serial Computers ✓
- Parallel computers ✓
- HPC infrastructure
- Parallel components in a cluster

Modern HPC infrastructures

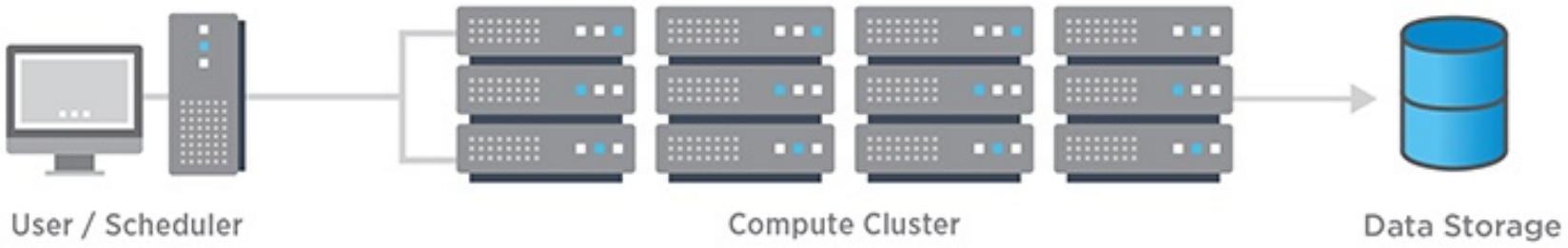
- Cluster of nodes (shared memory)



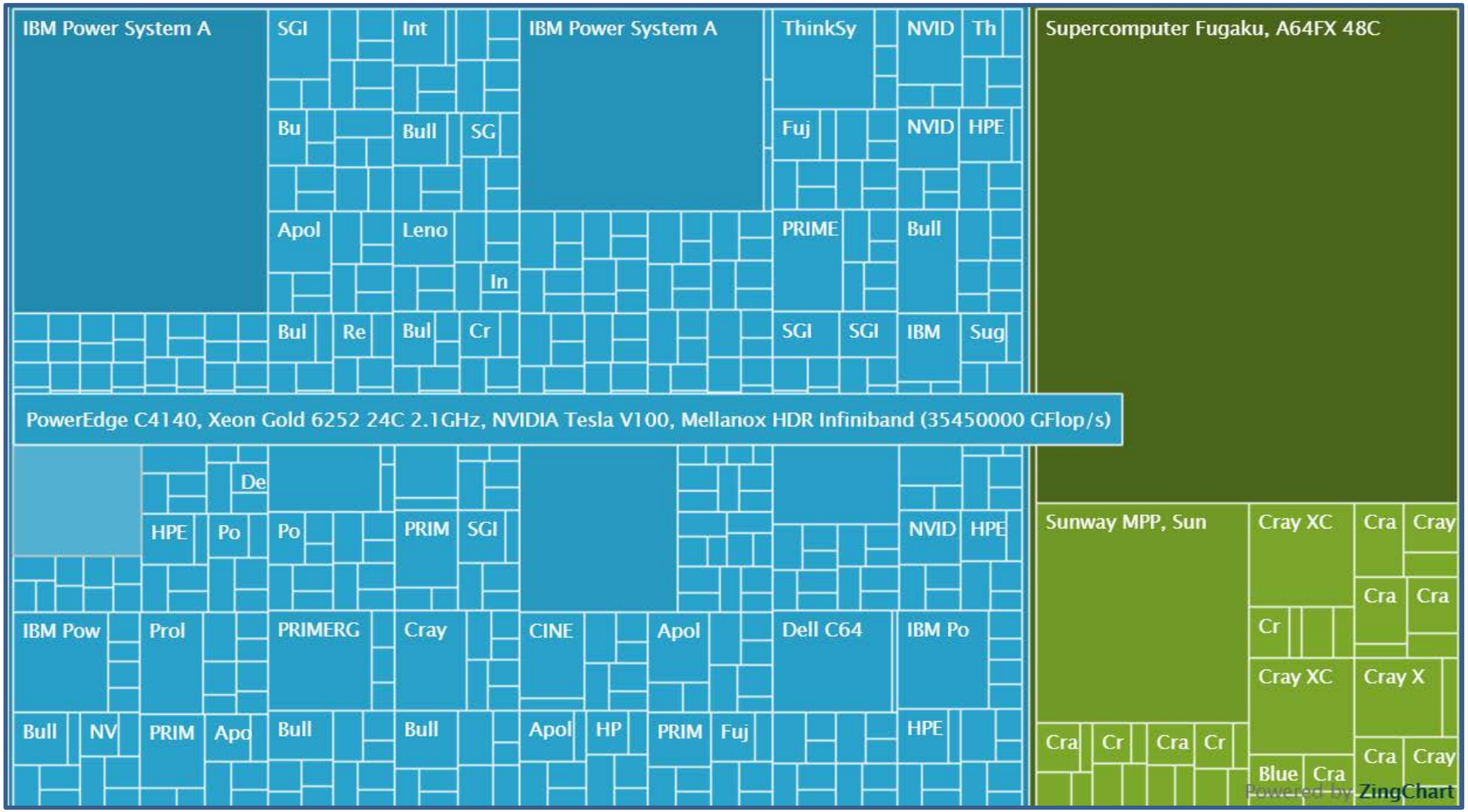
- Hybrid distributed/shared approach from memory point of view

Essential component of a cluster

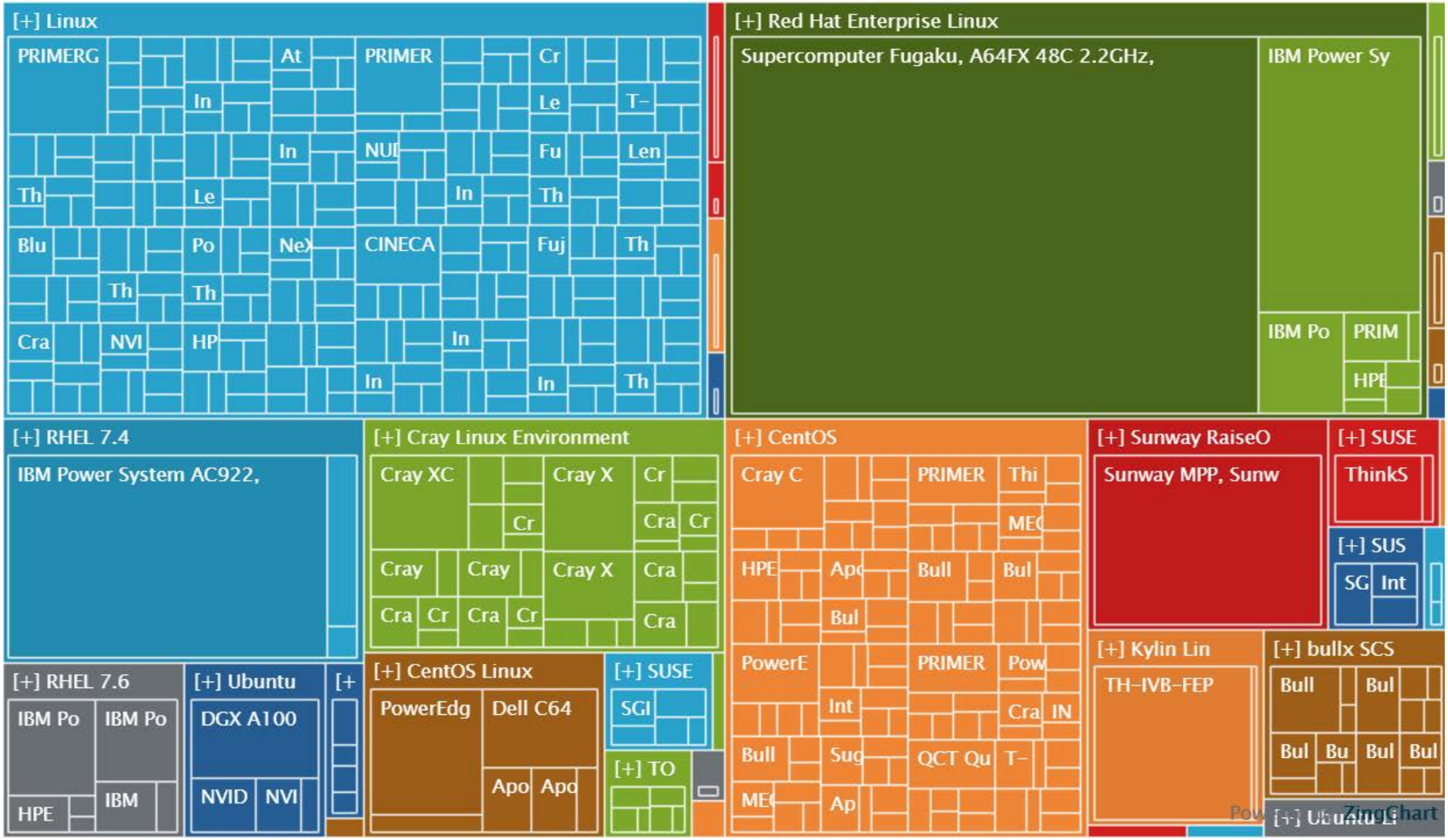
- Several computers (nodes)
  - often in special cases (1U) for easy mounting in a rack
- One or more networks (interconnects) to hook the nodes together
- Some kind of storage
- A login/access node..



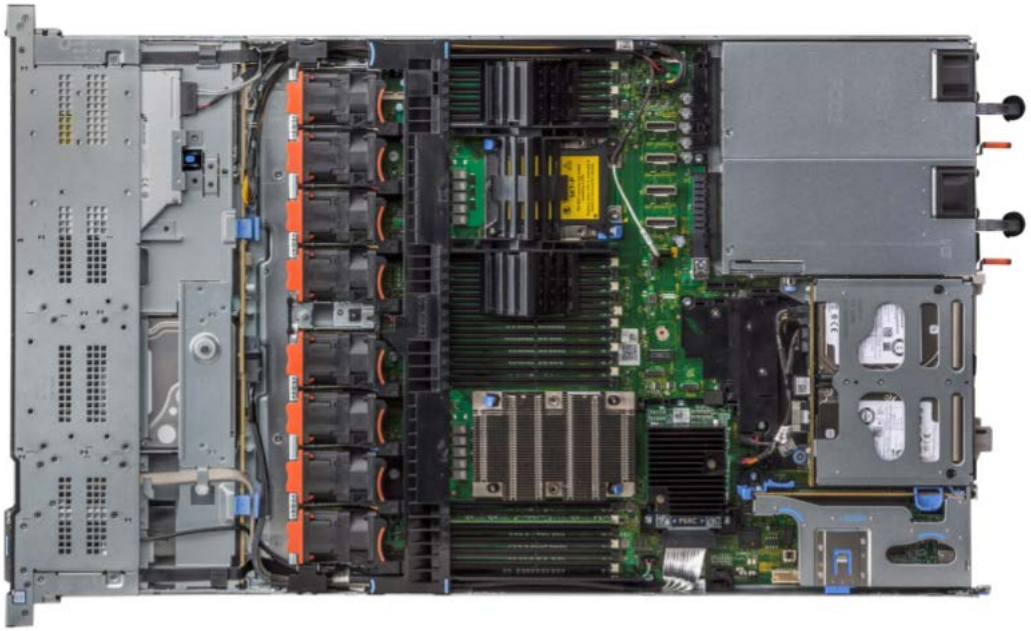
Even supercomputers are clusters !



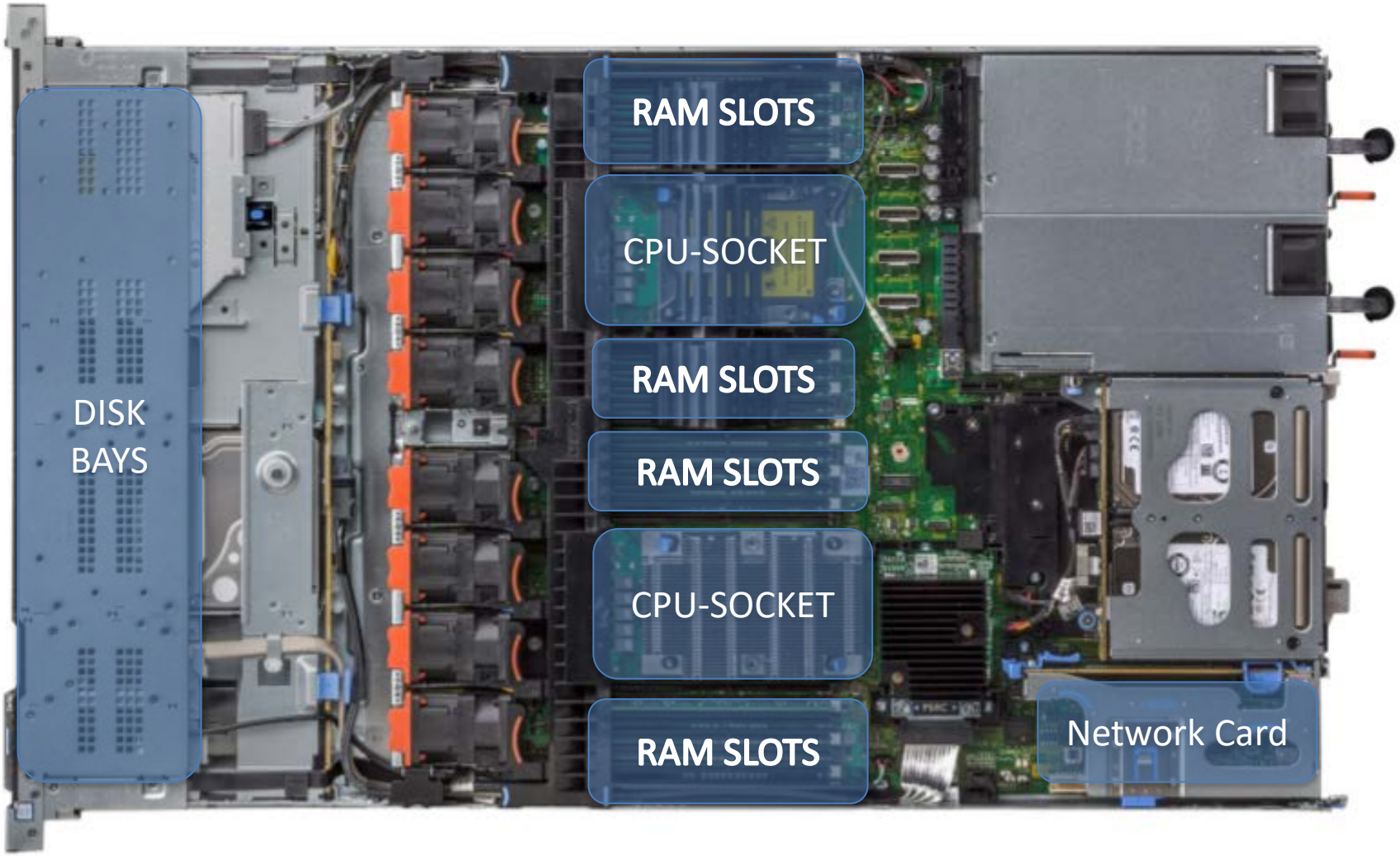
And cluster speaks the same language: LINUX



Modern 1U computing nodes

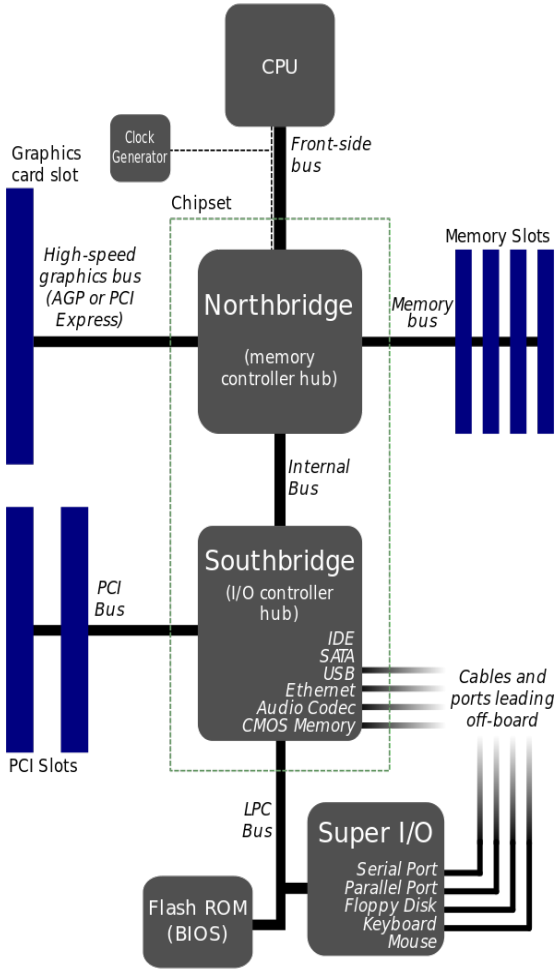


What does one node contain exactly ?



standard modern architecture

- All data communication from one CPU to another must travel over the same bus used to communicate with the Northbridge.
- All communication with RAM must pass through the Northbridge.
- Communication between a CPU and a device attached to the Southbridge is routed through the Northbridge.

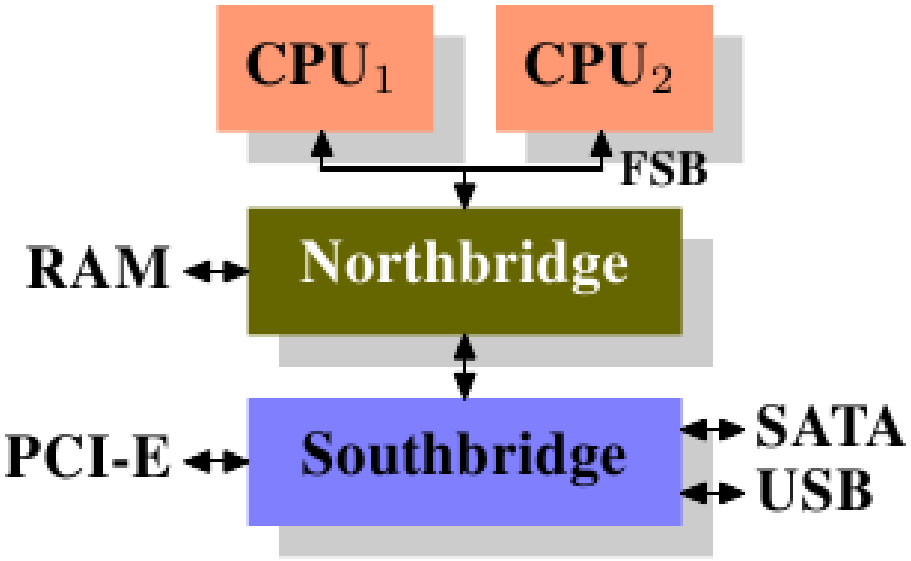


standard multisocket architecture

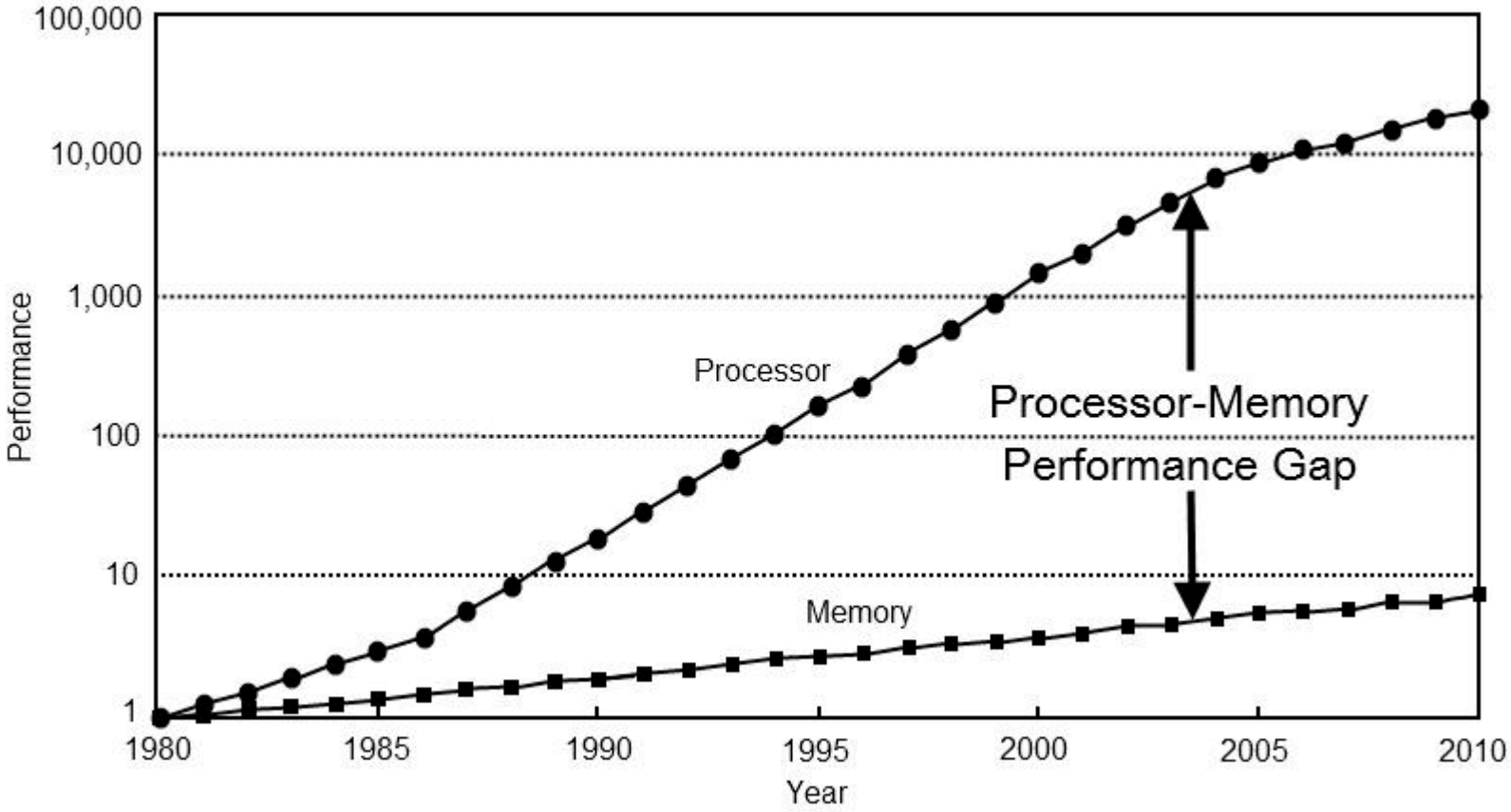
- Characteristics:

– more than one CPU !

– 64 bit address space



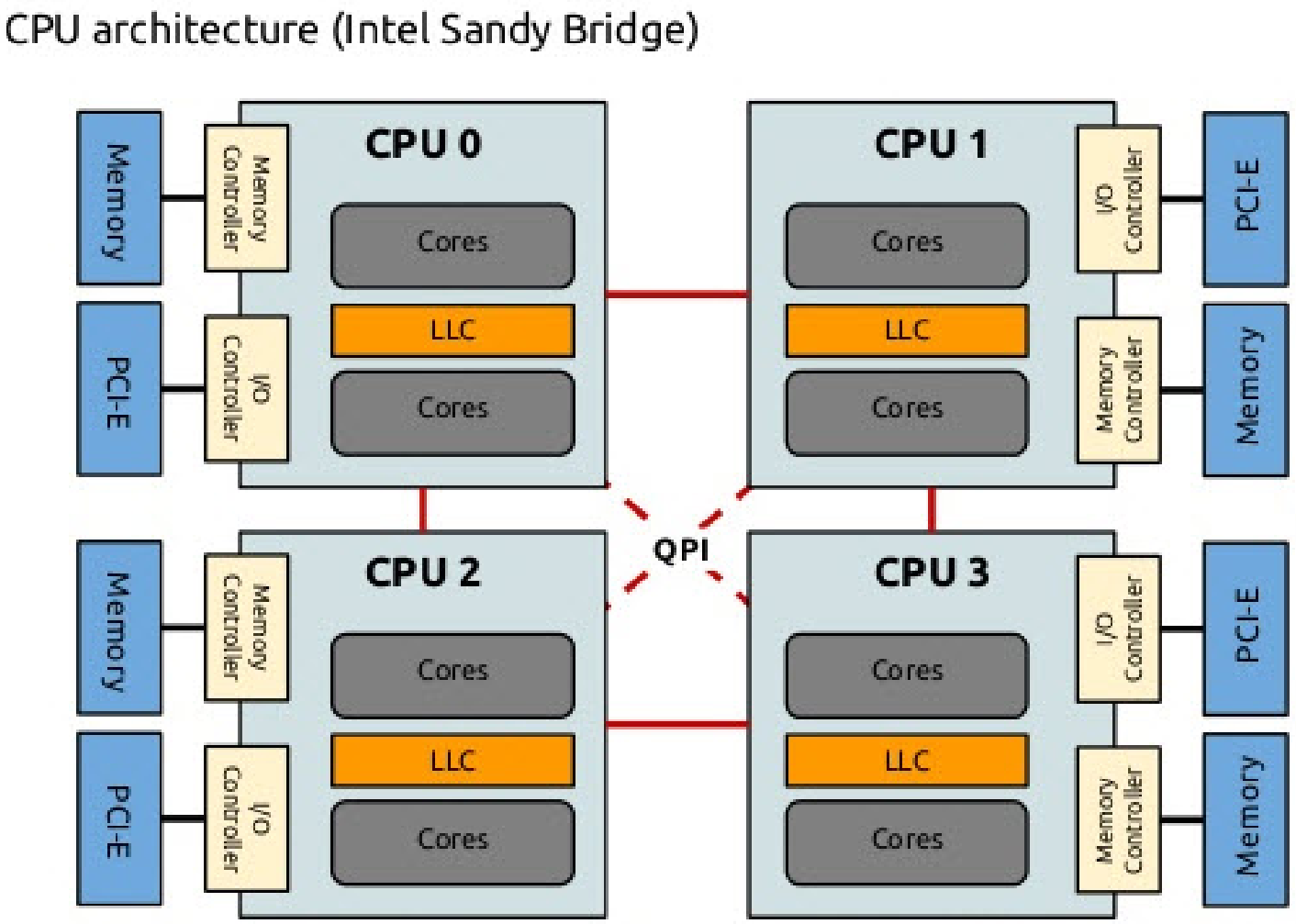
Memory wall problem



From SMP to NUMA

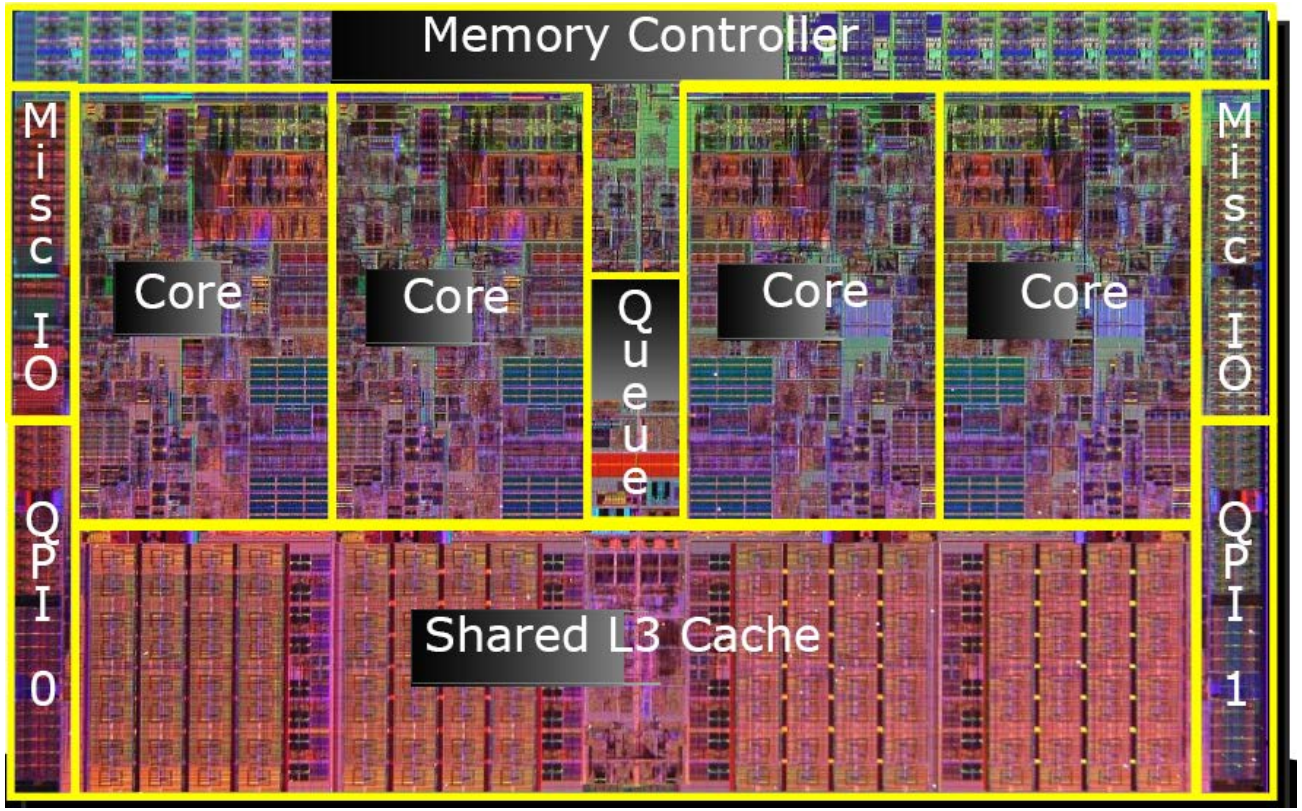
- FSB became rapidly a bottleneck: all the CPUs accessing memory through it
- SMP (UMA) approach no longer possible
- First NUMA architecture:
  - Hypertransport technology by AMD (2005)
- Intel came much later
  - Quick Path Interconnect (2009)
  - Fast Path Interconnect (2016)

How is it logically organized ?



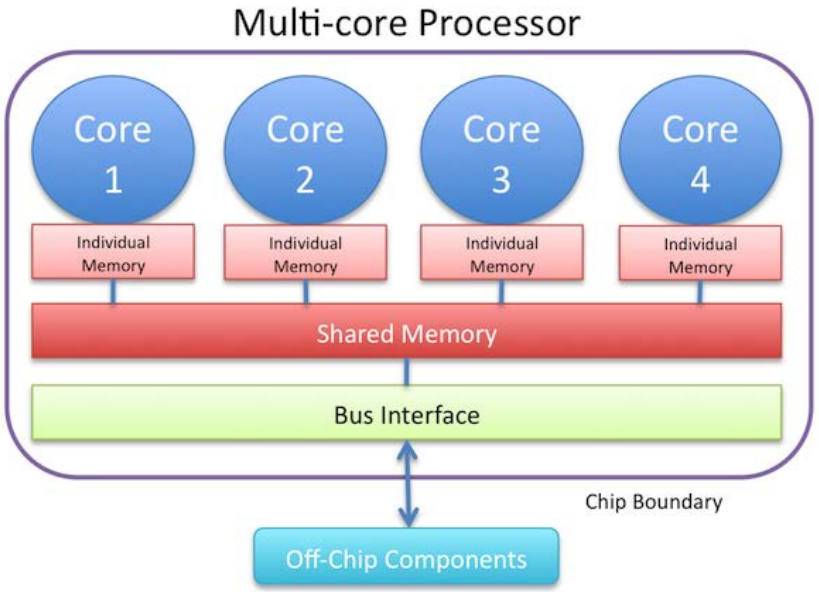
CPU level: Intel core I7

- CPU is multicore !



CPU are multicore processor

- Because of power, heat dissipation, etc increasing tendency to actually lower clock frequency but pack more computing cores onto a chip.
- These cores will share some resources, e.g. memory, network, disk, etc but are still capable of independent calculations

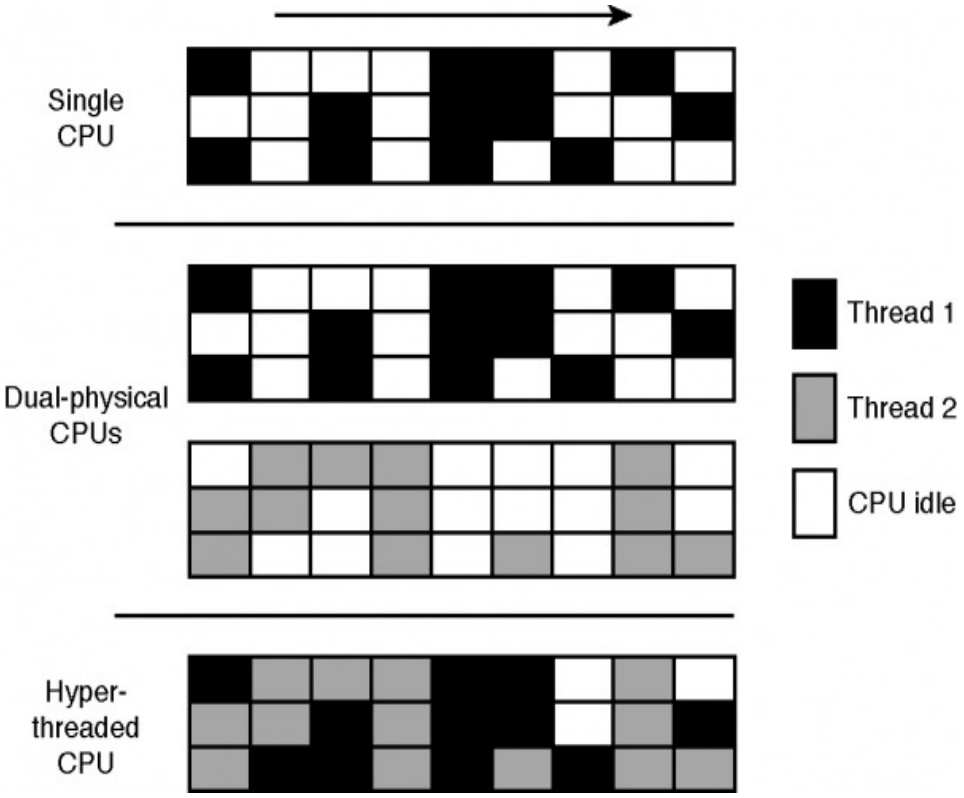


Core : definition

- A core is the smallest unit of computing, having one or more (hardware/software) threads and is responsible for executing instructions.

Hyper threading (HT)

- Intel® Hyper-Threading Technology uses processor resources more efficiently, enabling multiple threads to run on each core.
- O.S. “sees” two cores and transparently try to execute two program on two different “cores”
- Generally bad for HPC ?



Challenges for multicore

- Relies on effective exploitation of multiple-thread parallelism
  - Need for parallel computing model and parallel programming model
- Aggravates memory wall problem
  - Memory bandwidth
    - Way to get data out of memory banks
    - Way to get data into multi-core processor array
    - Memory latency
  - Cache sharing

a little bit of jargon..

- Multiprocessor = server with more than 1 CPU
- Multicore = a CPU with more than 1 core
- Processor = CPU = socket

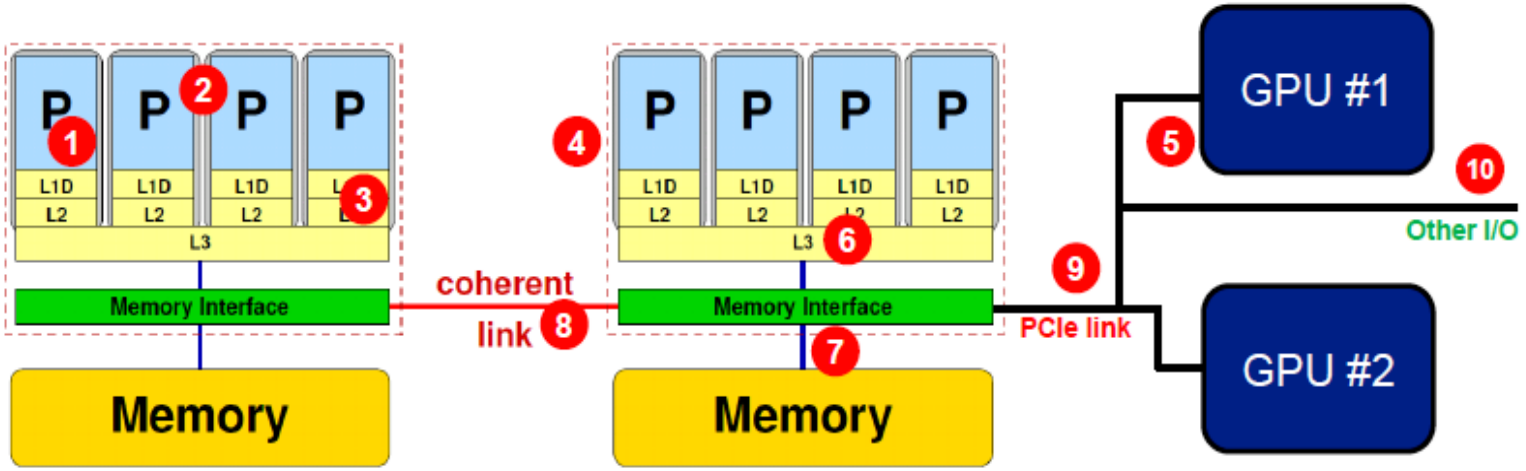
BUT SOMETIME:

- Processor = core
- a process for each processor (i.e. each core)

Agenda

- Why HPC is parallel ? ✓
- Serial Computers ✓
- Parallel computers ✓
- HPC infrastructure ✓
- Parallel components in a cluster

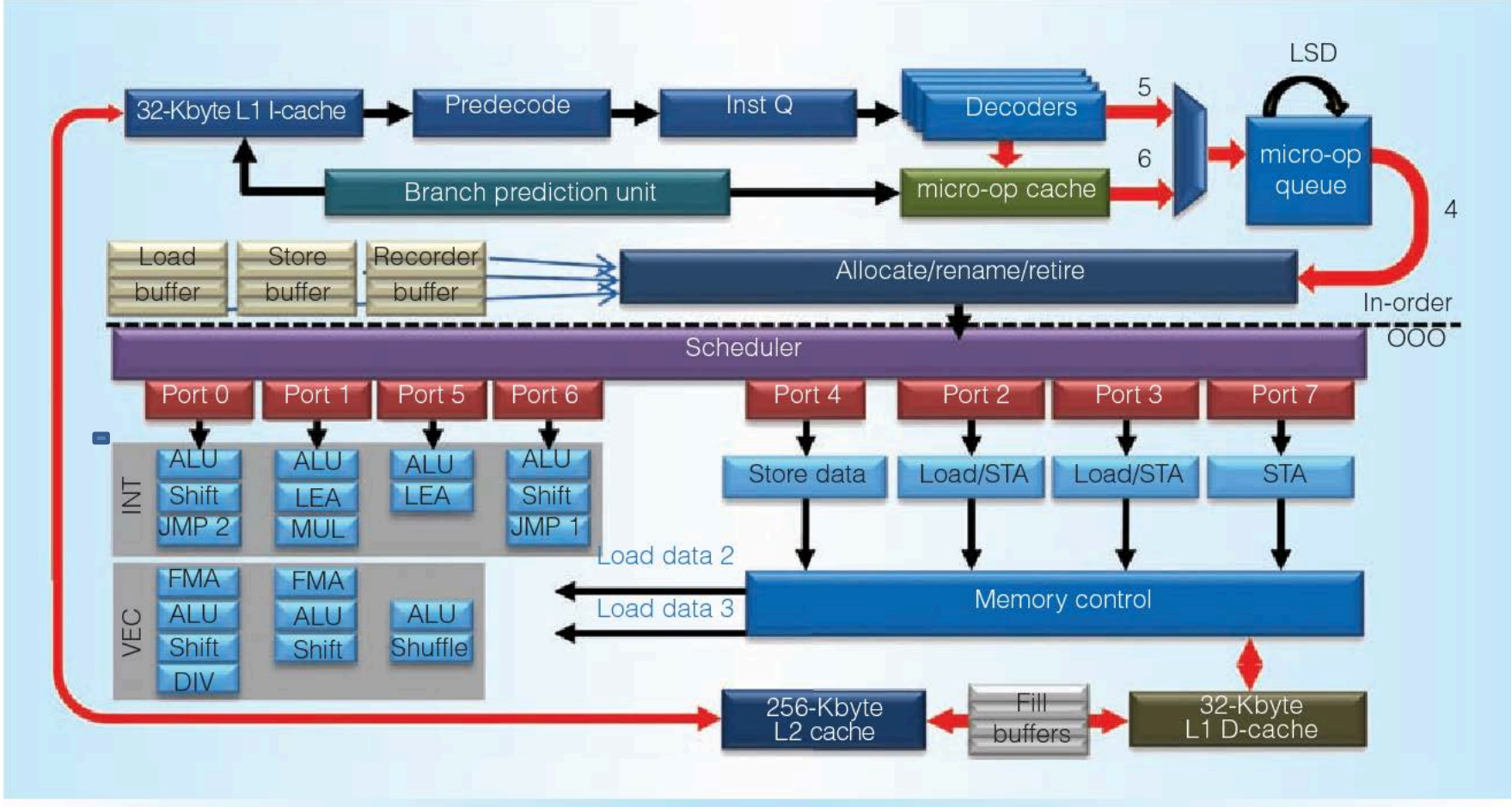
Parallellism within a HPC node



- Parallel resources
  - ILP/SIMD units (1)
  - Cores (2)
  - Inner cache levels (3)
  - Socket/ccNuma domains (4)
  - Multiple accelerator (5)

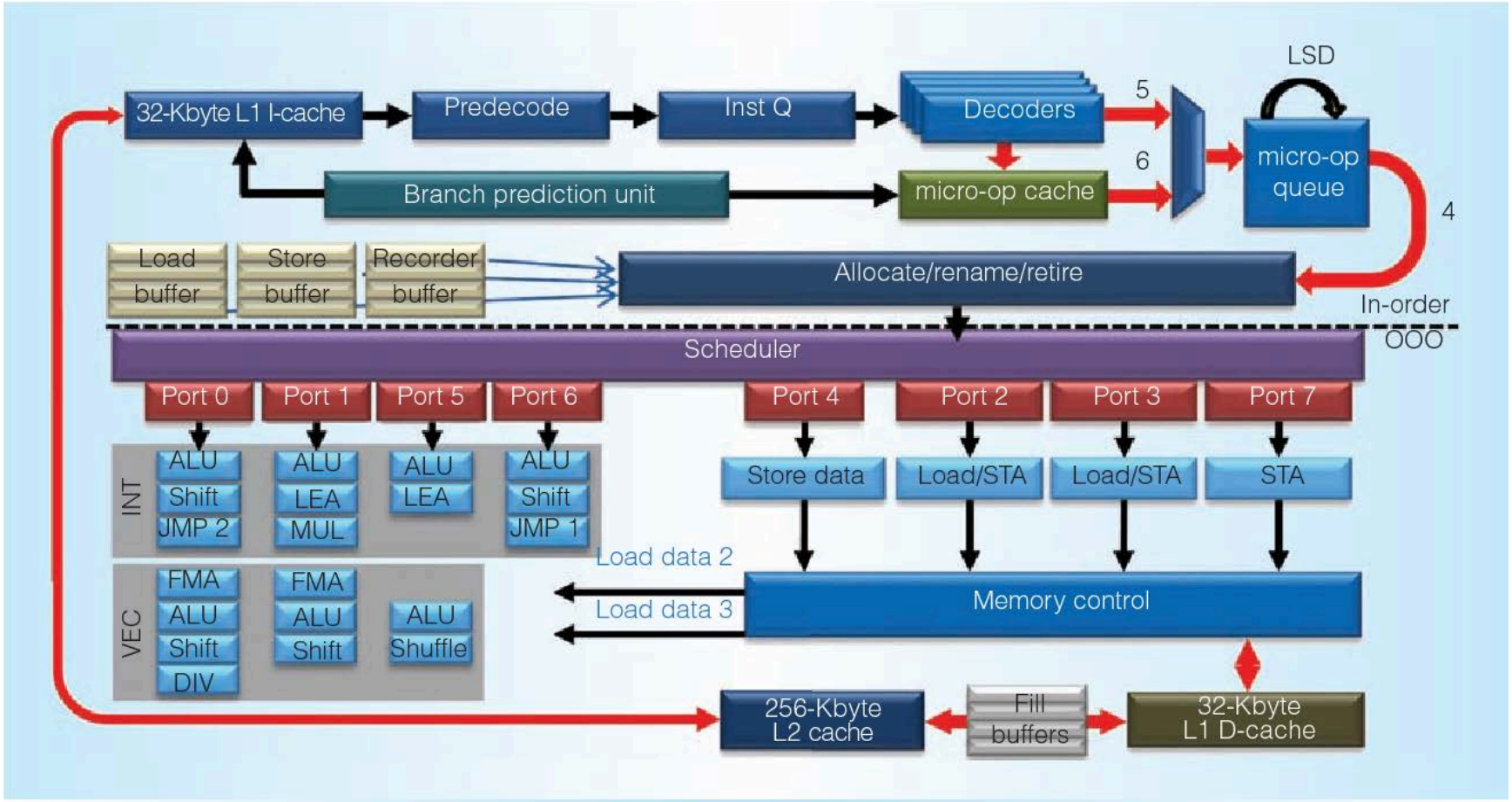
Core Level (1)

- Core can schedule instruction to more than one port at the same time

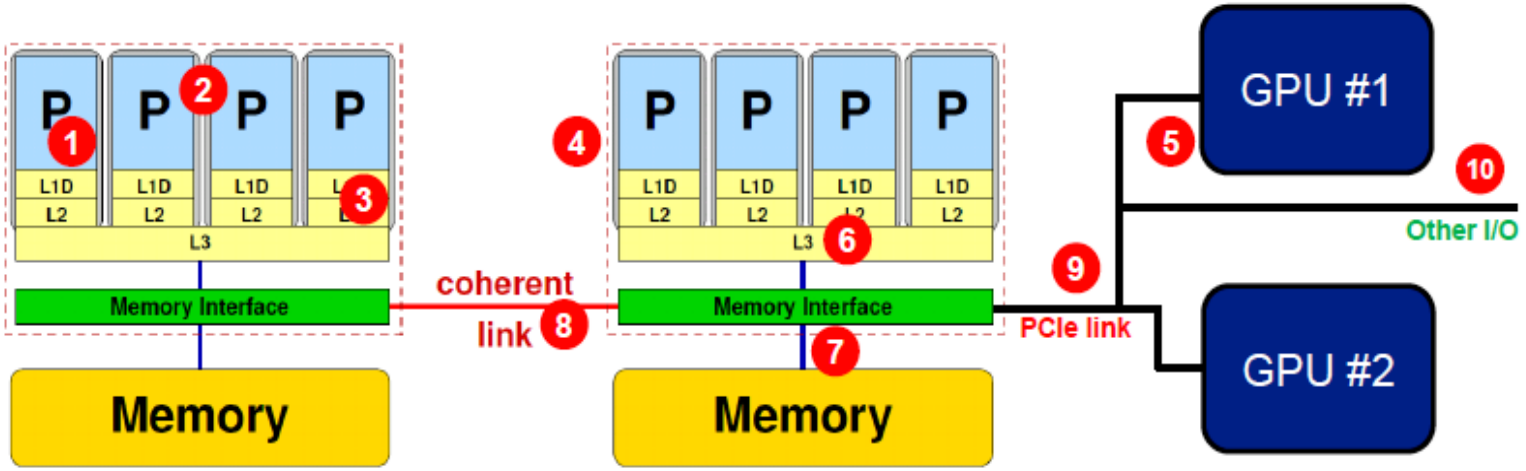


Core Level (2)

- Some port are/have SIMD devices..

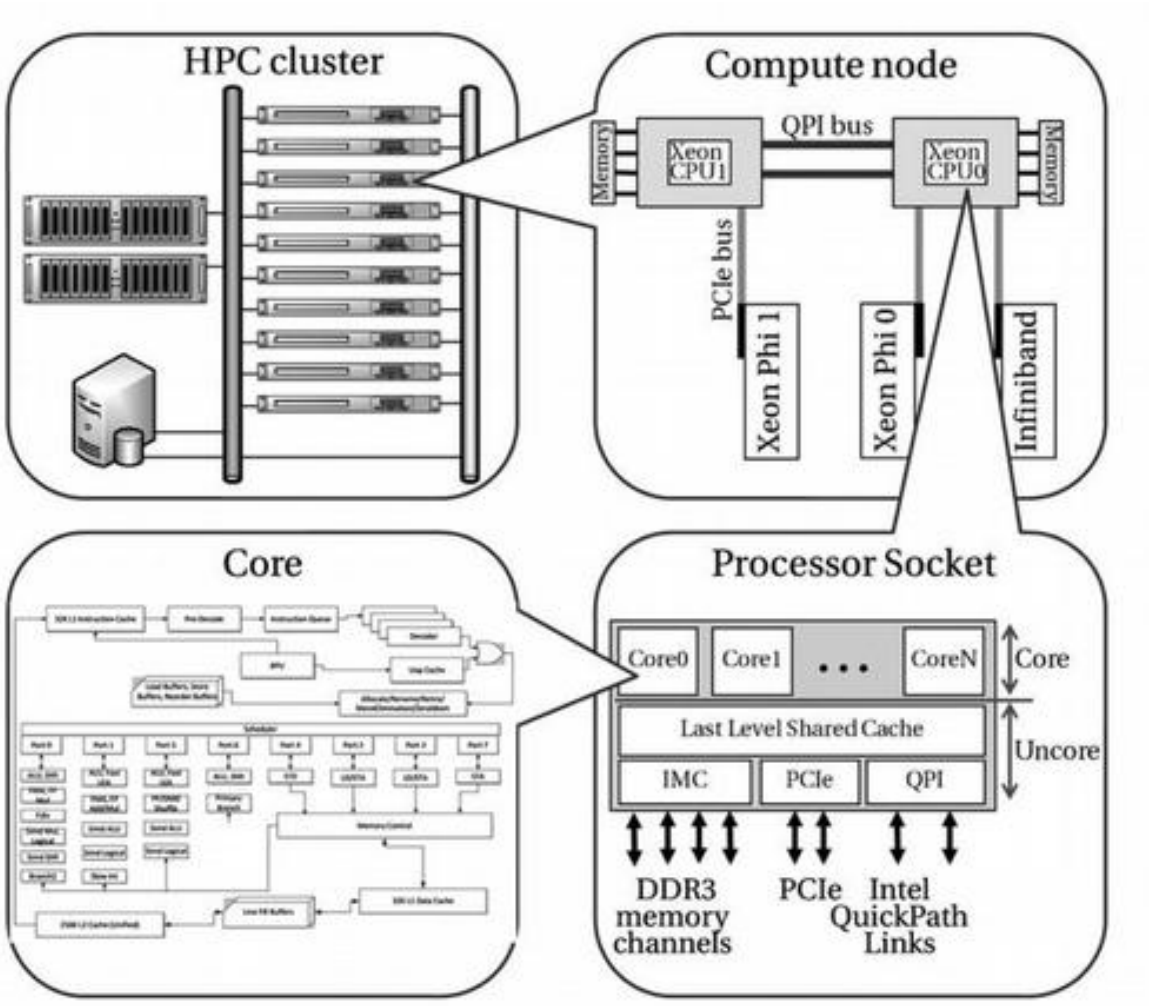


Parallellism within a HPC node

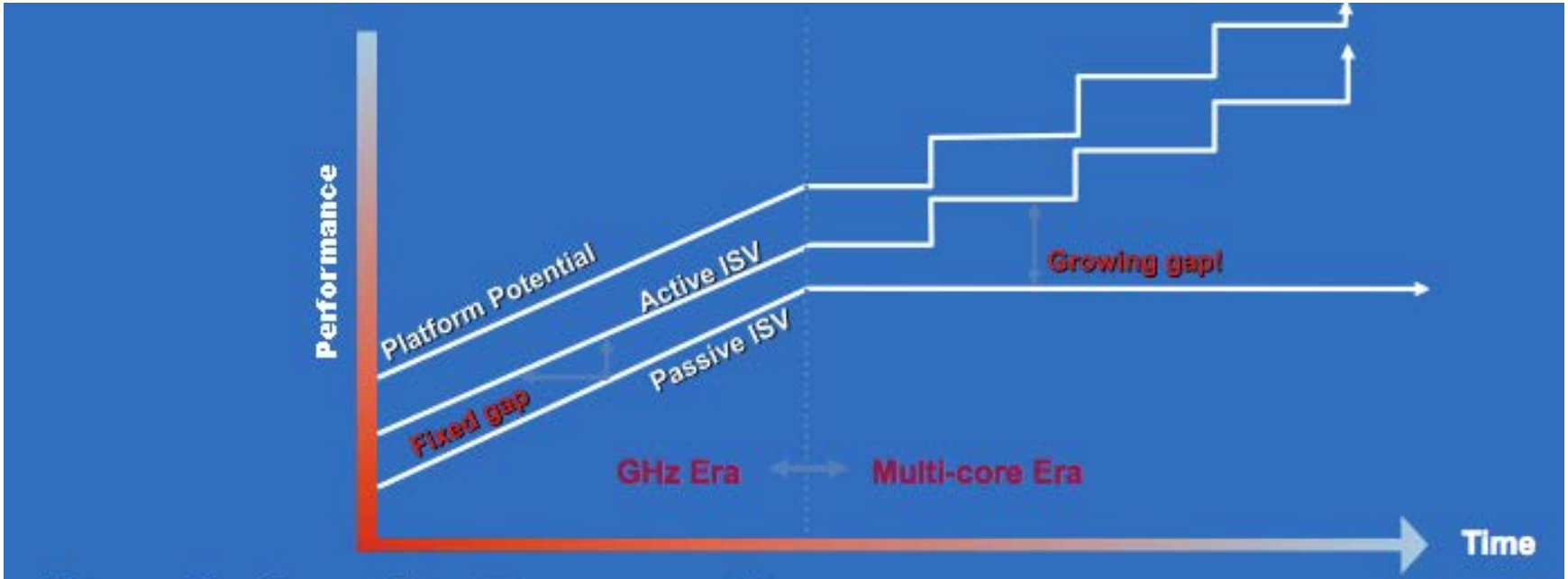


- Shared resources
  - Outer cache level per socket (6)
  - Memory bus per socket(7)
  - Intersocket link (8)
  - PCI-bus(es) (9)
  - Other I/O resources (10)

The building blocks of a HPC infrastructure (cluster)



An old picture from Intel..

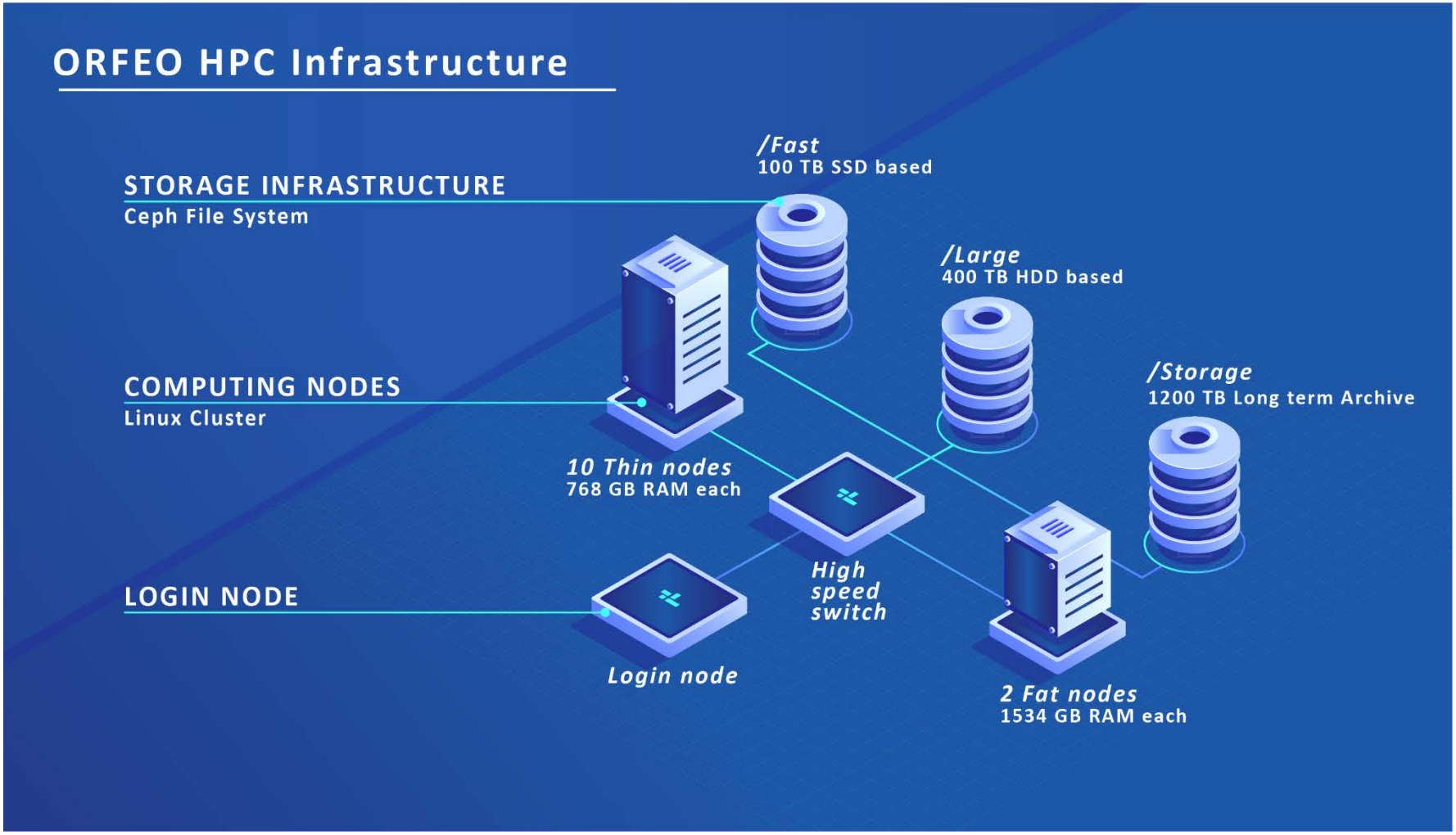


"Parallelism for Everyone"

Parallelism changes the game

- A large percentage of people who provide applications are going to have to care about parallelism in order to match the capabilities of their competitors.

A sophisticated Linux Cluster



All done !

- Why HPC is parallel ? ✓
- Serial Computers ✓
- Parallel computers ✓
- HPC infrastructure ✓
- Parallel components in a cluster ✓