HPC Architectures

Types of HPC hardware platforms currently in use

Partners

Funding

Horizon 2020
European Union Funding
for Research & Innovation

Reusing this material

Outline

- Shared memory architectures
  - Symmetric Multi-Processing (SMP) architectures
  - Non-Uniform Memory Access (NUMA) architectures
- Distributed memory architectures
- Hybrid distributed / shared memory architectures
- Accelerators

Architectures

- Architecture is about how different hardware components are connected together to make up usable machines

- Many factors influence choice of architecture:
  - Performance, cost, scalability, use cases, …

- Focus on the most important distinctions how processors and memory are situated and connected in HPC

- Discuss the role this plays in how parallel computing can be done on different architectures and how it can be expected to perform

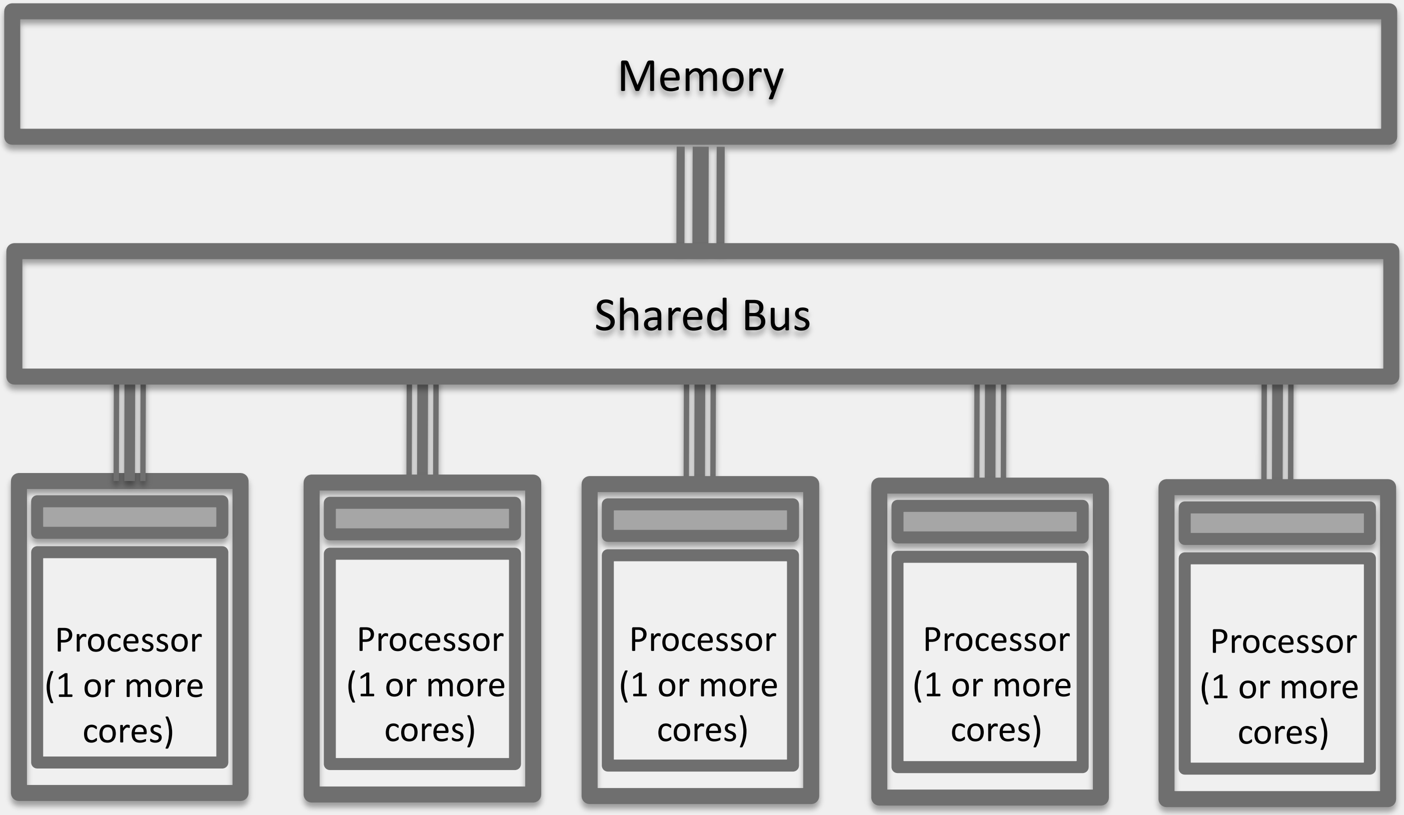
Shared memory architectures

Simplest to use, hardest to build

Shared-Memory Architectures

- Single memory that can be accessed by all processors / cores
- A single running instance of an Operating System (OS) controls the entire system (all memory, all processors / cores)
- Multi-processor shared-memory systems common in early 1990's
  - originally built from many single-core processors
- Modern multicore processors are really just shared-memory systems on a single chip
  - Nowadays can't buy a single-core processor even if you wanted one!

Symmetric Multi-Processing* Architectures

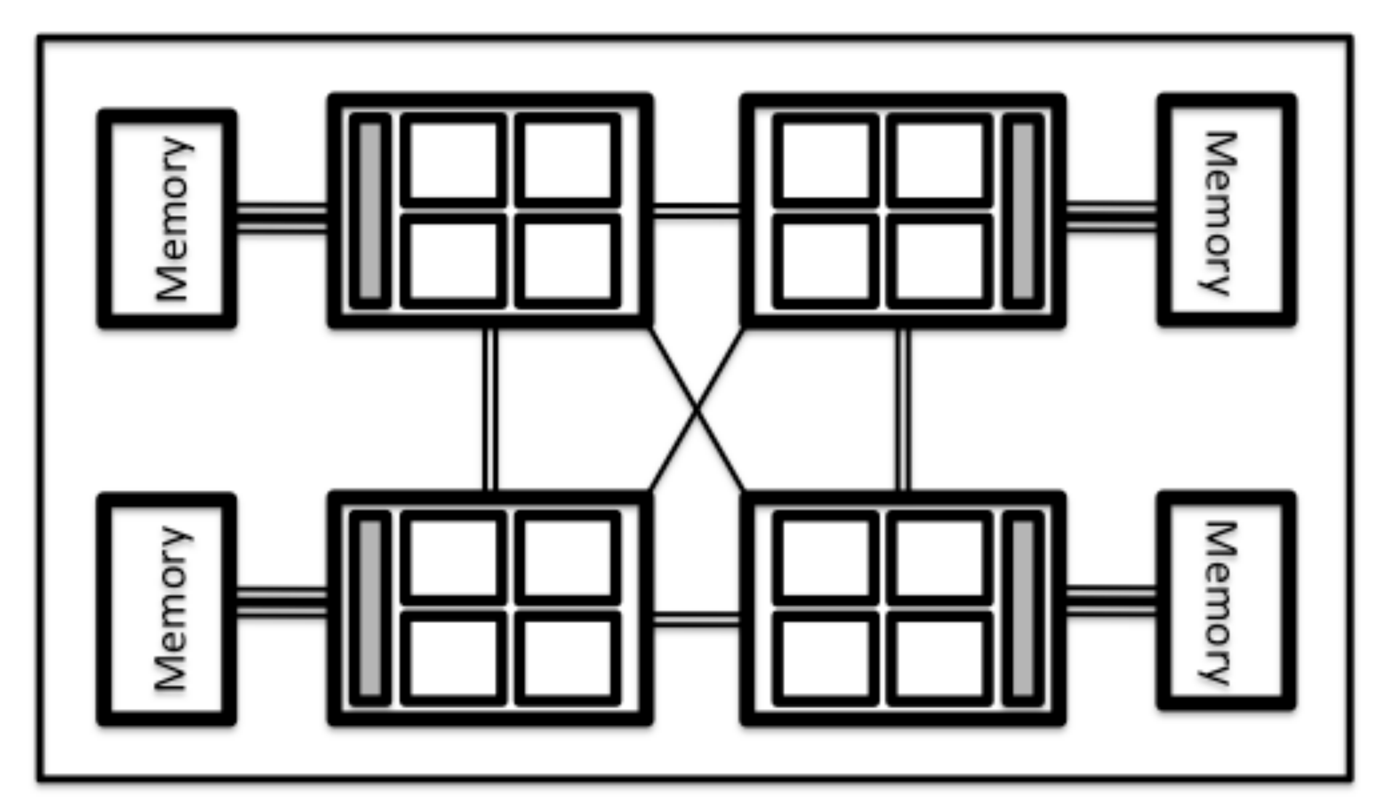

Memory
Shared Bus
Processor (1 or more cores)
Processor (1 or more cores)
Processor (1 or more cores)
Processor (1 or more cores)
Processor (1 or more cores)

*SMP

Symmetric: all cores access the same memory at the same speed

Example: a multicore laptop

Non-Uniform Memory Access* Architectures



*NUMA

Cores have access to memory used by other cores, but more slowly than access to their own local memory

Example: server, workstation, HPC node

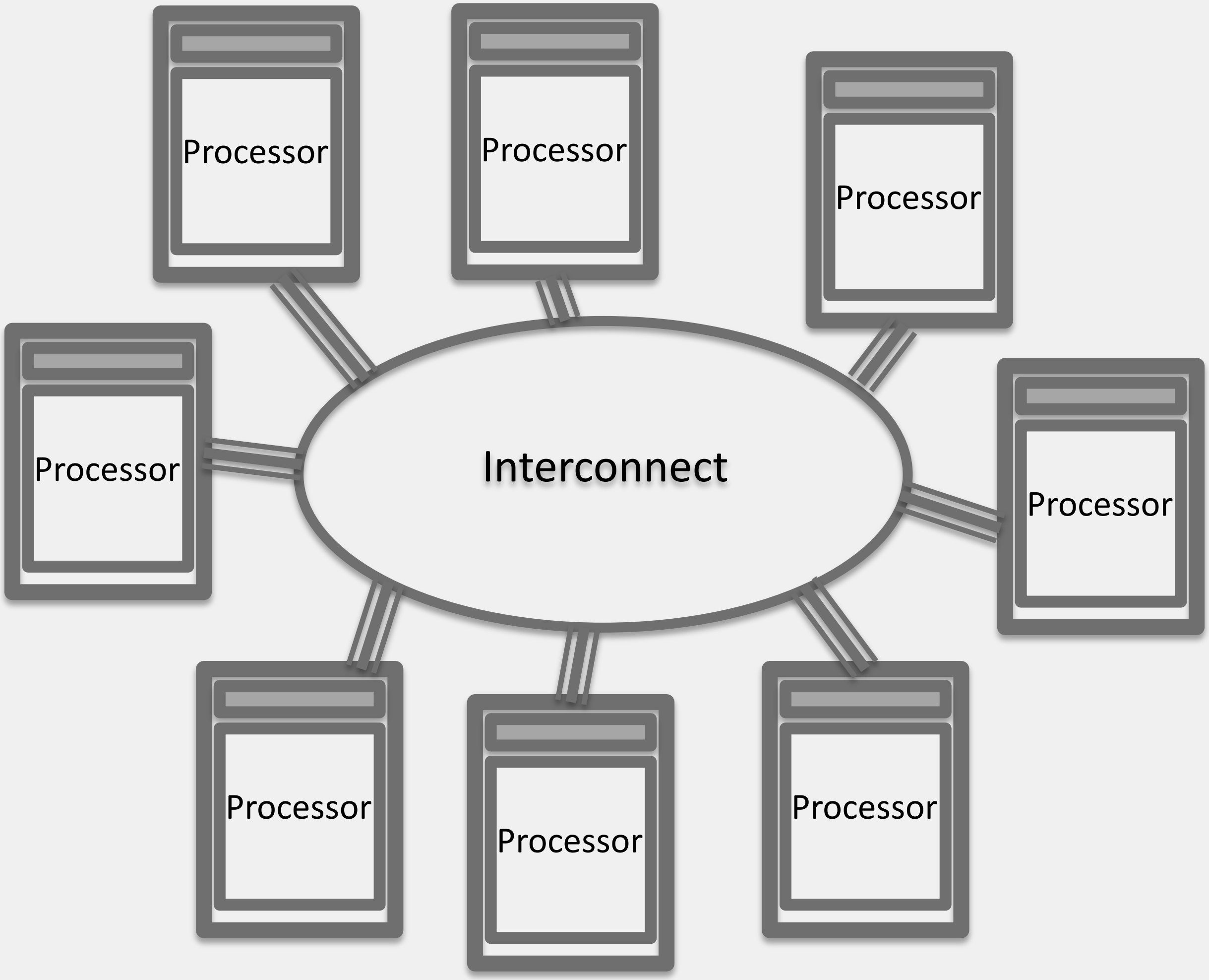
Shared-memory architectures

- Most small shared-memory computers (e.g. personal laptops, desktops) = SMP
  - Single processors (single socket multicore)
- "Serious" shared-memory computers (workstations, servers, HPC nodes) mostly NUMA
  - Multiple processors (multiple sockets multicore)
    - Even e.g. single socket AMD EPYC Zen2 (ARCHER2) = 4 NUMA regions
  - Few high-powered true SMP architectures
  - Program NUMA as if they are SMP – details hidden but performance impact
  - All cores controlled by a single OS
- Difficult to build shared-memory systems with large core numbers (> 1024 cores)
  - Expensive and power hungry
  - Difficult to scale the OS to this level

Distributed memory architectures

Clusters and interconnects

Multiple Connected Computers



- Each self-contained part is called a *node*.
  - each node runs its own copy of the OS

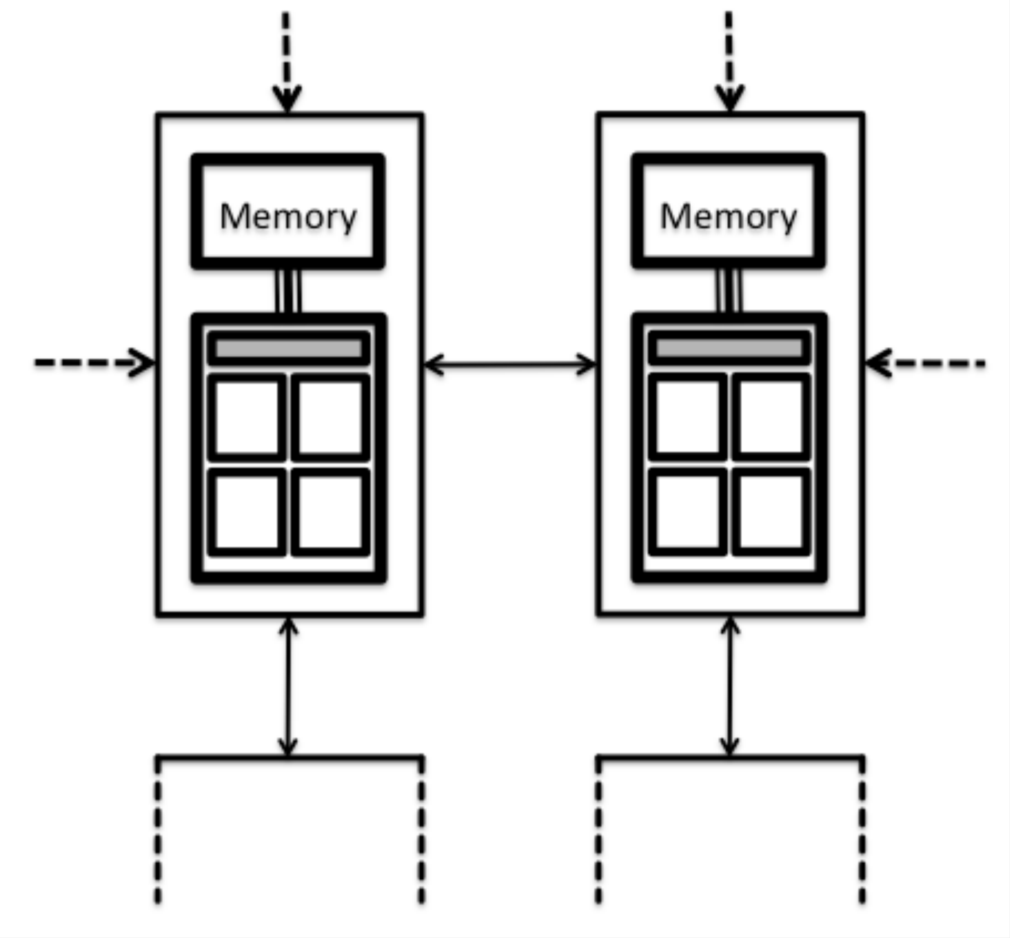
Distributed-memory architectures

- Almost all HPC machines are distributed memory
- Performance of parallel programs often depends on *interconnect* performance
  - Once interconnect a certain (high) quality, applications either CPU bound, memory bound, or IO bound
  - Lower quality interconnects (e.g. 10Mb/s – 1Gb/s Ethernet) do not usually provide enough performance for HPC
  - Specialist interconnects required to produce the largest supercomputers *e.g.* Cray Slingshot
  - Infiniband is dominant on smaller systems.
- High bandwidth relatively easy to achieve
  - low latency is usually more important and harder to achieve

Distributed/shared memory hybrids

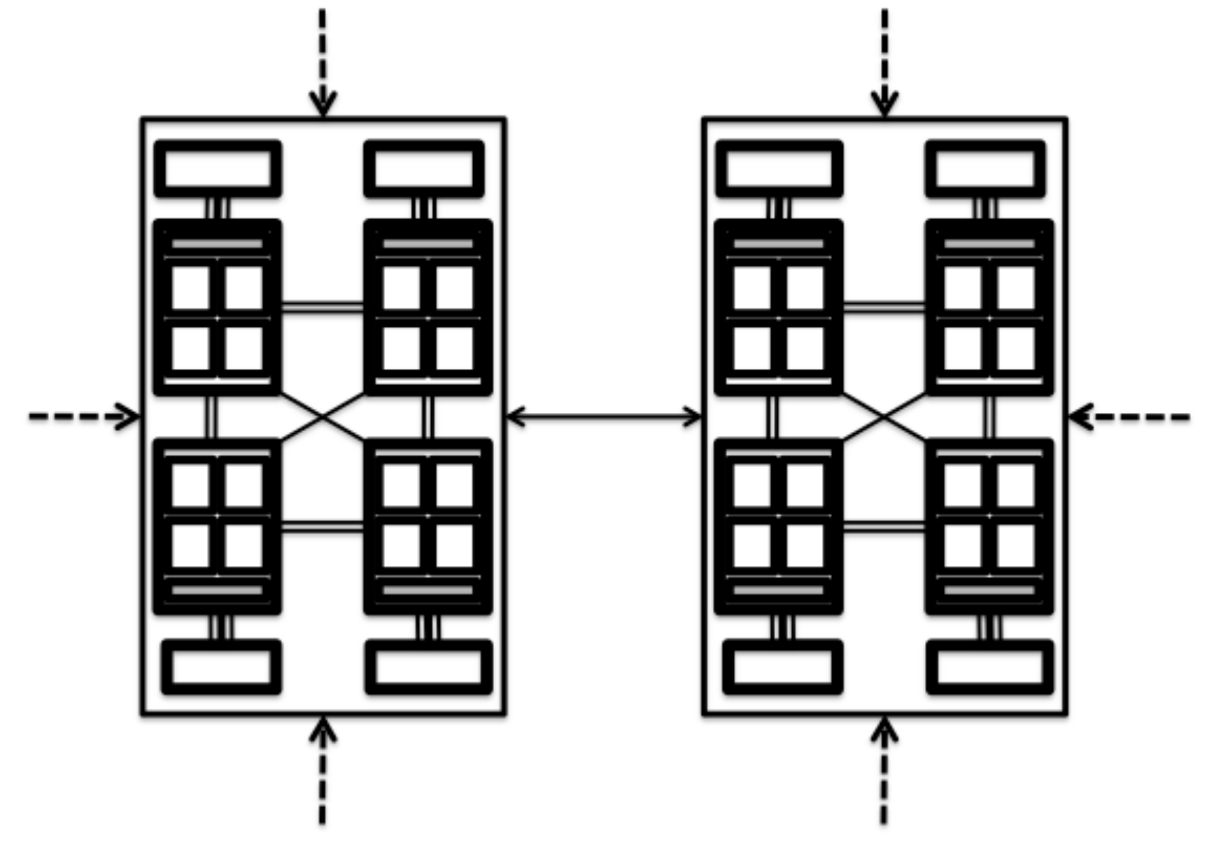
Almost everything now falls into this class

Multicore nodes



- In a real system:
  - each node will be a shared-memory system
    - e.g. a multicore processor
  - the network will have some specific topology
    - e.g. a regular grid

Hybrid architectures



- Now normal to have NUMA nodes
  - e.g. multi-socket systems with multicore processors
- Each node still runs a single copy of the OS

Hybrid architectures

- Almost all HPC machines fall in this class
- Most HPC applications include a message-passing (MPI) model for programming
  - Often use a single process per core
- Increased use of hybrid message-passing + shared memory (MPI+OpenMP) programming
  - Usually use 1 or more processes per NUMA region and then the appropriate number of shared-memory threads to occupy all the cores
- Placement of processes and threads can become complicated on these machines

Accelerators

How are they incorporated?

Including accelerators

- Accelerators usually incorporated into HPC machines using hybrid architecture model
  - A number of accelerators per node
  - Nodes connected using interconnects
  - Each accelerator typically has its own memory
- Communication from accelerator to accelerator depends on the hardware:
  - NVIDIA GPUs support direct communication (NVLINK)
  - AMD GPUs – Infinity Fabric includes
  - Intel Xeon Phi communication via CPU memory
  - Communicating via CPU memory involves lots of extra copy operations and is usually very slow

Summary

- Vast majority of HPC machines are shared-memory nodes linked by an interconnect.
  - Hybrid HPC architectures – combination of shared and distributed memory
  - Most are programmed using a pure MPI model (more later on MPI) - does not really reflect the hardware layout
- Accelerators are incorporated at the node level
  - May increasingly be able to avoid costly communication via node memory through direct GPU-GPU links