HPC Architectures

Types of resource currently in use

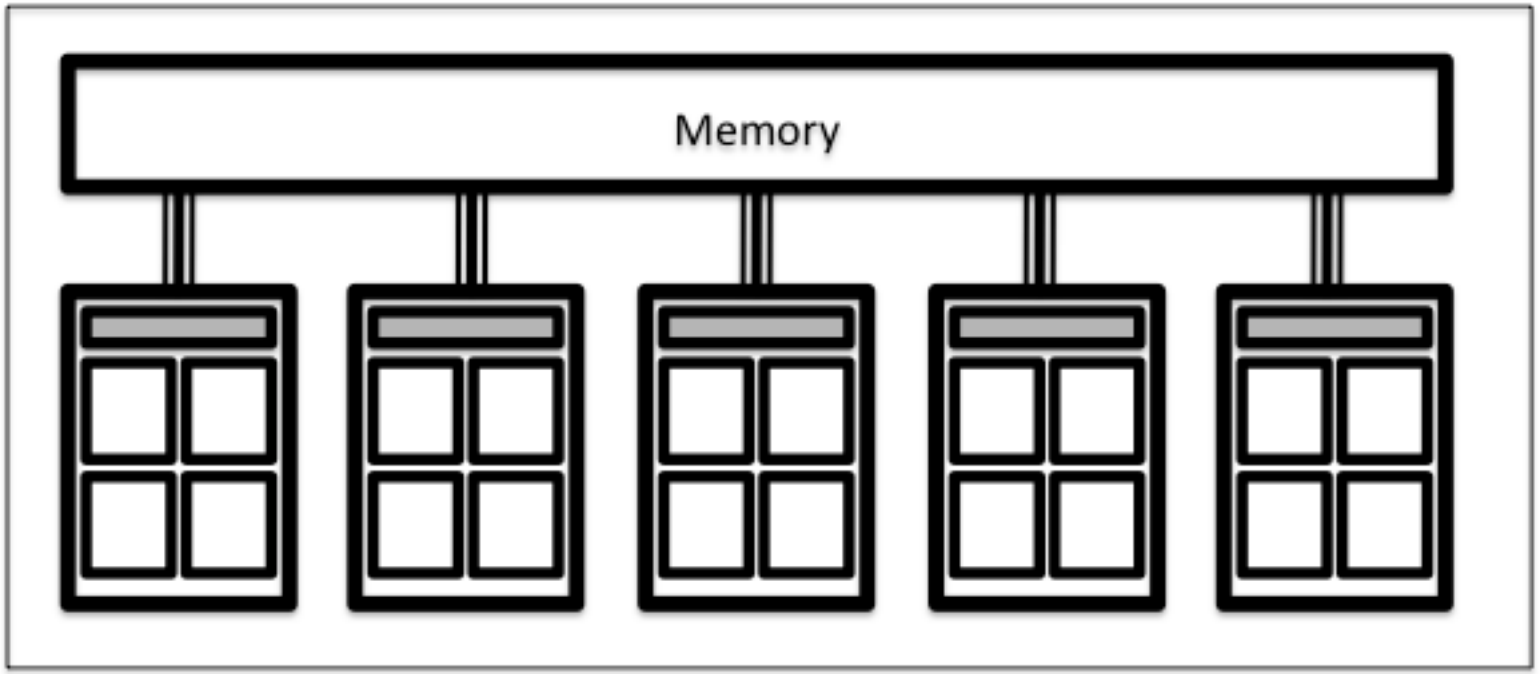
Outline

- Shared memory architectures
- Distributed memory architectures
- Distributed memory with shared-memory nodes
- Accelerators
- What is the difference between different Tiers?
  - Interconnect
  - Software
  - Job-size bias (capability)

Shared memory architectures

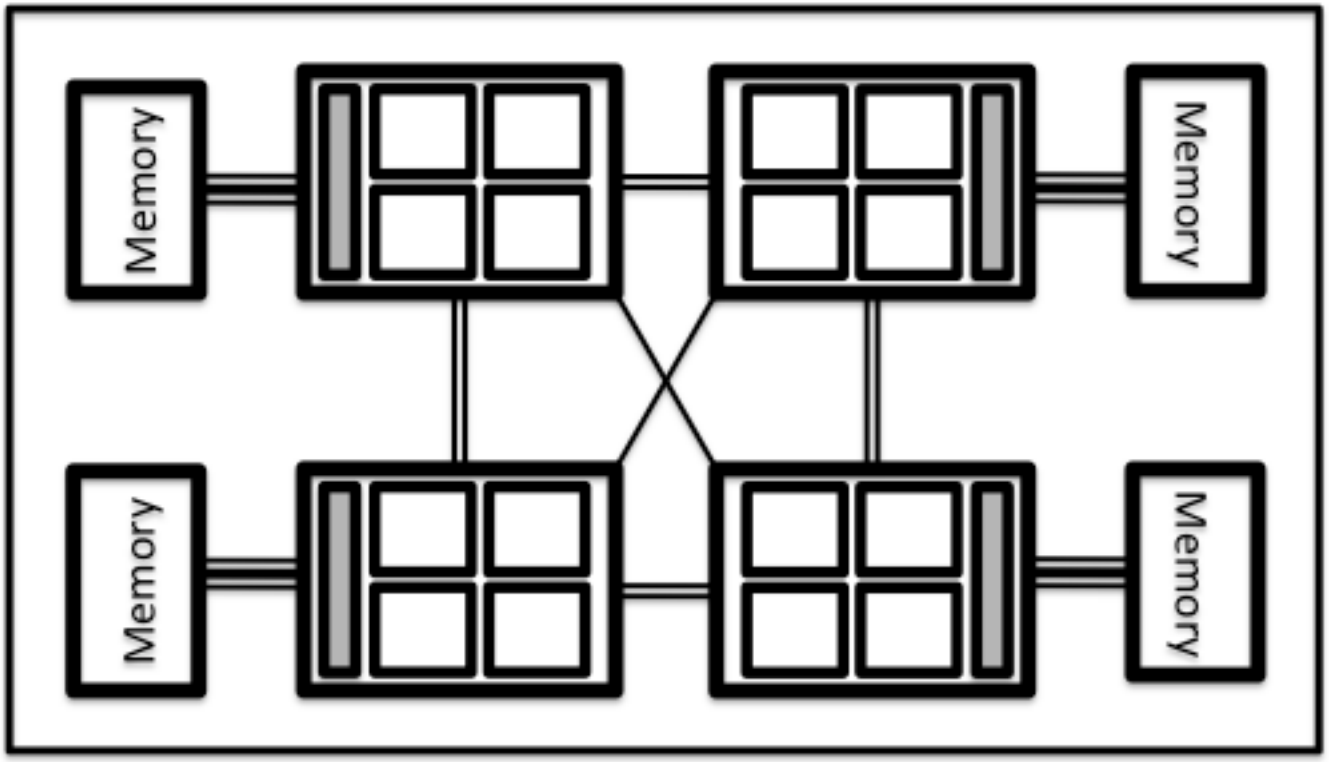
Simplest to use, hardest to build

Symmetric Multi-Processing Architectures



- All cores have the same access to memory

Non-Uniform Memory Access Architectures



- Cores have faster/wider access to local memory

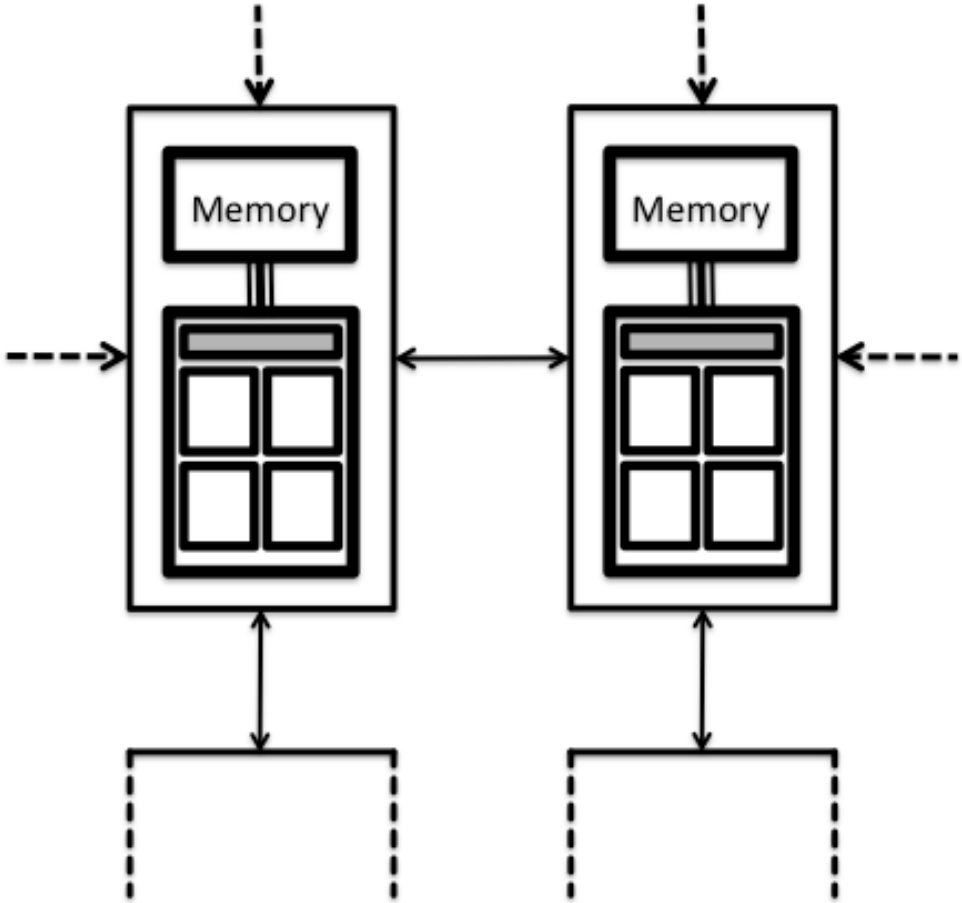
Shared-memory architectures

- Most computers are now shared memory machines due to multicore
- Some true SMP architectures…
  - *e.g.* BlueGene/Q nodes
- …but most are NUMA
  - Program NUMA as if they are SMP – details are hidden from the user.
- Difficult to build shared-memory systems with large core numbers (> 1024 cores)
  - Expensive and power hungry
  - Some systems manage by using software to provide shared-memory capability

Distributed memory architectures

Clusters and interconnects

Distributed-Memory Architectures



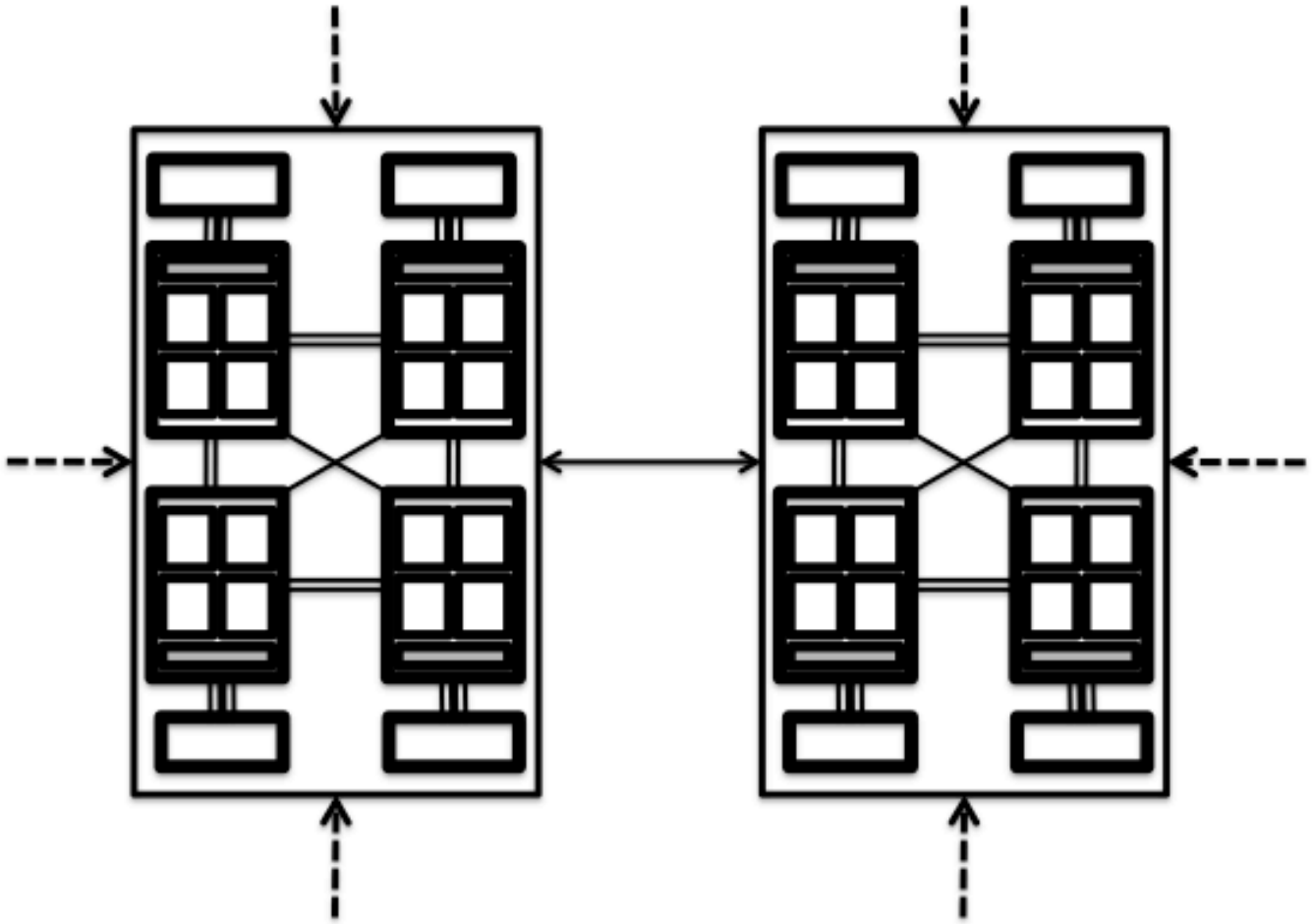
Distributed-memory architectures

- Each self-contained part is called a *node*.
- Almost all HPC machines are distributed memory in some way
  - Although they all tend to be shared-memory within a node.
- The performance of parallel programs often depends on the *interconnect* performance
  - Although once it is of a certain (high) quality, applications usually reveal themselves to be CPU, memory or IO bound
  - Low quality interconnects (e.g. 10Mb/s – 1Gb/s Ethernet) do not usually provide the performance required
  - Specialist interconnects are required to produce the largest supercomputers. *e.g.* Cray Aries, IBM BlueGene/Q
  - Infiniband is dominant on smaller systems.

Distributed/shared memory hybrids

Almost everything now falls into this class

Hybrid Architectures



archer

epcc

THE UNIVERSITY OF EDINBURGH

Hybrid architectures

- Almost all HPC machines fall in this class
- Most applications use a message-passing (MPI) model for programming
  - Usually 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

Example: ARCHER

- ARCHER has two 12-way multicore processors per node
  - Each 12-way processor is made up of two 6-core *dies*
  - Each node is a 24-core, shared-memory, NUMA machine

archer

epcc

THE UNIVERSITY OF EDINBURGH

Accelerators

How are they incorporated?

Including accelerators

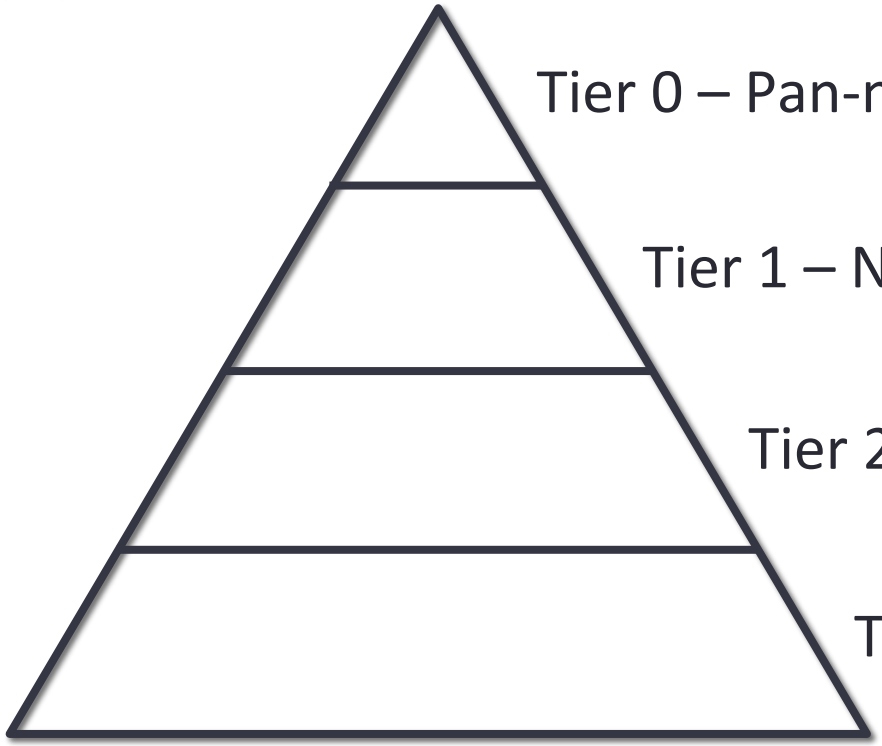
- Accelerators are usually incorporated into HPC machines using the hybrid architecture model
  - A number of accelerators per node
  - Nodes connected using interconnects
- Communication from accelerator to accelerator depends on the hardware:
  - NVIDIA GPU support direct communication
  - AMD GPU have to communicate via CPU memory
  - Intel Xeon Phi communication via CPU memory
  - Communicating via CPU memory involves lots of extra copy operations and is usually very slow

Comparison of types

What is the difference between different tiers?

HPC Facility Tiers

- HPC facilities are often spoken about as belonging to *Tiers*



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
- Shared HPC machines span a wide range of sizes:
  - From Tier 0 – Multi-petaflops (1 million cores)
  - To workstations with multiple CPUs (+ Accelerators)