

# Parallel and Distributed Systems / 5

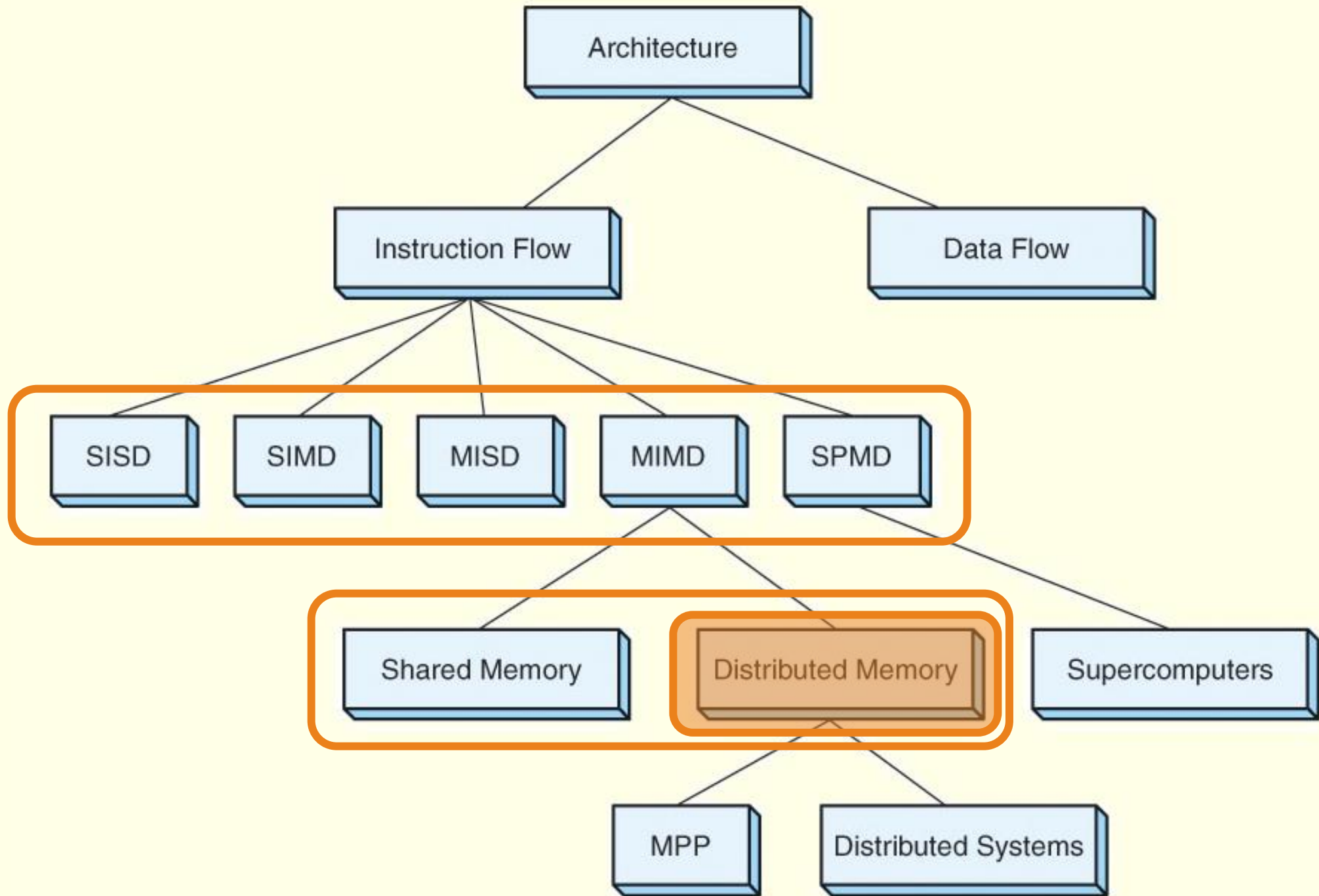


Pavel Krömer,  
Dept. of Computer Science,  
VSB – Technical University of  
Ostrava

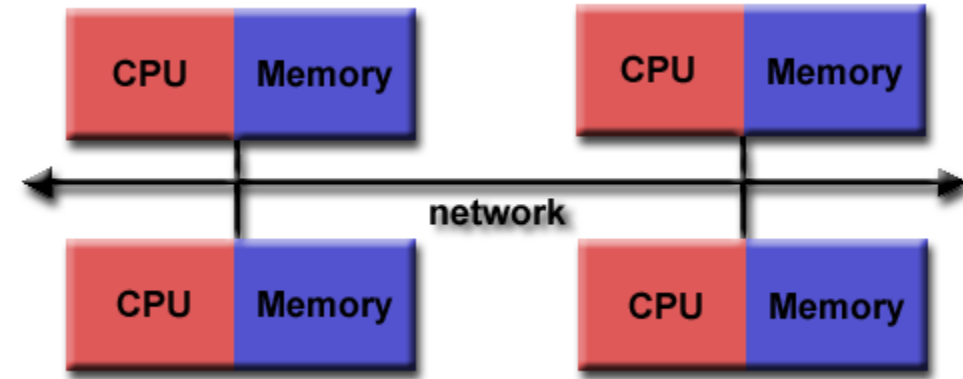
# Agenda

- Parallel systems with distributed memory
  - Computer clusters and HPC
- Literature
  - C. S. R. PRABHU: GRID AND CLUSTER COMPUTING. PHI Learning Pvt. Ltd., 2008





# Distributed memory systems



Use a communication network to connect inter-processor memory.

Processors have own local memory and memory addresses in one processor do not map to another processor. No global address space exists.

Processors operate independently, local changes do not affect other processors (and their memory). Cache coherency out of question.

IPC (data sharing + synchronization) explicitly defined by program/programmer.

Also **loosely coupled systems**

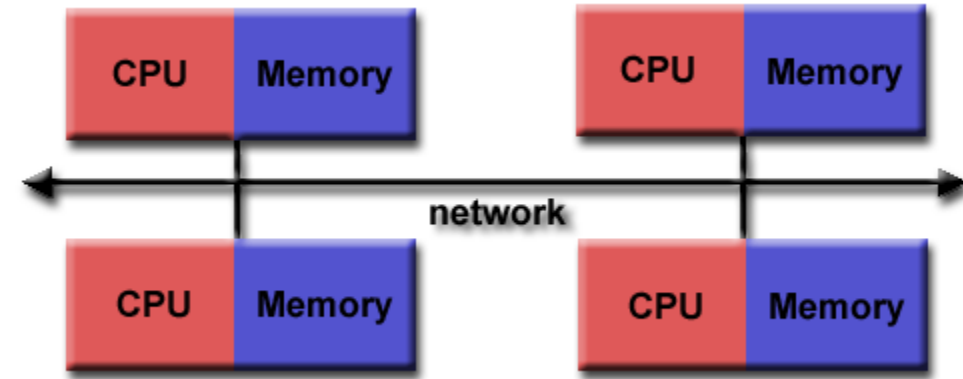
# Distributed memory systems

## Challenging IPC

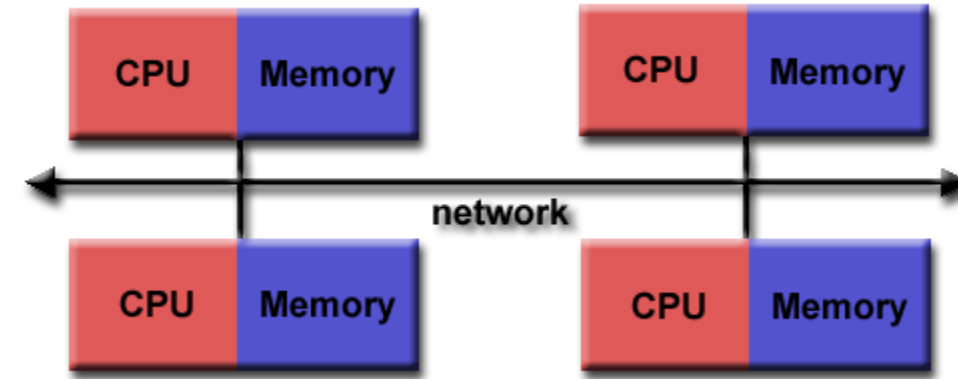
- Explicit IPC via message passing
- Communication usually a significant bottleneck
- Communication-driven algorithm design and considerations

## Advantages

- Memory scalable with the number of processors
- Each processor can rapidly access its own memory without interference and without any special overhead wrt. cache coherency
- Cost effective, can use commodity, off-the-shelf processors and networking



# Distributed memory systems

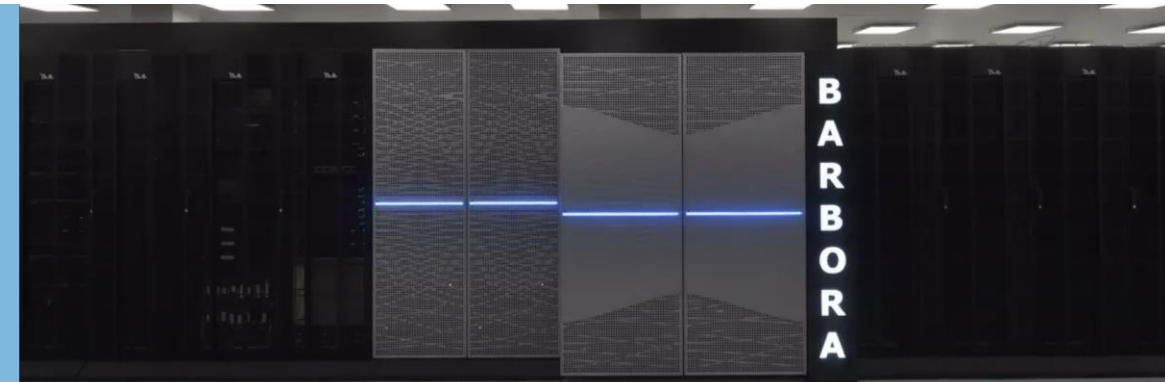


## Disadvantages

- Program/programmer responsible for all details associated with data communication between processors
- Difficult mapping of existing data structures to distributed memory for an **efficient execution**
- Non-uniform memory access times - data residing on a remote node takes longer to access than local data

## Barbora

- 2. 10. 2019, VSB-TUO / IT4Innovations
- 192x compute node w. 18-core CPU, 192 GiB RAM
- 8x GPU node w. 12-core CPU, 192 GiB RAM, 4x NVIDIA V100
- 1x fat node w. 8x16-core CPU, 6 TiB RAM







# Computer cluster

A **computer cluster** is a set of interconnected computers (compute **nodes**) that work together as if they were a single **orchestrated** resource.

Each **node** has its own OS and software stack and the cluster operation is achieved through additional software (middleware, communication, scheduling, etc.)

# Computer cluster

A *very* broad category



Mini-Wulf – 6x PC w. FreeBSD



1024xRPi - RasPi Supercomp



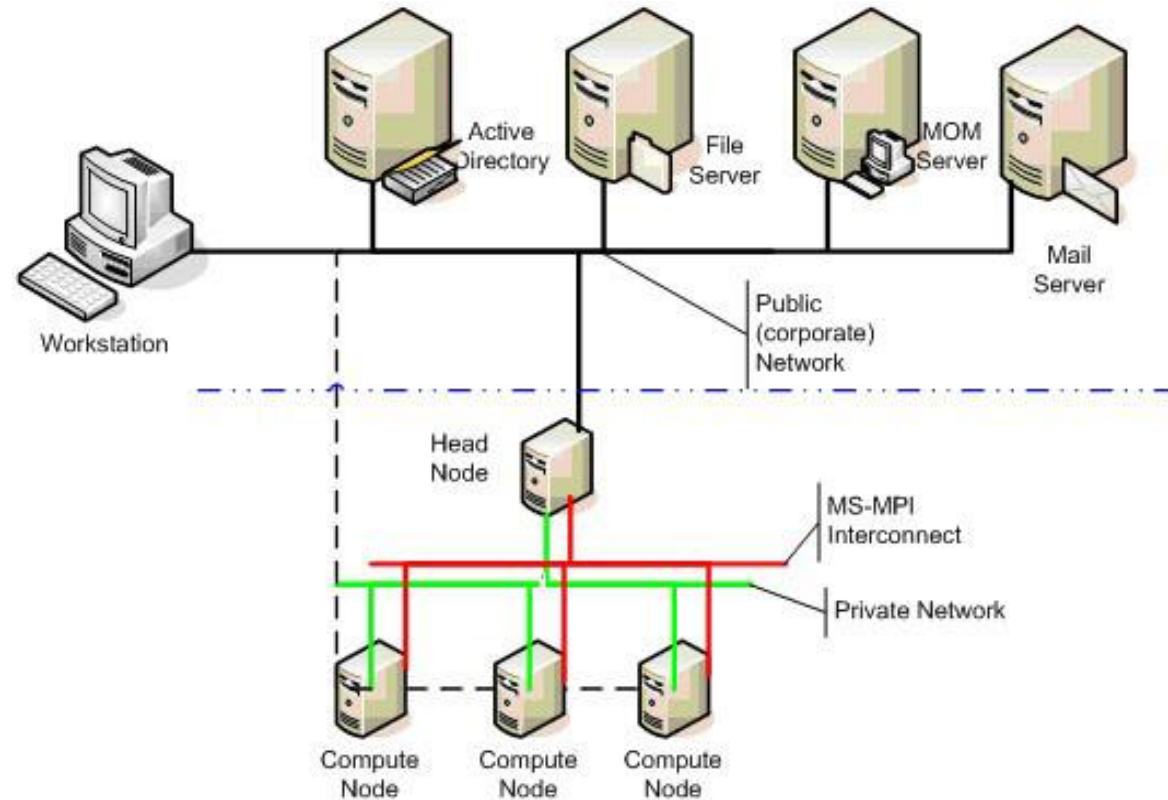
Archer – A UK National Resource Cray XC30

# Computer cluster

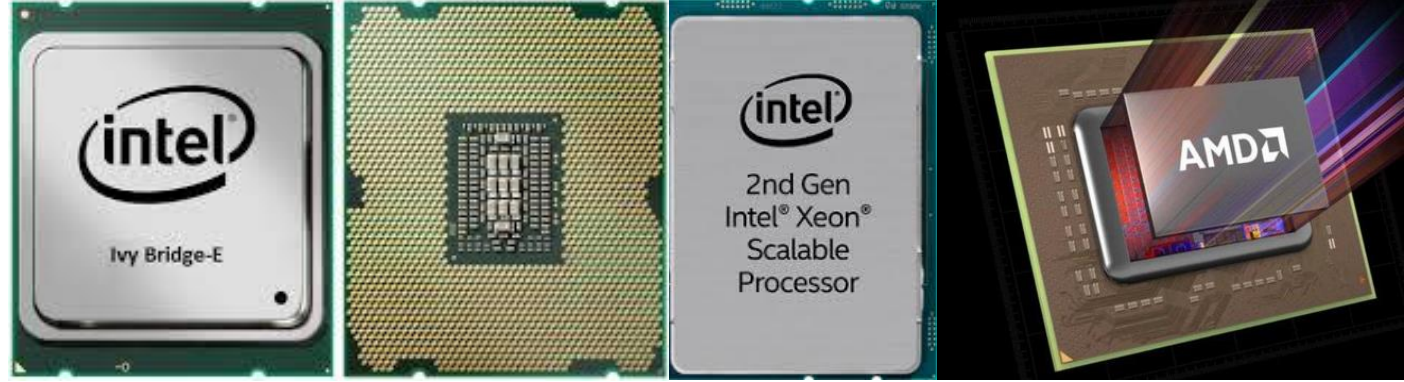
A set of loosely or tightly connected compute nodes that work together so that, in many respects, they can be viewed as a single system.

Each node (usually) performs the same task and is controlled by a software scheduler. They operate in a **non-interactive** (batch) manner

They (usually) use **high-performance networking hardware** interconnect networks). - 100 Gb Ethernet - Fiber Channel Standard (FCS) - Myrinet (past) - *Infiniband*



# Compute node(s)



## Archer

- two 2.7 GHz, 12-core E5-2697 v2 (Ivy Bridge) series processors
- 64/128 GB of memory (NUMA wrt. the processors)
- 4544 standard memory nodes (12 groups, 109,056 cores) and 376 high memory nodes (1 group, 9,024 cores)
- Aries interconnect

## Barbora

- 192x compute node w. 18-core CPU, 192 GiB RAM
- 8x GPU node w. 12-core CPU, 192 GiB RAM, 4x NVIDIA V100
- 1x fat node w. 8x16-core CPU, 6 TiB RAM
- Mellanox InfiniBand HDR 200 Gbps network

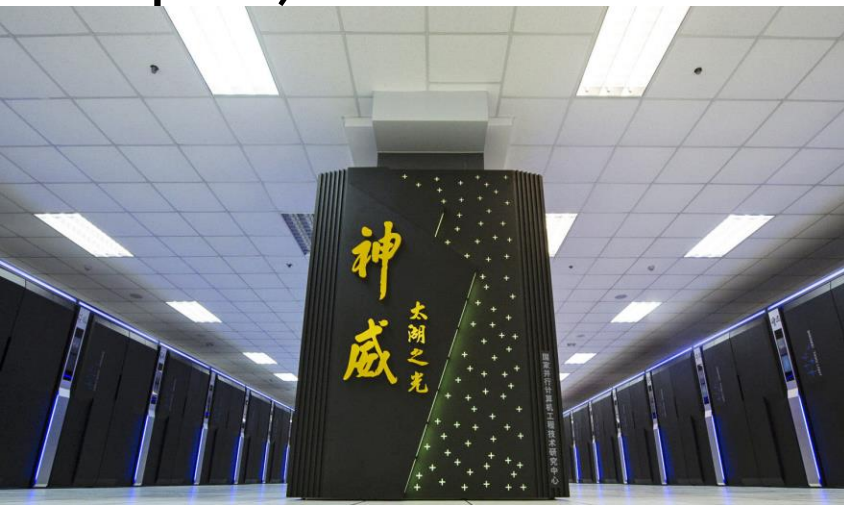
## Archer 2 (2020)

- Newly contracted to Cray
- 11,696 AMD Epyc Rome CPUs / x 64 cores
- total of 1.5m threads
- Europe's most powerful supercomputer

# Compute node(s)

## Sunway TaihuLight (2019: #3)


- 40,96 x 64-bit RISC processors (SW26010)
- Sunway architecture (2006+, based on DEC Alpha)



## Fugaku (2021)

- 48+2/4 ARM CPU by Fujitsu
- Out-of-order execution
- High-performance Arm8.2 CPU for HPC and AI



ISA	Armv8.2-A (AArch64 only) SVE (Scalable Vector Extension)	
SIMD width	512-bit	
Precision	FP64/32/16, INT64/32/16/8	
Cores	48 computing cores + 4 assistant cores (4 CMGs)	
Memory	HBM2: Peak B/W 1,024 GB/s	
Interconnect	TofuD: 28 Gbps x 2 lanes x 10 ports	



# Interconnect networks

Communication backbone of distributed systems. Connect compute (and other) nodes and aim to minimize bottlenecks caused by communication.

static (direct, p2p) vs. dynamic (indirect, switching, routing)