Lecture 9: MIMD Architectures

- **Introduction and classification**
- **Symmetric multiprocessors**
- **NUMA architecture**
- **Clusters**

Introduction

- A set of general purpose processors is connected together.
- In contrast to SIMD processors, MIMD processors can execute different programs on different processors.
  - Flexibility!
- MIMD processors works asynchronously, and don't have to synchronize with each other.
- By 90s, SIMD lost ground, since general purpose microprocessors are now very cheap and powerful.
  - MIMD machines could be built from commodity (off-the-shelf) microprocessors with relatively little effort.
- MIMD architectures can be highly scalable, if an appropriate memory organization is used.

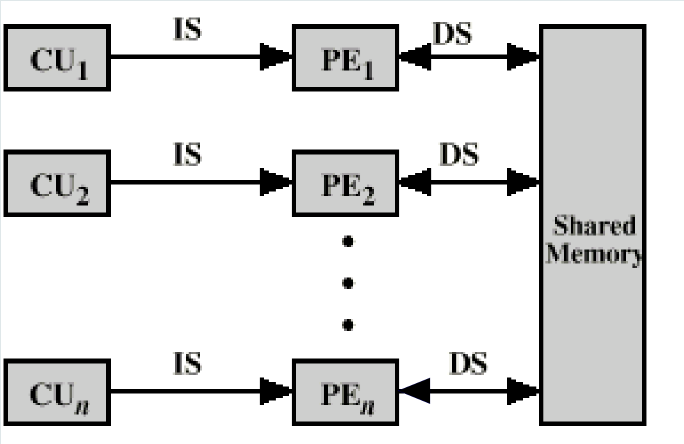
SIMD vs MIMD

- SIMD computers require less hardware than MIMD computers (single control unit).
- However, SIMD processors are specially designed, and tend to be expensive and have long design cycles.
- Not all applications are naturally suited to SIMD processors.
- Conceptually, MIMD computers cover SIMD need.
  - Having all processors executing the same program (single program multiple data streams - SPMD).
  - SPMD avoids the complexity of general concurrent programming.

MIMD Processor Classification

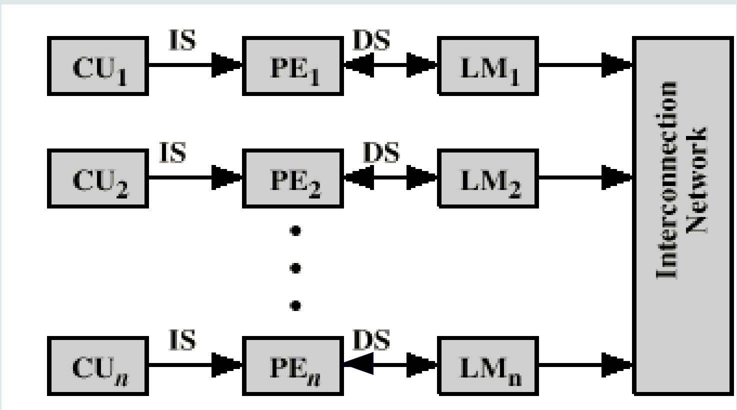
- **Centralized Memory:** Shared memory located at centralized location — consisting usually of several interleaved modules — the same distance from any processor.
  - **Symmetric Multiprocessor** (SMP)
  - **Uniform Memory Access** (UMA)
- **Distributed Memory:** Memory is distributed to each processor — improving scalability.
  - **Message Passing Architectures:** No processor can directly access another processor's memory.
  - **Hardware Distributed Shared Memory (DSM):** Memory is distributed, but the address space is shared.
    - Non-Uniform Memory Access (NUMA)
  - **Software DSM:** A level of OS built on top of message passing multiprocessor to give a shared memory view to the programmer.

MIMD with Shared Memory



- Tightly coupled, not scalable.
- Typically called **Multi-processor.**

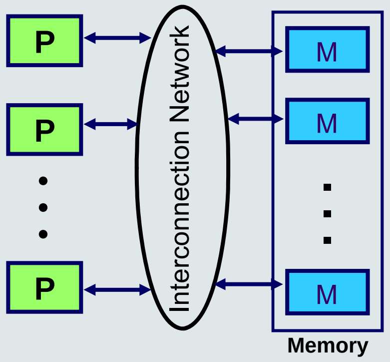
MIMD with Distributed Memory



- Loosely coupled with message passing, scalable
- Typically called **Multi-computer.**

Shared-Address-Space Platforms

- Part (or all) of the memory is accessible to all processors.
- Processors interact by modifying data objects stored in this shared-address-space.



- UMA (uniform memory access) = the time taken by a processor to access any memory word in the system is identical.
- NUMA, otherwise.

Multi-Computer Systems

- A multi-computer system is a collection of computers interconnected by a message-passing network.
  - Clusters.
- Each processor is an autonomous computer
  - With its own local memory.
  - There is no a shared-address-space any longer.
  - They communicating with each other through the message passing network.
- Can be easily built with commodity microprocessors.

MIMD Design Issues

- Design issues related to an MIMD machine are very complex, since it involves both architecture and software issues.
- The most important issues include:
  - Processor design
  - Physical organization
  - Interconnection structure
  - Inter-processor communication protocols
  - Memory hierarchy
  - Cache organization and coherency
  - Operating system design
  - Parallel programming languages
  - Application software techniques

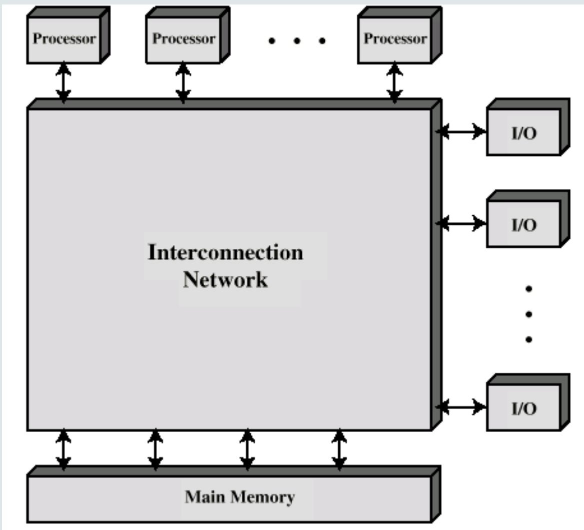
Lecture 9: MIMD Architectures

- **Introduction and classification**
- **Symmetric multiprocessors**
- **NUMA architecture**
- **Clusters**

Symmetric Multiprocessors (SMP)

- A set of similar processors of comparable capacity.
- All processors can perform the same functions (symmetric).
- Processors are connected by a bus or other internal connection.
- Processors share the same memory.
  - A single memory or a pool of memory modules
- Memory access time is (approximately) the same for each processor.
- All processors share access to I/O.
  - Either through the same channels or different channels giving paths to the same devices
- Controlled by an integrated operating system:
  - Providing interaction between processors
  - Interaction at job, task, file and data element levels

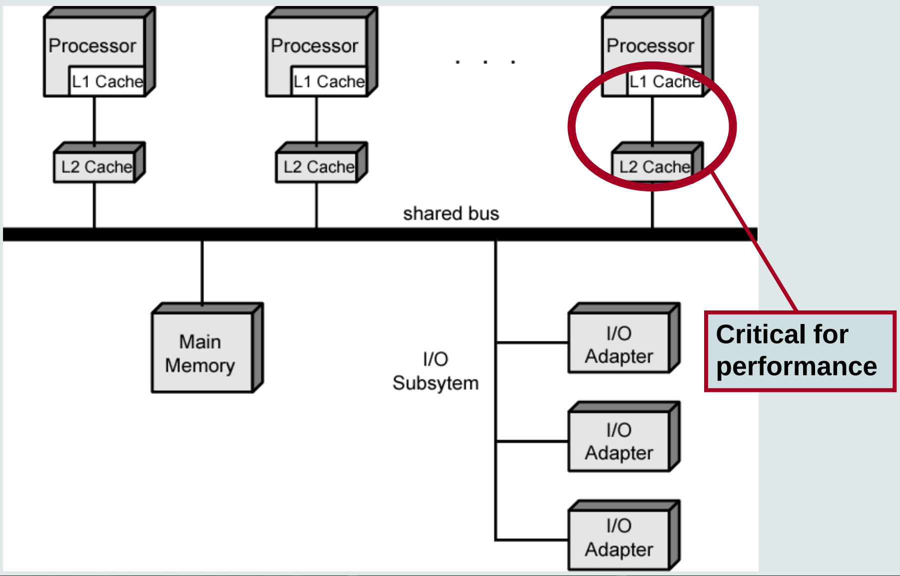
SMP Example



SMP Advantages

- High performance
  - If similar work can be done in parallel (e.g., scientific computing).
- Good availability
  - Since all processors can perform the same functions, failure of a single processor does not stop the system.
- Support incremental growth
  - Users can enhance performance by adding additional processors.
- Scaling
  - Vendors can offer range of products based on different number of processors.

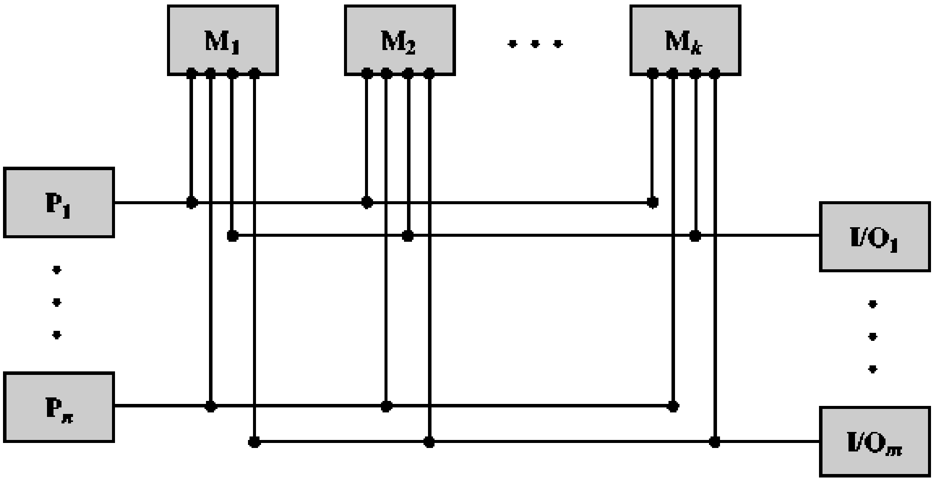
SMP based on Shared Bus



Shared Bus – Pros and Cons

- Advantages:
  - Simplicity.
  - Flexibility — Easy to expand the system by attaching more processors.
- Disadvantages:
  - Performance limited by bus bandwidth.
  - Each processor should have local cache
    - To reduce number of bus accesses
    - Can lead to problems with cache coherence
      - Should be solved in hardware (to be discussed later).

Multi-Port Memory SMP



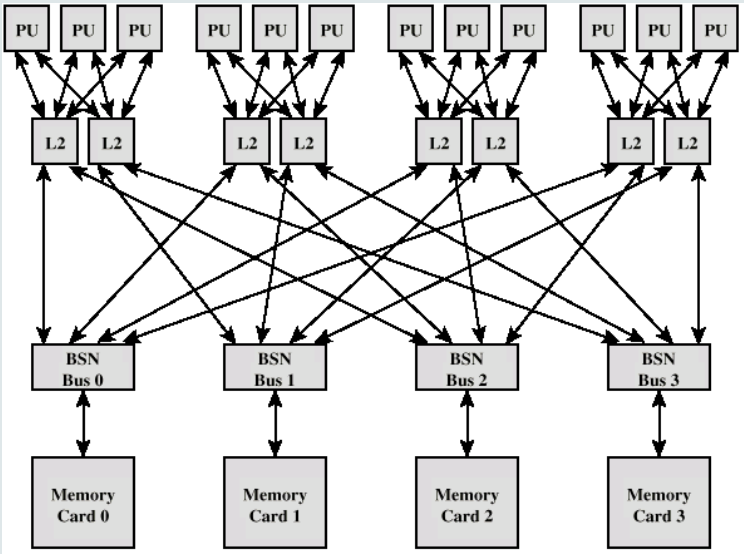
Multi-Port Memory

- Direct access of memory modules by several processors.
- Better performance.
  - Each processor has dedicated path to memory module.
- All the ports can be active simultaneously.
  - With the restriction that only one write can be performed into a memory location at a time.
- Hardware logic required to resolve conflicts.
  - Permanently designated priorities to each memory port.
- Can configure portions of memory as private to one or more processors
  - Increased security.
- Write through cache policy necessary to alert other processors of memory updates.

Operating System Issues

- An SMP OS manages processor resources so that the user perceives a single system.
- It should appear as a single-processor multiprogramming system.
- In both SMP and uniprocessor, multiple processes may be active at one time.
  - OS is responsible for scheduling their execution and allocating resources.
- A user may construct applications that use multiple processes without regard to whether a single processor or multiple processors will be available.
- OS supports reliability and fault tolerance.
  - Graceful degradation.

IBM S/390 Mainframe SMP



S/390 - Key Components

- Processor unit (PU)
  - CISC microprocessor
  - Frequently used instructions hard wired
  - 64k L1 unified cache with 1 cycle access time
- L2 cache
  - 384k
- Bus switching network adapter (BSN)
  - Includes 2M of L3 cache
- Memory card
  - 8G per card

Lecture 9: MIMD Architectures

- Introduction and classification
- Symmetric multiprocessors
- NUMA architecture
- Clusters

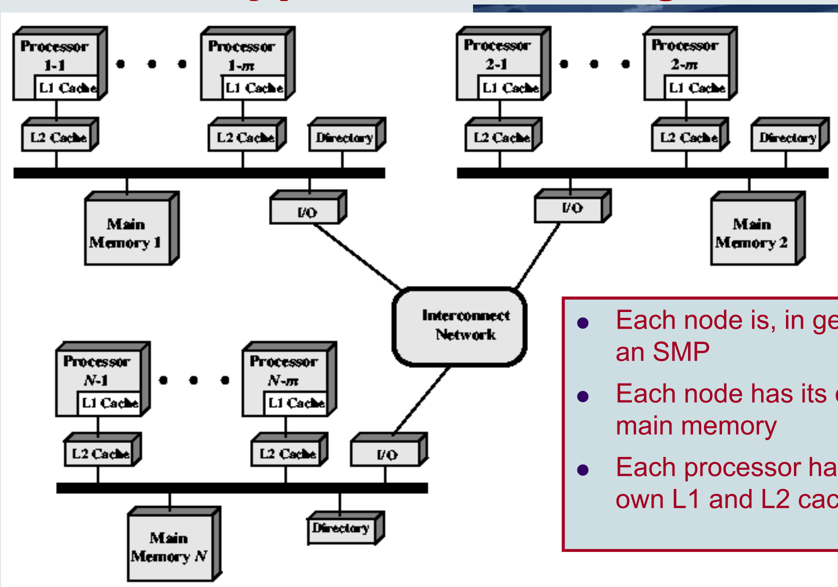
Memory Access Issues

- Uniform memory access (UMA), as in SMP:
  - All processors have access to all parts of memory.
  - Access time to all regions of memory is the same.
  - Access time for different processors is the same.
- Non-uniform memory access (NUMA)
  - All processors have access to all parts of memory.
  - Access time of processor differs depending on region of memory.
  - Different processors access different regions of memory at different speeds.

Motivation for NUMA

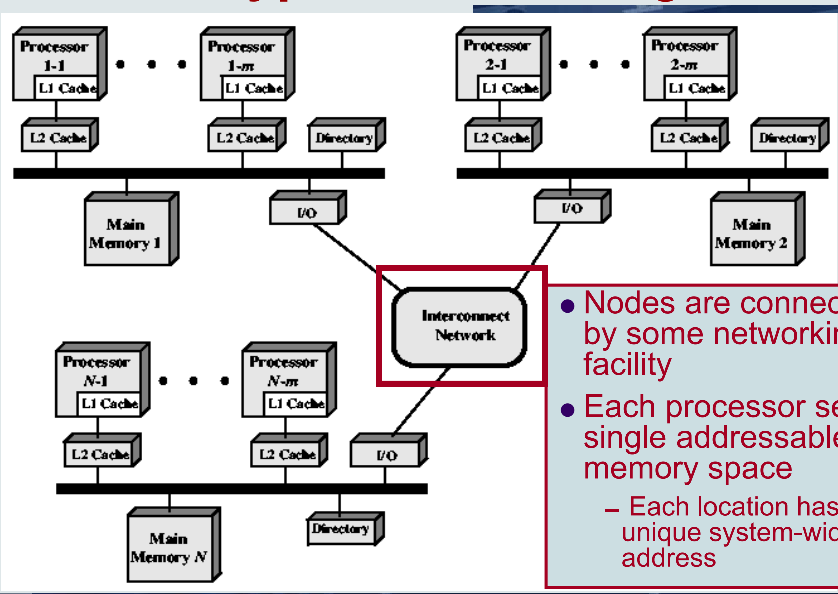
- SMP has practical limit to number of processors.
  - Bus traffic limits to between 16 and 64 processors.
  - A multi-port memory has limited number of inputs (ca. 10).
- In clusters each node has its own memory.
  - Applications do not see large global memory.
  - Coherence maintained by software not hardware (slow speed).
- NUMA retains SMP flavour while making large scale multiprocessing possible.
  - e.g. Silicon Graphics Origin NUMA machine (3000 Series Server) has 1024 MIPS processors (≥ 1 teraFLOPS peak performance!).
- Objective is to provide a transparent system-wide memory while permitting multiprocessor nodes each with its own bus or internal interconnection system.

A Typical NUMA Organization



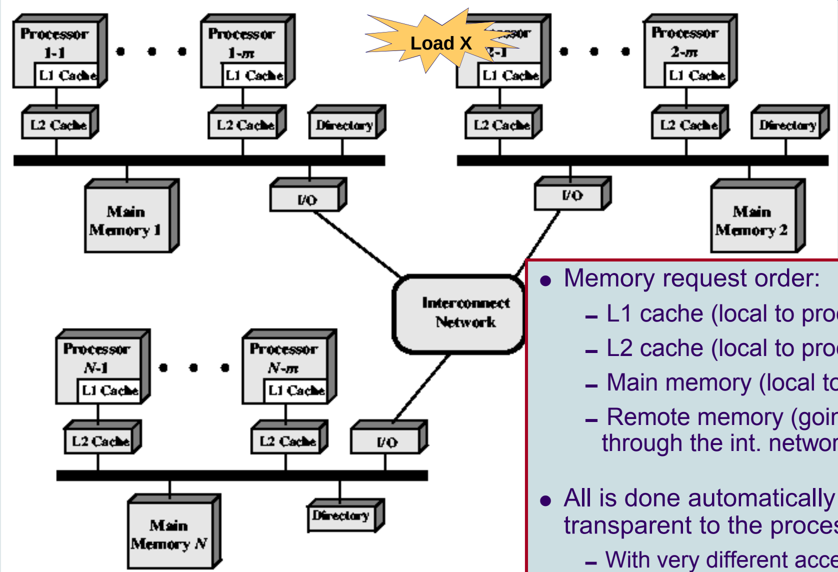
- Each node is, in general, an SMP
- Each node has its own main memory
- Each processor has its own L1 and L2 cache

A Typical NUMA Organization



- Nodes are connected by some networking facility
- Each processor sees a single addressable memory space
  - Each location has a unique system-wide address

A Typical NUMA Organization



- Memory request order:
  - L1 cache (local to processor)
  - L2 cache (local to processor)
  - Main memory (local to node)
  - Remote memory (going through the int. network)
- All is done automatically and transparent to the processor.
  - With very different access time!

NUMA — Pros and Cons

- Effective performance at higher levels of parallelism than SMP.
- However, performance can break down if too much access to remote memory. This can be avoided by:
  - Designing better L1 and L2 caches to reduce memory access.
    - Need also good temporal locality of software.
  - If software has good spatial locality, data needed for an application will reside on a small number of frequently used pages.
    - They can be initially loaded into the local memory.
  - Enhancing VM by including in OS a page migration mechanism to move a VM page to a node that is frequently using it.
- NUMA does not transparently look like a SMP.
  - Software changes needed to move OS and applications from SMP to NUMA.
  - Page allocation, process allocation and load balancing are needed.

Lecture 9: MIMD Architectures

- Introduction and classification
- Symmetric multiprocessors
- NUMA architecture
- Clusters

Loosely Coupled MIMD - Clusters

Cluster: A set of computers connected over a high-bandwidth local area network, and used as a parallel computer.

- A group of interconnected stand-alone computers
- Work together as a unified resource
- Give illusion of being one machine
- Each computer called a node
  - A node can also be a multiprocessor itself, such as an SMP.
- Message passing for communication between nodes

- **NOW** — Network of Workstations, homogeneous cluster.
- **Grid** — Computers connected over a wide area network.

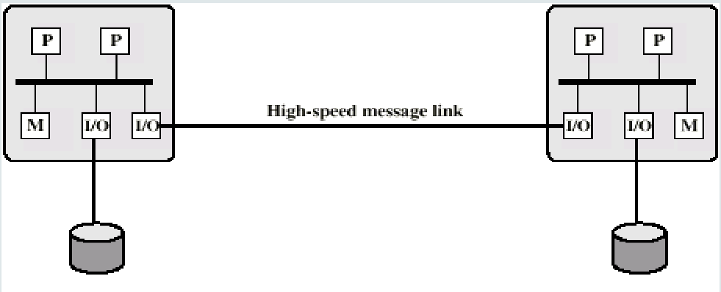
Cluster Benefits

- Absolute scalability
  - Cluster with a very large number of nodes is possible.
- Incremental scalability
  - A user can start with a small system and expand it as needed, without having to go through a major upgrade.
- High availability
  - Fault tolerance by nature.
- Superior price/performance
  - Can be built with cheap commodity nodes.
- Supercomputing-class commodity components are available
- The promise of supercomputing to the average PC User?

Cluster Configurations

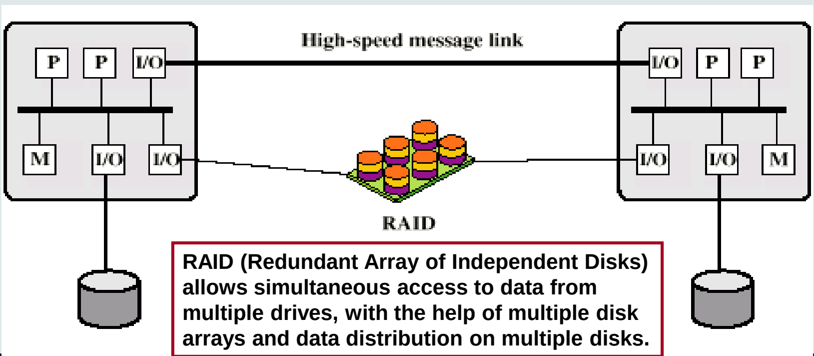
The simplest classification of clusters is based on whether the nodes share access to disks.

- Cluster with no shared disk.
  - Interconnection is by high-speed link
    - LAN — may be shared with other non-cluster computers
    - Dedicated interconnection facility



Cluster Configurations (Cont'd)

- Shared-disk cluster
  - Still connected by a message link
  - Disk subsystem directly linked to multiple computers
  - Disks should be made fault-tolerant
    - To avoid single point of failure in the system.

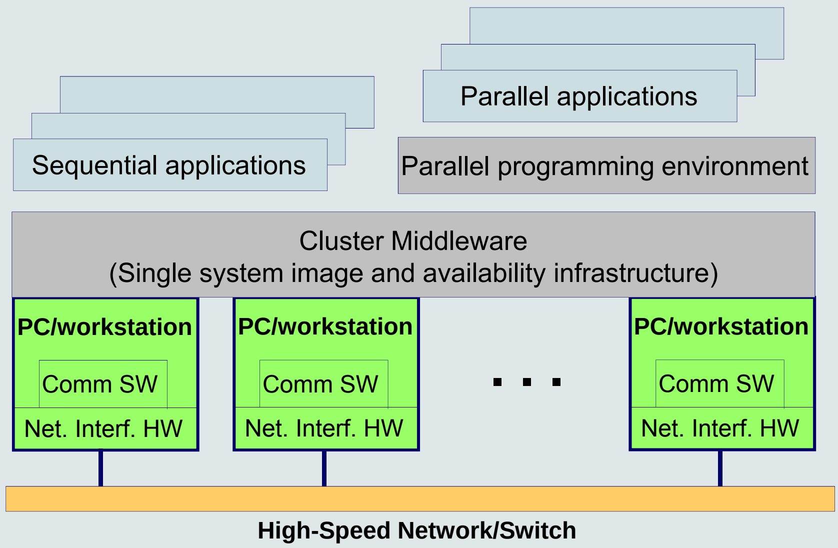


Parallelizing Computation

How to make a single application executing in parallel on a cluster:

- Paralleling complier
  - Determines at compile time which parts can be executed in parallel.
- Parallelized application
  - Application written from scratch to be parallel.
  - Message passing to move data between nodes.
  - Hard to program.
  - Best end results.
- Parametric computing
  - If a problem is repeated execution of algorithm on different sets of data.
    - e.g. simulation using different scenarios.
  - Needs effective tools to organize and run.

Cluster Computer Architecture



Cluster vs. SMP

- Both provide multiprocessor support to high demand applications.
- Both available commercially
  - SMP for longer time
- SMP:
  - Easier to manage and control
  - Closer to single processor systems
    - Scheduling is the main difference
    - Less physical space
    - Lower power consumption
- Clustering:
  - Superior incremental and absolute scalability
  - Superior availability
    - High degree of redundancy

Google Cluster of PCs

- Problem to solve:
  - Search queries
    - Tens of thousands per second from <u>many different users.</u>
  - A typical search query leads to
    - Read hundreds of megabytes of data located <u>in many locations.</u>
    - Tens of billions of CPU cycles
- Solution:
  - Cluster of commodity PCs
    - More than 15,000 nodes.
      - Rather than a smaller number of high-end servers.
  - Parametric computing
    - Same algorithm (PageRank to determine the relative importance of a web page).
    - Different sets of data.

Serving a Google Query

- User enters search query.
- DNS (Domain Name System) lookup:
  - Several clusters available worldwide
    - To handle catastrophic failures (earthquakes, power failures)
  - Selection of cluster
    - Closest (smallest roundtrip time) to the user
- Browser sends HTTP request.
- Load balancer selects a web server.
  - Hardware based
- Query execution.
  - Use inverted index to fine and locate the relevant documents.
- Web server returns result.

Summary

- Two most common MIMD architectures are symmetric multiprocessors and clusters.
  - SMP is an example of tightly coupled system with shared memories.
  - Cluster is a loosely coupled system with distributed memories.
- More recently, non-uniform memory access (NUMA) systems are also becoming popular.
  - More complex in terms of design and operation.
- Several organization styles can be also integrated into a single system.