Stallings 2012 edition
CS 320
Ch. 17 Parallel Processing

Multiple Processor Organization

The author makes the statement: "Processors execute programs by executing machine instructions in a sequence one at a time." He also says that this has never been entirely true. **Microcode or hardwired control do multiple operations in parallel by design and pipelined (even 2-stage) has multiple instructions operating in parallel.**

Flynn's taxonomy for parallel processors is **SISD, SIMD (Vector and array processors), MISD (MISD is not commercially implemented), and MIMD (Symmetric Multiprocessor or SMP or Nonuniform Memory Access Machines, NUMA).**

Symmetric Multiprocessors

A symmetric multiprocessor (SMP) is a system where m**ultiple processors share a single memory or pool of memory. The memory access time to any region of memory is the same for all processors.**

Non-uniform memory access machine or (NUMA) is a system where t**he memory access time to any region of memory is not the same for all processors.**

A computer cluster is a **collection of SMPs which communicate via a fixed path or a network.**

An SMP has five characteristics:
1) Two or more similar processors
2) Processors share memory and I/O and memory access time is about the same for all processors
3) All processors share all I/O devices
4) All processors can perform the same function
5) There is an integrated operating system that provides interaction between processors and programs.

The author lists four advantages of SMPs over uniprocessors:
1**) Performance**
2) Availability
3) Incremental growth
4) Scaling.

Incremental growth means adding one more processor to a system.
Scaling refers to vendors selling complete systems in various sizes.

For an SMP the existence of multiple processors is invisible to the user. This is done b**y the operating system which decides which processor does what.**

The most common form of multiprocessor system for personal computers is t**he time shared bus.**

The advantages and disadvantages of the time shared bus are:

Adv. Simple, flexible, and reliable.
DisAdv. performance.

Performance is a significant problem with the time shared bus system since t**he bus cycle time limits the speed of the system.**

A cache or multiple cache levels can greatly improve performance of time shared busses. **The local systems can cache information and reuse it without accessing the bus.**

There are five key features of a multiprocessor operating system:
Simultaneous concurrent processes
Scheduling
Synchronization
Memory management
Fault tolerance

For simultaneous concurrent processes, OS routines need to be reentrant s**o they can be used by multiple processors at what appears to be the same time.**

The OS must successfully assign waiting processes to available processors in an efficient manner. This is called *Scheduling.*

With shared memory and I/O event ordering and mutual exclusion become important issues. Synchronization is a term to describe this.

Fault tolerance is different on a multiprocessor than on a uniprocessor machine. **Multiprocessors allow a system to degrade if there is a failure while a uniprocessor system simply stops.**

Cache Coherence and MESI
When we discussed cache memory we talked about two write policies. **Write back and Write through. C**ache coherency is a problem for Write back but not write through.

Coherency means that the cache has out of date material with respect to the main memory. In Write through all cache writes are also done to the main memory. In Write back they are not.

One way to deal with cache coherency is called the "software compile time" approach. The c**ompiler performs an analysis of the code and marks those data items that may be unsafe for caching. This data is marked uncacheable.**

The disadvantage of the compile time approach is that the c**ompiler must be conservative so the process becomes inefficient.**

The advantages of the software solution to the cache coherency problem is that o**verhead is transferred from run time to compile time and design complexity is transferred from hardware to software.**

Hardware schemes for cache coherency can be divided into two approaches: Directory protocols and Snoopy protocols:

For directory protocols a local directory is kept for each line in the cache. Processes using cache lines must report such to the central directory and get permission to use that line.

In Snoopy protocols each local cache controller recognizes when its data is being shared. Announcements of shared data are broadcast over the network of cache controllers. To write to a line a process must first announce that over the network and get permission.

A hardware protocol for cache coherence is called MESI: **Modified, Exclusive, Shared, Invalid.**

For MESI e**ach line in the cache has a tag with a two bit code which specifies Modified (line different from main memory, Exclusive (line not present in other caches and coherent) Shared (line may be in other caches but is coherent), Invalid (line no longer has valid data).**

The figure below shows a MESI state diagram.

A) Figure **A is the state diagram in response to operations originating in the local processor. Figure B is the state diagram in response to operations originating from the bus.**

B) Notice that read misses (RME and RMS) only originate from the invalid state? **In the modified, shared, and exclusive state the data has to be in memory so you can't have a miss.**

C) Note that if you are in the modified state and you get a read hit (RH) you stay in the modified state. The data is modified but t**he cache is OK but the memory is out of date.**

D) Note the difference in what happens if there is a write hit in the shared state and the exclusive state. **In the exclusive state the processor owns the data so it simply writes to it and marks it as modified. In the shared state the processor must first gain control of the line by signaling its intention on the bus. It can then do the write and mark the data as modified.**

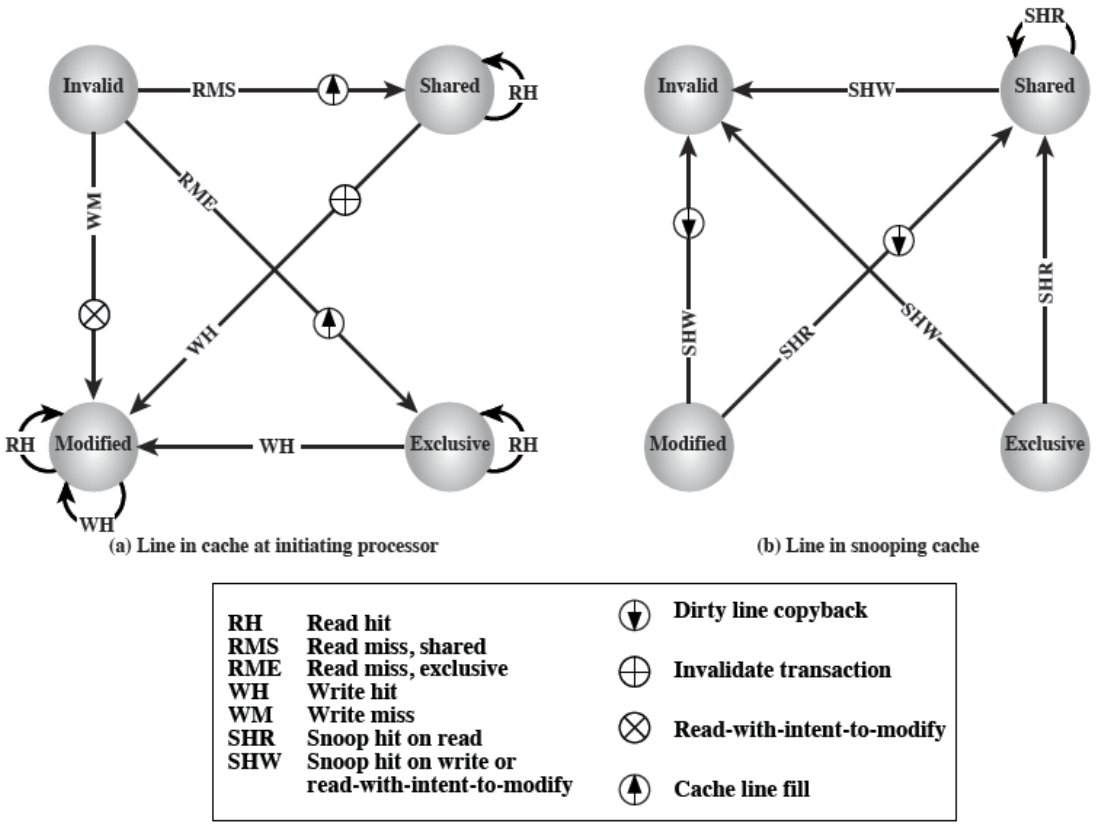

Figure 17.6 MESI State Transition Diagram

Multithreading and chip multiprocessors

MIPS rate = f_{clock} x instructions/cycle A measure of performance.

In multithreading the i**nstruction stream is divided into multiple smaller instruction streams known as threads.**

Stallings 2012 edition

With regard to multithreaded systems we define:

A) Process: **An instance of a program running on a computer.**
B) Process switch: **switching from one running process to another.**
C) Thread: **A Dispatchable unit of work within a process. It executes sequentially and is interruptible.**
D) Thread switch: **Switching processor control from one thread to another.**

Thread switches and process switches are similar but a thread switch is generally faster. **Threads share the same resources and processes do not.**

Explicit multithreading: **A user level thread defined by software**
Implicit multithreading: Typically kernel level.

A processor capable of multithreading must provide a separate program counter for each thread of execution to be executed concurrently. **Using a single PC is not feasible since the processes are truly concurrent.**

There are four principal approaches to multithreading:
Interleaved or fine-grained
blocked or coarse grained
simultaneous (SMT)
chip multiprocessing.

In interleaved each thread is given one cycle and the processor switches back and forth between the two.

In blocked a thread is executed until it becomes blocked at which time a switch is made.

In SMT threads are simultaneously issued to different pipes on a superscalar processor.

In chip the entire processor is replicated in hardware.

Clusters
A "cluster computer" is a g**roup of interconnected, whole computers working together as a unified computing resource that can create the illusion of being one machine.**

Beowulf Cluster
A Beowulf Cluster is a collection of parallel desktop level computers usually connected by Ethernet with an operating system that allows the machines to appear as a single machine. It has been called a *poor man's supercomputer.* The Beowulf cluster originated at NASA around 1994. Several different operating systems have been used but Linux is used by most clusters.

There are four benefits to cluster computers*:*
***A*bsolute scalability**
Incremental scalability
High availability
Superior price/performance.

Absolute scalability refers to creating a new system bigger or smaller than previous systems.
Incremental scalability refers to adding a new computer to an existing cluster.

There are 5 clustering methods.

Passive standby
Active secondary
Separate servers
Servers connected to disks
Server shared disks.

Passive standby: **A secondary machine takes over if the primary machine stops producing a "heartbeat". Passive standby does no increase performance.**

Active secondary: **Similar to passive standby except that the second machine is used for processing task and is not just idle. A**ctive secondary i**ncreases performance and reduces cost per unit of processing.**

Separate servers: **Each computer is a separate server with its own disk and memory. Hardware or software is used to schedule the servers for efficient client usage.**

Servers connected to disk: **Similar to separate servers but in addition there is a common disk.**

Server shared disks: **Multiple servers share partitions in a common disk.**

Operating System Design Issues:
For clusters there are two general methods of dealing with failures.
Highly available clusters
Fault-tolerant clusters.

"Highly available clusters": **Each node is an independent computer and the operating system recognizes a failed machine and works around it by using others.**
"Fault-tolerant" clusters: **One that uses redundancy such as RAID and other techniques to reduce the chance of a system failure.**

One of the functions of an OS in a cluster system is that of failure management. In this regard, we define Failover and Failback
Failover is the process of transversing data and resources from a failed system to an alternative system.
Failback is the reverse after the failed system has been fixed.

Another function of the OS in a cluster system is load balancing. **Load balancing is an attempt to make all of the computers in the cluster equal in terms of their work load.**

Cluster computers are often used to run a single application. When this is done the OS may be responsible for finding ways to run such an application in parallel on several computers. The author lists three approaches to this problem: Parallelizing compiler, Parallelized applications, and Parametric computing.

Parallelizing compiler: **The compiler figures out at compile time which parts of an application can be done in parallel.**

Parallelized application: **The programmer writes the application from the outset in modules which can be run in parallel.**

What is parametric computing: **This is used for applications that have an algorithm that must be run a large number of times while gathering statistics for results.**

Figure 17.10 shows a typical cluster computer architecture.

Middleware p**rovides a unified system image to the user rather than a set of independent computers.**

Since machines communicate over this line and only two machines can talk at any one time, high speed is essential or machines end up waiting on communications.

The middleware occasionally does check pointing. **This saves the state of the machine to allow rollback and recovery should there be a failure.**

The middleware and communications network make it possible for any node to access any other node such as a disk or I/O device without knowing where it is.

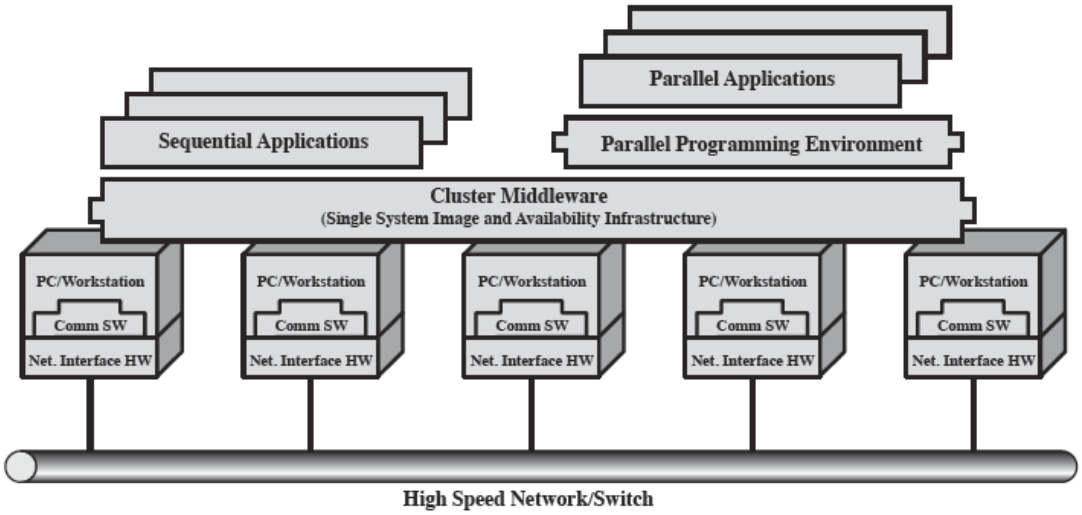

Figure 17.10 Cluster Computer Architecture [BUYY99a]

Advantages and disadvantages of clusters over SMPs?

SMP is easier to manage.

Clusters can provide superior performance and are superior with regard to scalability and availability.

Stallings 2012 edition

NonUniform Memory Access

CC-NUMA: **Cache-Coherent NUMA. Cache coherency is maintained among the caches of the various processors in the NUMA.**

SMPs are ultimately limited by the speed of the bus. With a cluster each node has its own private memory and does not see a large global memory. NUMA is an attempt to fix both of these problems.

Figure 17.12 shows a typical NUMA architecture.

The node is the basic building block of the NUMA. **Each node is effectively an SMP with its own L1 and L2 cache and main memory.**

Each processor sees only a single system wide memory. The local main memory in each node has its own system address. Thus all memory is in one address space.

Each node has a directory associated with it. **Keeps track of information with regard to cache coherency.**

Cache changes are broadcast and recorded in node directories for cache coherency.

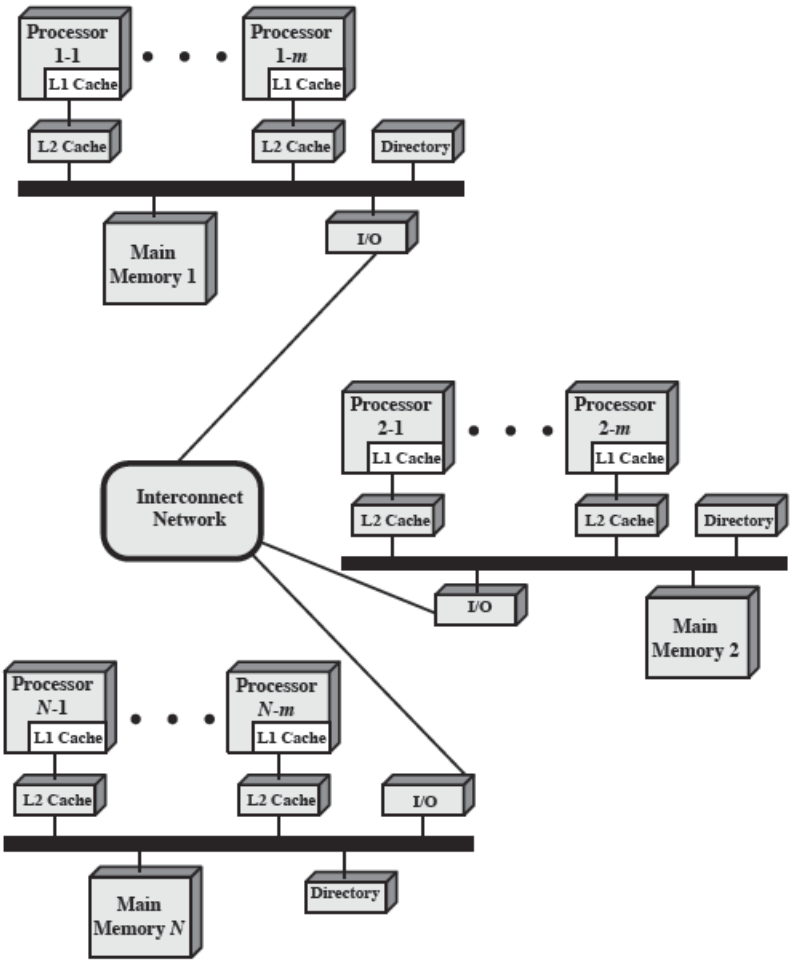

Figure 17.12 CC-NUMA Organization

The advantages of NUMA: **Good performance at a higher level of parallelism than an SMP without requiring major software changes.**

Stallings 2012 edition

Vector Computation:

Vector computers work on problems that are generally classified as *continuous-field simulation* problems. These might u**se finite element analysis to simulate and solve problems. Weather forecasting is an example.**

Vector processors operate on vectors where desktops operate on one unit at a time.

Vector processors have one unit that operates on vectors whereas parallel processors have multiple units each of which does some part of a task.

Parallel processors use two primitive operators called *Fork* and *Join*.
Fork splits a process into two branches which can be done in parallel.
Join is the inverse.

In parallel processors the control and register units are duplicated. In Parallel ALU's only the ALU is duplicated.

Cloud Computing

Cloud computing is a **model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.**

Cloud computing has five essential characteristics:
Broad network access
Rapid elasticity
Measured service
On-demand self-service
Resource pooling.

Rapid elasticity is the a**bility to expand and reduce resources according to your specific service requirements.**

Resource pooling means that the p**rovider's resources are pooled to serve many clients.**

There are three Cloud Computing Service models:
Software as a Service
Platform as a Service
Infrastructure as a Service.

Infrastructure as a Service p**rovides access to underlying cloud infrastructure such as disc storage or execution on supercomputers.**

Cloud Computing has four deployment models:
Public cloud
Private Cloud
Community Cloud
Hybrid Cloud.

Stallings 2012 edition

Public cloud is available to the general public where Community cloud may be available only to those in one company.

Hybrid cloud is a c**omposition of two or more clouds – say a private cloud and a public cloud**