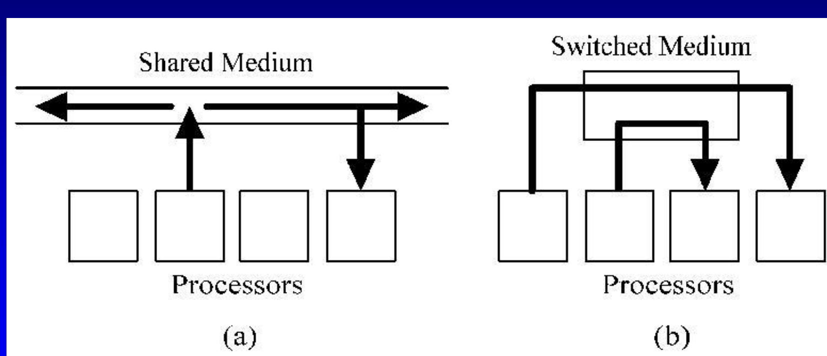
Interconnection Networks

- Using interconnection networks we can
  - Connect processors to shared memory
  - Connect processors to each other
- Interconnection media types
  - Shared medium
  - Switched medium

Shared versus Switched Media



Shared Medium

- Allows only message at a time
- Messages are broadcast
- Each processor "listens" to every message
- Collisions require resending of messages
- Ethernet is an example

Switched Medium

- Supports point-to-point messages between pairs of processors
- Each processor has its own path to switch
- Advantages over shared media
  - Allows multiple messages to be sent simultaneously
  - Allows scaling of network to accommodate increase in processors

Switch Network Topologies

- View switched network as a **graph**
  - Vertices = processors **or switches**
  - Edges = communication paths
- Two kinds of topologies
  - Direct
  - Indirect

Direct Topology

- Ratio of switch nodes to processor nodes is 1:1
- Every switch node is connected to
  - 1 processor node
  - At least 1 other switch node

Indirect Topology

- Ratio of switch nodes to processor nodes is greater than 1:1
- Some switches simply connect other switches

Processor Arrays Multiprocessors and Multicomputers

- Criteria to understand effectiveness in implementing efficient parallel algorithms on real architecture are:

1. Diameter: It is the largest distance between two nodes in the network. Low diameter is better as it puts a lower bound on the complexity of parallel algorithms.

2. Bisection width of the network: It is the minimum number of edges that must be removed in order to divide the network into two halves. High bisection width is better. Data set/Bisection width puts a lower bound on the complexity of parallel algorithms.

3. Number of edges per node: It is better if the number of edges per node is a constant independent of the network size. Processor organization scale well with a organization having more processors.

4. Maximum edge length: For better scalability, it is best if the nodes and edges are laid out in 3-D space so that the maximum edge length is constant independent of the network size.

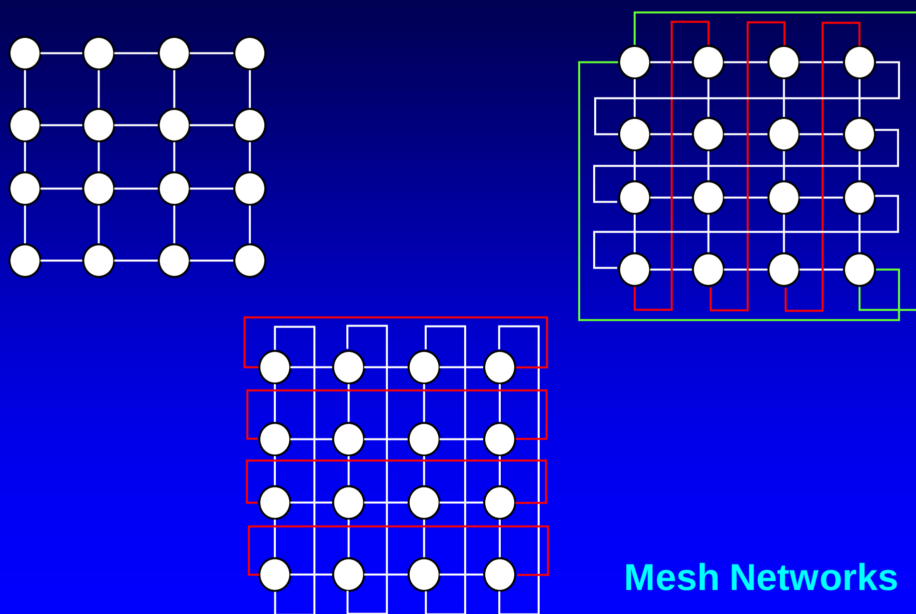
Processor Organizations:

Mesh Network: (1 of 8)

1. q-D lattice
2. Communication is allowed only between neighboring nodes
3. May allow wrap around connections
4. Diameter of a q-D mesh with k^q nodes is q(k-1) (Difficult to get polylogarithmic time algorithm)

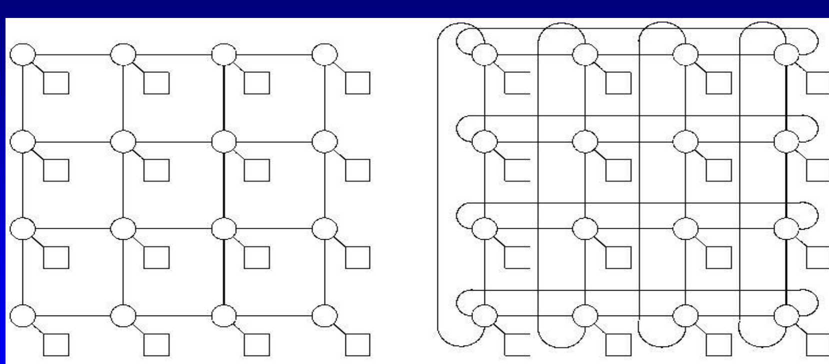
5. Bisection width of a q-D mesh with k^q nodes is k^{q-1}
6. Maximum edges per nodes is 2q
7. Maximum edge length is a constant

Ex. MarPar's MP-1, Intel's Paragon XP/S



Mesh Networks

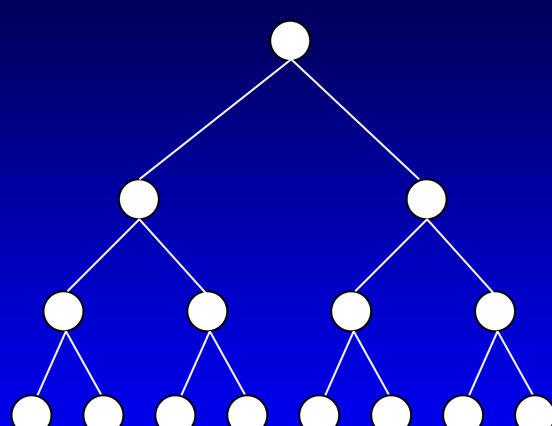
2-D Meshes



(a) (b)

Binary tree: (2 of 8)

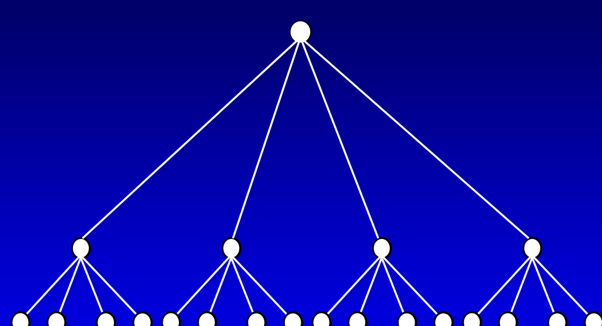
1. 2^k-1 nodes are arranged into a complete binary tree of depth k.
2. A node has at most 3 links
3. Low diameter of 2(k-1)
4. Poor bisection width

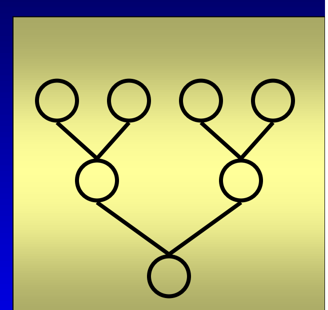


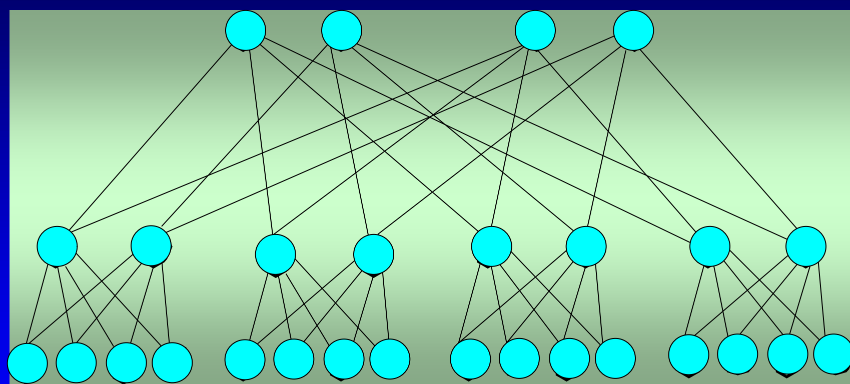
Tree Network

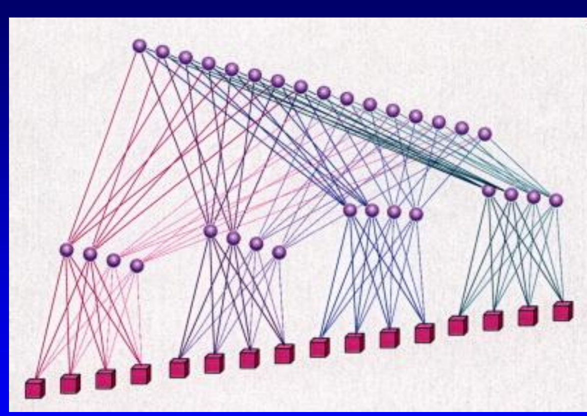
Hypertree Network: (Ex. data routine net of CM-5) (3 of 8)

1. Low diameter of binary tree with Improved bisection width
2. A 4-ary hypertree with depth d has 4^d leaves and $2^d(2^{d+1}-1)$ nodes
3. Diameter is 2d and bisection width is 2^{d+1}
4. No. of edges per node is never more than 6
5. Maximum edge length is an increasing function of the problem size.



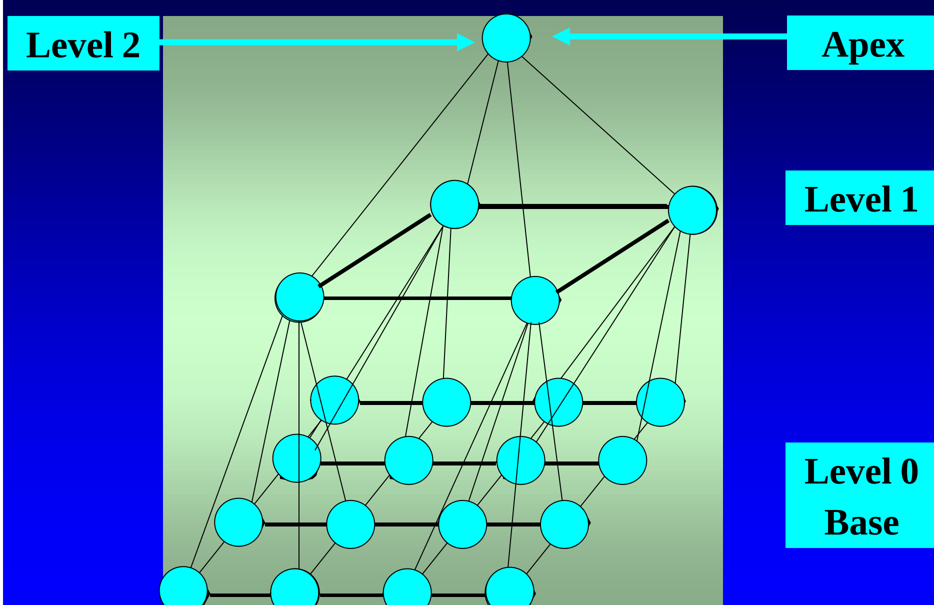






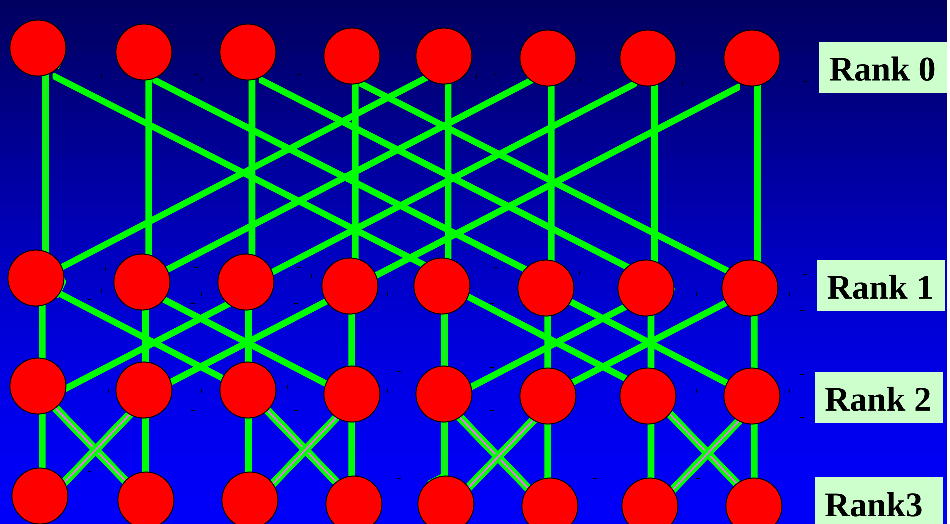
Pyramid Network: (4 of 8)

1. Mesh Network + Tree Network
2. Network of size k^2 is a complete 4-ary rooted tree of height $\log_2 k$
3. Total no. of processors of size k^2 is $(4/3)k^2-(1/3)$
4. Level of the base is 0, apex of the pyramid has level $\log_2 k$.
5. Every interior processor is connected to 9 other processors
6. Pyramid reduces the diameter, 2 log k
7. Bisection width is 2k

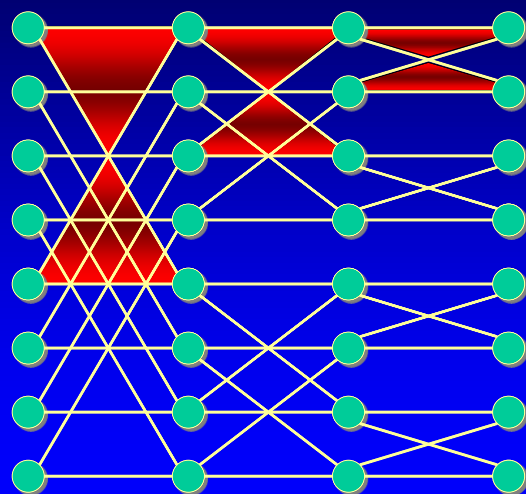


Butterfly Network: (Ex. BBN TC2000) (5 of 8)

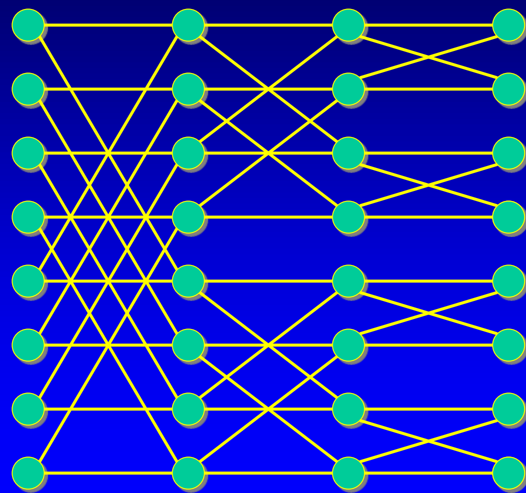
1. It consist of $(k+1)2^k$ nodes divided into k+1 rows or **ranks**
2. Each row contains $\mathbf{2}^k$ nodes
3. If node(i,j) denotes j^{th} node on i^{th} rank $0 \le i \le k$ and, $0 \le j < n$ then node(i,j) on rank i>0 is connected to two nodes on rank i-1, nodes (i-1,j) and (i-1,m), where m is the integer found by inverting the i^{th} ***msb*** in binary representation of j.
4. Diameter of the net is 2k
5. Bisection width is 2^{k-1}



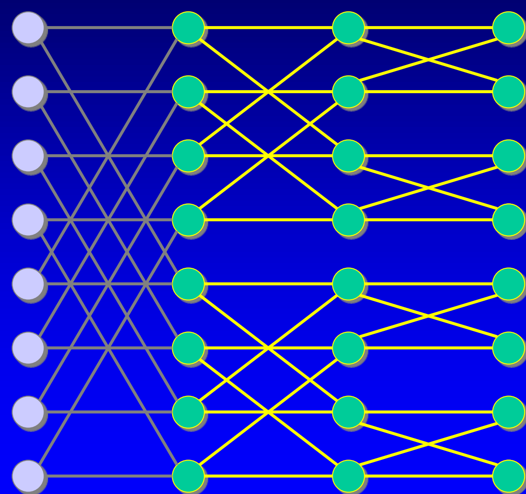
Butterflies



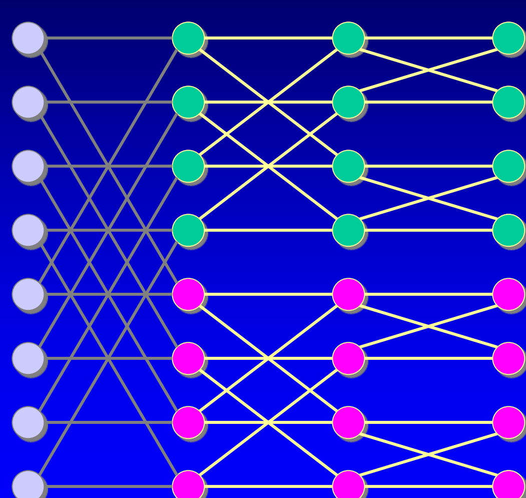
Decomposing a Butterfly



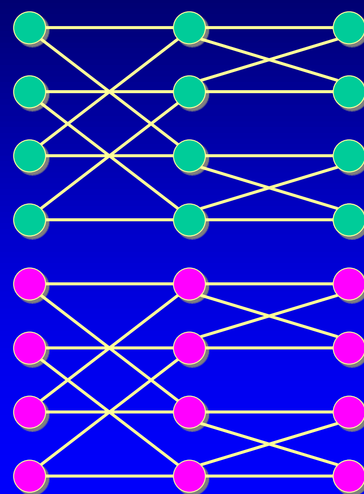
Decomposing a Butterfly



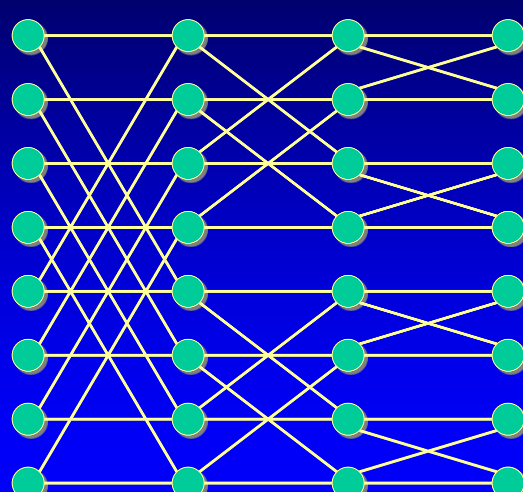
Decomposing a Butterfly



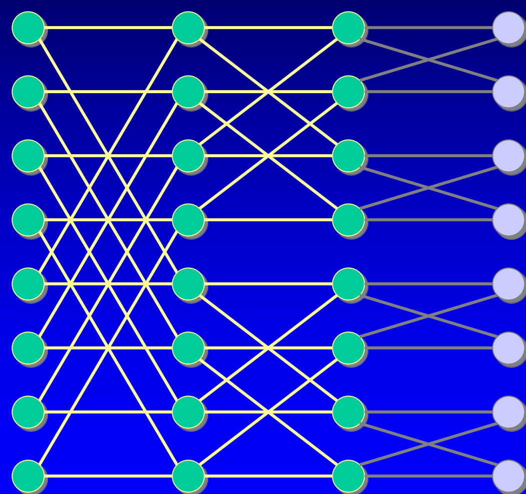
Decomposing a Butterfly



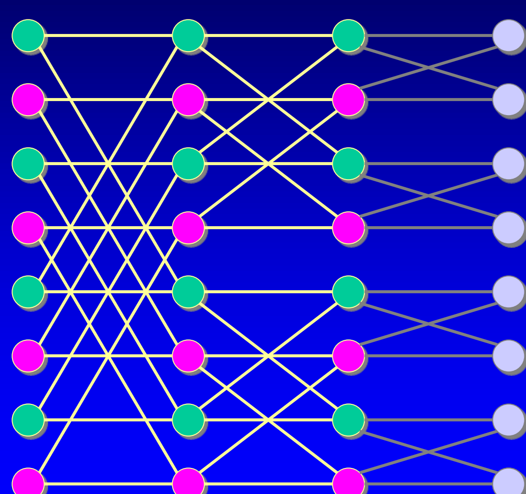
Decomposing a Butterfly II



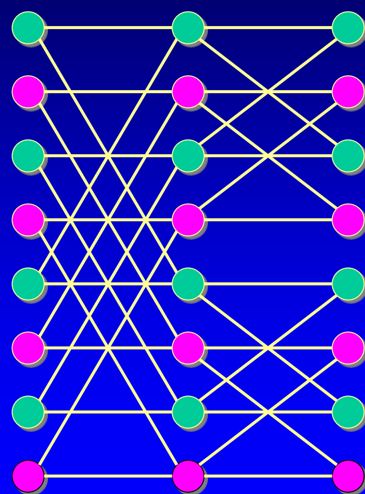
Decomposing a Butterfly II



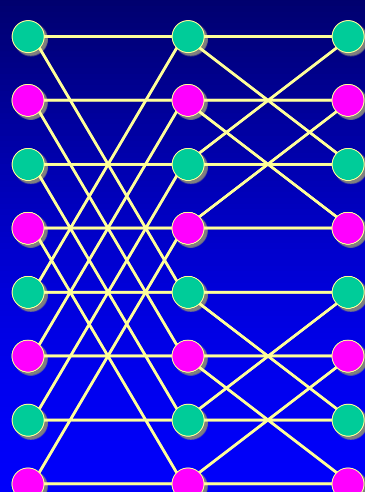
Decomposing a Butterfly II



Decomposing a Butterfly II



Decomposing a Butterfly II



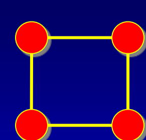
Hypercube (Cube Connected) Networks: (6 of 8)

1. 2^k nodes form a k-D network
2. Node addresses 0, 1, …, 2^k-1
3. Diameter with 2^k nodes is k
4. Bisection width is 2^{k-1}
5. Low diameter and high bisection width
6. Node *i* connected to *k* nodes whose addresses differ from *i* in exactly one bit position
7. No. of edges per node is k-the logarithmic of the no. of nodes in the network **(Ex. CM-200)**

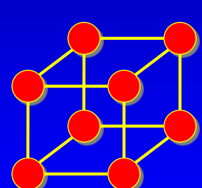
Hypercube

$k = 0$
$N = 1$ (2^k)

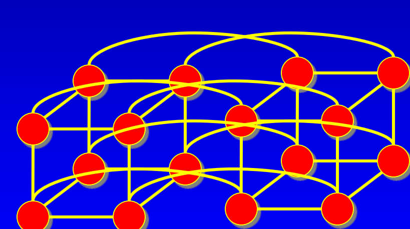
$k = 1$
$N = 2$



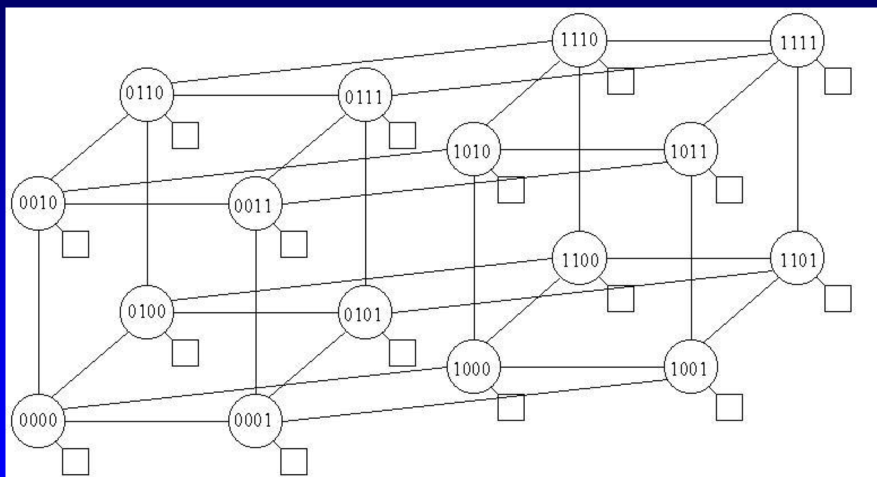
$k = 2$
$N = 4$



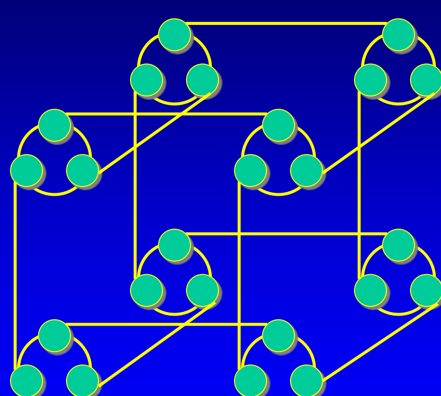
$k = 3$
$N = 8$



$k = 4$
$N = 16$



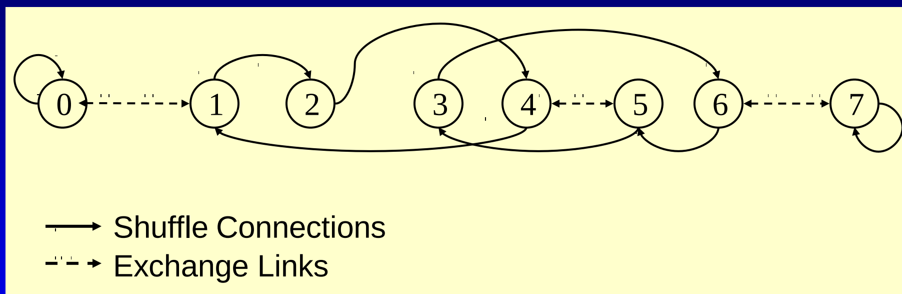
Cube-Connected Cycles



Shuffle Exchange Network: (7 of 8)

1. Consist of $n = 2^k$ nodes numbered 0,...,n-1 having two kind of connections called ***shuffle*** and ***exchange***.
2. Exchange connections link pairs of nodes whose numbers differ in their last significant bit.
3. Shuffle connection link node i with node 2i mod (n-1), with the exception that node n-1 is connected to itself.

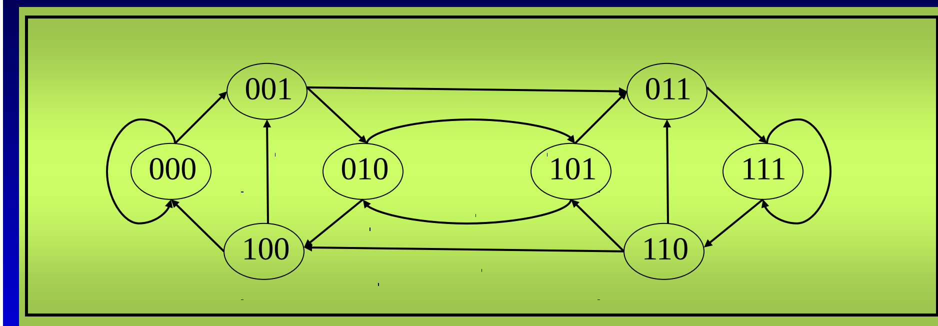
4. Let $a_{k-1}a_{k-2}...a_0$ be the address of a node in a perfect shuffle network, expressed in binary. A datum at this address will be at address $a_{k-2}...a_0a_{k-1}$.
5. Length of the longest link increases as a function of network size.
6. Diameter of the network with 2^k nodes is 2k-1
7. Bisection width is $2^{k-1}/k$



de Bruijn network: (8 of 8)

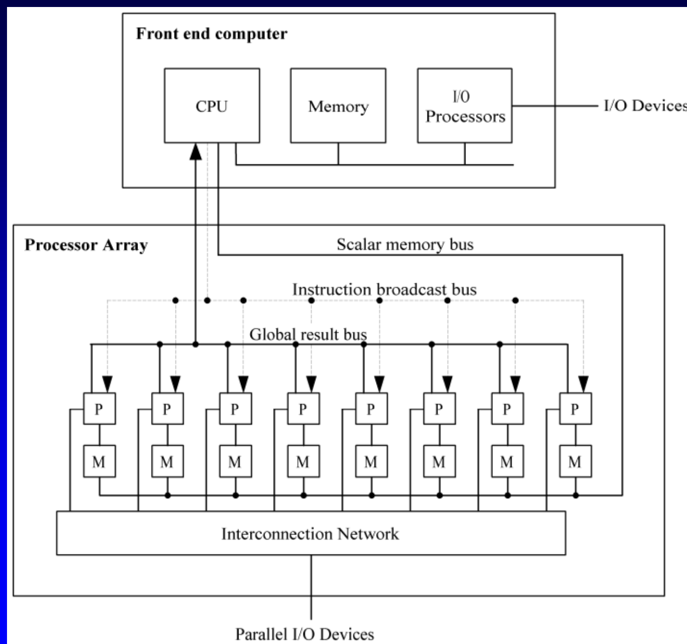
1. Let n = 2^k nodes and $a_{k-1}a_{k-2}\ldots a_0$ be the addresses
2. Two nodes reachable via directed edges are

   $a_{k-2}a_{k-3}\ldots a_0 0$ and $a_{k-2}a_{k-3}\ldots a_0 1$
3. The number of edges per node are constant independent of the network size.
4. Bisection width with 2^k nodes is $2^k/k$
5. Diameter is k



Processor Arrays

- It is a vector computer implemented as a sequential computer
- connected to a set of identical synchronized processing elements
- capable of performing the same operation on different data
- sequential computers are known as ***Front Ends***.



Parallel Computing (Intro-03): Rajeev Wankar 47

Processor Array Shortcomings

- Not all problems are data-parallel
- Speed drops for conditionally executed code
- Don’t adapt to multiple users well
- Do not scale down well to “starter” system
  - (Cost of the high bandwidth communication networks is more if fewer processor)
- Rely on custom VLSI for processors
  - (Others are using semiconductor technology)
- Expense of control units has dropped

Parallel Computing (Intro-03): Rajeev Wankar 48

Multiprocessors

- Multiple-CPU computers consist of a number of fully programmable processors, each capable of executing its own program
- Multiprocessors are multiple CPU computers with a shared memory.

- Based on the amount of time a processor takes to access local or global memory, shared address-space computers are classified into two categories.
- If the time taken by a processor to access any memory word is identical, the computer is classified as ***uniform memory access (UMA)*** computer

- If the time taken to access a remote memory bank is longer than the time to access a local one, the computer is called a ***nonuniform memory access (NUMA)*** computer.

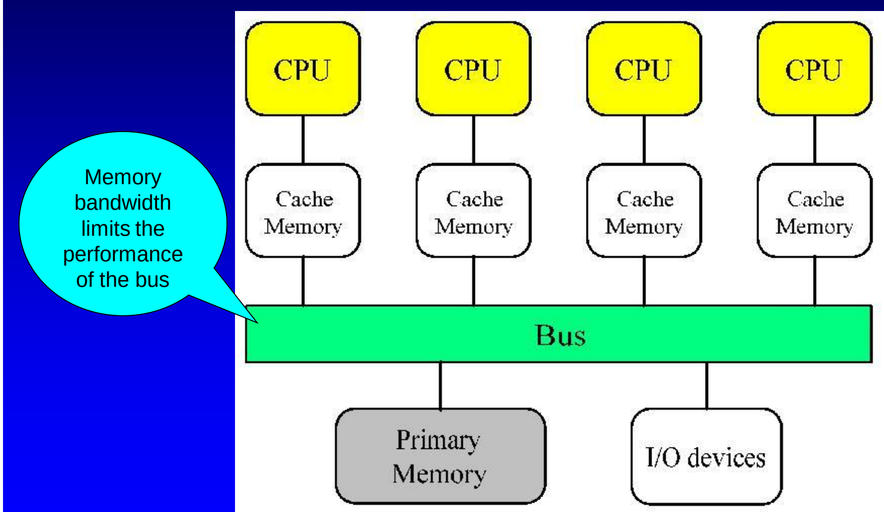
UMA

- Central switching mechanism to reach shared centralized memory
- Switching mechanisms are Common bus, crossbar switch and packet switch net

Centralized Multiprocessor

- Straightforward extension of uniprocessor
- Add CPUs to **bus**
- All processors share same primary memory
- Memory access time same for all CPUs
  - Uniform memory access (UMA) multiprocessor
  - Symmetrical multiprocessor (SMP)

Centralized Multiprocessor



Private and Shared Data

- Private data: items used only by a single processor
- Shared data: values used by multiple processors
- *In a multiprocessor, processors communicate via shared data values*

Problems Associated with Shared Data

- Cache coherence
  - Replicating data across multiple caches reduces contention
  - How to ensure different processors have same value for same address?
  - Snooping/Snarfing protocol
    - (Each CPU's ***cache controller monitor*** snoops bus/addressline)
    - Write invalidate protocol (processor sending an invalidation signal over the bus)
    - Write update protocol (processor broadcasts a new data without issuing the invalidation signal)
- Processor Synchronization
  - Mutual exclusion
  - Barrier

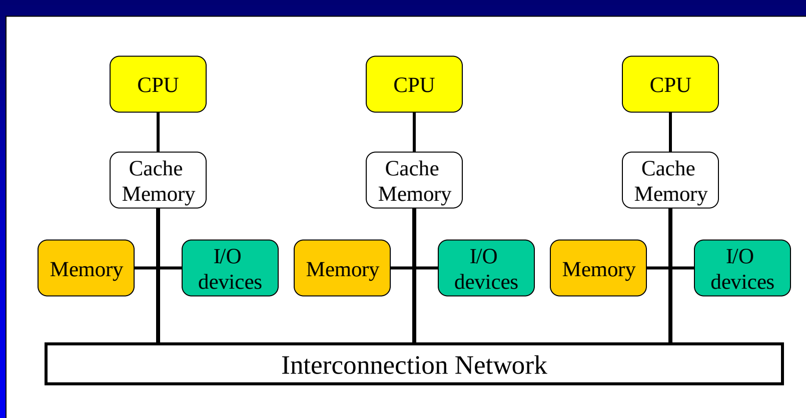
- **NUMA Multiprocessors**

- Memory is distributed, every processor has some nearby memory, and the shared address space on a NUMA multiprocessor is formed by combining these memories

Distributed Multiprocessor

- Distribute primary memory among processors
- Possibility to distribute instruction and data among memory unit so the memory reference is local to the processor
- Increase aggregate memory bandwidth and lower average memory access time
- Allow greater number of processors
- Also called non-uniform memory access (NUMA) multiprocessor

Distributed Multiprocessor



Cache Coherence

- Some NUMA multiprocessors do not have cache coherence support in hardware
  - Only instructions, private data in cache
  - Large memory access time variance
- Implementation more difficult
  - No shared memory bus to “snoop”
    - Snooping methods does not scale well
  - ***Directory-based protocol needed***

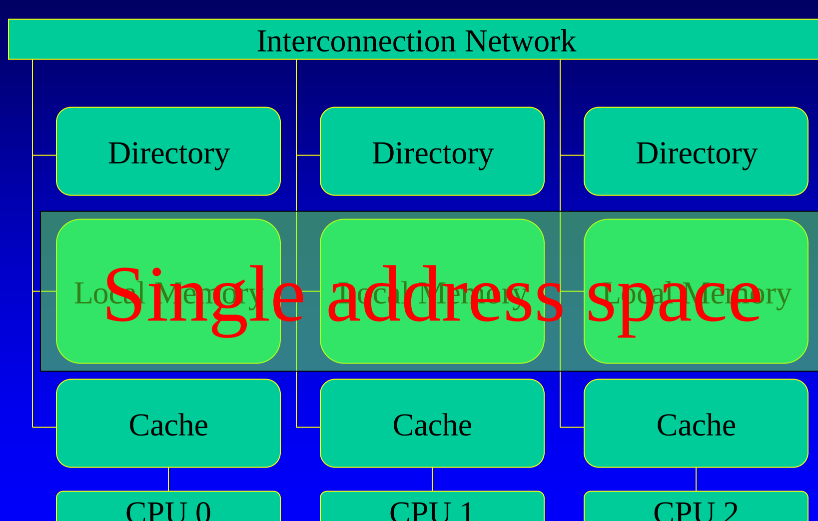
Directory-based Protocol

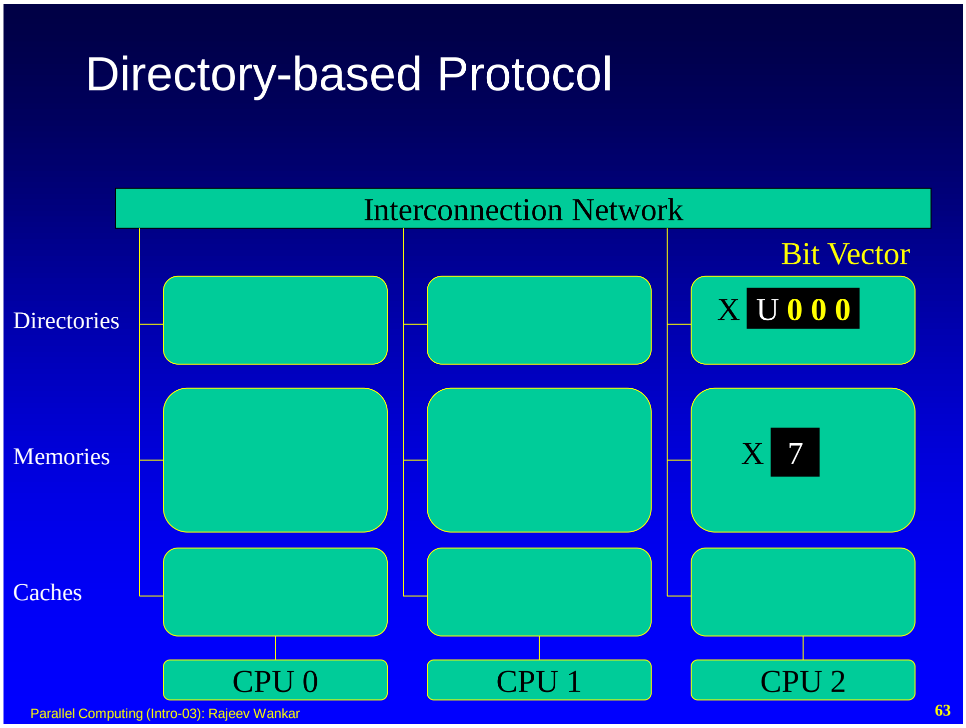
- Distributed directory contains information about cacheable memory blocks
- One directory entry for each cache block
- Each entry has
  - Sharing status
  - Which processors have copies

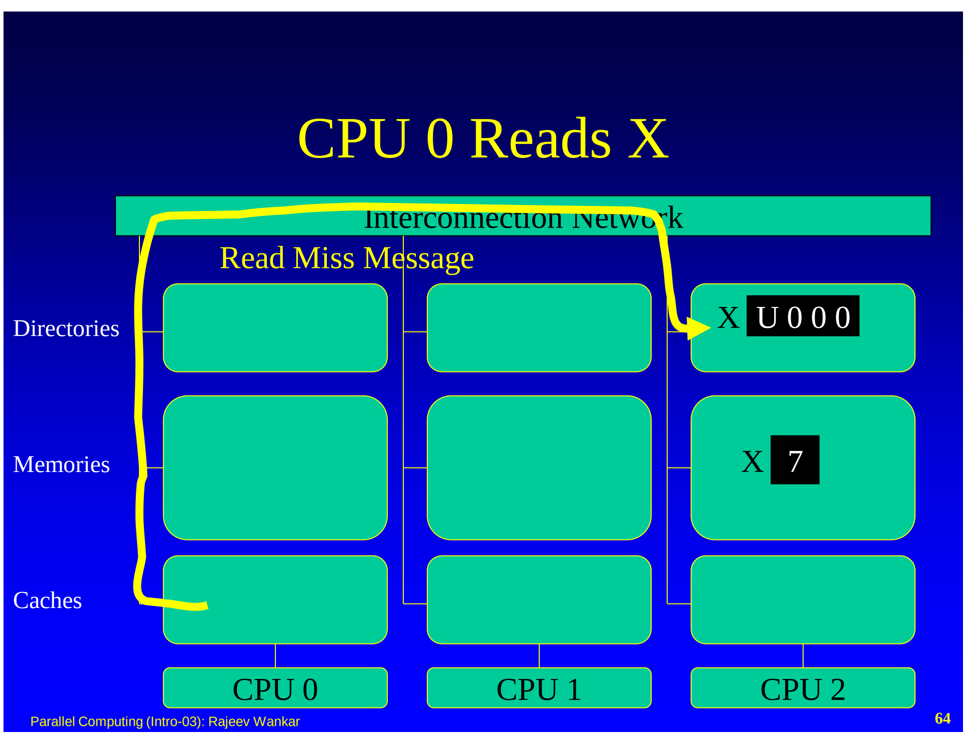
Sharing Status

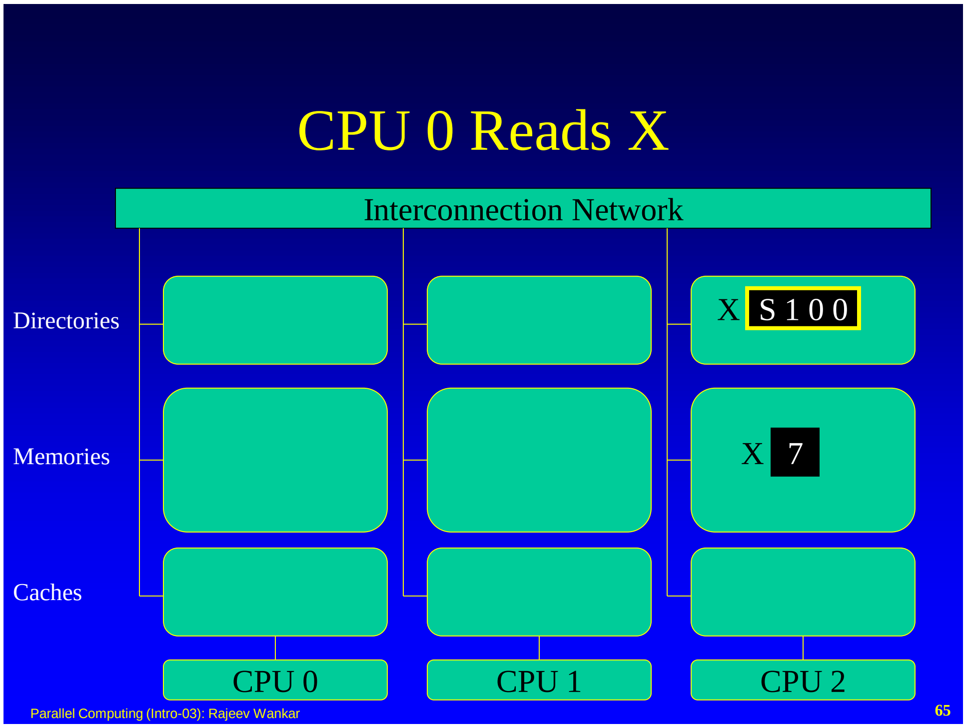
- Uncached
  - Block not in any processor's cache
- Shared
  - Cached by one or more processors
  - Read only
- Exclusive
  - Cached by exactly one processor
  - Processor has written block
  - Copy **in memory** is obsolete

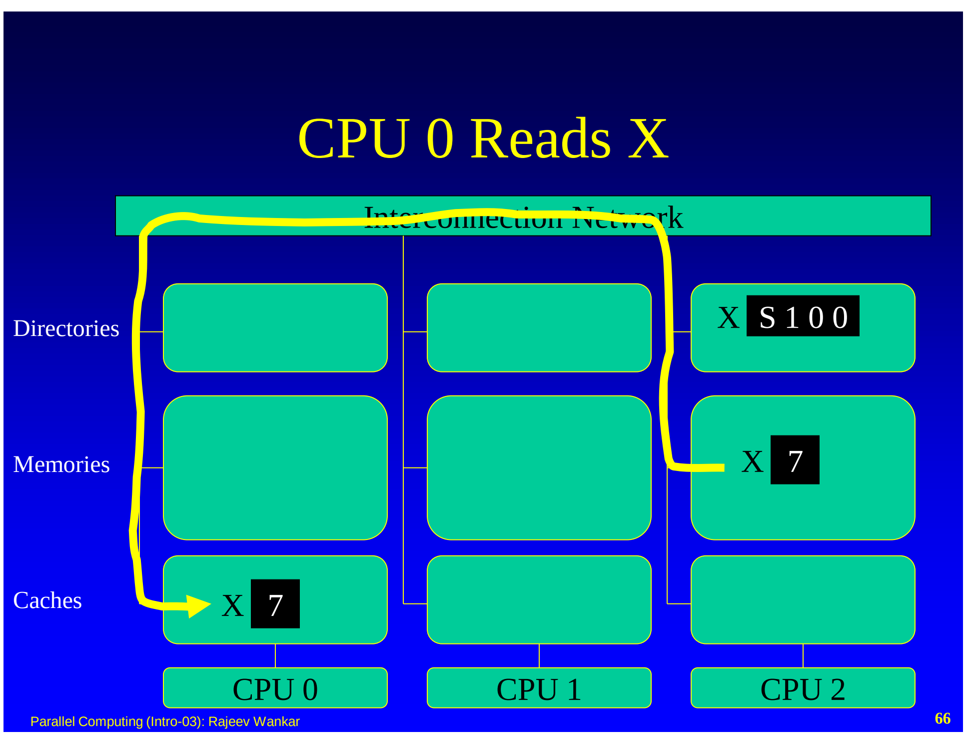
Directory-based Protocol

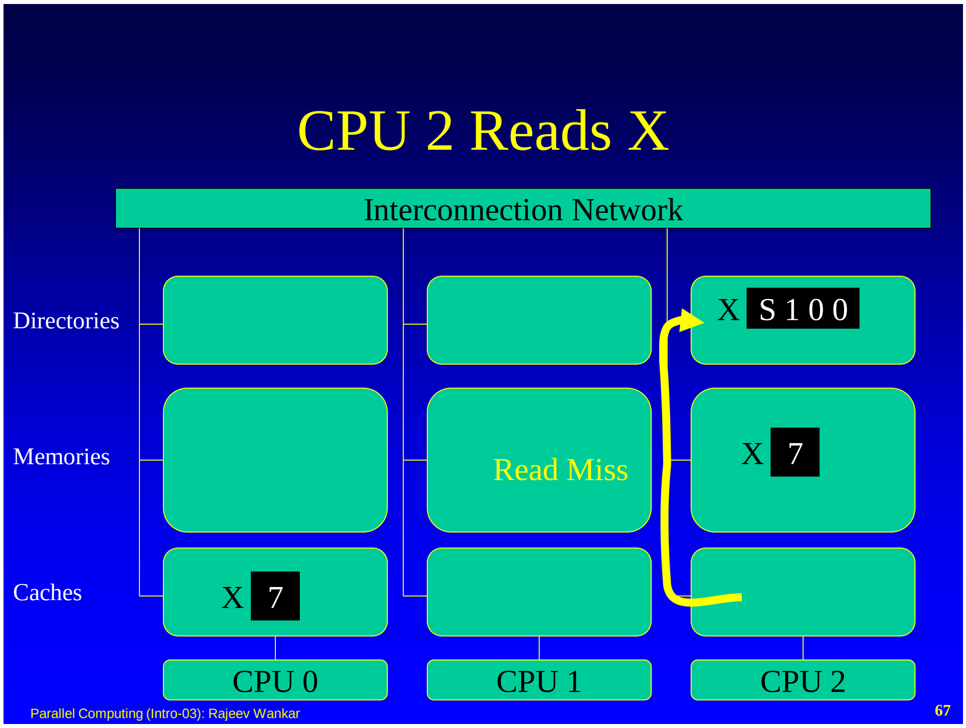


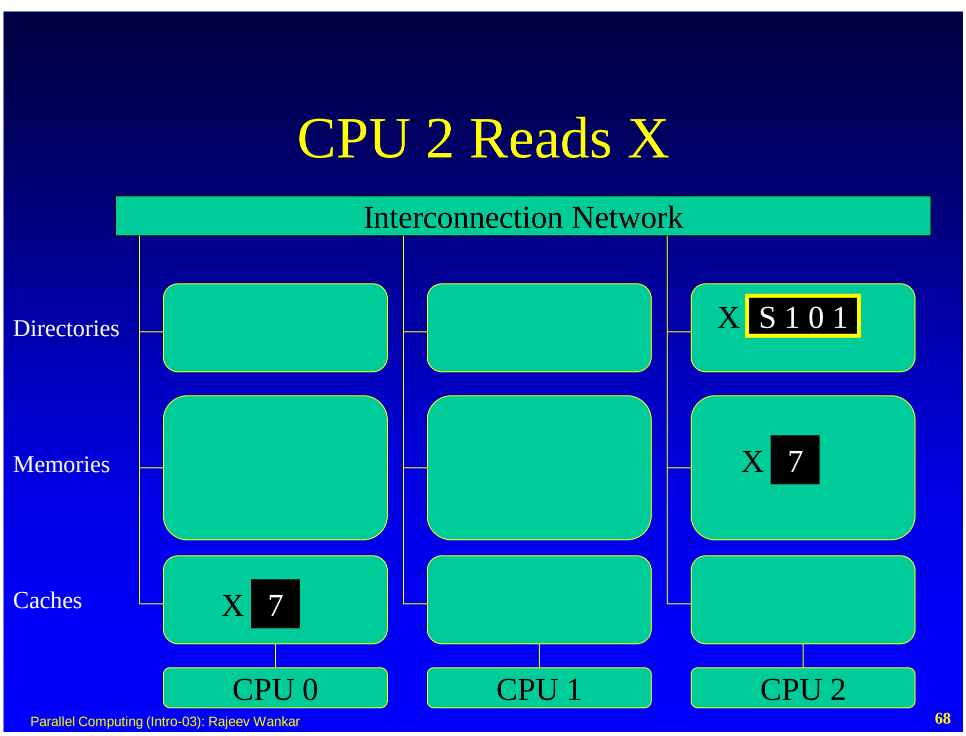

Directory-based Protocol
Interconnection Network
Bit Vector
Directories
X U 0 0 0
Memories
X 7
Caches
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
63

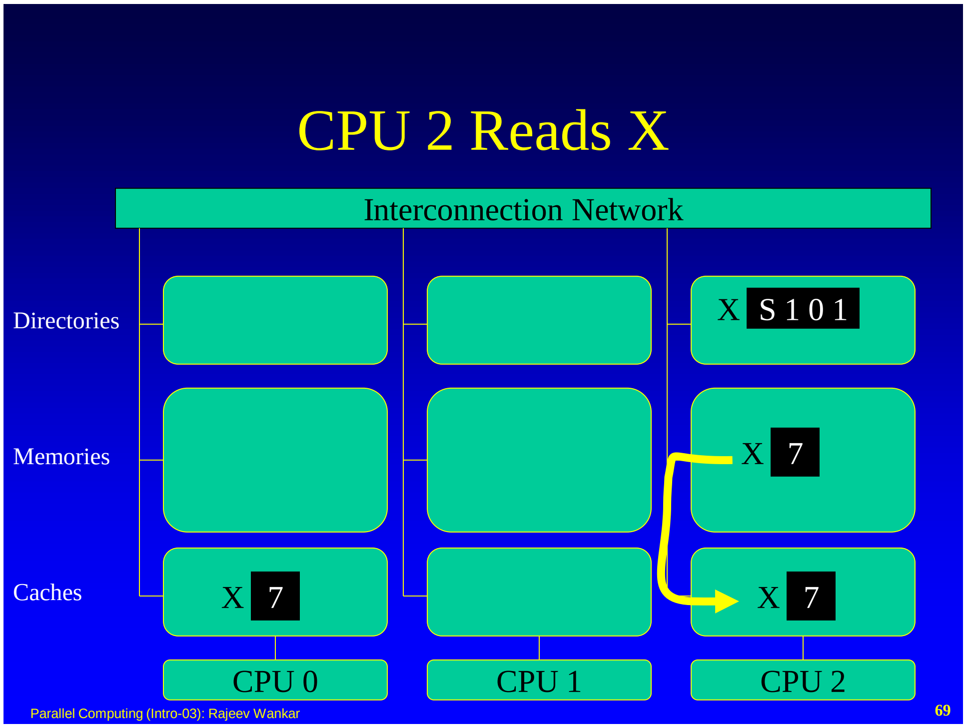

CPU 0 Reads X
Interconnection Network
Read Miss Message
Directories
X U 0 0 0
Memories
X 7
Caches
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
64

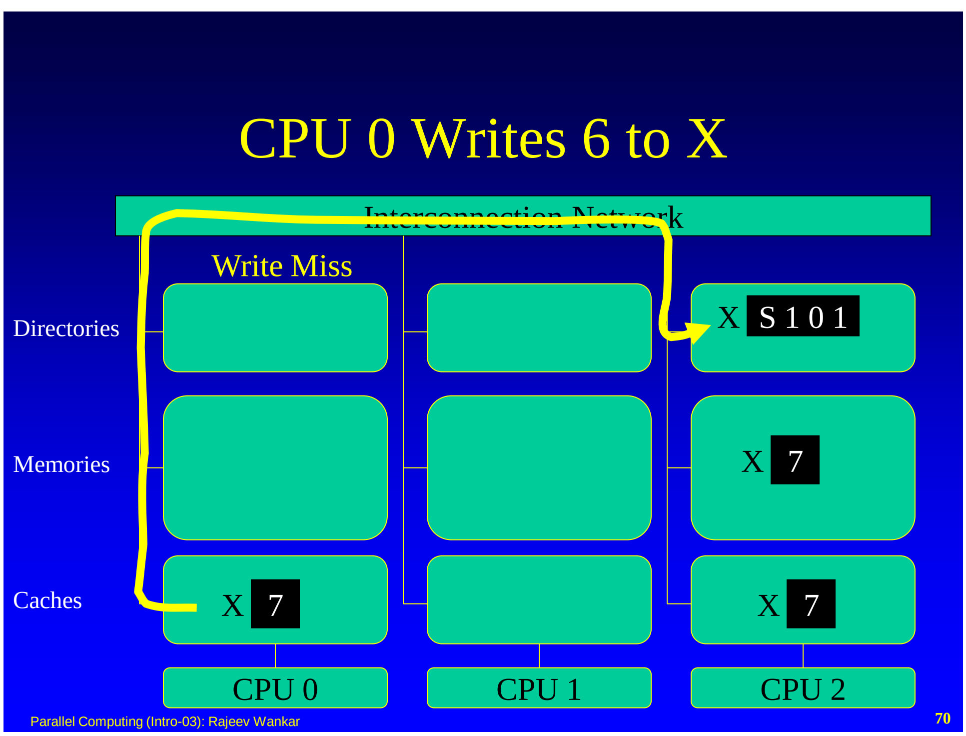

CPU 0 Reads X
Interconnection Network
Directories
Memories
Caches
X
S 1 0 0
X
7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
65

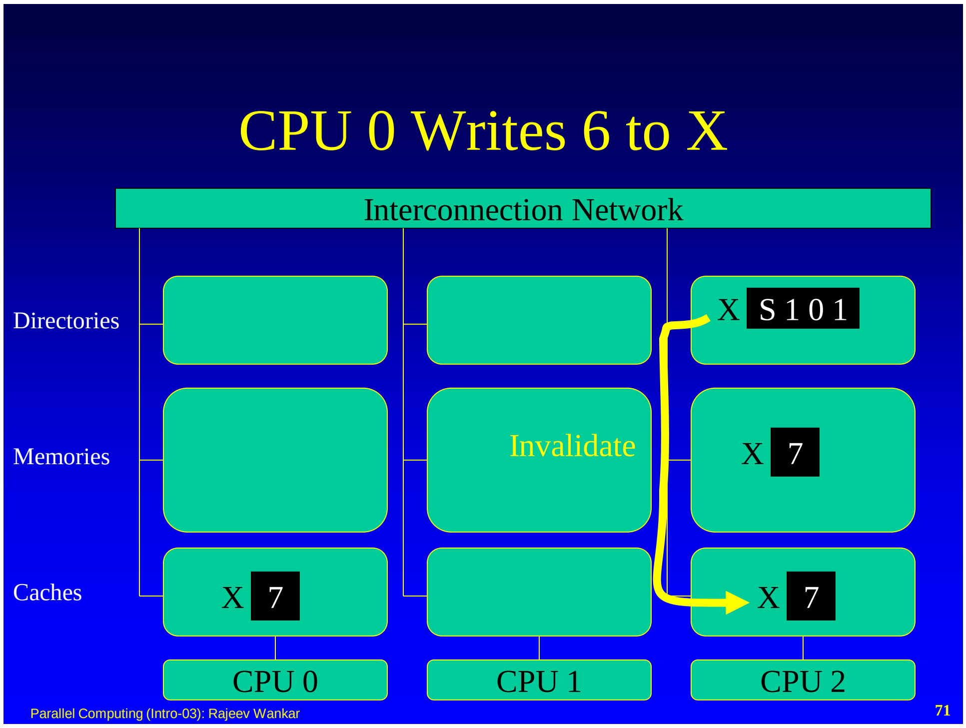

CPU 0 Reads X
Interconnection Network
Directories
Memories
Caches
X
S 1 0 0
X
7
X
7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
66

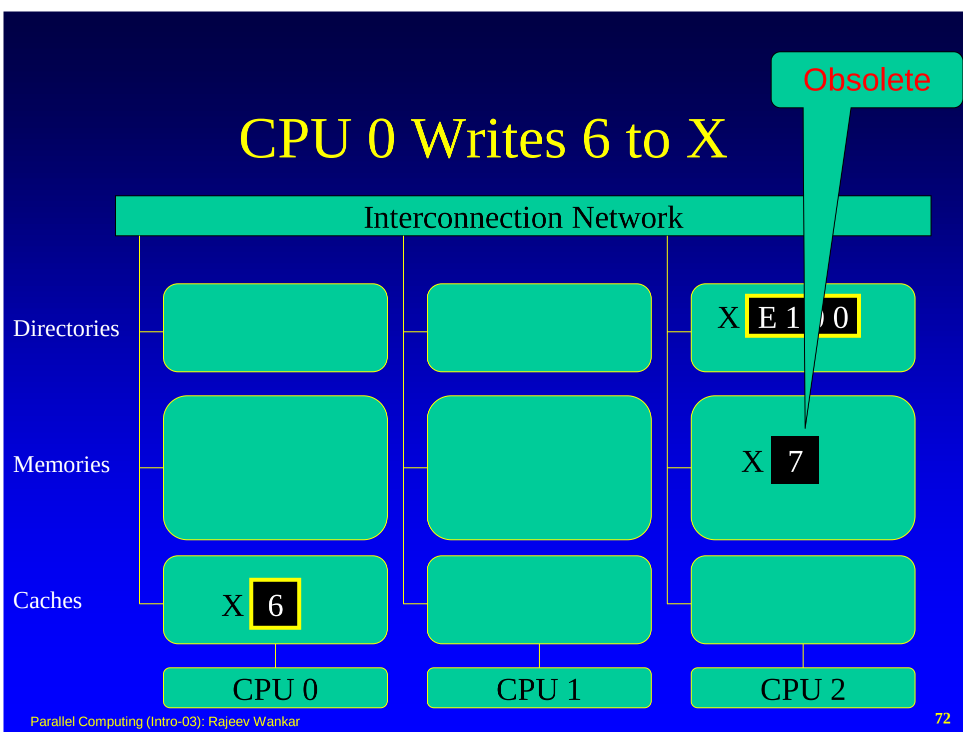

CPU 2 Reads X
Interconnection Network
Directories
Memories
Caches
X
S 1 0 0
Read Miss
X
7
X
7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
67

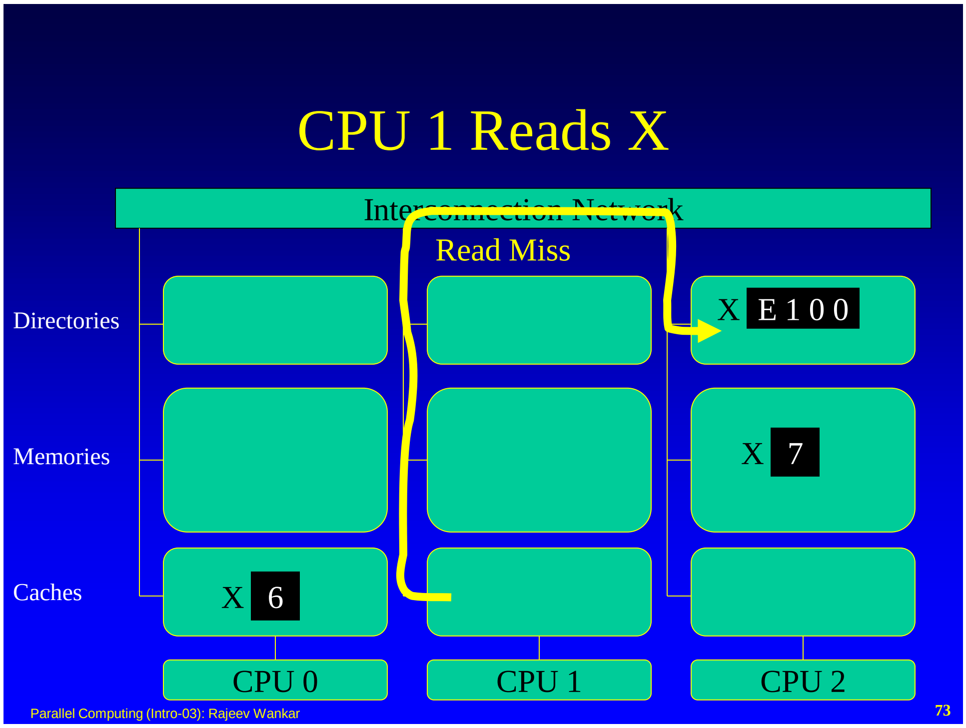

CPU 2 Reads X
Interconnection Network
Directories
Memories
Caches
X
S 1 0 1
X
7
X
7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
68

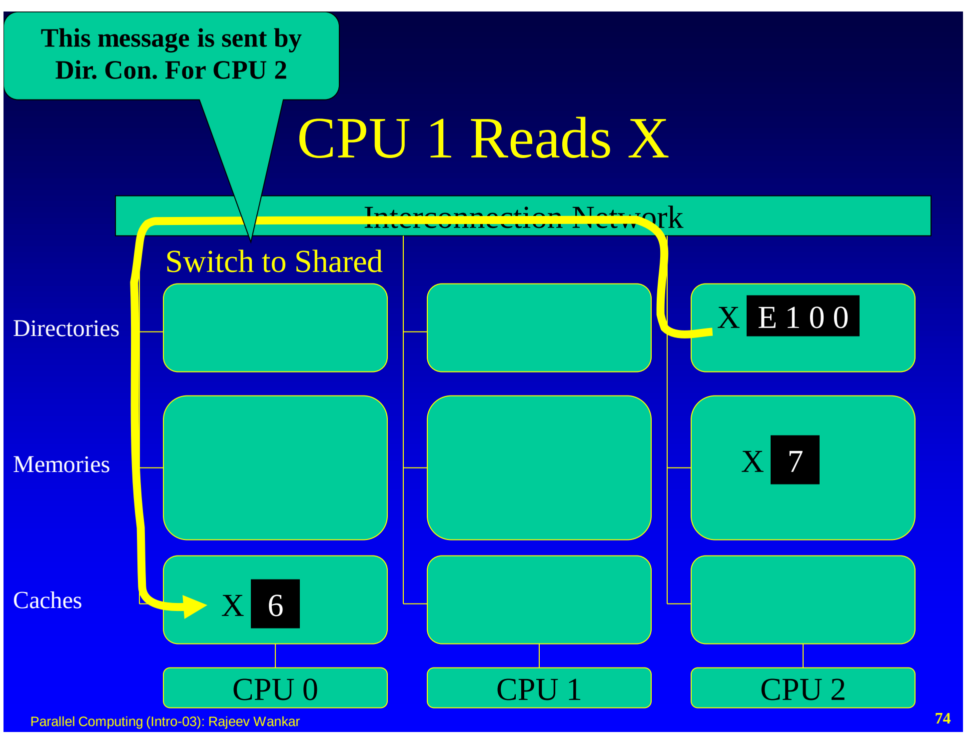

CPU 2 Reads X
Interconnection Network
Directories
Memories
Caches
X S 1 0 1
X 7
X 7
X 7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
69

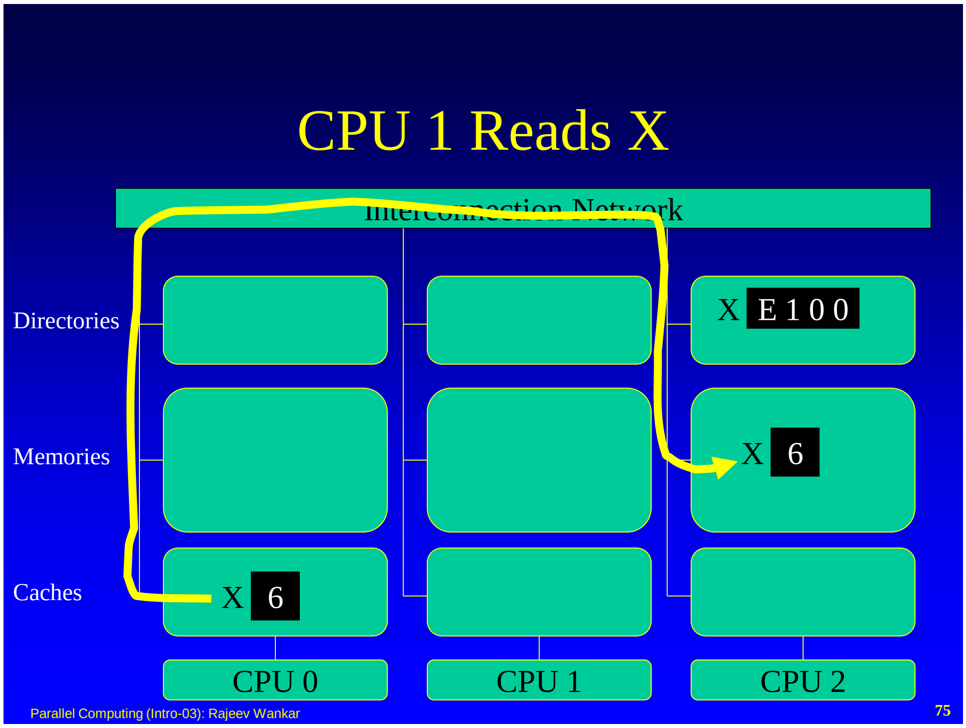

CPU 0 Writes 6 to X
Interconnection Network
Write Miss
Directories
Memories
Caches
X S 1 0 1
X 7
X 7
X 7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
70

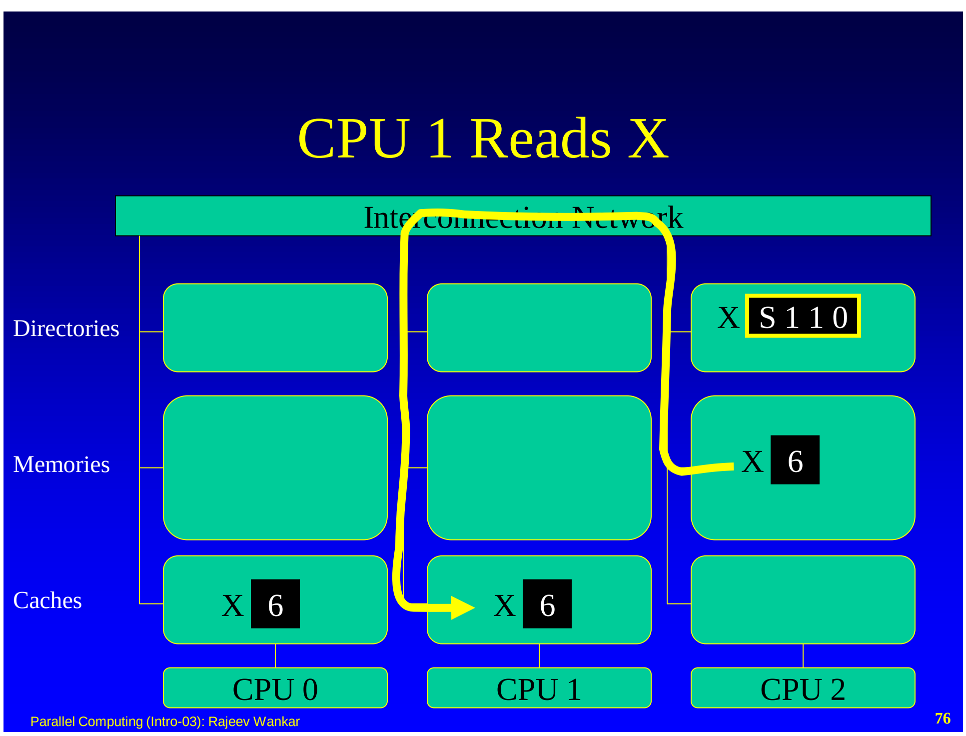

CPU 0 Writes 6 to X
Interconnection Network
Directories
Memories
Caches
X S 1 0 1
Invalidate
X 7
X 7
X 7
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
71

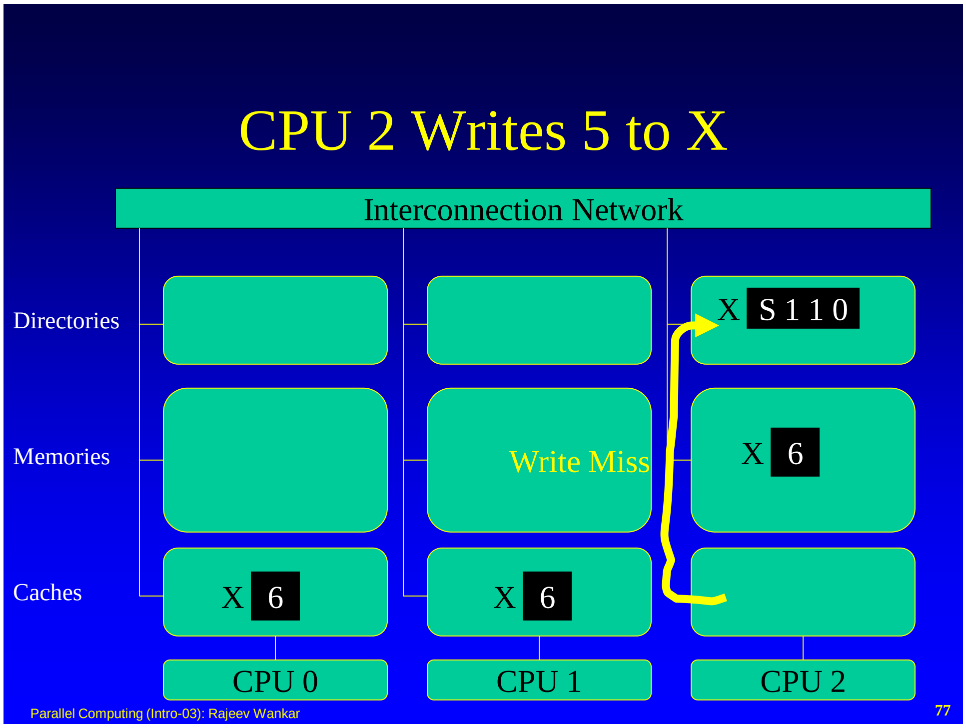

Obsolete
CPU 0 Writes 6 to X
Interconnection Network
Directories
Memories
Caches
X E 1 0 0
X 7
X 6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
72

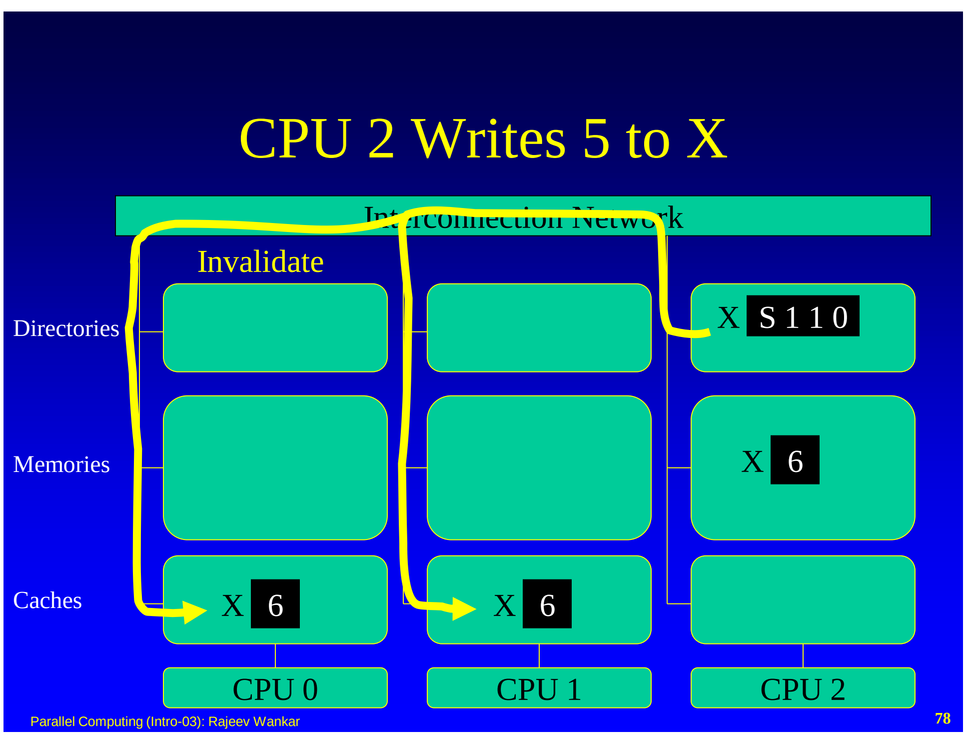

CPU 1 Reads X
Interconnection Network
Read Miss
Directories
Memories
Caches
X E 1 0 0
X 7
X 6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
73

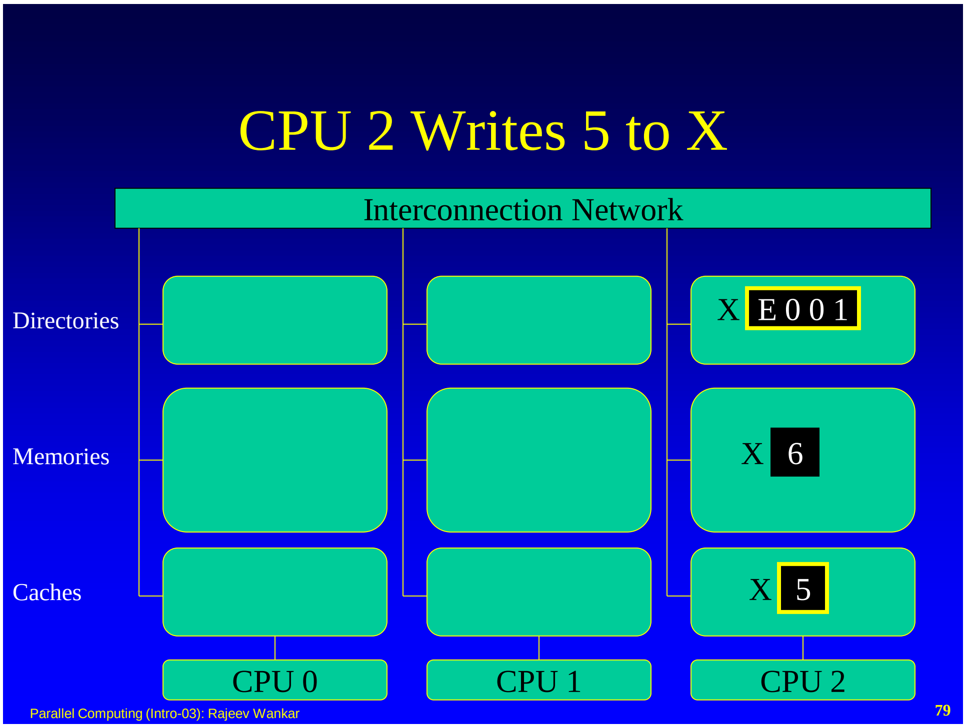

This message is sent by Dir. Con. For CPU 2
CPU 1 Reads X
Interconnection Network
Switch to Shared
Directories
Memories
Caches
X E 1 0 0
X 7
X 6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
74

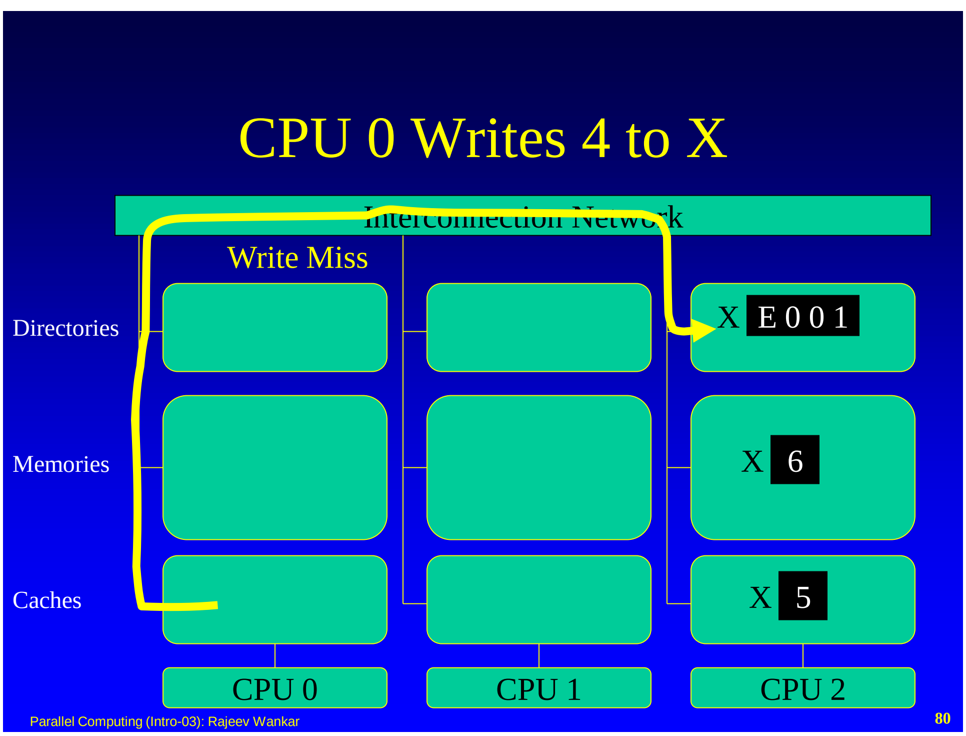

CPU 1 Reads X
Interconnection Network
Directories
Memories
Caches
X E 1 0 0
X 6
X 6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
75

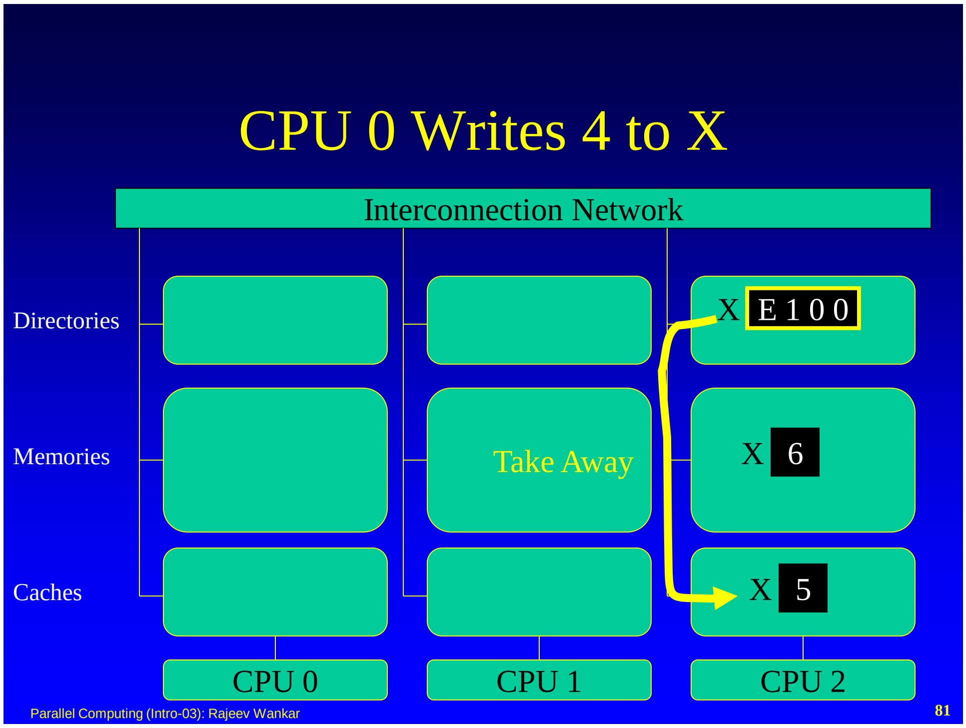

CPU 1 Reads X
Interconnection Network
Directories
Memories
Caches
X S 1 1 0
X 6
X 6
X 6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
76

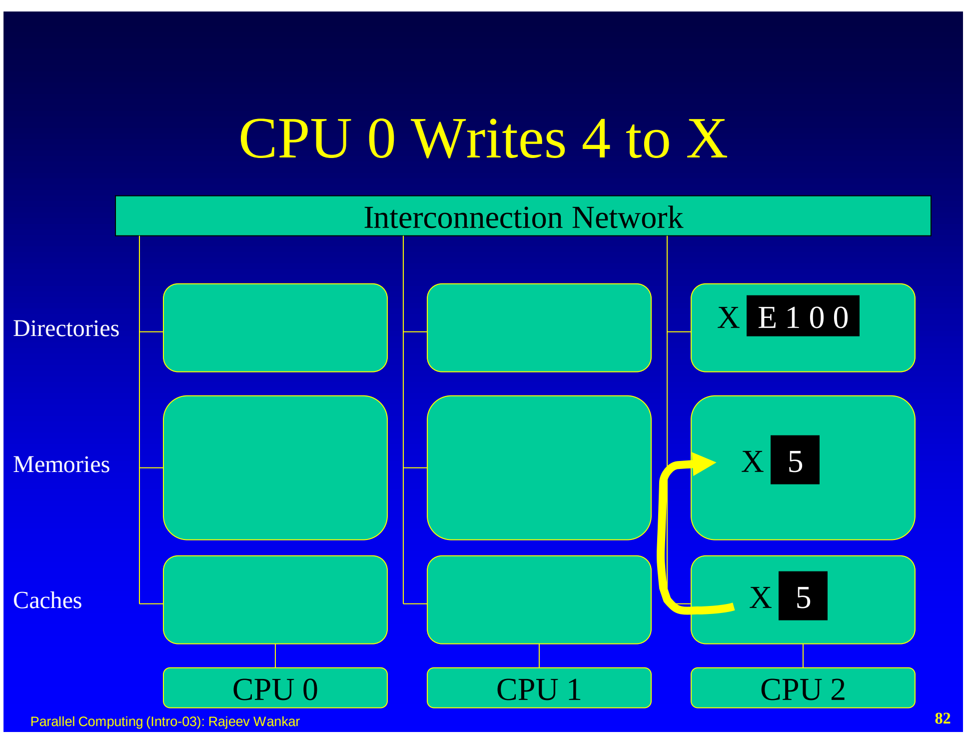

CPU 2 Writes 5 to X
Interconnection Network
Directories
Memories
Caches
X
S 1 1 0
Write Miss
X
6
X
6
X
6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
77


CPU 2 Writes 5 to X
Interconnection Network
Invalidate
Directories
Memories
Caches
X
S 1 1 0
X
6
X
6
X
6
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
78

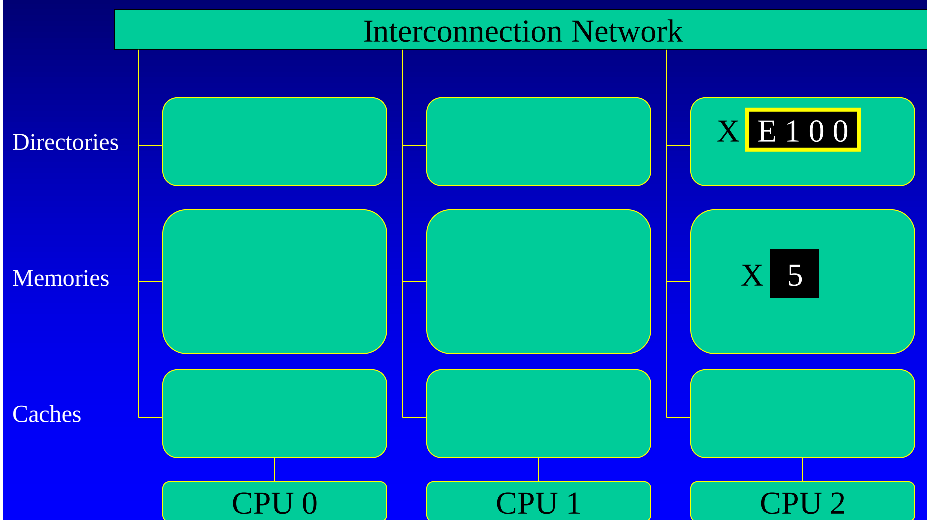

CPU 2 Writes 5 to X
Interconnection Network
Directories
Memories
Caches
X
E 0 0 1
X
6
X
5
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
79


CPU 0 Writes 4 to X
Interconnection Network
Write Miss
Directories
Memories
Caches
X
E 0 0 1
X
6
X
5
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
80


CPU 0 Writes 4 to X
Interconnection Network
Directories
Memories
Caches
X
E 1 0 0
Take Away
X
6
X
5
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
81

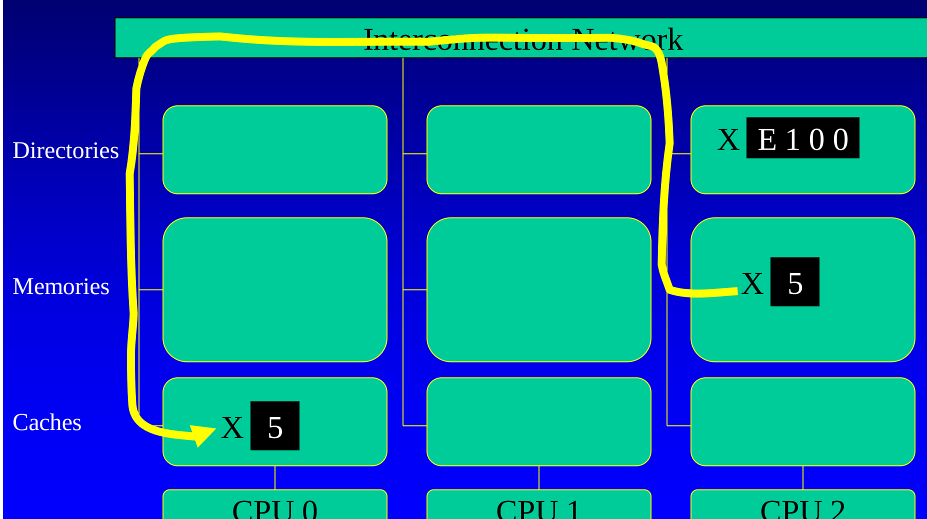

CPU 0 Writes 4 to X
Interconnection Network
Directories
Memories
Caches
X
E 1 0 0
X
5
X
5
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
82

CPU 0 Writes 4 to X

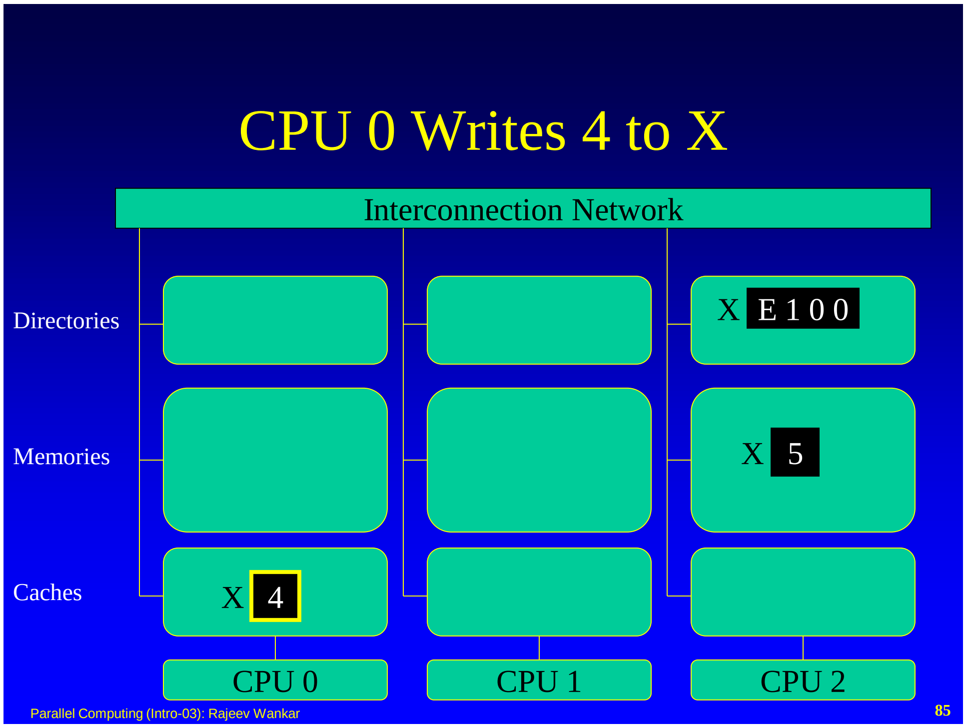


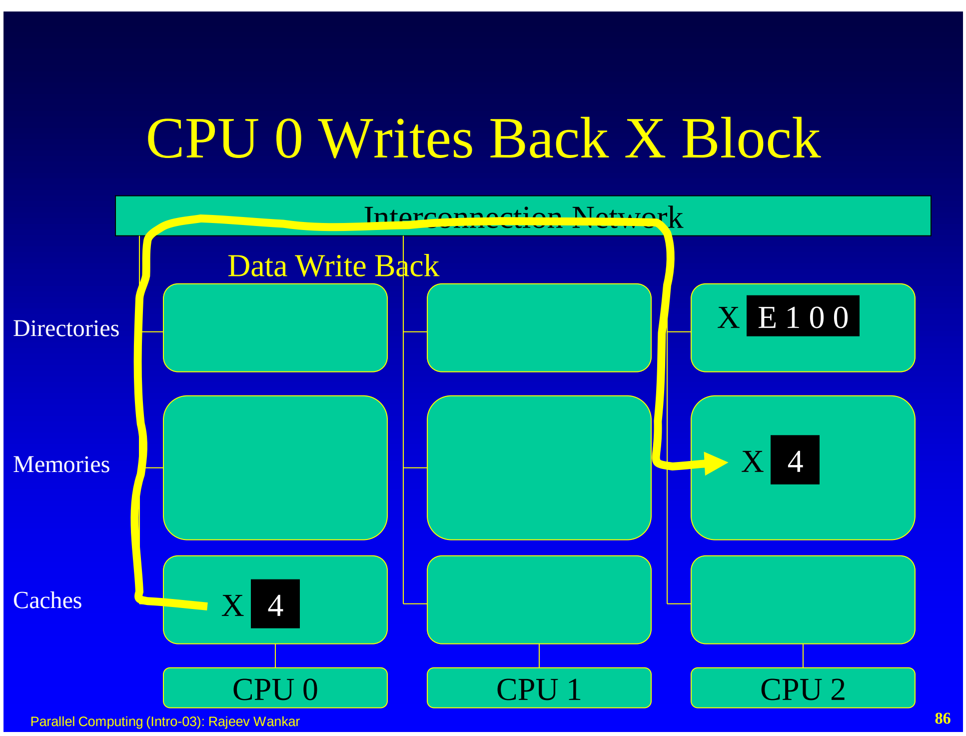
Parallel Computing (Intro-03): Rajeev Wankar 83

CPU 0 Writes 4 to X

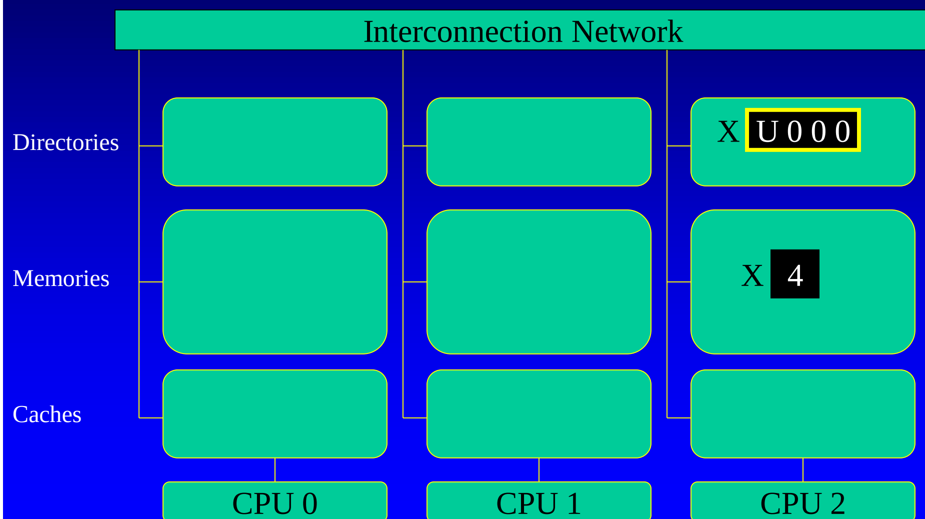


Parallel Computing (Intro-03): Rajeev Wankar 84


CPU 0 Writes 4 to X
Interconnection Network
Directories
Memories
Caches
X E 1 0 0
X 5
X 4
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
85


CPU 0 Writes Back X Block
Interconnection Network
Data Write Back
Directories
Memories
Caches
X E 1 0 0
X 4
X 4
CPU 0
CPU 1
CPU 2
Parallel Computing (Intro-03): Rajeev Wankar
86

CPU 0 Writes Back X Block

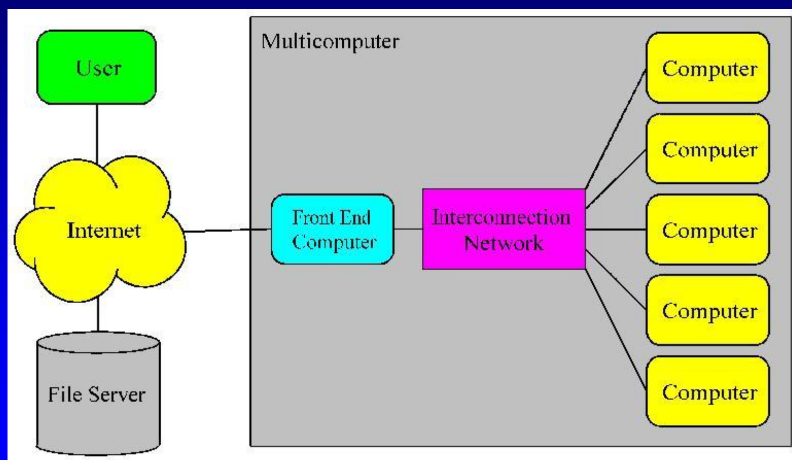


Multicomputers

- It has no shared memory, each processor has its own memory
- Interaction is done through the message passing
- Distributed memory multiple-CPU computer
- Same address on different processors refers to different physical memory locations
- Commodity clusters
- Store and forward message passing

★ Cluster Computing, Grid Computing

Asymmetrical Multicomputer



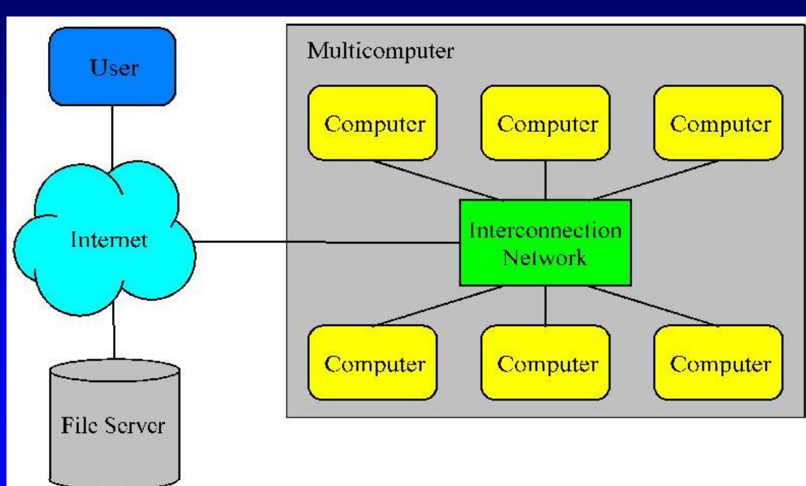
Asymmetrical MC Advantages

- Back-end processors dedicated to parallel computations ⇒ Easier to understand, model, tune performance
- Only a simple back-end operating system needed ⇒ Easy for a vendor to create

Asymmetrical MC Disadvantages

- Front-end computer is a single point of failure
- Single front-end computer limits scalability of system
- Primitive operating system in back-end processors makes debugging difficult
- Every application requires development of both front-end and back-end program

Symmetrical Multicomputer



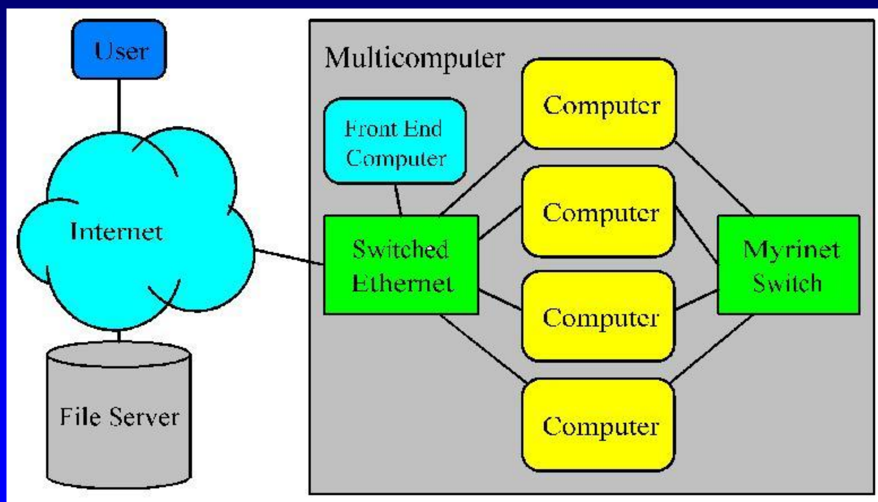
Symmetrical MC Advantages

- Improve performance bottleneck caused by single front-end computer
- Better support for debugging (each node can print debugging message)
- Every processor executes same program

Symmetrical MC Disadvantages

- More difficult to maintain illusion of single "parallel computer"
- No simple way to balance program development workload among processors
- More difficult to achieve high performance when multiple processes on each processor

ParPar Cluster, A Mixed Model



Commodity Cluster

- Co-located computers
- Dedicated to running parallel jobs
- No keyboards or displays
- Identical operating system
- Identical local disk images
- Administered as an entity

- *Will discuss in detail in the next chapter*

Network of Workstations

- Dispersed computers
- First priority: person at keyboard
- Parallel jobs run in background
- Different operating systems
- Different local images
- Check-pointing and restarting important

Speedup is the ratio between the time taken by the parallel computer, executing fastest sequential algorithm and the time taken by that parallel computer executing it using p processors

Efficiency = speedup/p

Parallelizibility is the ratio between the time taken by the parallel computer, executing parallel algorithm on one processor and the time taken by that parallel computer executing it using p processors