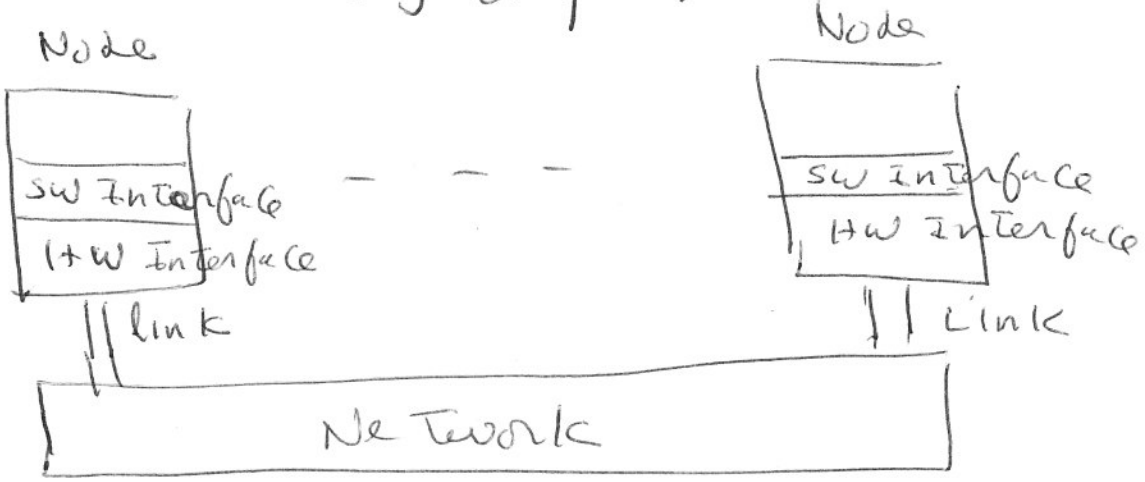
Networks

Introduction: Networks Goal is to connect Computers together and treat them as one big Computer



Type of Networks:

1- Massively parallel processor MPP network: connect Thousands of nodes, Max distance = 25 m

2- Local Area Network (LAN): connect hundreds of Computers, distance up to few km.

3- Wide Area network (WAN) connect Computers through the world, thousands of Computers, distance is Thousands of km.

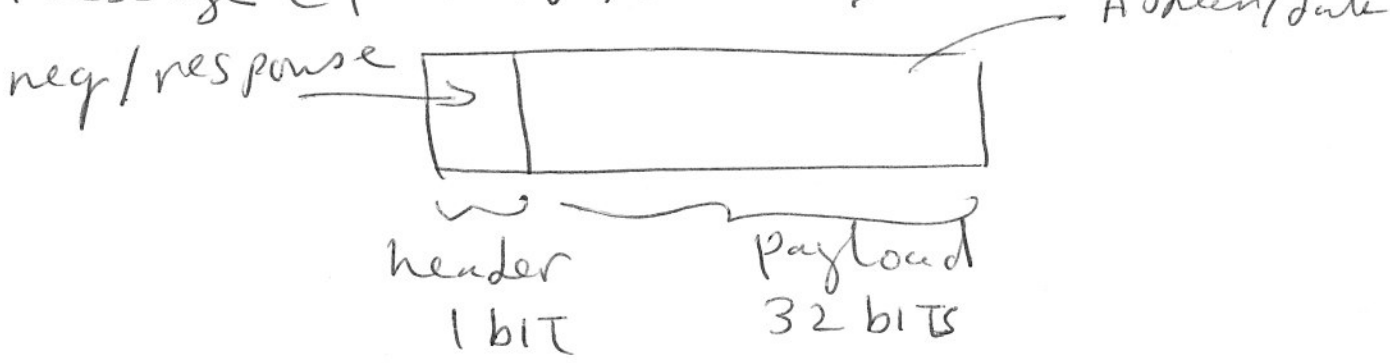
Simple Network

Design

- Use FIFO queue to hold Data and use Full Duplex Communication Link (Network)
- Need protocol for Rules of Communication.

Packet: message sent that include Commands (information needed to deliver message) and Data

Message (packet) Format:



header: If = 1 reply
If = 0 req

Example

0	F000 0000

req data From Comp Mem address F0000000

reply Data

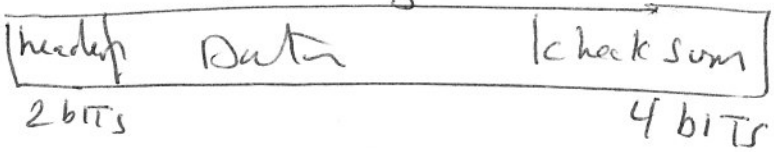
1	FE20576A

Data requested = FE20576A

Need The following:

- Error detection (checksum)
- Acknowledge req/reply (Exp-d header)

Message Format



0 0 = req
0 1 reply
1 0 = Ack req
1 1 = Ack reply

Communication protocol:

Send:

1 - The application copies data to be sent to OS buffer
2 - OS calculate checksum and includes it in message, Then starts the Timer.

3 - OS sends message to Network interface hardware

Receive:

1 - System copies data from Network interface hardware to OS buffer
2 - System calculates checksum and send back Ack if correct, else deletes message

Network Performance

Network parameters :-

1- Bandwidth: Maximum rate of information once message enters the Network

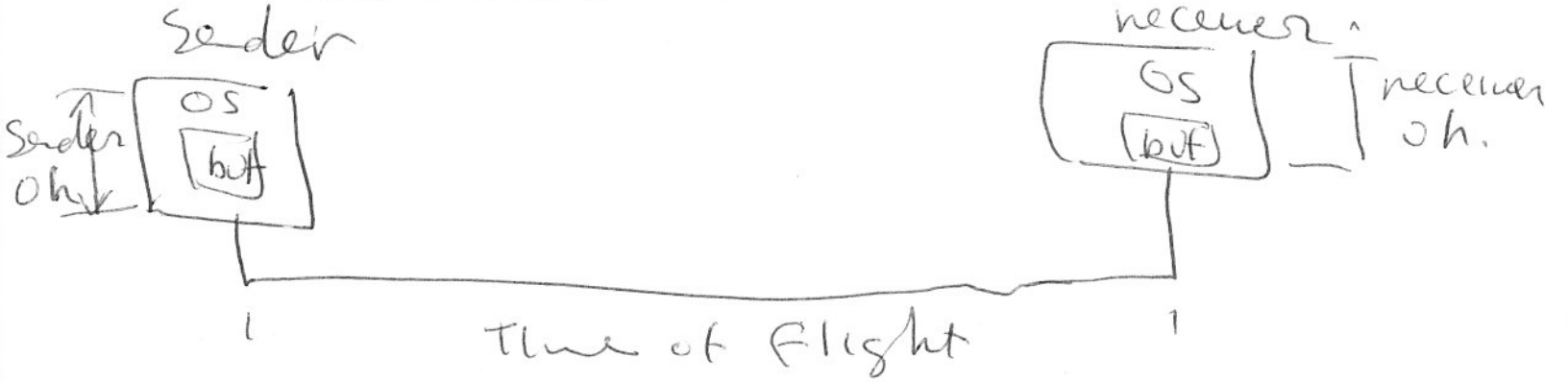
2- Transmission Time: Time for message To go through the Network = $\frac{\text{message size}}{\text{B.W}}$

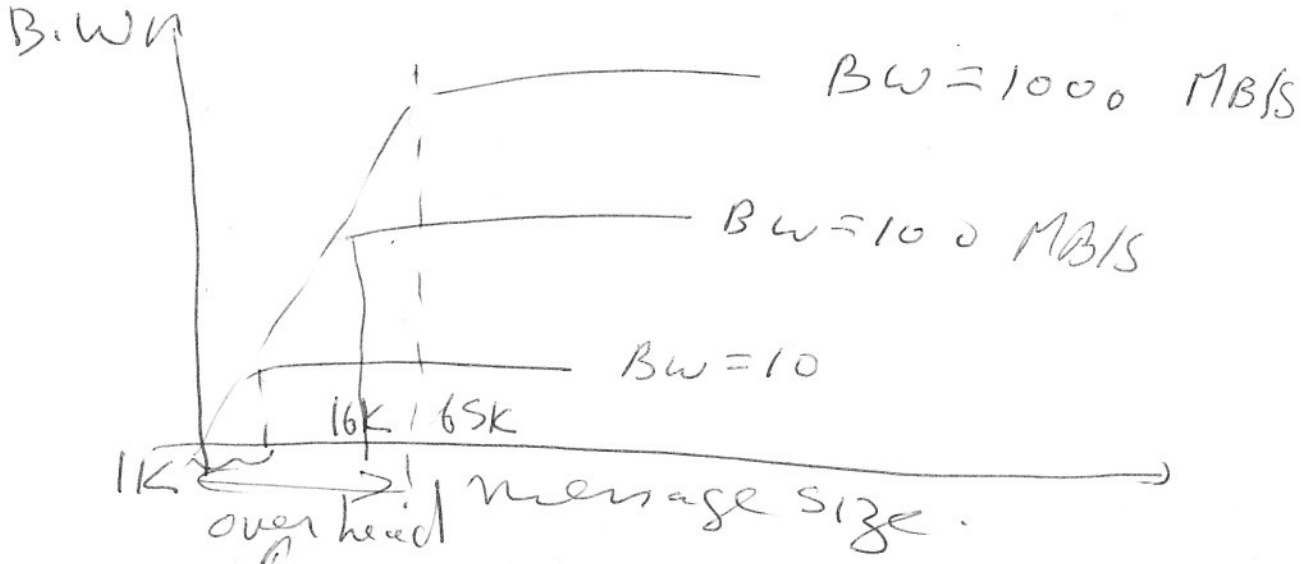
3- Time of Flight: Time of first bit to arrive at receiver (delay of N.W).

4- Sender overhead: Time for procesor To inject message into network.

5- Receiver overhead: time for procesor to pull message from network

Total Latency = Sender overhead + Time of Flight + $\frac{\text{Message size}}{\text{B.W}}$ + Receiver overhead





need large size of message for high B.W.

Network Media

1- Twisted pair:

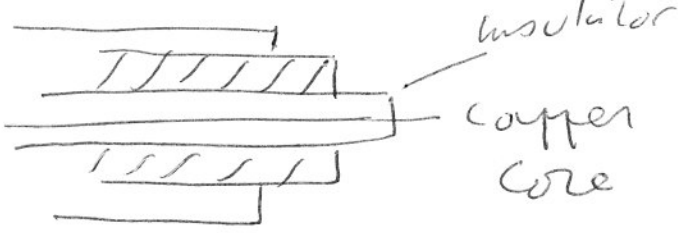
two insulated wires each about 1 mm and twisted to reduce interference.

B.W = few MBITS/sec

distance = few km

used in Telephone, LANS

2- Coxial Cable:

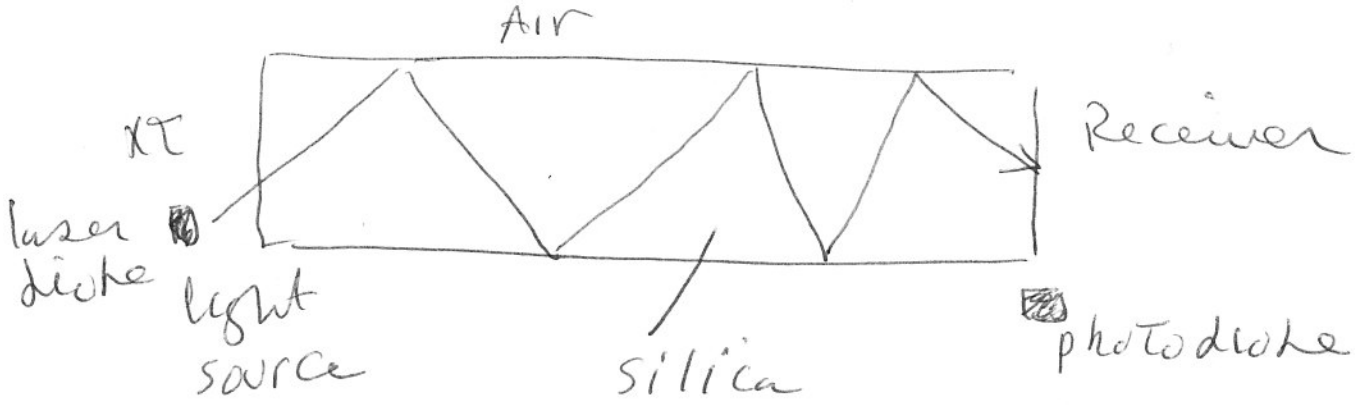


high B.W, noise immunity

BW = 10 MBITS/sec

distance = over 1 km

3- Fiber optic Network:



three components:

1- Fiber optic cable (Transmission media)

2- light source

3- light detector

only one way (simplex)

Fiber cable (two types):-

1- Multimode - uses LEDs
distance = up to 2 km
Bw = up to 600 Mbits/sec

2- single mode - use laser diodes
(expensive)
Bw = 6 BITS/sec
dist = 100 kms

problems: less reliable, expensive,
difficult to bend and attach connector.

Network Media

type	BW	dist	Cost/m	cost terminator	cost components interface
1 - Twisted pair	1 Mb/s	2 km	$.23	$4.6	$2
2 - Coax cable	10 Mb/s	1 km	$1.64	$220	$5
3 - Multimode Fiber	600 Mb/s	2 km	$1.03	$11.80	$1000
4 - Single Mode Fiber	2000 Mb/s	100 km	$1.64	$23.40	$1000

Network Topology

Topology: Define the structure of Interconnect

shared versus switched Media

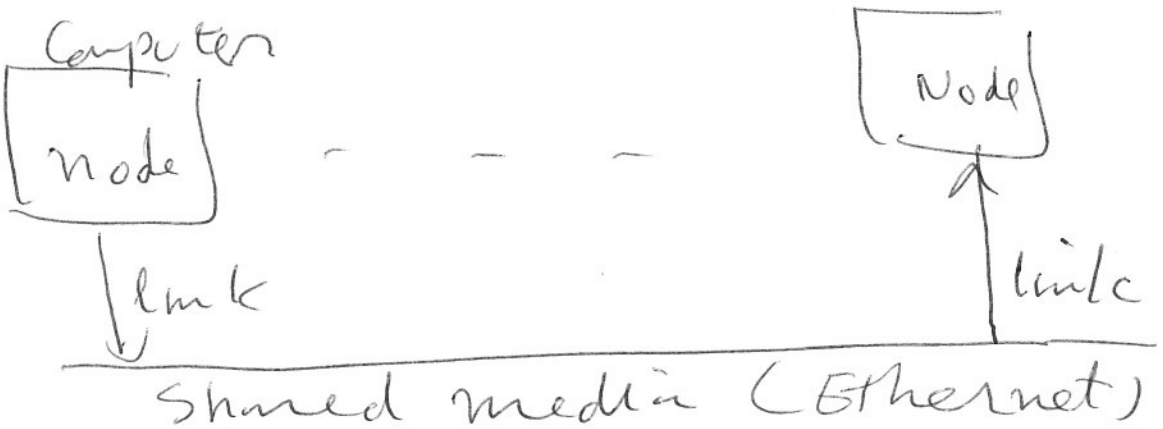
1- Shared :- simplest way to connect computers

Example: LAN using Ethernet

Ethernet: is a bus that can be shared by hundreds of computers
- only on message is sent at a time.
- it uses distributed Arbitration

Disadvantages: - performance degradation because of shared media (limited BW).

Advantage: inexpensive



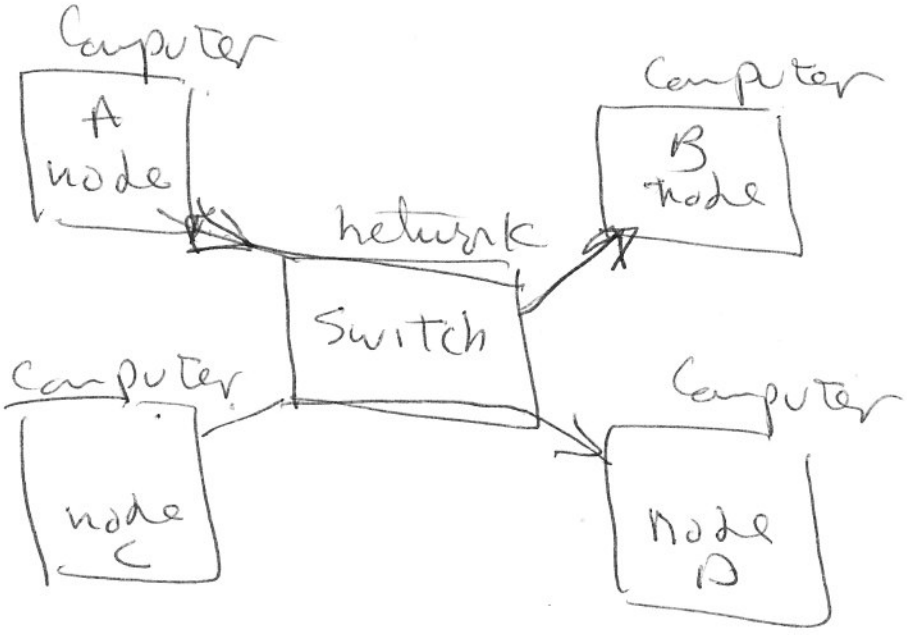
2 - Switched Media

Each node has a dedicated link to a switch that provides connection to all destination

Advantage: - Aggregate bandwidth is many times that of shared media
- provides direct connection between source and destination (no interference from other nodes).

switched media Advantages:-

- Faster because there is no Arbitration
- it allow multiple nodes to Communicate simultanously.
- it scales to large numbers of nodes

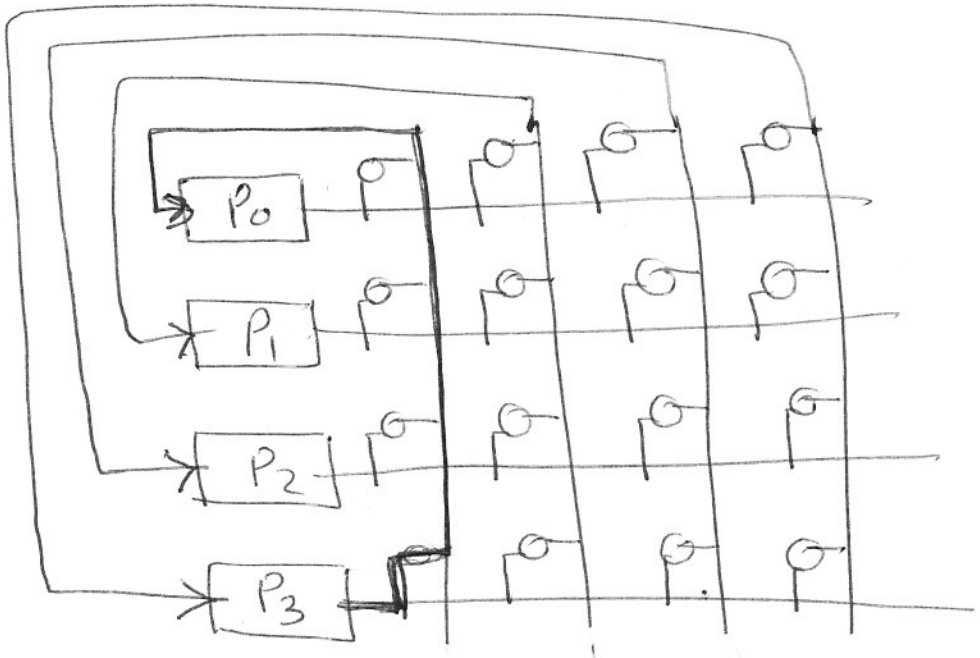


Switched Media (ATM)

switch topology

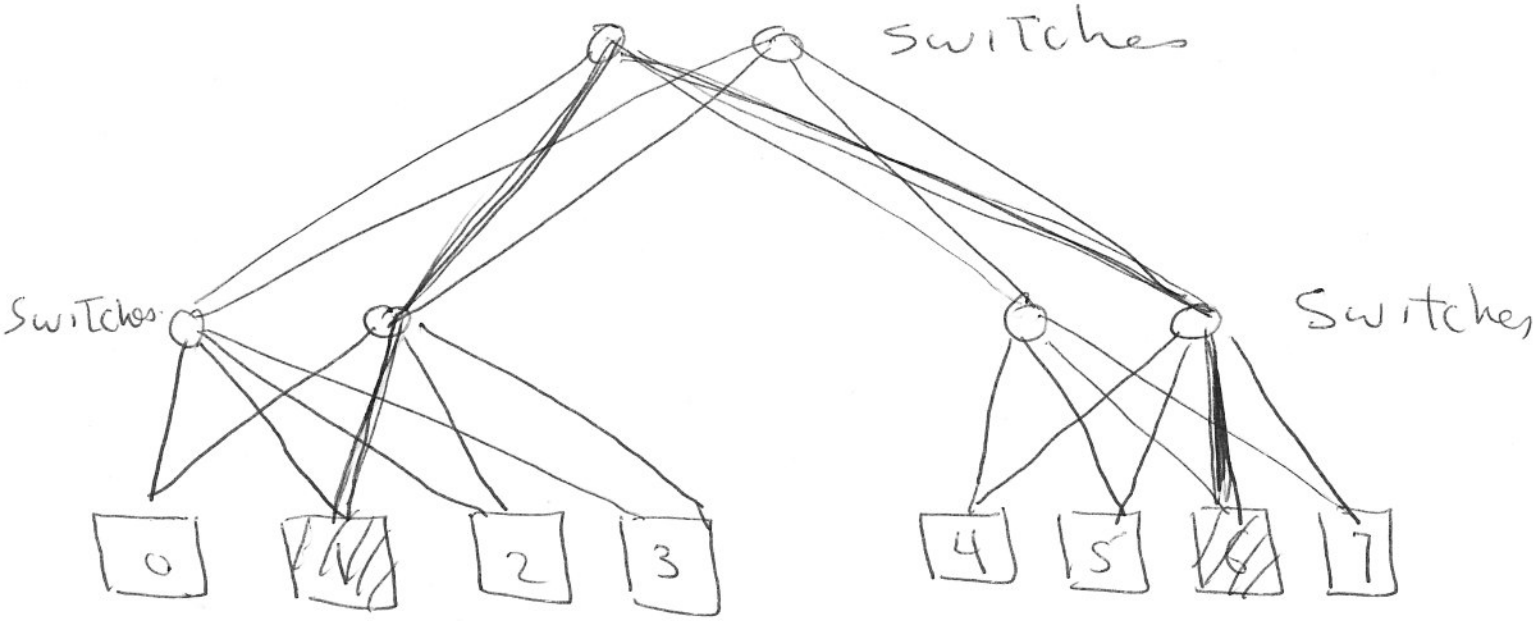
Define the structure of interconnects.

1- Cross bar switches uses nxn switches



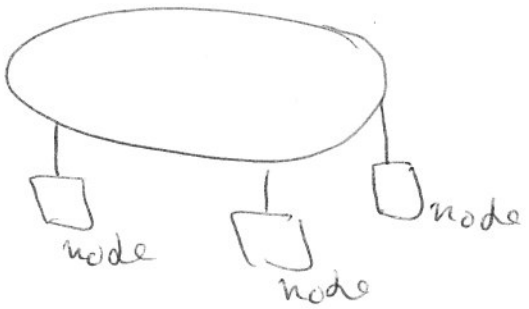
Used for Mpp

2- Fat-Tree: allows multiple paths between nodes.

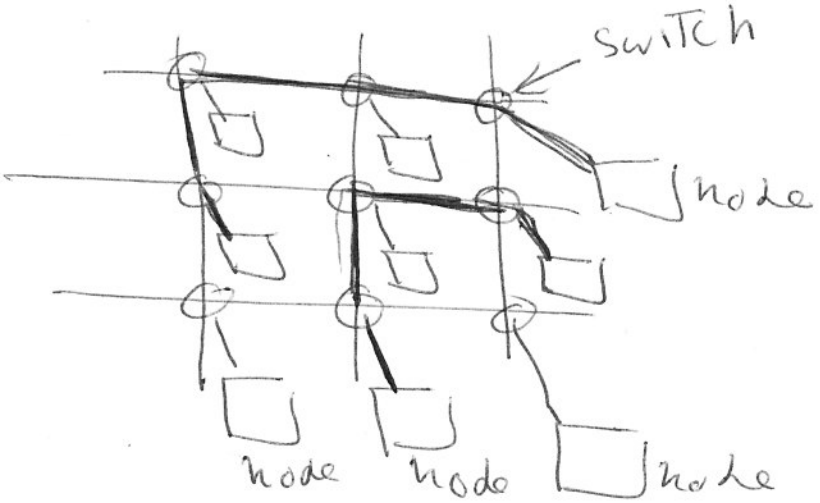


3- Ring Topology

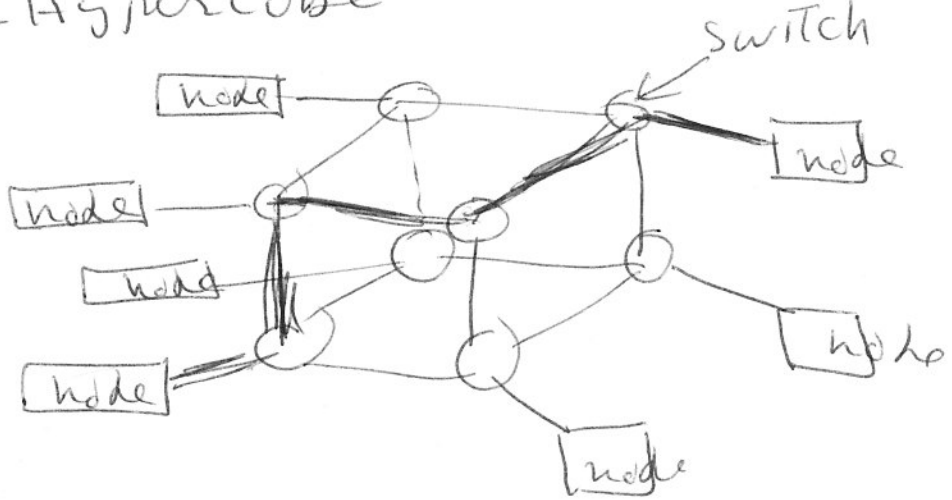
Low cost, single Token is passed around the ring to determine which node is allowed to send message (Arbitration)



4- 2D mesh



5- Hypercube



Cost/performance for 64 nodes

	Bus	Ring	2D	6-Cube	Fully Connected
performance B.W	1	2	16	32	1024
cost/SW	NA	3	5	7	64
number of Lnks	1	128	192	256	2080

Distributed shared Memory Multiprocessor system

Motivation: Scalable architecture for large number of processors

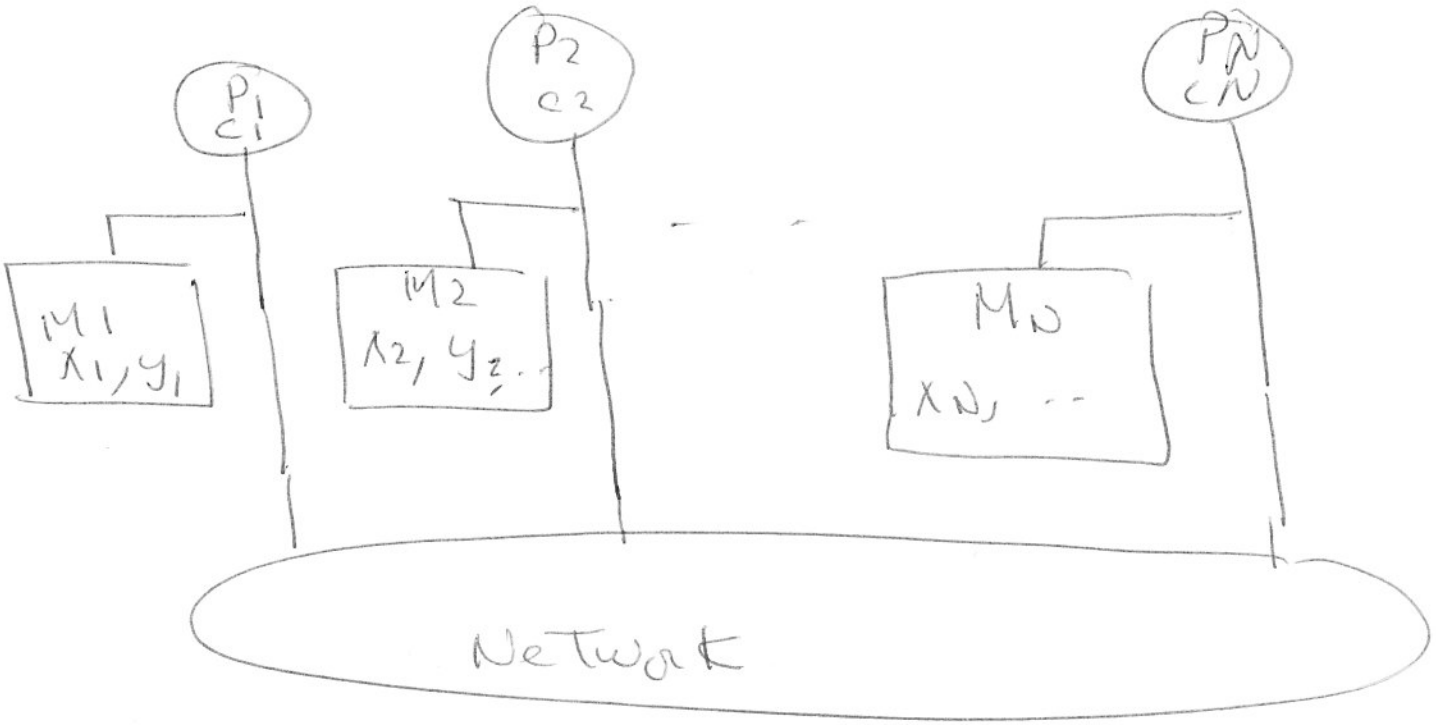
- It uses distributed memory among processor nodes with scalable switched network.

It uses messages to communicate between processors.

Two types of Accesses:

1- Local procesor to local memory

2- Local processor to Remote memory if Adress is not in local memory.



cost / performance

Bus performance = 1 .

6-cube performance = 32 .

Disadvantage of snooping cache protocol:

1- requires communication on every cache miss / write

2 - Arbitration for the bus

3 - one shared memory

EXAMPLE: Calculate Scalability of snooping protocol that uses 64 bit bus at 200 MHZ, if each procesor runs at 1 6HZ, and cache miss + write = 1%, block size = 32 B.

Solution:

bus bandwidth = $\frac{8\,B}{5\,ns} = 1.6\,GB/sec$

Scalability = $\frac{1.6\,GB/sec}{32\,B \times 0.01 / 1\,ns} = \frac{1.6\,GB/sec}{.32\,GB/sec} = 5$

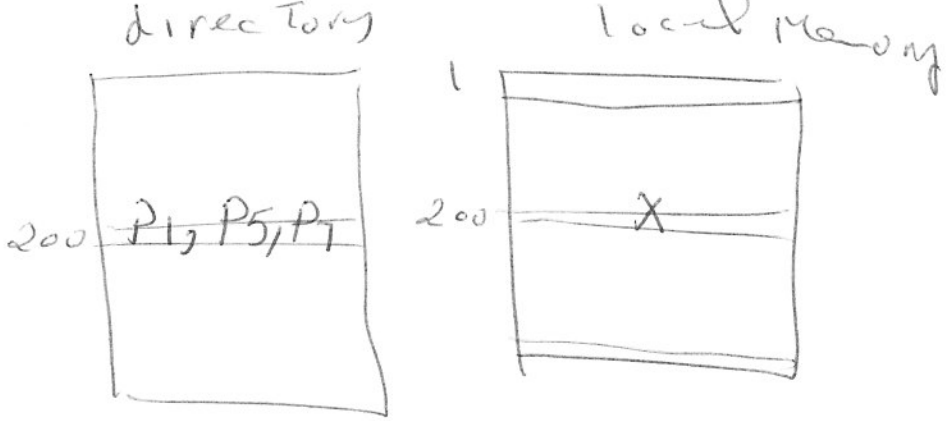
Concept of Distributed Memory Architecture

- Memory is distributed among processors.
= Multiple accesses between processors could use Network - d distributed memory.
- Directory is used to keep coherency. Each memory block has an entry in the directory that indicates the status of block.

Cost of directory = cost of memory locations x number of used processors

EXAMPLE

memory location 200 is used/shared by P_1, P_5, P_7



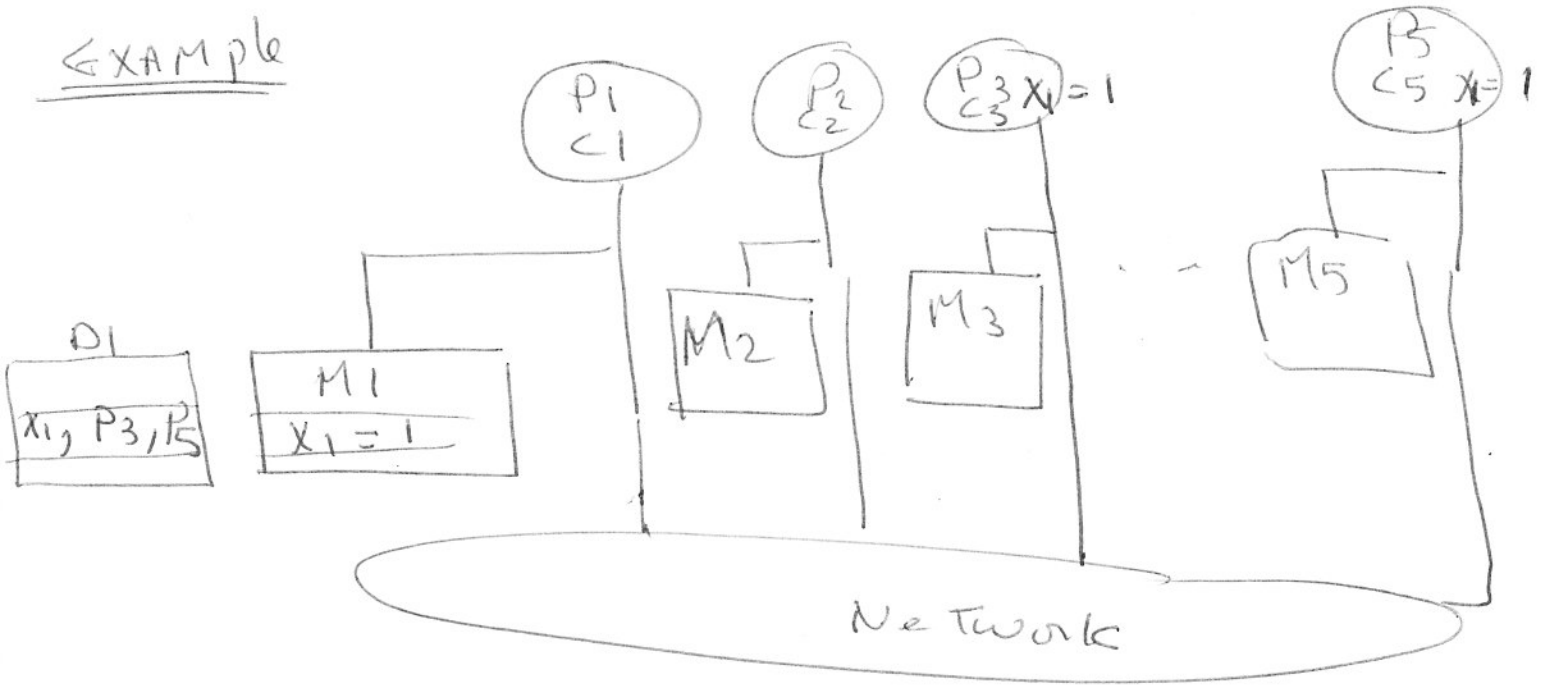
Directory entry for 200 has vector bits for P_1, P_5, P_7 set to 1.

Directory Based coherency protocol

Three different states of blocks in cache.

1- Shared: one or more processor has the block in their cache. value in memory is up to date

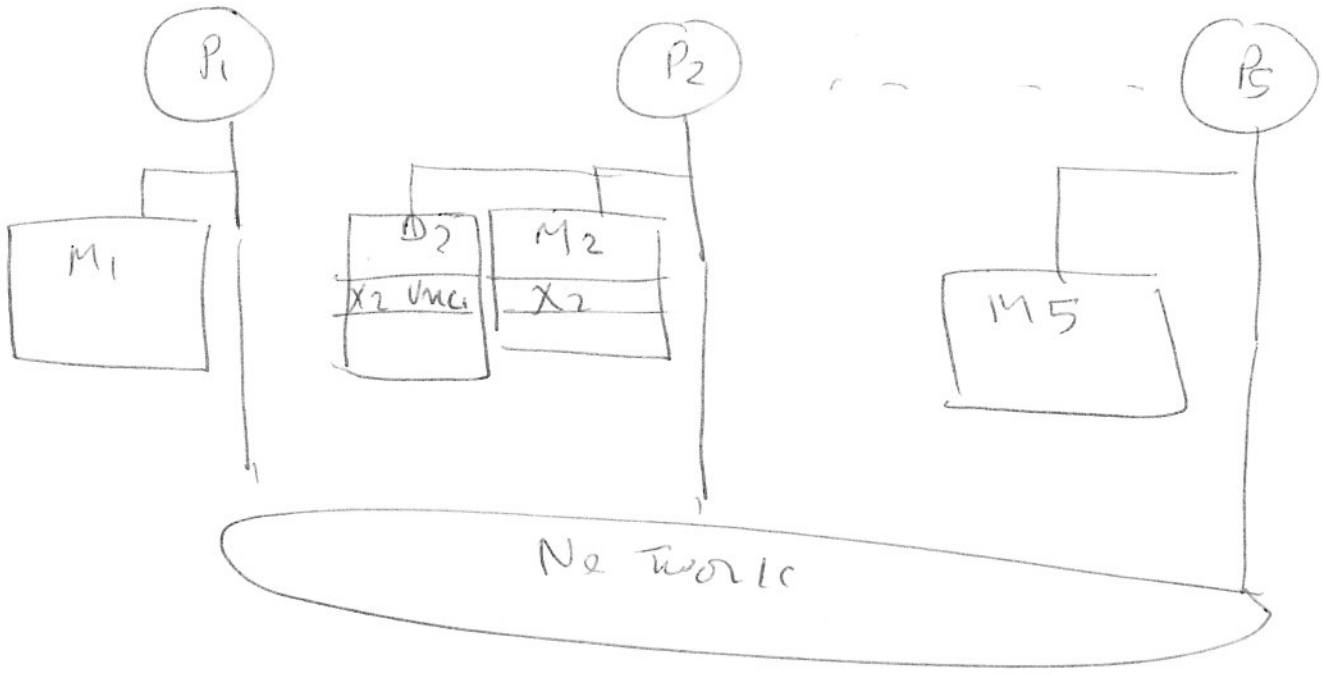
EXAMPLE



X1 is shared between P3, P5

2- Uncached: No Processor has a copy of the location.

EXAMPLE



X_2 in M_2 only, D_2 = Uncached for X_2.

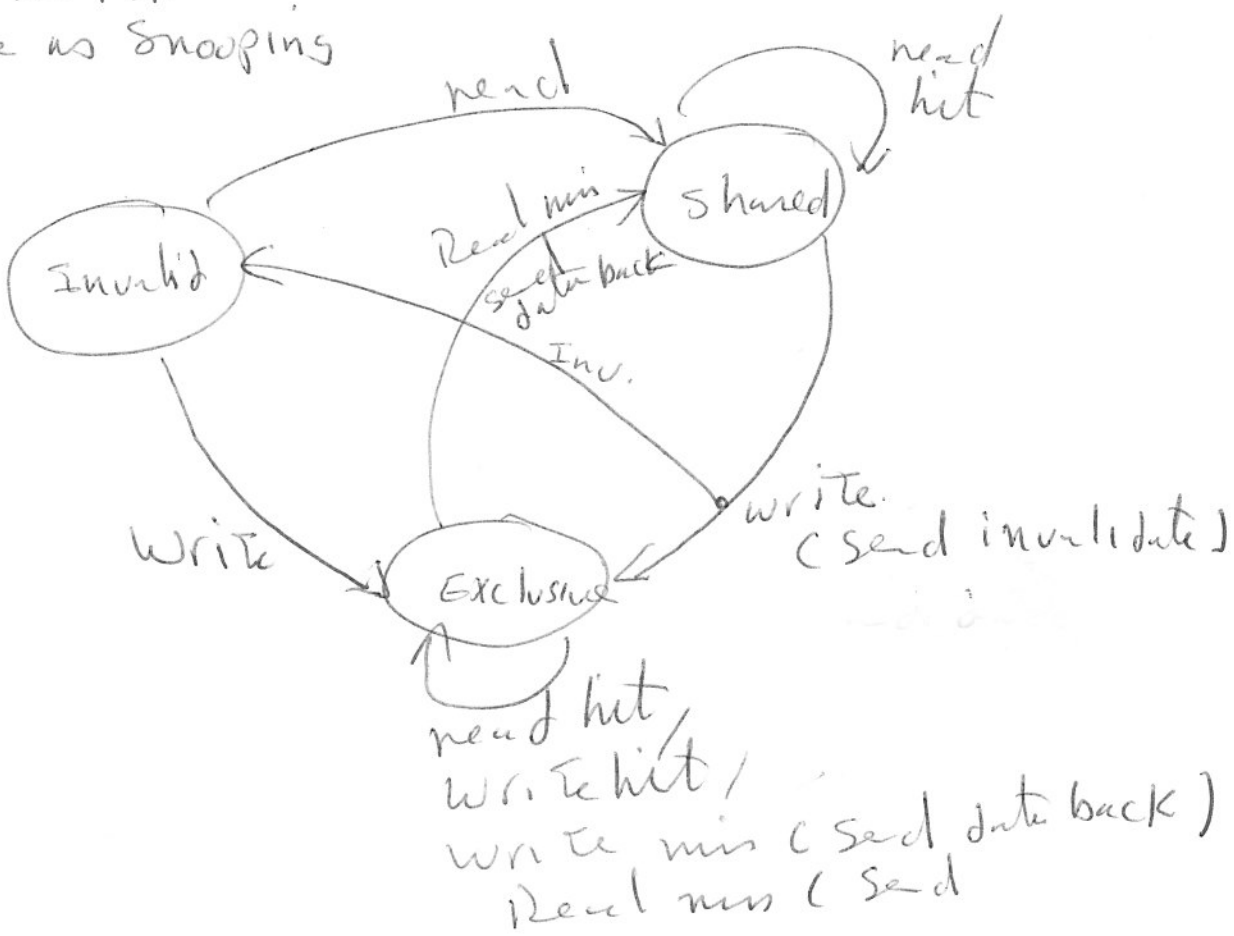
3- Exclusive:

One Processor has a copy of block in its Cache after it has written to it.

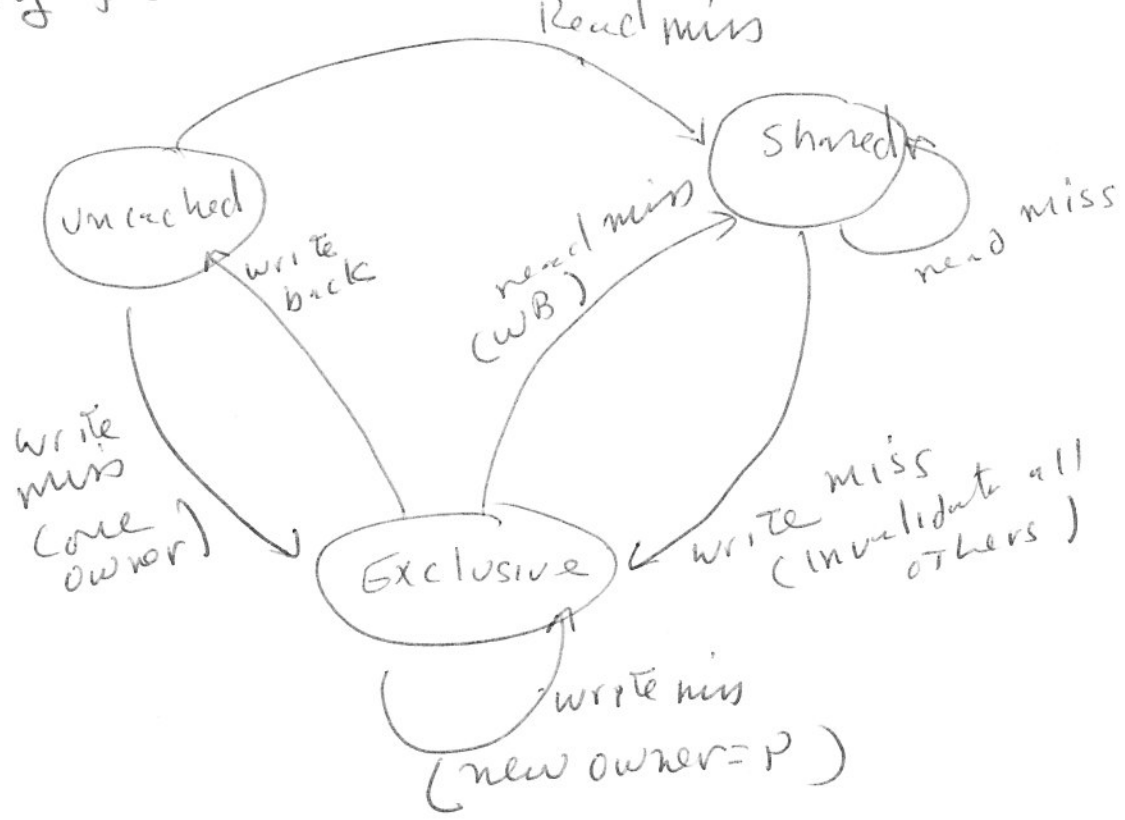
Memory is out of date, Processor is the Owner.

$X_1 = 3$ in P_3, P_3 is the owner.
X_1 is Exclusive!

1. Cache coherency protocol
For NUMA
= Same as Snooping



2- Directory protocol



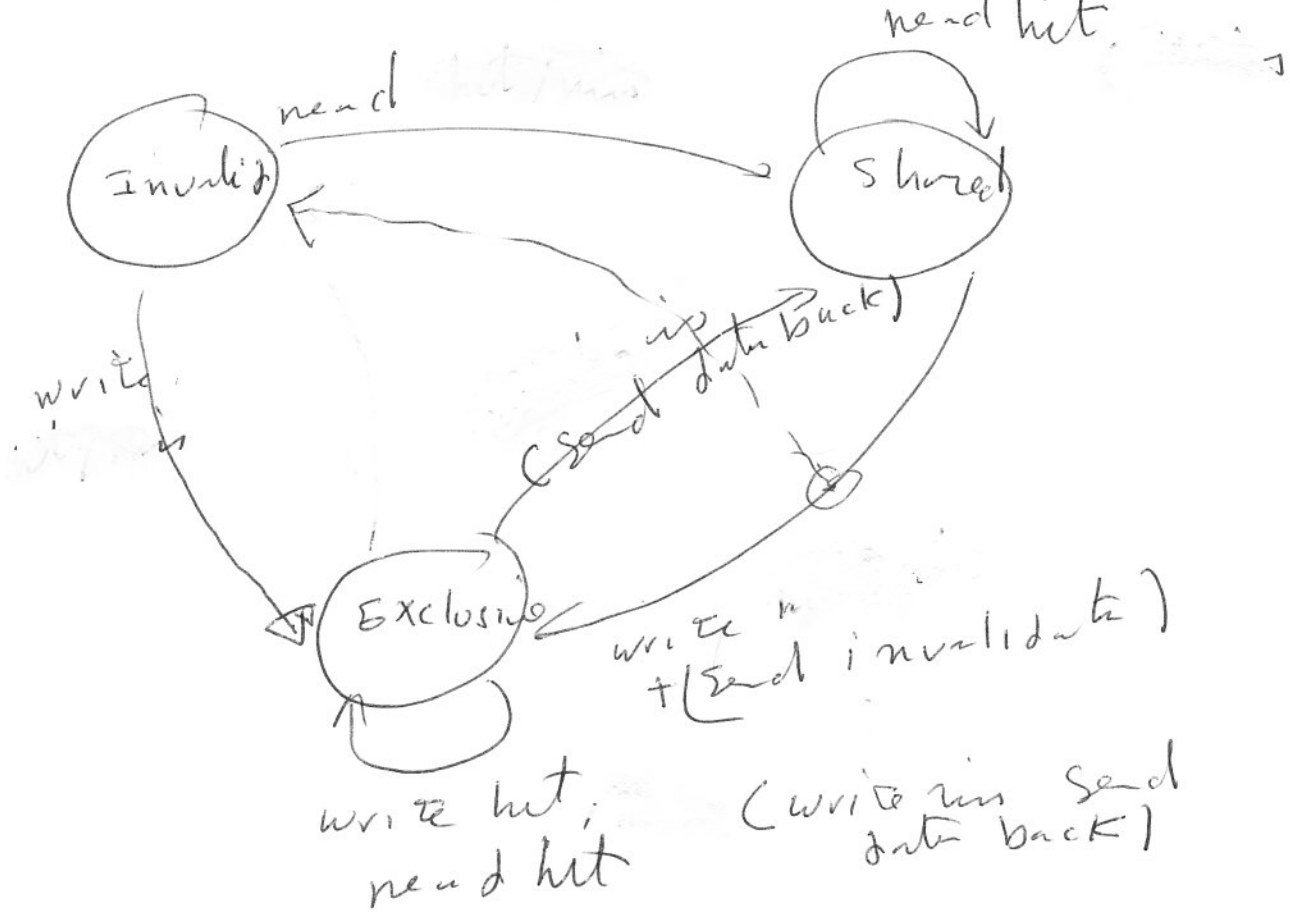
Different operations:

1- Read Miss: requesting processor send Address, read.
- State of block in cache is made shared.
- State of block in home directory is made shared and Processor number is added.
- If block replaces an Exclusive block, it must be written back to its own memory and change its state. (uncached)

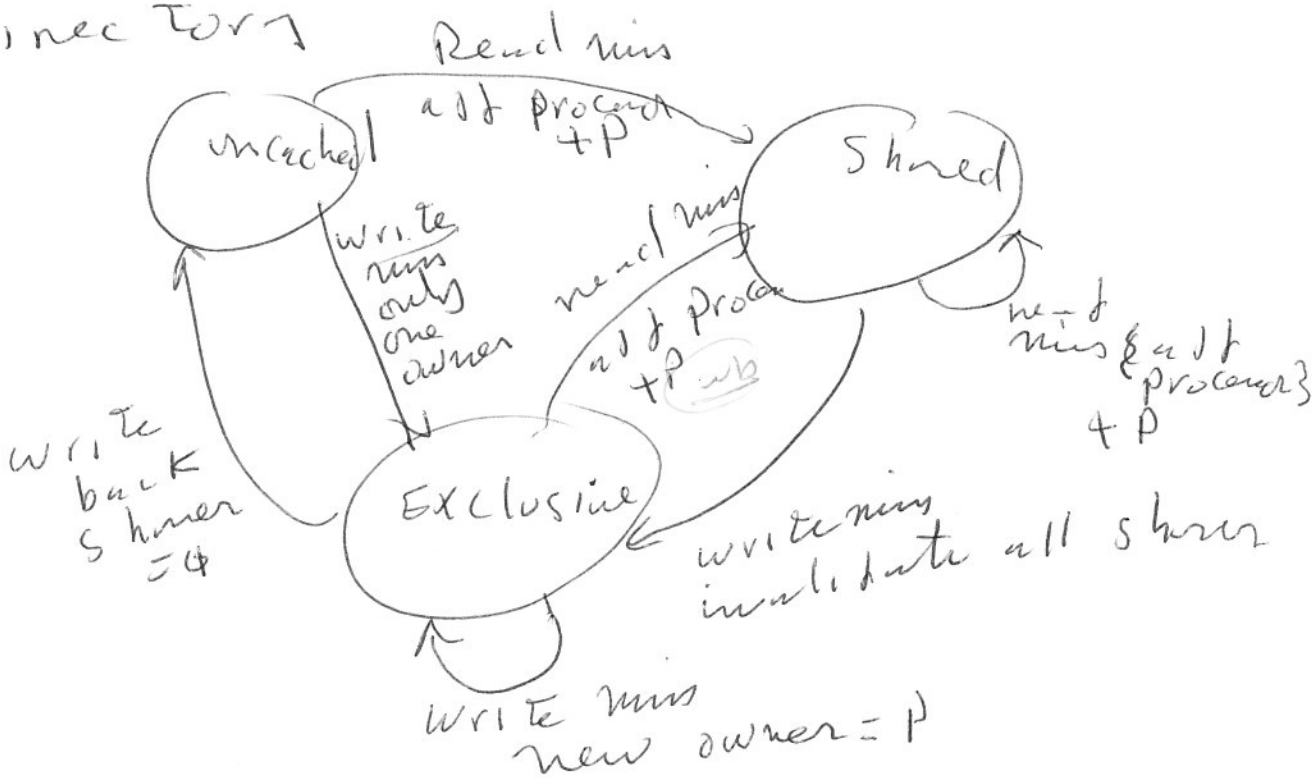
2- write Miss: requesting processor send Address, write.
- State in cache is made Exclusive,
  Invalidate other blocks in Processor Caches
- State in home directory is made Exclusive and update owner.
- If replaces an Exclusive block, it must be written back.

EXAMPLE: P_3 writes to X1.
X1 becomes Exclusive in C_3,
Directory of P1 indicates that P_3 is the owner (State = Exclusive).

1- Cache coherency protocol
= same as in snooping



2- Directory



Cache coherency could be maintained by software, assuming all shared data are ~~non~~ uncacheable, Then copy them into private data, but compiler for Implement software coherency are limited and programmer has to deal with coherency of each shared block.

Disadvantages of snooping cache protocol:

- requires communication on every cache miss including writes

Example: 64 bit bus, with clock = 200MHz

Bw = 1.6 GB/sec.

if cache misses including writes is 1%

of a 1GHz processor, with block = 64 Byte

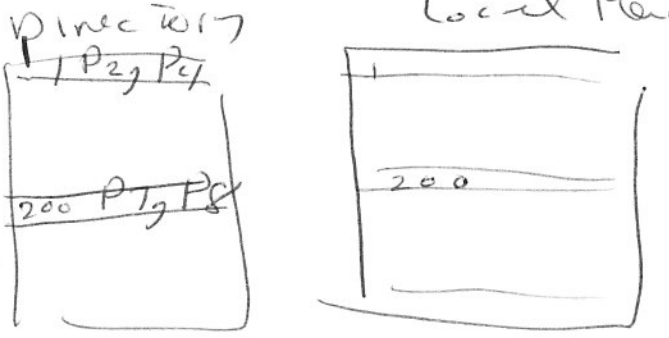
$$\text{scalability} = \frac{1.6\,GB/sec}{64B \times .01 \times 1\,GB/sec} = 2 \text{ processors}$$

Solution: use Directory protocol

In directory, information is kept where to get data (Network)

Centralized directory That keeps state of all shared cache blocks and their processors

Each memory block has an entery in directory (NOT Cache)



Cost of directory

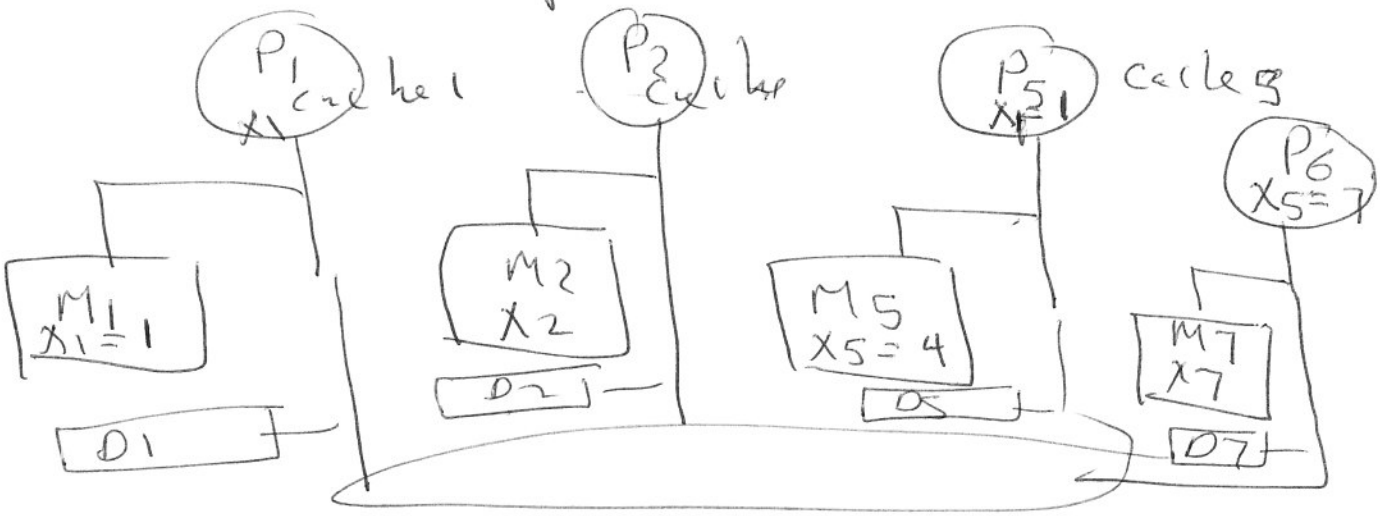
- number of memory blocks X number of processors

large Directory size (could be scaled)

Directory coherency protocol

States of cache blocks in Directory:

1- Shared: one or more processor have the block in their cache and the value in memory is up to date.



X1 is shared on P5.

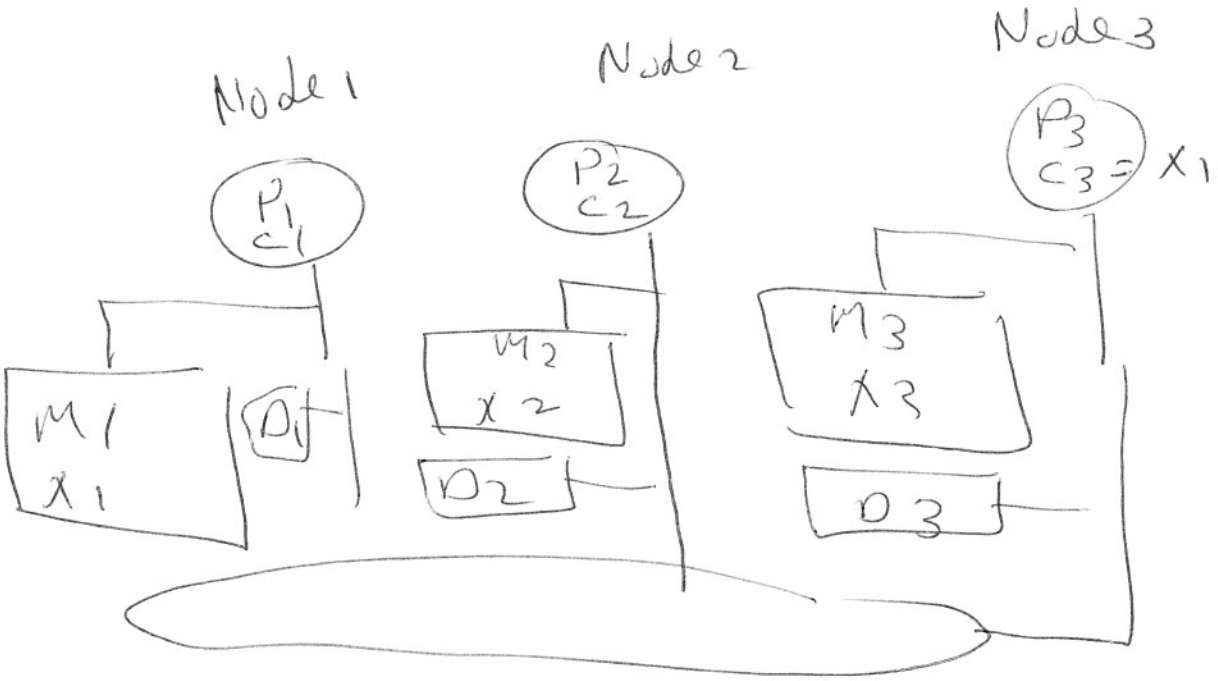
2- Uncached: No processor has a copy of X2

3- Exclusive: One processor has a copy of block (it has written to it), memory is out of date (Processor is owner)

X5=7 in cache 6 and P6 is owner

Protocol Principles:

- write to shared or uncached data will generate a miss, processor must block until it completes write (not a snooping..)



Node 1 is home node for X1

if Node 2 request a read to X1, Node 2 is called local node and if P3 has X1, then Node 3 is called remote node

Different Directory operations:-

1- Read miss - requesting processor sent Address, read, state of block is made Shared in Directory of home processor.

Example: P3 reads X2, generates a msg coping X2 to C3, makes X2 shared in M2 by P3

If block is Exclusive, it must be written back to its home memory.

2 - write miss:- requesting processor sent Address, write. state of block is made Exclusive and owner updated in directory; other processor cache invalidate

Example P3 writes to X1, write msg, X1 becomes Exclusive in C3, and D1 indicates that P3 is only owner.

IF Exclusive, must be written back to its owner, and change owner in directory.