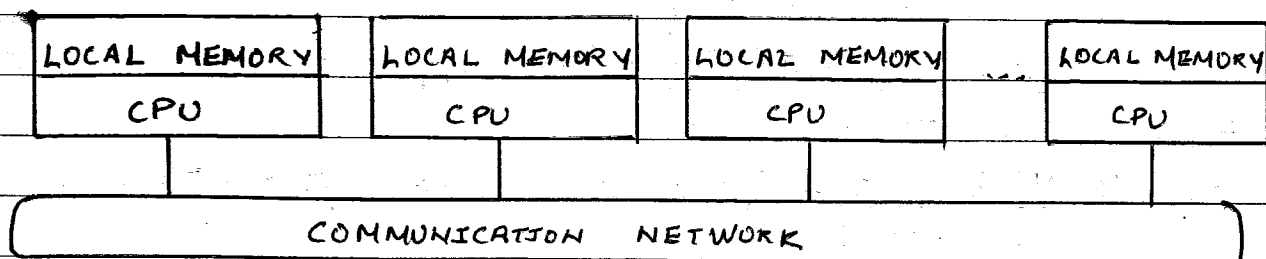


INTRODUCTION TO DISTRIBUTED SYSTEMS

① Distributed Systems -

It is a collection of independent computers that appear to the users of the system as a single computer.

② Architecture for Distributed System -



Distributed computing system (loosely coupled systems) is a collection of processors (nodes) interconnected by a communication network in which each processor has its own local memory and other peripherals, and the communication between any two processors of the system takes place by message passing over the communication network.

② Goals of Distributed System -

- (1) Making Resources Accessible for the users (and applications) to access remote resources, and to share them in a controlled way and efficient way.
- (2) Distribution Transparency that is to hide the fact that its processes and resources are physically distributed across multiple computers.
- (3) Openness or open distributed system is a system that offers services according to standard rules that describe the syntax and semantics of those services.
- (4) Scalability with respect to its size (more users & resources), geographically and administratively.



③ Advantages of Distributed Systems -

- own centralized systems
- (1) Economics - Microprocessors offer a better price/performance than mainframes
 - (2) Speed - A distributed system may have more total computing power than a mainframe
 - (3) Inherent Distribution - Some applications involve spatially separated machines
 - (4) Reliability - If one machine crashes, the system as a whole can still survive.
 - (5) Incremental Growth - Computing power can be added in small increments
 - (6) Data Sharing - Allow many users access to a common data base
 - (7) Device Sharing - Allow many users to share expensive peripherals
 - (8) Communication - Make human-to-human communication easier
 - (9) Flexibility - Spread the work load over the available machines in the most cost effective way.
- own isolated (personal) computers

④ Disadvantages of Distributed Systems -

- (1) Software - little software exists at present for distributed systems.
- (2) Networking - The network can saturate or cause other problems
- (3) Security - Easy access also applies to secret data.

⑤ Hardware Concepts -

All distributed systems are MIMD (multiple instruction streams and multiple data streams) but not all MIMD are distributed systems.

MIMD computers are divided into two groups -

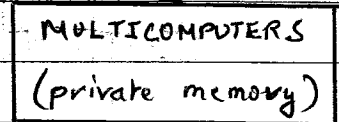
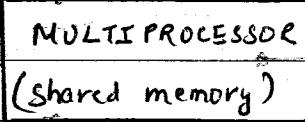
- (1) Multiprocessors (have shared memory)
- (2) Multicomputers (have private memory)

MIMD

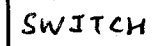
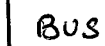
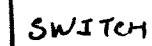
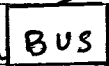
PARALLEL AND DISTRIBUTED COMPUTERS

Tightly Coupled

Loosely Coupled



Divided based on interconnection networks

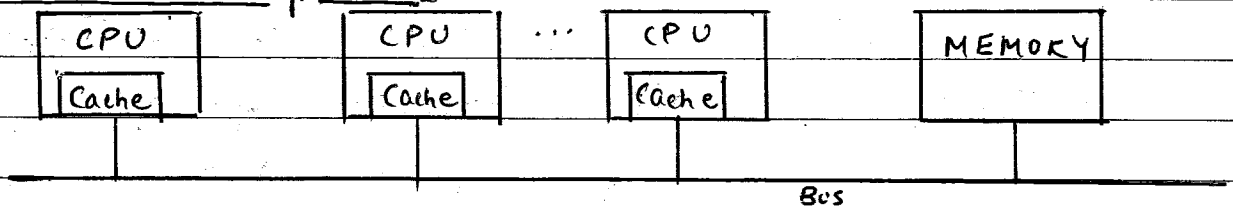


Sequent, Encore
Ultrasystems RPS

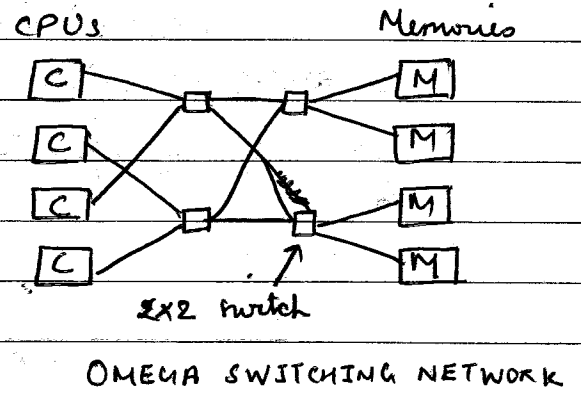
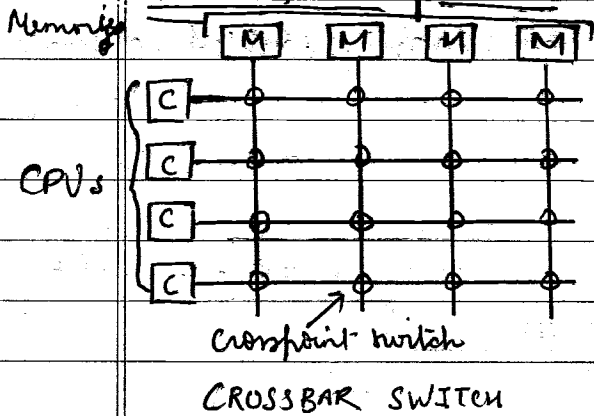
Workstations on a LAN

Hypercube, Transputer

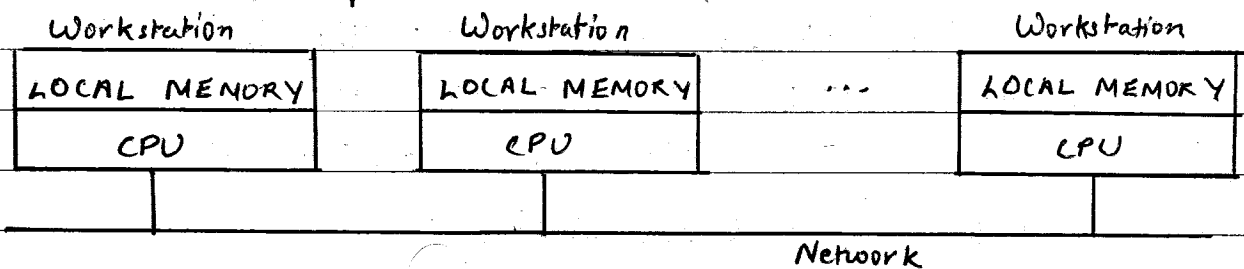
Bus-based multiprocessors -



Switched Multiprocessors -



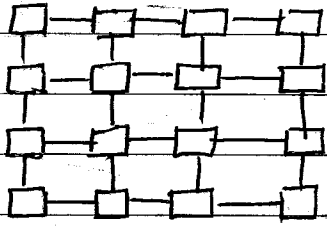
Bus-based microcomputers -



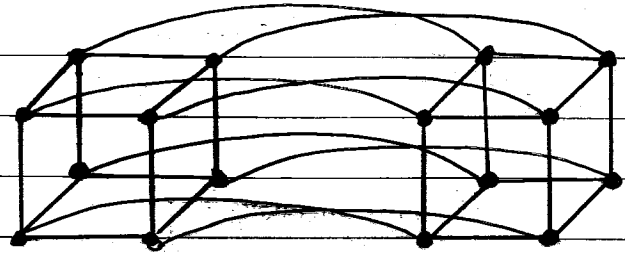
A multicomputer consisting of workstation on a LAN

The file system is supported by one or more machines called file servers.

Switch-microcomputers -



GRID



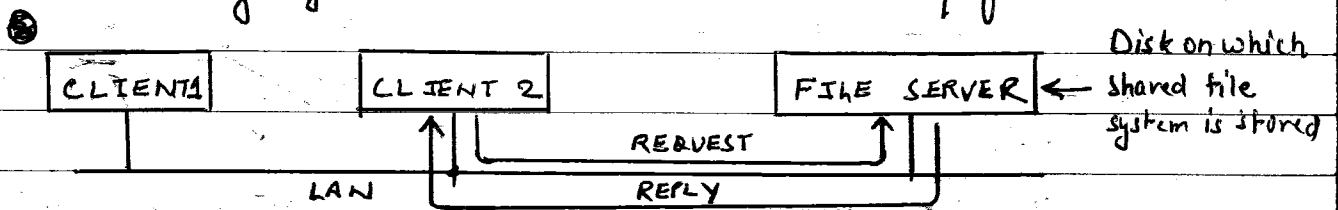
4-DIMENSION HYPERCUBE

⑥ Software concepts -

- Two kinds of operating systems for multiple CPU systems are -
- (1) loosely coupled software allows machines and users of a distributed system to be fundamentally independent of one another, but still to interact to a limited degree where that is necessary.
 - (2) Tightly coupled software have one single operating system or at least the feel of one operating system.

Networks operating systems -

loosely coupled software on loosely coupled hardware
Eg: Network of workstations connected by a LAN in which a user can remotely login to into another workstation / file server.



True Distributed Systems -

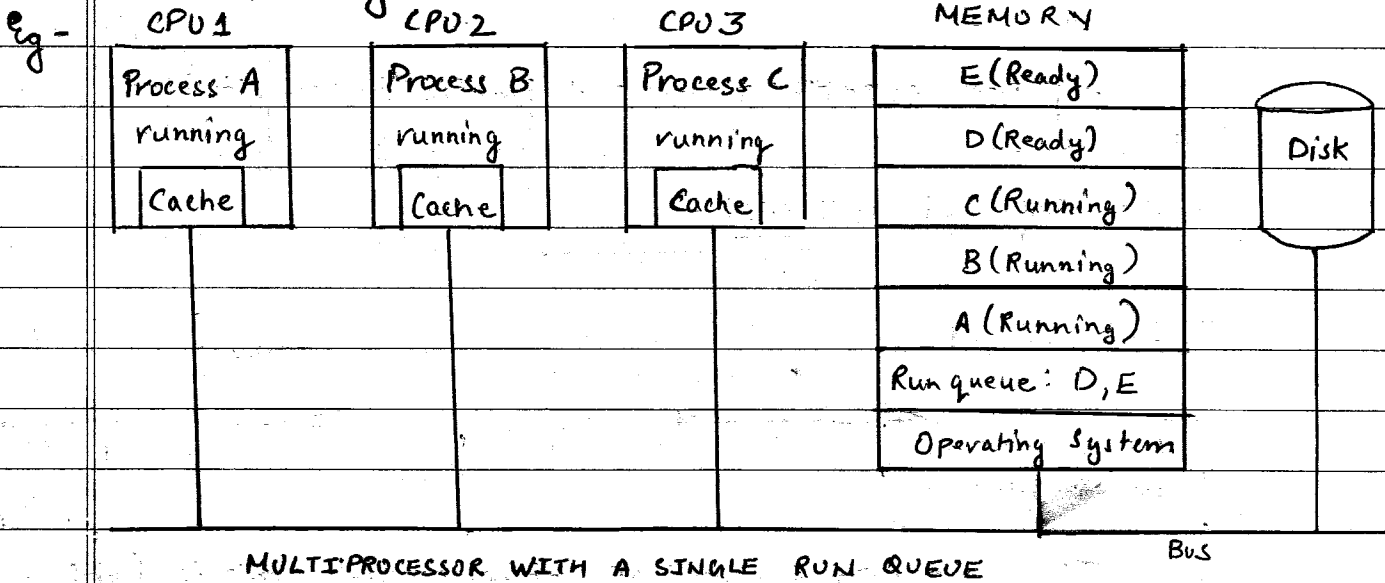
Tightly coupled software on loosely coupled hardware. The goal of such a system is to create the illusion in the minds of the users that the entire network of computers is a single timesharing system, rather than a collection of distinct machines.

Multiprocessor Timesharing Systems -

It is same as true distributed system but have only one operating system

Multiprocessor Timesharing Systems -

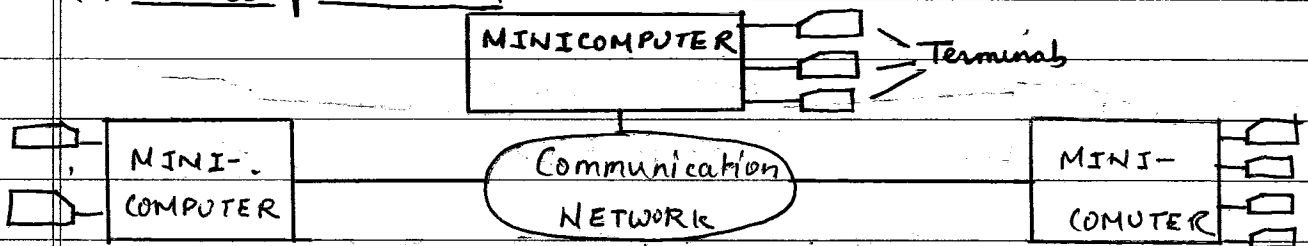
Tightly coupled software on tightly coupled hardware. Key characteristic is the existence of a single run queue that is a list of all the processes in the system that are logically unblocked and ready to run. The run queue is a data structure kept in the shared memory.



⑦ Distributed Computing Model -

It is classified into five categories -

(1) Minicomputer Model -



It is a simple extension of the centralized timesharing system. Each user can remotely access to other minicomputers.

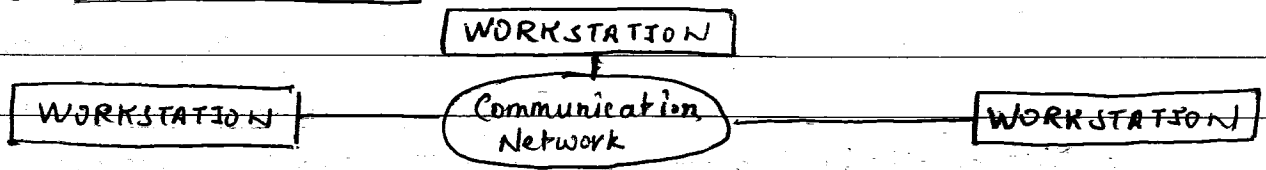
It is basically used for resource sharing that are available on some machine other than the one on to which the user is currently logged.

Several terminals are used for multiple users simultaneously logged on to a single minicomputer.

Eg: - Early ARPAnet.

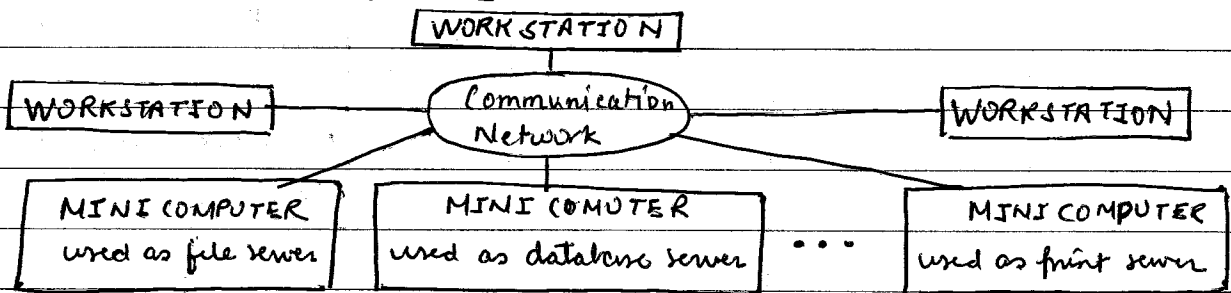
myCOMPANION

(2) Workstation model -



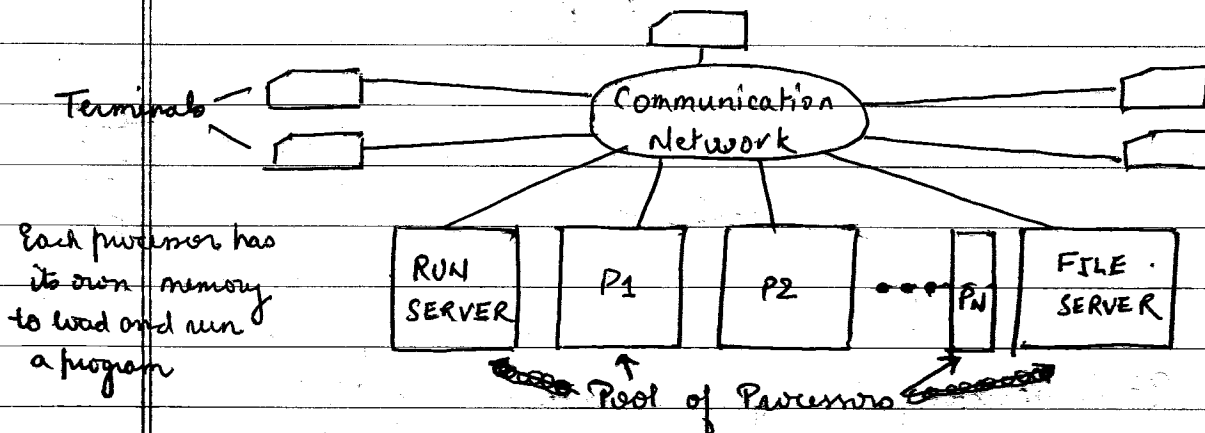
When the user's workstation doesn't have sufficient processing power, it transfers one or more of the processes from the user's workstation to some other workstation that is currently idle and gets the process executed there, and finally the result of execution is returned to the user's workstation. Eg - Sprite system

(3) Workstation-server Model -



Those workstations that do not have disk that is diskless workstations use minicomputers for different types of services. Eg - V-system.

(4) Processor-Pool Model -



Each processor has its own memory to load and run a program

When a very large amount of computing power is needed in short time then in this model, the processors are pooled together to be shared by the users as needed.

Run server - allocates & manages the processors in the pool to different users on a demand basis.

my companion Eg - Amoeba, Plan 9



(5) Hybrid Model -

Combining the workstation-server model and processor-pool model generates a model known as Hybrid model.

Many users or group of users needing massive computation that hybrid model is used.

Workstation-server model is ideal for simple usage whereas processor pool model is ideal for massive computation.

(8) Issues in designing distributed system -

(1) Transparency - Issues like location, migration, replication, concurrency and parallelism transparencies.

Location Transparency - Users cannot tell where resources are located

Migration Transparency - Resources can move at will without changing their names.

Replication Transparency - Users cannot tell how many copies exist.

Concurrency Transparency - Multiple users can share resources automatically.

Parallelism Transparency - Activities can happen in parallel without users knowing.

(2) Flexibility - form minimal services - ^{MAIN} [Ease of modification
Ease of enhancement]

(i) An interprocess communication mechanism

(ii) Some memory management

(iii) A small amount of low-level process management and scheduling

(iv) Low-level input/output.

(3) Reliability - Various aspects of reliability are availability, security and fault tolerance.

(4) Performance - Various performance metrics are response time, throughput, system utilization and amount of network capacity consumed.

(5) Scalability - Resources should not run out, handles significant increase in number of users, avoid centralized components, tables and algorithms.