

## B.E.

Eighth Semester Examination, Dec.-2006

### DISTRIBUTED SYSTEM

**Note :** Attempt any five questions.

**Q. 1. What are distributed systems? What are the objectives and goals of these systems? How are these systems helpful? Illustrate.**

**Ans. Distributed systems :** A non centralized network consisting of numerous computers that can communicate with one another that appear to users as a single computer.

**Advantage/goal of system :**

**1. Openness :** It is a property of distributed systems much as each subsystem is continually open to interaction with other systems.

**2. Scalability :** One can easily alternate it means easy to accommodate with users, resources and other dimensions there are reliable, and scalable according to loads, geographic and administrative scalability etc.

Load scalability means it is easy for us to expand and contract its resources pool to a conduct heavier or higher loads.

Geographic scalability means are that maintains its unfutures and arability etc.

**Administrative scalability :** No matter how many different organisation need to share a single distributed system, it should still be easy to use & manage.

**Q. 2. Explain the following :**

**(a) ATM Networks**

**(b) UNIX emulation in MACH.**

**Ans. (a) ATM (Asynchronous Transfer Mode) :** A network technology based on transferring data is cells or packets of a fixed size. The cell used with ATM is relatively small compared to units used with other technologies. The small constant cell size allows ATM equiv. to transmit video, audio and computer data over the same network, and assume that no single type of data page the line. ATM creates a fixed channel or route between two points whenever data X for begins. It differs from TCP/IP.

UNIX emulation has also been implemented on the V kernel systems.

Strict binary compatibility require that are binary files conducted to run on a conventional implementation of a remains of UNIX.

This system required following requirements :

(i) Address space layout

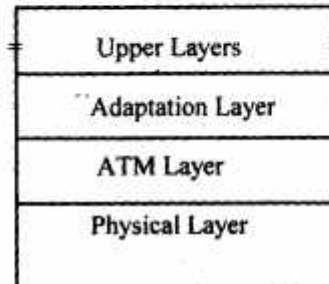
(ii) System call processing.

(i) The emulation must provides the regains expected by the program. If the code is nonrelatable, the m/c instructions answer that regions such as the program text and head occur certain expected address ranges.

**Advantage of ATM are as :**

- A single Network can now be used to transport an arbitrary mix of voice, data broadcast television, videotapes, radio etc.
- It is best for home & business, only one wire (fiber), use for all communication & information needs.

**ATM Reference Model :**



**(i) ATM Physical Layer :**

This layer based on SONET-Synchronous optical network, it put its cells into the payload portion of SONET frames In Europe are called it SDH-Synchronous Digital Hierarchy, that is clearly related to SONET.

**(ii) ATM Layer :**

Here we used 48 byte data field & 63-byte cell is used have.

**(iii) ATM Adaptation Layer :**

Originally adoption layer were defined for 4 classes of traffic :

1. Constant bit rate traffic (video + audio)
2. Variable bit rate traffic
3. Connection-oriented data
4. Connection data.

**Q. 3. (a) Define a distributed system? What are advantages of D.S. over centralized systems.**

**Ans.** A collection of independent computes that appear to the users of system as a single computer.

**First aspect deals with hardware :** The m/s are autonomous.

**Second deal with software :** The uses think system as a single computer. Both are essential.

**Advantages of Distributed over centralized system :**

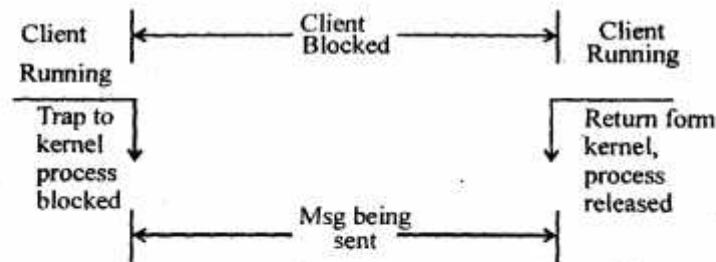
1. **Economic :** Microprocessor offers a better price/performance ratio than main frames.
2. **Speed :** A distributed systems have more speed than that of main frame.
3. **Inherent Distribution :** Some applications involve spatially separated m/cs.
4. **Reliability :** If one m/c cracks, other m/c can handle its load.
5. **Incremental Growth :** Computing power can be added in small increments.

**Q. 3. (b) Briefly discuss a blocking primitive for sending messages in client server model.**

**Ans.** Blocking primitive for sending messages in client server model.

The blocking primitive are also called synchronous primitives. When a process call send it specifies a destination while the message is being sent, the sending process is blocked. The information following the call to send is not executed until the message has been completely sent.

Similarly, a call to receive does not return control until a message has been actually received & put in the message buffer pointer by the parameter. The process remain suspended in receive until a message arrives, even if it takes hours. In same system, the receiver can specify from when it wishes to receive, in which case it remains blocked until a message from the sender arrives.



**Fig. (a) Blocking Primitive**

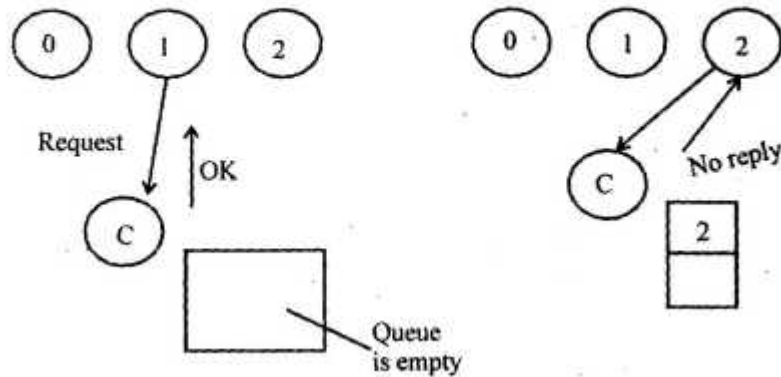
**Q. 4. Discuss any two algorithms for ensuring the mutual exclusion in distributed systems.**

**Ans.** The following two algorithms we discussed as :

1. **A Centralized Algorithm :** The most straight forward way to achieve mutual exclusion in a distributed system is to simulate how it is done in a one-processor system.

One process is elected as the coordinator, when a process want to enter in critical region it sends a request message to the coordinator stating which critical region it wants to enter and asking for permission. If No. one wants then coordinator send back a reply granting permission.

As shown in figure :



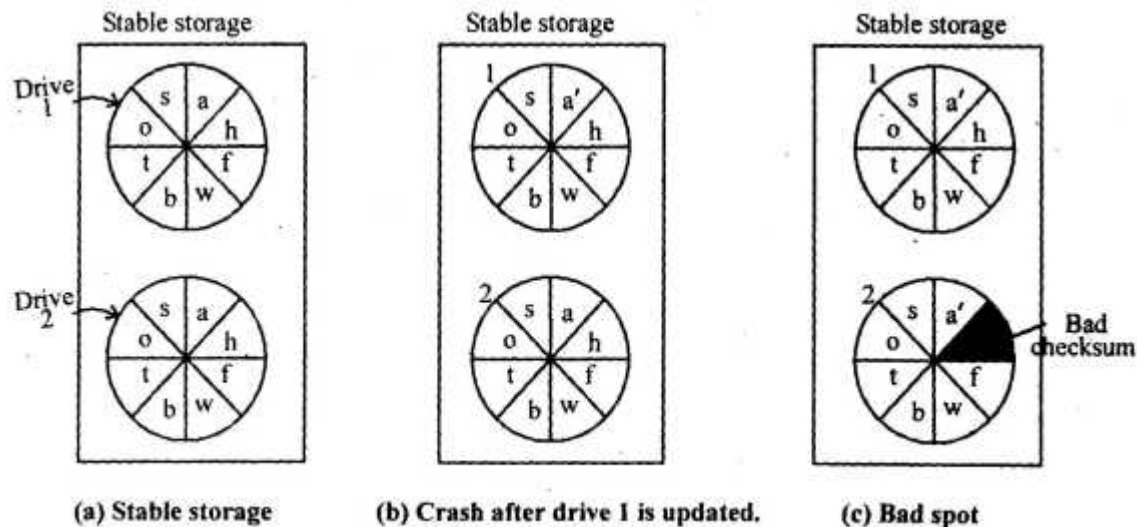
## 2. Distributed Algorithm :

**The algorithm works as follows :** When a process wants to enter a critical region, it builds a message containing the name of the critical region it wants to enter, it builds a message containing the name of the critical region it wants to enter and send the message to all other processes.

### Q. 5. (a) What is stable storage? How it is important for distributed systems?

**Ans.** Storage comes in 3 categories first we have ordinary RAM memory, which is wiped out when power fails or a m/c crashes. Next we have disk storage, which services CPU failures but which can be lost in disk head crashes.

Finally, we have stable storage, which is designed to service anything except major calamities such as floods & earthquakes. Stable storage can be implemented with a pair of ordinary disks.





**Q. 5. (b) Discuss Bully's Algorithm for choosing a co-ordinator process.**

**Ans.** When a process notices that the coordinator is No. larger responding to Requests, it initiate an election.

A process, P holds an election as follows :

1. P send an ELECTION message to all processes with higher No's.
2. If No. are responds, P wins the election and becomes coordinator.
3. If one of higher-ups answers, it takes over. P's job is done.

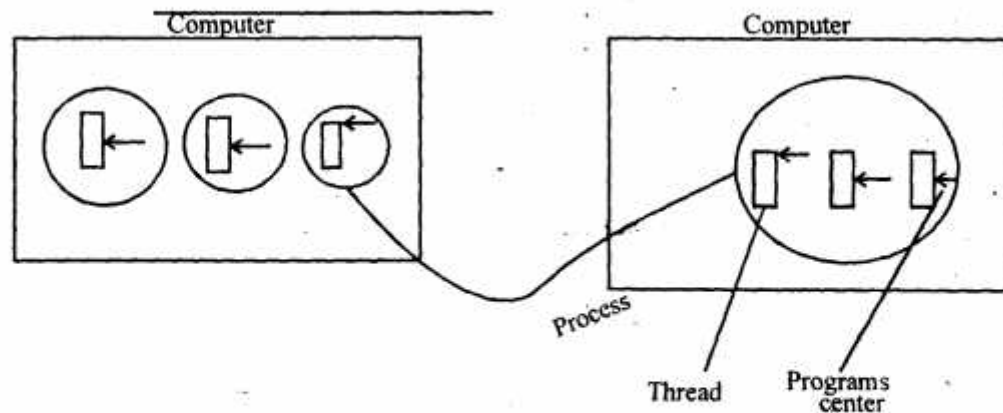
At any movement, a process can get an ELECTION message from are of its covert numbered colleagues. When such a message back to the sender to indicate that the is alive and will take over. The receiver then holds an election, unless it is already holding one. Eventually all process give up but one, and that one is the new coordinator.

**Q. 6. (a) What do you mean by a thread? How threads are implemented in user's space?**

**Ans.** Thread is a light weight process that share an address space with one or more other threads.

**Implementation of Threads :**

The threads are as given in following figures :



**Fig. Three processes with one thread each**

Each process has its own program counter, its own run stack, its own register set, and its own address space. The processes have nothing to do with each other, except that they may be able to communicate through the

systems interprocess communication primitives such as semaphores, monitors or messages. The following items are used for per thread items.

Per thread items.
Program Counter
Stack
Register Set
Child Threads
State

Threads can be in any one of the following states :

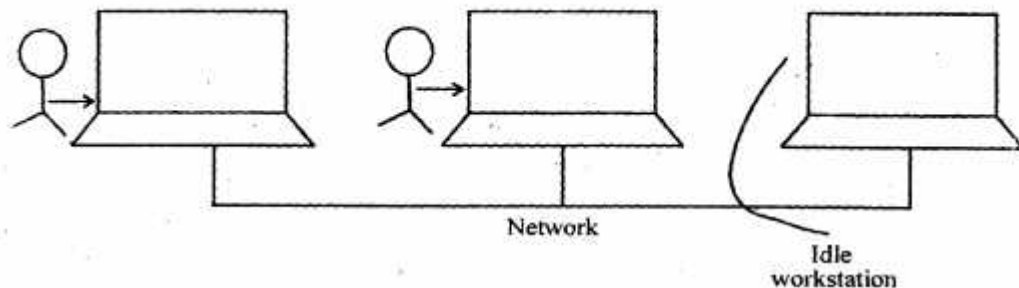
- Running
- Blocked
- Ready
- Blocked/Terminated.

A running thread which currently has the CPU and is active. A blocked thread is waiting for another thread to unblock it (on semaphore). A ready thread is scheduled to run and it will take its turn as soon as possible.

**Q. 6. (b) Discuss the workstation model of system briefly.**

**Ans. Work station model :**

**Workstation model is straight forward :** The system consists of workstations highland personal computers, called throughout a building or campus & connected by a high speed LAN, as shown in fig. 1.



In some systems, the workstation have local disks & in others they level. The latter was universally called. Diskless workstations, but the former are variously known as Diskless workstations or dusky workstation. If the station are workless, then we can use remote sever.

**Advantages :** Diskless workstation are popular is colleagues, they are cheap also sending of message is assumed to be relate that is every message is acknowledge.

When a process receive a request process message from another process, the action it takes depends on its state w.r.t. the critical region named is the message. There cases are :

1. If the receiver is not in the critical region & does not want to enter it, if sends back on OK message to the sender.
2. It the receiver is already in the critical region, it does not reply. Instead if queue the request.
3. If the receiver wants to enter the critical region but has not yet done so, it compares the timestamp in the incoming message with the one contained in the message that it has sent everyone the lowest one wins. If the incoming message is lower, the receiver sends back are OK.

**Q. 7. (a) Discuss page based distributed shared memory model briefly.**

**Ans.** The page based distributed shared memory model is an given : There system are built on top of multi computers that is, processors connected by a specialised message passing N/ws, workstations on a LAN or similar designs. The essential element have is that No. processor can directly access any other processors' memory. Such system are sometimes called No. RMP. Non-Remote Memory Access, system to constant them with NUMA systems.

The main difference between NUMA and NORMA is that in the former, every processor, can directly reference every word is global address space just by reading or writing it.

Pages can be randemly distributed among memories without effecting the results that programs give. When a processor references a remote page, the system has the option of fetching it or using it remotely. The decisions affects the performance but not correctness. NUMA m/cs are true multiprocessors the n/w allows every processor to reference every word is the address space without s/w intervention.

**Q. 7. (b) Write a short note on Non Uniform Memory Access.**

**Ans.** To reduce the cost by giving to hierarchical systems, some memory is associated with each CPU. Each CPU can access its run local memory quickly, but accessing anybody else's many is slower. This design give rises to what is known as NUMA-Non-Uniform Memory Access.

NUMA m/c have better average access time than m/c based on merge new.

**Disadvantage :**

1. NUMA m/c require complex algorithms for good s/w placement.
2. NUMA switches which are crosstans are very expensive and large average n/w & are experience & slew.

The conclusion is very clean : Building a larger, tightly-compiled, shared memory multiprocessor is possible but it is difficult & experience.

**Q. 8. Briefly discuss the following :**

**(i) Unit emulation in MACH.**

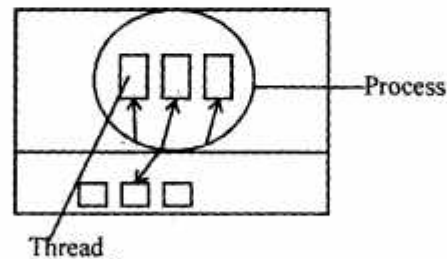
**(ii) Implementation of C threads in MACH.**

**Ans. (i) Goals of Mach :**

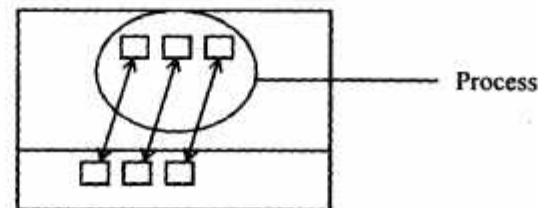
1. Providing a base for building other operating system (UNIX).
2. Supporting large sparse address space.
3. Allowing transparent access to N/w resources.
4. Exploiting parallelism is both the system & application.
5. Making Mach portable to a large collection of m/cs.

**(ii) Implementation of C threads in MACH :**

This approach time shared all the C threads over one kernel thread as given below figure.

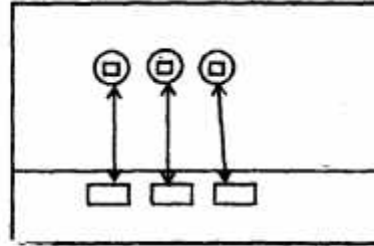


**Fig. (a) All C threads use ONE kernel thread**

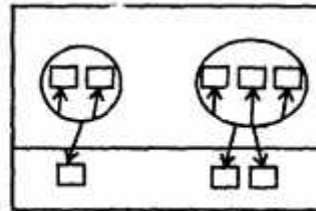


**Fig. (b) Each C thread has its own kernel thread**





**Fig. (c) Each C thread has its own single-threaded process**



**Fig. (d) Arbitrary mapping of user threads to kernel threads**

The main practical value of the first approach is that because there is No true parallesosm, successive run gives reproductive results, easy debug, third one is not normally used.

Truth one, although the most complicated, give greater flexibility and it normally used of present.