Anjuman-i-islam's

M.H. Saboo Siddik College of Engineering

8, Saboo Siddik Polytechnic Road, Byculla, Mumbai 400 008.

Department of Computer Engineering

(Affiliated to University of Mumbai)

Seat No.	7281020		
Subject Name	Distributed computing	**Date of Exam**	03 /06/2021
Subject Code	52752	**Semester**	VIII
Invigilator Signature		**Total No. of Pages**	15.
Student Signature			

Q2 A. Solve any Two.

1.) Write short note on - Group Communication.

→ - Group ~~communication~~ is a group of users sharing some common interest.

- In Group communication, messages sent to a group of processes will deliver to all the members of the group.
- 3 Types of group communication are as follows:

1. ONE-TO-MANY-COMMUNICATION:

- There exist a single sender & multiple recievers.
- Also called as multicast communication.
- It includes.

a. group management:
- Two types of groups i.e. closed & Open groups.
- where based on where & how they can send the message they are divided.

b. Group addressing: Multicast address also known as the N/w address is used.

c. Buffered & Non Buffered Multicast facilities are used according to the type of N/w.

d. Semantics:
- Send-to-all, o Bulletin semantics.

e. flexibility in reliability
- 0 reliable : No response expected
- 1 reliable : One ack expected.
- M out of N: depends on sender

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

2. MANY-TO-ONE-COMMUNICATION.
- Only one receiver at any given time.
- Reciever can be.
  a. Selective: Specifies the unique sender.
  b. Non-Selective: The receiver is ready to accept messages from any where.

3. MANY-TO-MANY COMMUNICATION:
- Multiple senders send messages to multiple receiver.
- Important issue is ordered message delivery.
- It includes.
  a. No ordering: In this semantic the ordering is not maintained on the receiver end.
  b. Absolute Ordering:
     This semantics ensures that all messages are delivered to all receiver process in the exact order in which they were sent.
  c. Consistent ordering:
     This semantics make sures that all messages are delivered to all receiver process in same order.
  d. Casual Ordering:
     This semantics ensures that if the event of sending one message is casually related to the event of sending another message, the two messages are delivered to all receivers in correct order

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

(iii) Short Note on NFS.

→ - NFS stands for Network File system.
- NFS is a platform independent remote file system technology created by sun microsystems.
- It is a client server system that provides shared file storage for client across the network.
- Implemented using RPC protocol.
- It is available through the N/w via a virtual file system.
- It is generally implemented in layered architecture on the Top of TCP/IP layer.
- Allows an application to access files on remote hosts.

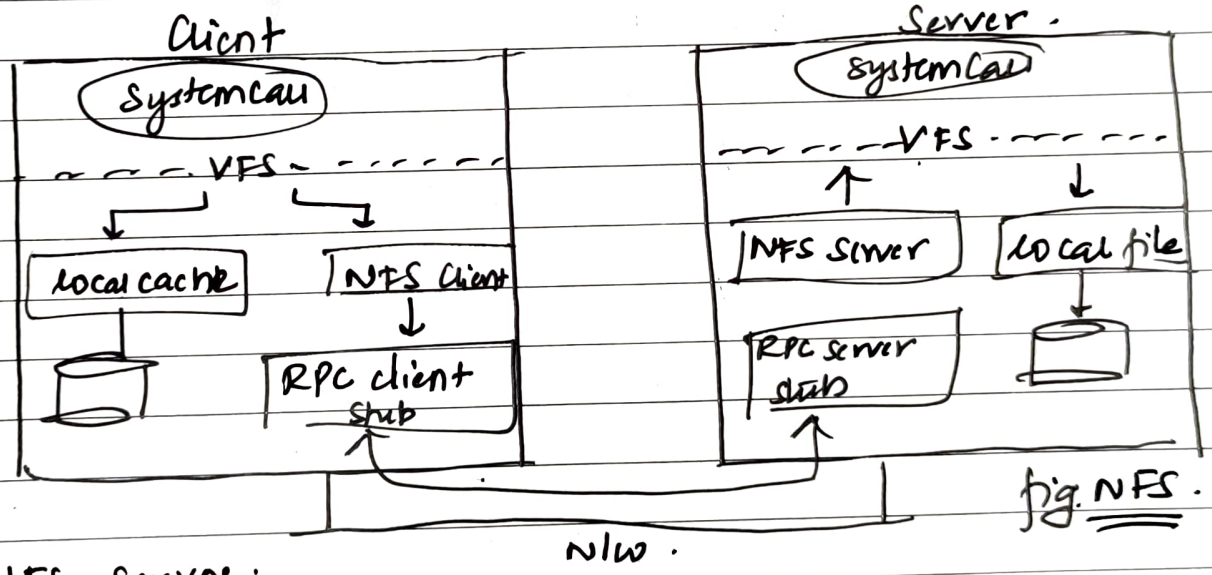


fig. NFS.

- NFS Servers:
- They share the files. The remote file servers are mounted on these servers.

- NFS Simplifies Management:
- Instead of creating multiple directories & duplicates it creates one copy that is accessible to all in the system.
- Simplify backup procedures. Only the server disk backup is needed

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

NFS clients: computers that share files.
- NFS uses a mixture of kernel support & user-space daemons on the client side.
- Multiple client can mount the same remote file systems so that the user can share files.

- Protocol: It uses. Remote Procedure Call. mechanisms.
  - It uses stateless protocol which simplifies the over all working as previous state information is not required.
  - It also simplify crash recovery.

- Goals:
  1. compatability: same semantics to all the system.
  2. Easy deployment: Implementation should be easy.
  3. Machine & OS independence: NFS clients can also run on non-unix platforms.
  4. Efficient: It should be good enough to satisfy the needs. as fast as the local File system.

(B) Solve any 1:

(ii) Describe file caching schemes in brief.

→
- In order to improve the IO performance in centralized time sharing systems, file caching is implemented.
- main goal of a file caching system is to hold the recently accessed data in the main memory.
- so repeated access can be efficiently handled.

Seat No.	7281020	Semester	VIII
Subject name & Code	OC-52752	Student Signature	

On file caching we used the following for propogation of modifications.

a. write through schemes.
when cache entry is modified, the new value is immediately sent to the server for updating the master copy of file.

b. Delayed-write schemes.
- when cache entry is modified it remains in the cache only.
- client system just makes a note of the cache entry that has been created.
- All the cache entries are collected & sent to the server later on

a. write on ejection from cache:
- Modified data in the cache is sent to the server when the cache replacement policy has decided to eject it from client side

b. Periodic write:
- Cache is scanned on regular intervals. if modified the last scan is sent to the server.

c. write on close:
- modification to cache is sent to the server when the corresponding file is closed by the client.

There are certain validation schemes available too when it comes to file caching.
- If file resides on the multiple node then it becomes necessary for the server that is the client copy consistent or it has been modified. Here validation is required.
- validation is done in two scheme.

Seat No.	7281020	Semester	VIII
Subject name & Code	OC-52752.	Student Signature	

- **Client initiated approach.**

-Here client contacts the server that the locally cached data is consistent with the copy present at the server or not.

- The semantics depends on frequency of check.

a. check before every access: not preferred as the server will be contacted every time.

b. periodic check: checks after a finite interval (pre-defined)

c. check on file open: when file is opened then check.

- **Server initiated approach.**

-Here the server tracks and modifies itself as and when it recieves a certain request.

A Typical file caching system could be shown as follows.

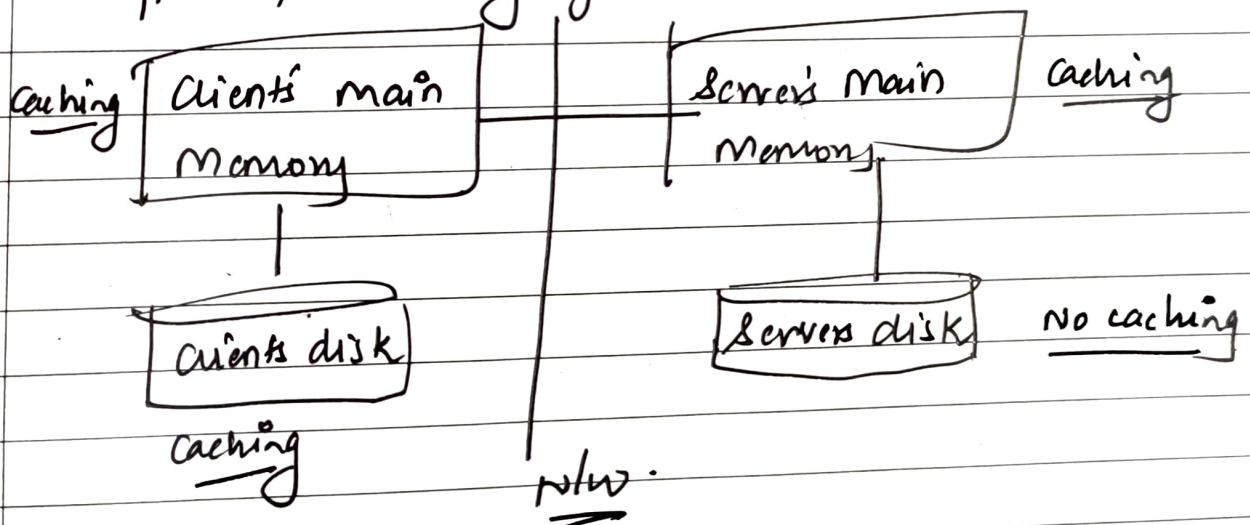


fig:- File caching system.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

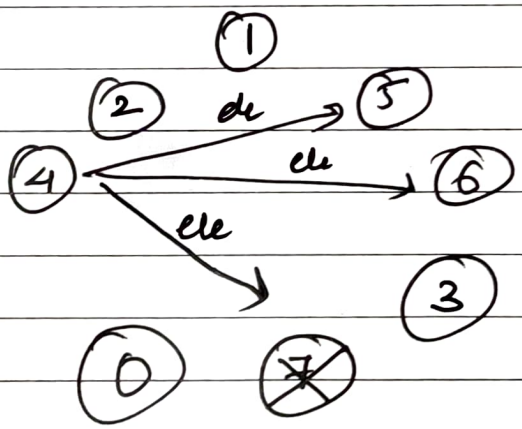
Q3. Solve the following questions

A. Solve any two

(i) Discuss the Bully algorithm with example. State its advantages & disadvantages

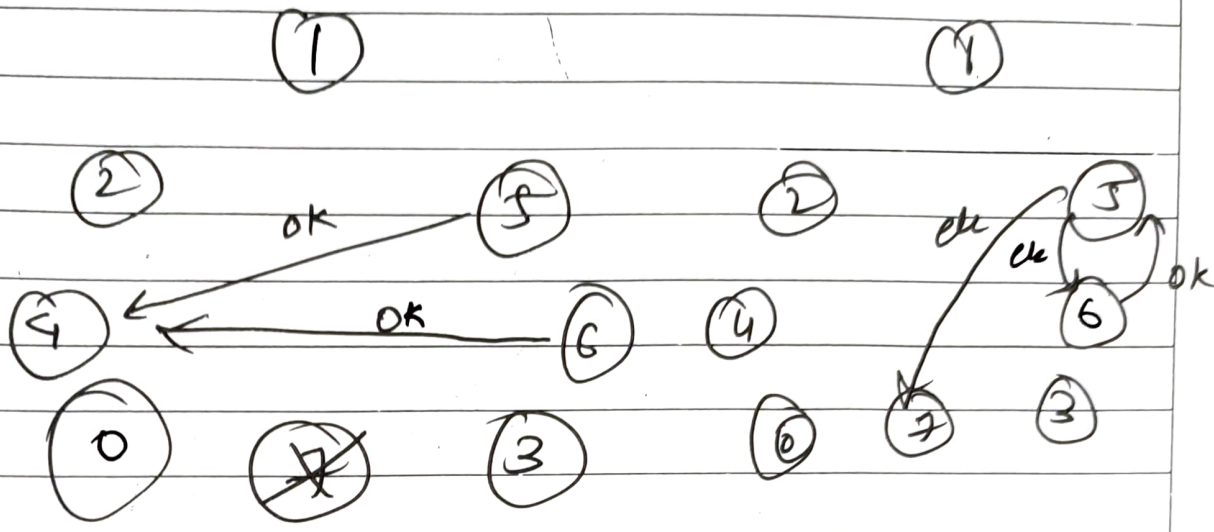
- Bully algorithm was proposed by Garcia-Molina.
- It is based on assumptions that.
  a. Each process has a unique id.
  b. Each process knows the no of remaining processes.
  c. Scenarios are synchronous in nature.
  d. Highest process with no or id of the process is considered as the coordinator.

Consider the following example.

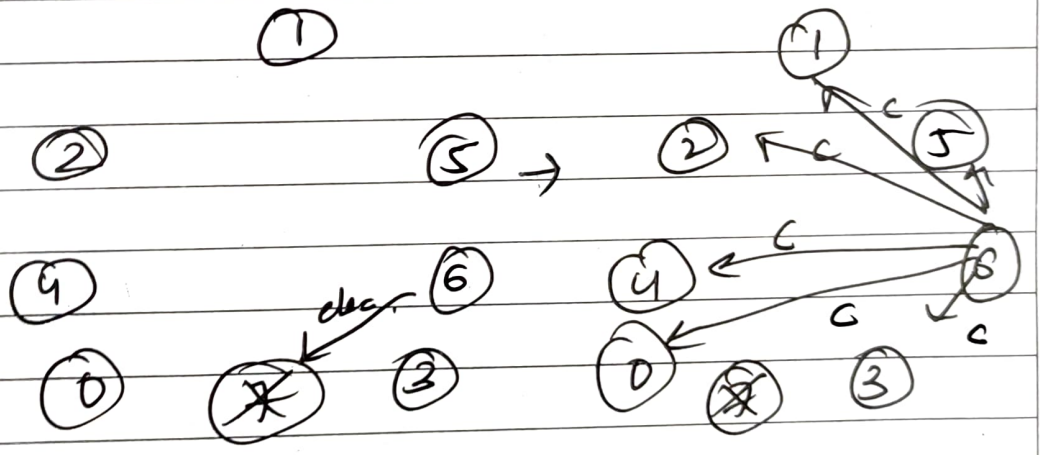


- Process (4) calls an election. when the coordinator is no longer responding.
- It sends election message to all higher-no process as shown.
- If no process responds process 4 becomes coordinator
- But if a process responds. then 4's role is terminated as the coordinator.
- Suppose 5 & 6 responded.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC - 527520	Student Signature	



- Now 5 wants to be coordinator it sends election message to higher process as show in figure
- But 6 responds and then 6 transmits the election message again as the node 7 is dead 6 becomes the coordinator & broadcasts that it is the coordinator

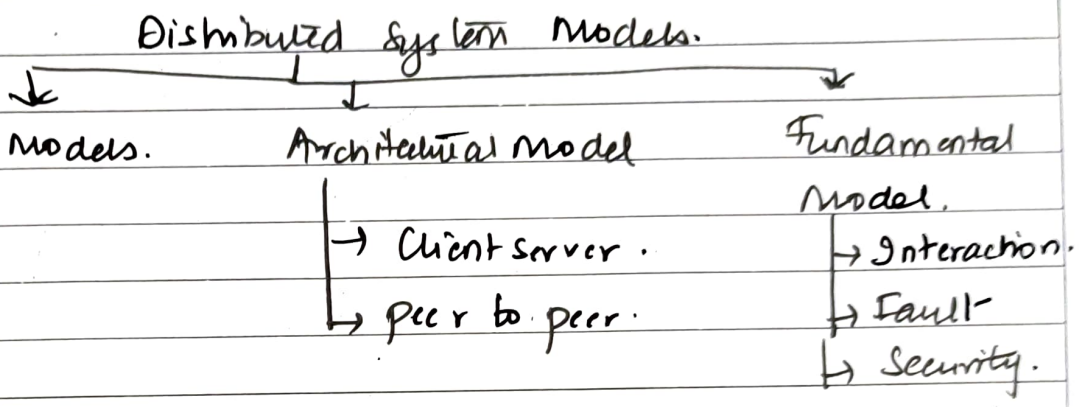


Advantage	Disadvantage
- Simple to implement. - less probability of crashing - only process with high priority are involved in election. so its quick.	- No of stages to decide a new leader is huge. - number of broadcast message is more reducing the overall bandwidth. - Increased n/w traffic.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752.	Student Signature	

(ii) Different models of the Dismbuted system? Explain.

→ Following diagram shows the various models of the dismbuted system.



(1) Physical model.

- Representation of the underlying hardware of the dismbuted system.
- Hardware composition is captured.
- 3 generations.
  - (i) Early dismbuted systems:
    - Emerged in 1970's & 1980's
    - consists of 10 to 100 nodes connected by LAN.
    - local printer; file systems etc.
  - (ii) Internet scale dismbuted systems.
    - Emerged in 1990's.
    - High no of nodes across organizations.
    - Highly heterogenous.
  - (iii) Contemporary Dismbuted Systems
    - ultra large scale
    - location independent Node.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-50752	Student Signature	

(2) Architectural model.

- Depicts the way in which the components of the system are interacting with each other.
- It simplifies & abstracts the functionality.
- Two types
  1. Client server model.
     - processes are divided as client & server processes.
     - server implements a specific service for eg. file system or database services.
     - usually based on the simple request / reply protocol.

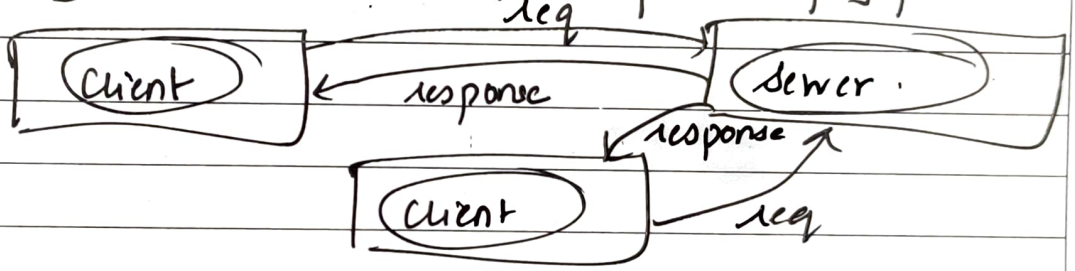


fig: client server model

2. Peer to Peer model
   - No centralized control in this mode.
   - Each node can either be a client or a server based on its requirement.
   - If its requesting it is client; if it is providing it is known as server.
   - All the processes has same capabilities & responsibilities in this model

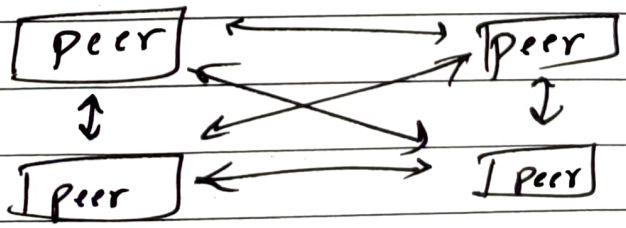


fig: Peer-to-Peer model

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

3) Fundamental Model.

- To make all the explicit & relevant assumptions about the systems used for modelling.

(1) Interaction model:

- Deals with performance & handling time in a Distributed system.
- Broadly divided into 2 categories Synchronous & A Synchronous.
- In synchronous bounds like upperbound on message, ordered message delivery, local step Based execution are defined.
- In Asynchronous there are no bounds defined explicitly.

(2) Fault model.

- Failures can occur in both process & communications channels.
- Reasons can be both S/w & H/w faults.
- failure models defines how failures occur in system.

a. Omission fault: when a process or channel fails to do something that is expected to do.

b. Arbitrary fault or Byzantine fault:
   - Any error can occur and cause a fault.
   - Broadly followed fault model.

c. Timing fault:
   - Based on the synchronizational fault that occurs in the system

(3) Security model:

- Based on Trust worthiness.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

- Each component is defined a specific role like trusted client, trusted servers, trusted administrators etc.
- Used for
  - protecting comm^n channel
  - protecting interactions.
  - access to objects is protected

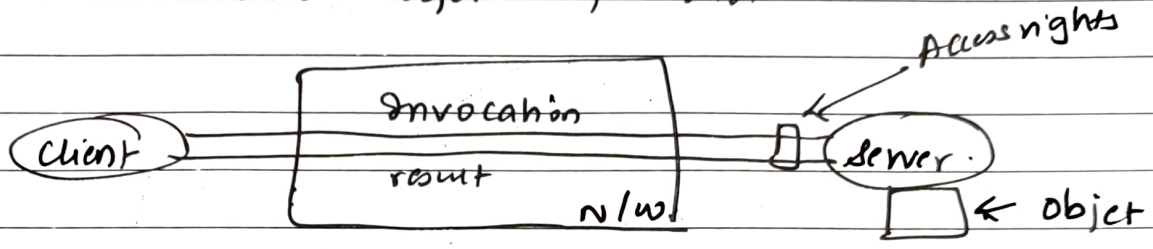


fig: Protecting objects.

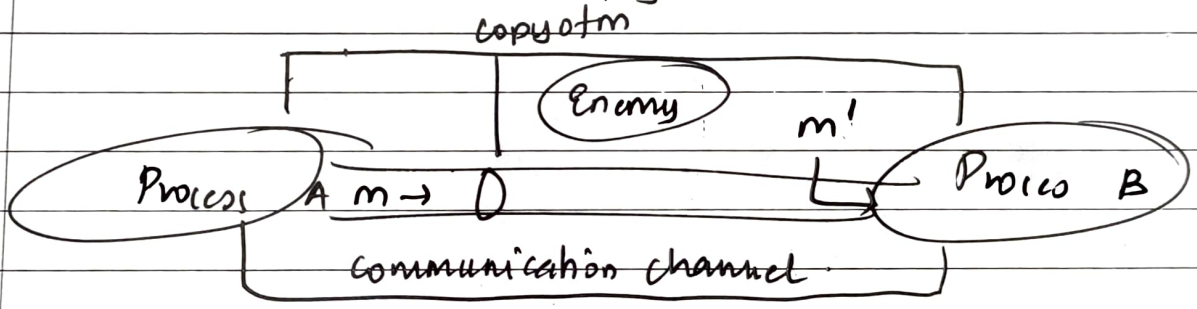


fig: protecting channel.

B. Solve any one.

(ii) Define Remote Procedure Call (RPC)? Describe the working of RPC in detail

→ - RPC stands for Remote Procedure call.
- Also known as Remote functional call or Remote Sub-routine call.
- Used for designing several distributed Applications.
- features of RPC are
  simple syntax; familiar semantics, Ease of use, generality, efficiency.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

- Generic RPC Model.

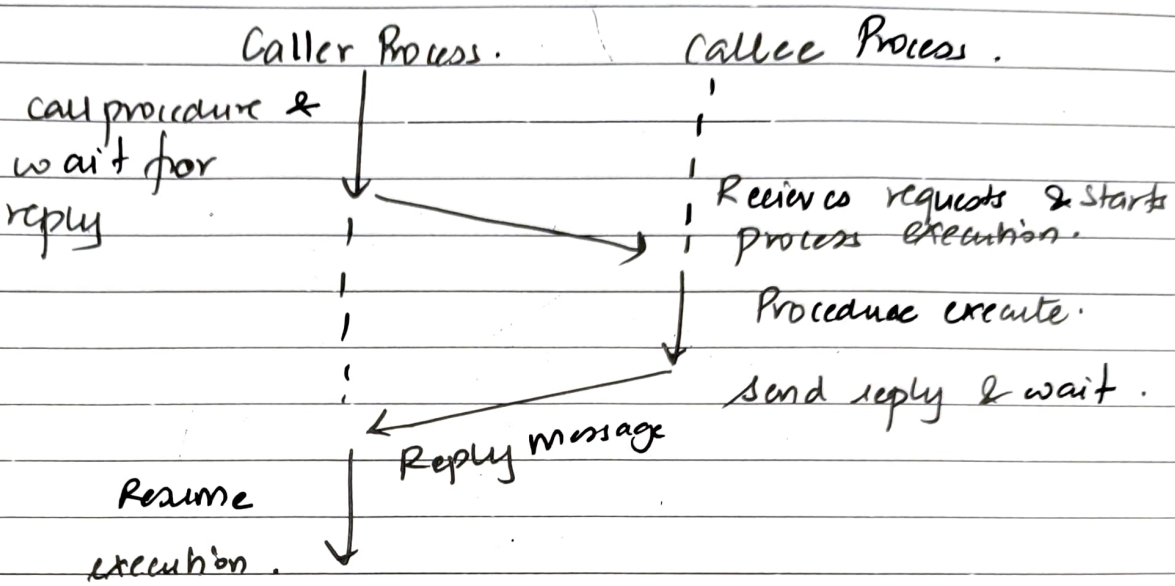


fig: Generic RPC Model

- The basic idea of RPC is to make a remote procedure call look transparent.
- It should not be aware that the procedure is executing on a seperate machine.
- Transperency is achieved in RPC.
- Caller sends a call (request) message to the callee (Server process) & wait for its reply.
- The server process executes the procedure execution & returns the result.
- Once the reply message is recieved the result is extracted & the caller's process is resumed execution.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC-52752	Student Signature	

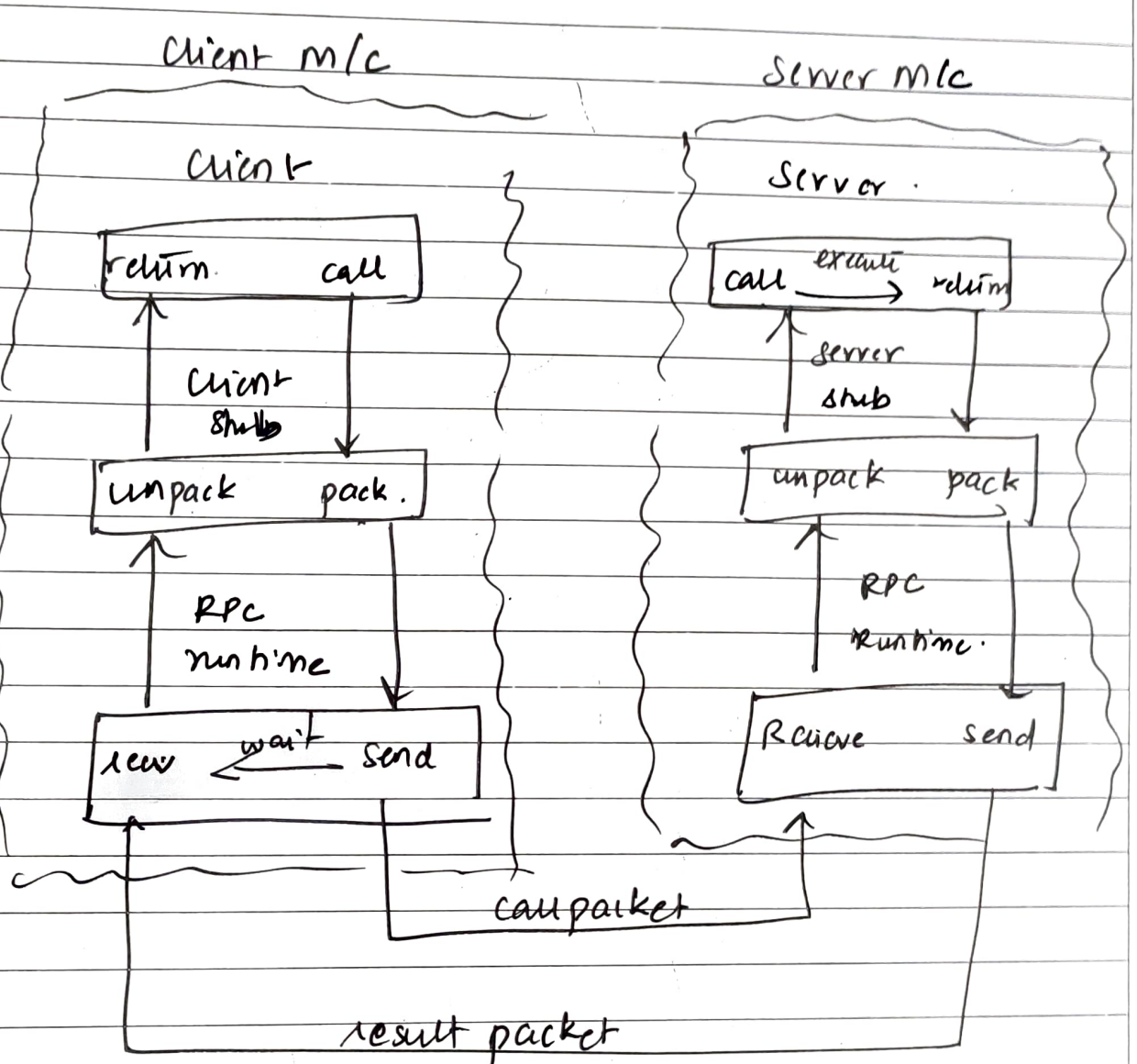


fig: RPC model

- Implementation of RPC

- To achieve the goal of semantic transperency the implementation of an RPC mechanism is based on the stubs.
- Stubs provide a local procedure call abstraction.
- Implementation usually involve 5 elements.

(1) Client:- initiates RPC.

Seat No.	7281020	Semester	VIII
Subject name & Code	DC- 57752	Student Signature	

(ii) Client stub: packing the arguements in procedure

(iii) RPC runtime:- handles transmission of messages.

(iv) Server stub: The job of server stub is similar to client

(v) Server:- Execute & returns the result back to client

- Advantages of RPC:

- RPC supports process oriented & thread oriented models
- Internal message passing mechanism is hidden from the user.
- Effort to rewrite & redevelop is minimum.
- Can be used in distributed environments

- Disadvantages of RPC

- less flexible because of strict h/w architecture
- Increase in cost because of the remote procedure call.
- Not a standard, it is implemented in different ways in diff architecture.