VIT | Vidyalankar Institute of Technology

End Semester Examination (MU)

Evaluation of University Question Paper Solution Prepared by Faculty

Subject (Write in full) : Distributed computing (Regular / KT)

Exam : May 2024 / Nov. 20 ___ Exam Date : 15/05/24 Q. Paper Code : C 801

Department : CMPN Year : ~~FE/SE/TE~~/BE/ME/MMS Semester : 08 Scheme : CBSGS/CBCS

Handwritten Solution Prepared by : Umesh Kulkarni

Name of the Subject Cluster :

Name of the Cluster Mentor / Assessor :

1st Assessment :

	Marks Obtained				Total
				(f)	

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Q.1 A How does middleware handle communication failure and ensure message delivery in distributed environment.

Soln ~~Message~~ ~~Po.~~ Middle ware communicates ~~Middle~~ play a usital role in handling fault, by various serval mechansium.

- Message queuing.
2) Realiable message protocol
3) Transaction managment
Retry mechansium
Error Handling
Load balancing
~~fault~~ tolerance
Redenancy.

Q2 B. Explain goals of distributed system and how they differ from

solu:- the primary goal of distributed system

- scalibility :-
- Reability
- ~~Availability~~
- Performance
- consistency
- Partition Tolerance.

Total Marks of Question no.		Examiner
		Moderator
		Re-Assessor

Space for Marks	Question No.	START WRITING HERE

Difference from centralized system.

⇒ single point of failure :- - centralized systems have a single point of failure. a central server or database, which can ~~load~~ lead to system wide . outages If it fails. - Distributed system distribute data and computation across- multiple nodes reducing the risk of a single point of failure

⇒ Scalability Limination : Centralized system may face scalability Liminations. as they rely on a single server. or database to ~~have~~ handle all request. Distributed system can scale horizontaly by adding more nodes, allowing to ~~handl~~ handle increase load more effeccitvely.

⇒ Reability and Avaibility
Centralize system is less reability and availbility compared -

⇒ performance - ~~Stroggle~~ struggle to handle large volume of data or requests. efficiently, especially of load increase

⇒ consistency challenges :- - Archieving consistency, in relatively straightforward

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

as there is a single source of truth
maintaining consistency across multiple
nodes, is more due to factors like a
network latency and concurrency.

Q.1. c) Describe the client-server and peer-to-peer model of distributed system

→ Client-server model :-
In the client-server model, the system is organized into two type of node.
client & server.

1) client - The client is typically a user facing device or application that request service or resource from a server. client communication by sending request to server

2) server :-
The server is a decidated node of service that response to client request by providing services or resource, server

Characterise of client-server model.
1) centralised control
2) scability.
3) Reability & availability.
4) clear separation of concerns

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

peer-to-peer (P2P) model.

- characteristics of P2P model.

1. Decentralized
2. Resource sharing.
3. Autonomy
4. Dynamic Topology
5. Scability challenges.

Q2-A Discuss the significance of group communic

→ significant of group communication

1. coordinated interaction.
2. fault Tolerance
3. scability.
4. Consistency and Reablity
5. Distributed computing Paradigms

Decentralized control.

Adventage of Group communication

- Efficency
- Reability
- flexibility
- fault Tolerance

Group commution is a fundamental building blocks, of distributed system enabling, efficient coordination fault tolerance scability.

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Q B- Explain the concept of IPC

IPC is a mechansium that allows different process to communicate with each other and synchronize their action in a shared environment. such as a multitasking operating system or distributed system: IPC enables process to exchange data, coordinate their acitivities,

key componets of IPC

- process IPC involves communication between different processes running concurrently on a computer system,

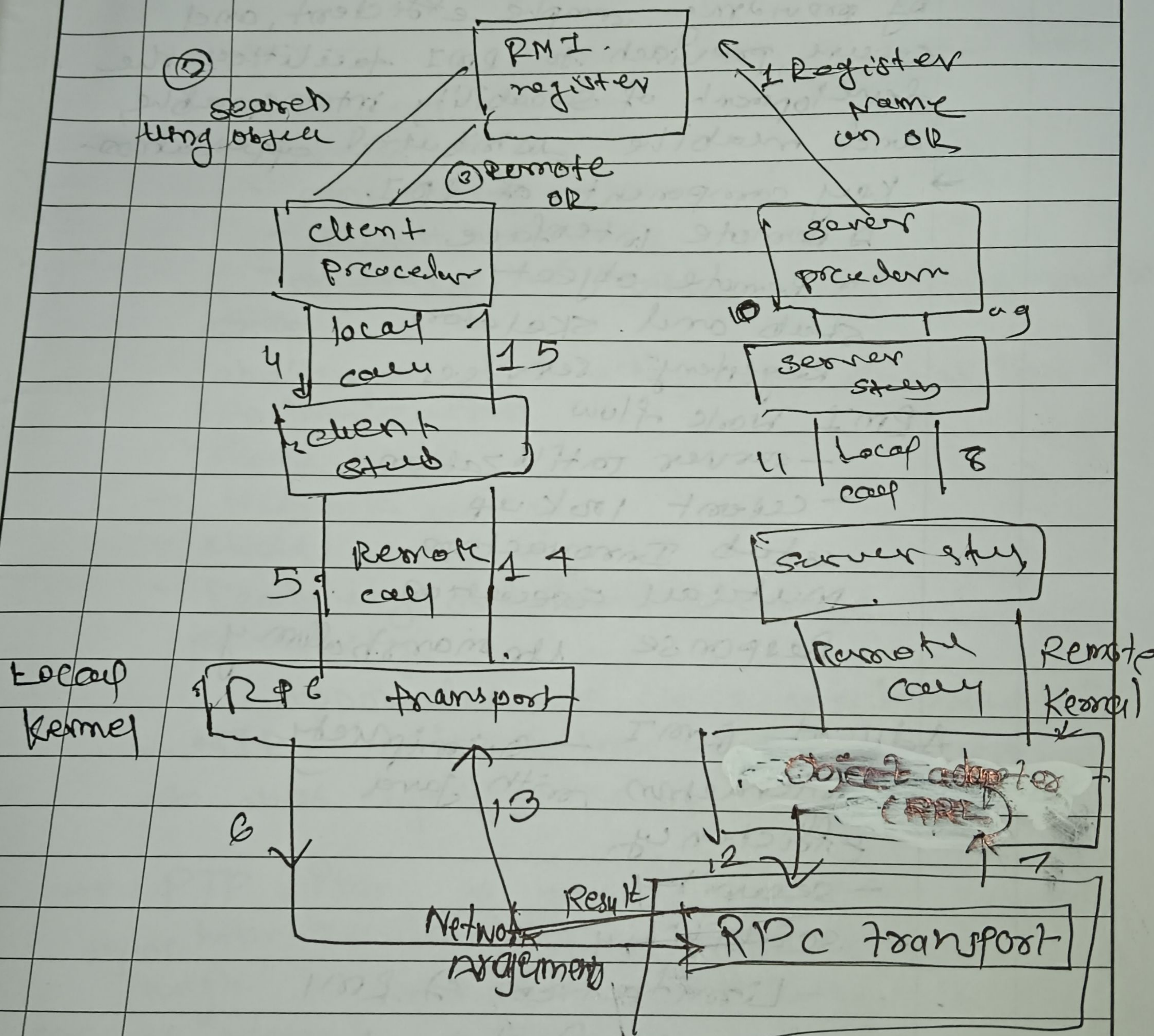
→ Communication channels-
these channel can take various forms including shared memory, message passing

→ synchronizing:-
Types of IPC
- shared memory
- message passing
- pipes & fifo
- Socket
- signals.

Importance of IPC
- concurrency
- Resource sharing
- Modularity
- Distributed Computing.

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE
	Q2C	Discuss RMI.



RMI

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

RMI is a powerful mechanism for building system in java, enabling seamless communication and interaction process by providing simple efficient, and secure, proviach to RMI facilitles the development of scabiliy, interoprable, and reabile distributed application.

→ key components of RMI
- Remote interface
- Remote object
- stub and skeleton
- Registery services

RMI work flow
- server Intilization
- client look up
- stub Innovaction
- Mutual Execution
- Response Unmarshalling

Adv of RMI — simplicity
- Interaction with Java
- Efficiency
- security
- scability
- Limitation of RMI
- fore wall Restrication
- performance overhead

use case — middleware, Enterprise App

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

START WRITING HERE

Space for Marks	Question No.

Q3A. Discuss the importance of clock synchronization in distributed system explain challenges.

Solution — Important of clock synchronization
~~clock~~ event ordering
- coordination
- consistency protocols
- fault tolerance
- security and ~~[illegible]~~ Authentication.

~~[illegible]~~

challenges ~~of~~ in achieving Global Time synchronization.
- clock drift
- network devices
- clock skews
- Reability and fault tolerance
scability

Approaches to clock synchronization
- NTP - network time protocols - NTP ~~to~~ clock synchronization.

- PTP (Pres. is more precise clock synchronization protocol design for high performance computing. it achives micro second accuracy by using H/W ~~to~~ time stamp. and synchronised clock,

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

START WRITING HERE

Space for Marks	Question No.

- clock synchronization algorithms
- I. christian algorithm
- I. Berkeley algorithm
- Lampor algorithm.

Lamport Logical clocks provide different trade between accuracy precision. and overhead in distributed system.

- Hybrid: approach - combine multiple clock synchronization technique to address the ~~process~~ ~~chances~~ challenge of accuracy, fault tolerance, scability.

clock synchronization is essential for ensuring the consistenty, coordination, reability in distirbution system

Q3 B Mutual Exclusion Algorithm

soln - Requirments of mutual- exclusion Algorithm

i) no deadlocks - Gurantee that process ~~too~~ not deadlock while attempting to enter the critical section, deadlock

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

occurs when two or more processes are indefinately block. waiting for access to reasources, ~~heald~~ held by each other,

ac

no starvation :- ~~that~~ should ensure that every process eventually gets a chance, to enter the critical section, ~~even~~ starvation occurs where a process is continuously denied to enter into critical section even though, it is ready and waiting to enter.

- Fairness - Mutual exclusion algorithms should provide fairness in resource allocation, ensuring that processes are granted access to the critical section in a fair and predictable manner. Fairness prevents some processes from monopolizing the critical section at the expense of others.

- Fault Tolerance - Mutual exclusion algorithms should be resilient to failures, such as process crashes or network partitions. They should ensure that if a process fails while holding the lock, other processes still access the critical

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

section after appropriate recovery procedures

Evaluation of performance Measures for mutual Exclusion Algorithms

1. concurrency overhead: concurrency overhead measures the impact of mutual exclusion such as contention, context switching and resource utilization

2. latency: latency measures the time taken by processes to aquire and release lock for entering and exciting the critical section. lower latency indicates faster access to resources and better responsiveness in distributed systems.

3. Throughput - measures the rate at which processes can enter and exit the critical section over time. Higher throughput indicates better utilization of system resources and increases scalability.

4. Scalability - evaluates the ability of mutual

Total Marks of Question no.		Examiner
		Moderator
		Re-Assessor

Space for Marks	Question No.	START WRITING HERE

exculsion algorithm to maintain performance as the number of processes or nodes in the system ~~re~~increases. scalable algorithms should exhibit consistent performance under varying workloads and system sizes.

5. Fault Tolerance - measures resilience of mutually exclusion algorithms to failure such as process crashes, network partitions, or communication errors. Robust algorithms should recover gracefully from failures and maintain correctness and consistency in distributed systems.

6 - Fairness - evaluates the equitable distribution of resources among competing ~~proce~~ process in the system. Fair algorithm should prevent ~~alg~~ process from being unfairly starved or prioritized in ~~and~~ accessing critical section.

7. Overhead Analysis - quantifies additional computational and communication costs incurred by mutual exclusion algorithms beyond the basic requirements of mutual exclusion ~~algorithms be~~ It ~~has~~ includes factors such as message complexity, synchronization overhead and protocol ~~overhea~~ overhead.

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

4a) Load balaning.

→ 1) Optimal Resource Utilization - Load balancing distributes incoming request or task across multiple resources, such as servers, CPUs or storage device in a balanced manner. By evenly distributing the work load. It ensures that all resources are utilized efficiently and none are over burden while other remain under utilize the optimal resource allocation maximize the throughput capacity of the system, leading to improving over a performance.

→ Improved reponse time. load balancing reduces response time by evenly distributing request amount available resources by spreading work load across servers multiply all processing unit. Reducing the queuing delay and processing time for individual request, resulting in faster response for the endute. This enhances user experience and satisfaction.

→ Enhance stability.
facilitates horizontal scaling by adding or removing resources dynamically

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

base on work load demand for resource increase. load balancer can allocate addn resources to handle increasing load. without handling system performance. Similary when work load decreases. Excess resource can be realise to optimize resource utilization at reduce cost

→ fault tolerance & high availability
By distributing request across redundent resource or ~~/~~ fallover clusters in case of failure or outstages, outrages in one part of system load balancer can redirect traffic to healty ~~xxx~~ resources, ensuring continous operation & minimal distruption to users. thus redundancy & ~~failure~~ capability enhances the reability and resistance of system.

→ dynamic load adapting. - Dynamy ~~adapt~~ adapt to changes in workload or system condition in realtime. by monitoring resource utilization matrix. such as cpu load memory usage and network traffic, load balancers can adjust the distribution of incoming frequency to ensure that

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

START WRITING HERE

Space for Marks	Question No.

resources are allocated effectively and efficiently., optimizes performance and maintain stability.

Q4 B - ~~general~~ Real time example where task assignment play a critical role in optimization.

→ Task directly affect the performance scalability & cost effectiveness of the service in real time by optimizing resource allocation can deliver better service quality. Enhance user experience and ~~to~~ achieve higher resource. utilization rate while minimizing operational expenses.

e.g. cloud - virtual machine placement in cloud.

- Importance of task assignment
  - Resource utilization
  - Performance
  - energy efficiency
  - Techniques
  - Load balancing method
  - Resource placement policy
  - Dynamic resource adjustment

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Reasoure utilization

Efficient task assignment ensures that virtual machine are allocated to physical servers, in a way that maximised resource utilization by balaceng the work load across servers task assignments prevents over utilization of some servers and under utilization of some servers there by optimizing reasoure usage.

performance — effect the running on the virtual machine, placing VM on server with suitable H/W specification and minimum contention, ensure that application can access the required computational reasource memory and n/w bandwidth leading to improved performance and responsiveness.

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE
	Q4 c	The technique used for load sharing in ds:
		1) Distributed
		2) Centralised.
		Load sharing involve distributing the work load across multiple nodes to optimise resource utilization, improve performance and ensure fault tolerance
		Centralize - A central controller or load balancer is responsible for distributing incoming request or task amont available node in the system. The Load balancer monitors all each nodes and makes decision about task assignment base on predefined algorithm or policy
		Adv - centralize control
		- global view
		- Dynamic adaption
		challenges - single point of failure
		- Scalability limitation.
		- N/W overhead.

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Distributed –

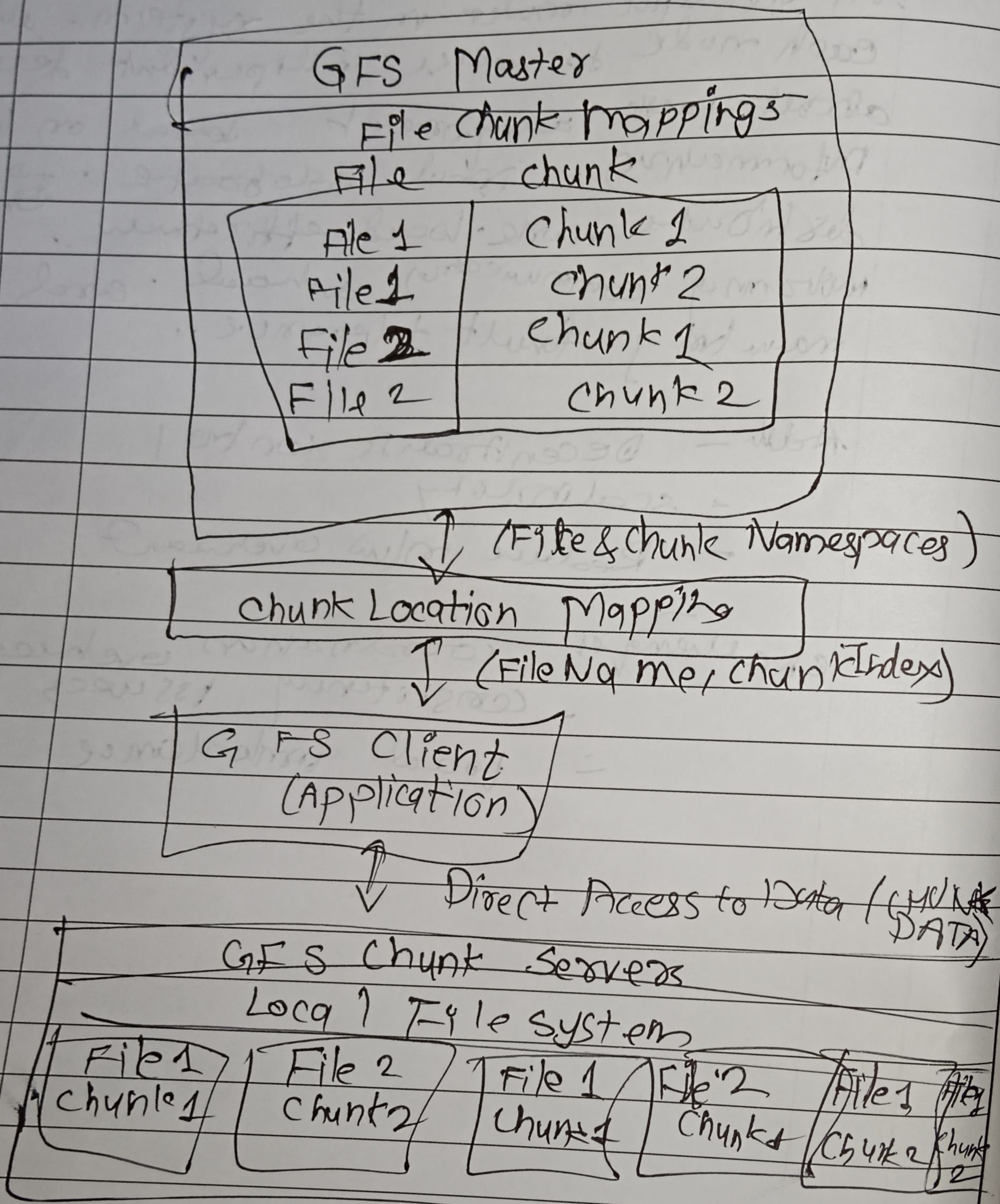
In this distributes the load in multiple nodes in the system. allowing each node to make independent decision about task assignment based on local information. And collaborate to distribute work-load efficiently, thus minimizing communication overhead and maintaining fault tolerance,

Adv – Decentralise control
– scalability
– Reduce n/w overhead

challenges – coordination overhead
– consistency issues
– load imbalance

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE
Q5	A	Discuss:



Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Google has develope several file system to meet the storage & scalability requirements of massive infrastructure components exampley are GFS and the successor colossus by the system.

GFS -

- Master chunk server Architecture.

GFS follows follow master chunk architecture server. the system consist of single master server, and multiple chunk servers. The master manages metadata including the name space and chunk space location access control, chunk server store data ~~and~~ and ~~schedule~~ chunk replication -

GFS replicates each chunk multiple times ~~across~~ different chunk servers to ensure fault tolerance and data durability the default replication factor is typically.

Client interaction -

client interact with master for metadata operations, such as file creation deletion and renaming. data access in once involves directory

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE
		communication between client and
		chunk servers bypassing master
		after initial meta data lookup
		!
		- Scalability
		- Reliability
		- Simplicity

Total Marks of Question no.	

Space for Marks	Question No.
	Q5 B

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

Q5 B Describe the file mode · used in dfs · Argue Adv and limitations.

1 Large ~~file~~ mode
simplicity
flexibility

limitation — Lack of structure
— scability.

2) Hierarchical model
~~Adv~~ — organization — eas of navication
limitation — single point of failure
concurrency.

(3) Distributed file system
— Adv. — scalability
— fault tolerance.
limmation — consistency
— complexity

(4) Object base file model

Adv — Granularity
— scalability
limmitation — compatibility
metadata overhead

Total Marks of Question no.		Examiner	
		Moderator	
		Re-Assessor	

Space for Marks	Question No.	START WRITING HERE

(5) Database file model.
- structural data management
- transaction support

limitations -
complexity
- overhead.

each
file model or distributed file system
has its own set of advantages and limitations
and choice depends
on factors such as
- nature of data
- scalability
- requirements
- fault tolerance
- consideration
- intended usage case.

Understanding the characteristics
and trade of different file models
is essential for designing
and implementing effective and
distributed storage solution