Distributed

1. **

A distributed system is a collection of independent computers, interconnected with network capable of collaborating on a task.

A distributed system can be characterized as a collection of multiple autonomous computers that communicate over a communication network.

** Features:

(i) No common physical clock enhanced Reliability
(ii) Increased performance
(iii) Increased cost ratio.
(iv) Access to geographically remote data and resources.
(v) scalability.

Example:

1. telephone
2. computer network such as internate
3. ATM machine
4. Network of workstation.
5. mobile computing.

Example of ATM:

- primary requirments: security and reliability
- consistancy.
- concurrent transaction.
- Fault tolarance.

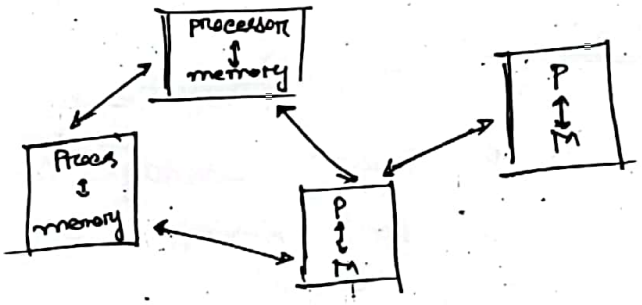
** Advantages:

1. Information sharing.
2. Resource sharing.
3. Shorter response time
4. Higher output.
5. Higher Reliability.
6. Better flexibility.
7. Better price and performance Ratio.
8. scalibility.
9. Transparency.

** Disadvantage:

1. Difficulities of developing distributed software
2. Networking problem.
3. Security problem.
4. performance.
5. openness.
6. Reliability and fault tolerance.

Figure:

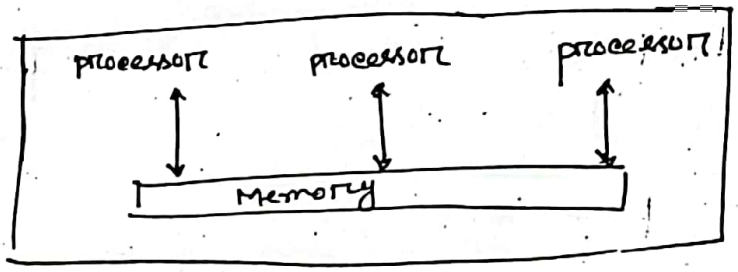


Q3. what is parallel system? Merits and Demerits of it.

parallel system:

A system is said to be a parallel system in which multiple processors have direct access to shared memory. This forms a common address space. Generally tightly coupled system are known as parallel system. Loosely coupled are distributed system. In parallel it use two or more processor to solve a problem.

Figure:



Example:

1. two servers that shared the workload routing mail, solving mathematical problem etc.
2. Supercomputers in parallel system architecture.
3. Terminals connected to single server.

Advantage:

1. provide concurrency
2. Taking advantage of non local resources
3. cost savings
4. overcoming memory constrain.

5. save time and money.
6. user friendly.

** Disadvantage :

1. lack of scalability between memory and cpu
2. programmer responsibility for synchronization
3. Difficult and expensive to design and produce shared memory.

02.

** Difference between parallel and Distributed?

parallel	Distributed
1. Tightly coupled system shared memory	1. loosely or weakly coupled Distributed clk control.
2. global clock control	No global clock control
3. processors: order of Tbps	Gbps.
4. focus on performance scientific computing	focus on performance reliability/availability resource.
5. single computer required.	use multiple computer → multiple computer
6. multiple processor performs	

Why parallel and distributed computing?

⇒ Moore's law: the number of transistor double every 18 months.

⇒ The good use of increasing number of transistor is to increase the width as:

4 bit > 8 bit > 16 > 32 > 64

⇒ The fundamental limit of the speed of a sequential processor

- power wall
- latency "

power wall and latency wall indicate the era of single thread performance improvement. more transistor on a chip are now applied to increase thorgput system.

—

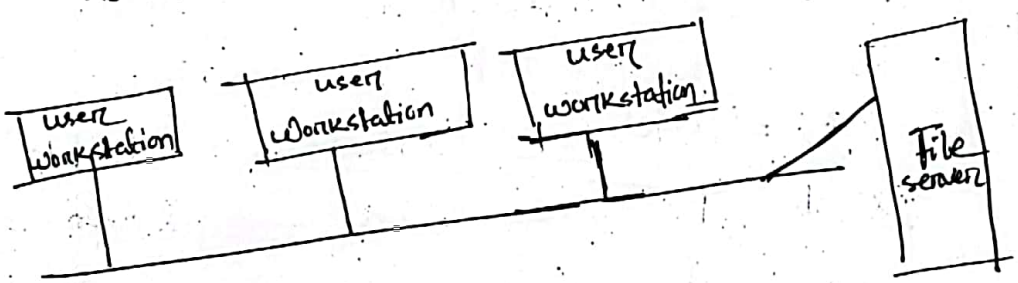
parallel computing provides concurrency and saves time and money. In distributed a single task is divided among different computers. where in parallal multiple processor perform multiple task.

Hardware Model

Q9. Explain Work station server model.

It is a distributed system that is composed of a network of workstations where each workstation provides local processing capability and an interface to the network.

Some workstation assigned dedicated roles as file server, print server, time server, name server. Workstation maybe diskless, depending only on the communication interface and operating server nodes.



Advantage:

1. Diskless Workstation is cheap, quite and easy to maintain.
2. This model provide high reliability and scability.
3. Backup and hard ware maintenance are easier to perform with few large disk.
4. New release of sw are easily installed.

2. processor pool :

A processor pool model consist of a bank of cpu's each with its own local memory o operating with shared memory. In this model all processing is done by the processing model. It comes from the queuing theory.

λ - number of request / sec.

μ - process request / sec.

T - time between request and process.

then,

$$T = \frac{1}{\mu - \lambda}$$

Which model best ?

Depends on nature of workload. Algorithm with large amount of parallism such as simulations then parallel. If the use of it for resource sharing then Distributed.

A hybride based on the both model may be suitable solutions.

05. Explain Different type of Architecture:

Software Architecture

there are many hardware and software architecture available - they are -

In the lower level it is necessary to interconnect multiple cpu with some sort of network. In the higher level it is necessary to interconnect process running on the cpu.

Client Server:

It is a model of software construction and process interaction which allows the seperation, decomposition and potential distribution of system and application.

A process fills the role of a server if it behaves:

1. provide service through interface
2. Hides implementation and complexity
3. Hybride based on both model.

A process fill the role of client if

1. present a standard interface to the server and user.
2. Initiates communication with server.
3. performs data analysis.

A client server has two types—

1. two-tier architecture:

If two types of node i.e clients and servers. This architecture referred as two tier architecture.

2. Multi-tier :

If the network consist of three different kind of nodes like node, client application server, database which store data (which process data) for application are called three tier arch

the n-tier or multi tier archi. may deploy any number of distinct services using new bussiness ~~business~~ logic. The advantage of it is, it is more scaleable. The disadvantage of it is, it puts a greater load on the network much more diffipult to program.

10. Explain Different type of cluster.

** Cluster:

A cluster is a group of loosely coupled computer that work together closely and viewed as it were a single computer. It is commonly but not always connected with local area network. It is used to improve speed and reliability. It is much more cost effective.

there are many types of cluster. they are

High availability (HA):

HA cluster are used for improving the availability of services. They operad by redundant nodes, which are used to provide service when system component fails. The most common size for nodes is 2 nodes that the minimum required node to redundant. There are many commercial implementation. such as— Linux - HA cluster.

Load balancing (LB)

operate by having all workload come through one or more load balancing front ends. then distribute it to a backend services. Implemented for improve performance.
Ex- (LVS), HPC

High performance (HP):

HP clusters are used to provide improve performance by splitting a computational task across many different nodes. They are mostly used in scientific computing.

Ex- Running linux as the os.

This is often refferred as Beowulf cluster.

** Grid computing:

Grid computing or grid clusters are a technology closely related to cluster computing. The main differents between grids and traditional cluster are grids connect collections of computer which do not trust each other. grid computing is optimised for workloads which consist of many independent jobs that do not have to share data. Resources such as storage may be shared by all nodes.

** peer to peer:

P2P computer network is a network that relies on the computing power and the bandwidth of the participants. This is useful for many purposes such as —
sharing content files containing audio, video, data or anything in digital format

6. Define Network security. Draw security model.

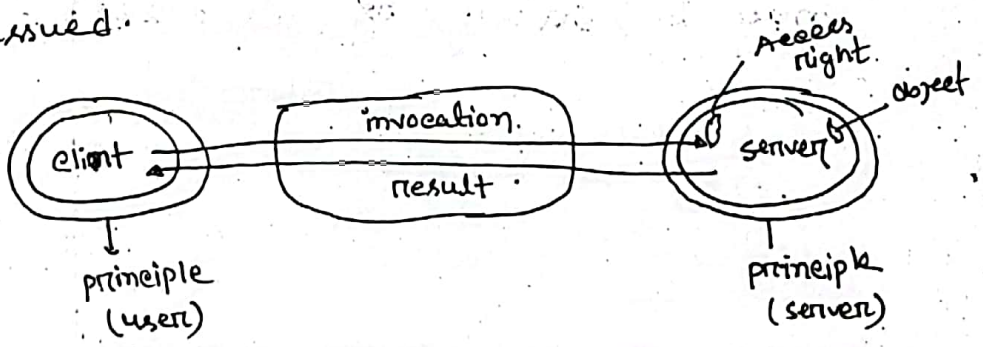
⟹

In distributed computing network security means the set of measures to guarantee the privacy, integrity and availability of resources. It involves protection of object and securing process and communication channels. The policy is to specify who is authorized to access resources.

MODEL:

Object: Intended for use by different clients via remote invocation.

principal: authority on whose behalf invocation is issued.



7. what is threat and attack? Explain different of threat.

→

Threat:

A potential for violation of security, which exists when there is a circumstances, capability, action or event that could break security and cause harm. Threat is a possible danger that might exploit vulnerability.

Attack:

An assault on the system security and derives from an intelligent threat to violate the security policy of a system.

types:

1. Eavesdropping:
   Obtaining copies of messages without authority.

2. Masquerading:
   Sending and receiving messages using the identity of another principle without their authority.

3. Message tampering:
   Intercepting and altering messages.

4. Replaying:

intercepting, storing, replaying message.

5. Denial of service:

flooding a channel with request to deny access to others.

8. Network auditing? How we configure firewalls.

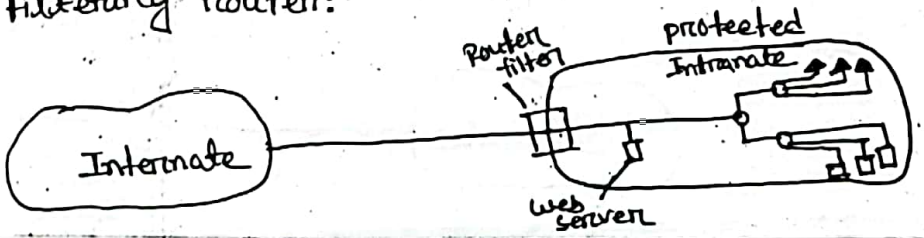
Network auditing is the collective measures done to analyze, study and gather data about a network with the purpose of ascertaining its health in accordance with network requirments.

It works through a systematic process where a network is analysed for:

1. security
2. Implementation
3. Availability
4. Management
5. performance

Configation firewall:

a. Filtering router:



4. Replaying:

intercepting, storing, replaying messages

5. Denial of service:

flooding a channel with request to deny access to others.

8. What Network auditing? How we configure firewalls.

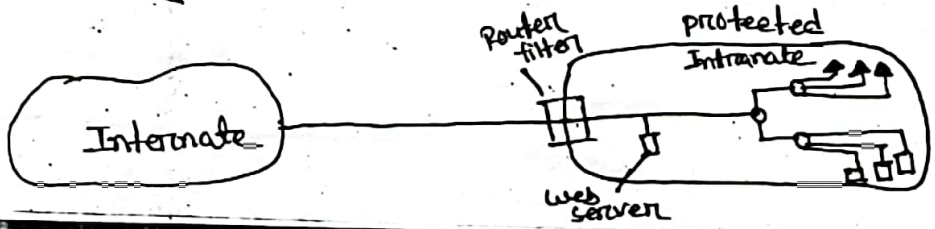
Network auditing is the collective measures done to analyze, study and gather data about a network with the purpose of ascertaining its health in accordance with network requirments.

It works through a systematic process where a network is analysed for:

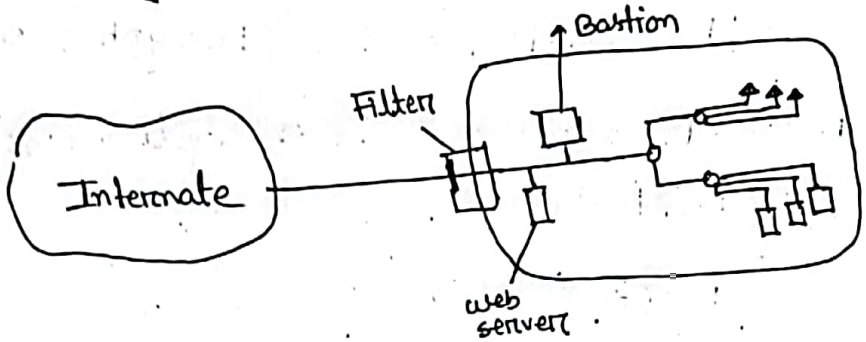
1. security
2. Implementation
3. Availability
4. Management
5. performance

Configation firewall:

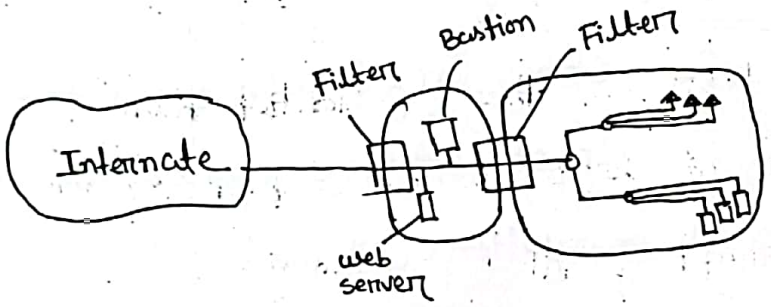
a. Filtering router:



(b) Filtering router and bastion:



(c) Screen subnet for bastion:



09. Compare among Different symetric encryptographic algorithm.

TEA:

TEA stands for tiny encryption algorithm.

1. Simple and concise.
2. secure and reasonably fast
3. Simple, symmetric algorithm.
4. written in C.
5. key 128 bits

DES:

DES stands for Data Encryption standard

1. Us standard for business applications.
2. 64 bit
3. cracked in 1997.
4. secure but poor performance.

RSA:

RSA stands for theres scientist name.

1. pair of keys. one public and one private
2. encryption with public key.
3. decryption passible only if private key known.
4. Factorizing large number.

AES:

AES stands for Advance encryption standard.

1. invitation for proposal 1997
2. in progress
3. key size 128, 192 and 256 bits.

Main security techniques for Distributed.

1. Access control:
   - Implement resource protection
   - essential in Distributed system.

2. Firewall:
   monitor traffice into and out of internale

3. Cryptographic Algorithm.
   - Digital signature.
   - Authentication.

Access control:

ensure that user access computer resources in a controlled and authorized manner.

- protection domain:
  set of rights for each resource

- Two implementation of protection domain:
  - capabilities.
  - Access control.

Access control how. works? Reference

⇒ intercepts all access attempts.

⇒ authenticates request

⇒ Applies Access control.

- if yes, access proceeds.
- if no, " denied.

Firewall

Monitor and control all communication or traffice into or out of intranet.

= service control:

Filter request for service
reject http request unless offic

= Behaviour control:

prevent illegal behaviour.
filter "spam" message.

= User control:

Allow access to authorized groups.

- Symmetric key cryptography
  - requires sender, receiver know shared key.

- Public key cryptography.
  - sender, receiver do not share secret key.
  - public key known to all.
  - private " " by receiver.

- Cryptographic Algorithm:
  - Encryption:
    - Apply rules to transform plain text to ~~this~~ cipher text.
    - Define with a function F and key K.
    - $F_K(M) = \{M\}_K$, message M encrypted with (K)
  - Decryption:
    - uses inverse function
    - $F^{-1}_K(M_K) = M$
    - can be symmetric or asymmetric
    - protected by ip packet filtering.

Digital signature:

- alternative to handwritten signatures.
- authenticate, difficult to forget.

Works:

- relies on secure hash functions, called digest.
- Sender encrypts digest
- Receiver verify signature.
- generally public key is used

Cryptographic protocol:

is an abstract protocol that perform security related function and applies cryptographic method. This discribes how the data structure and algorithm should be used.

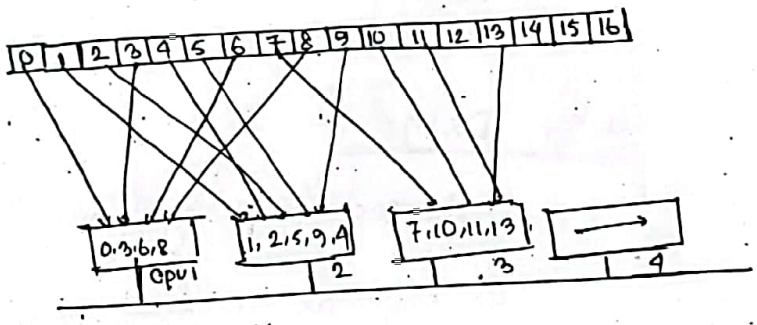
Uses:

1. key agreement on establishment
2. Entity authentication.
3. symmetric encryption.
4. Non-repudation
5. secure application level data

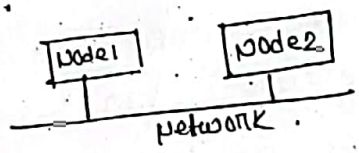
Distributed Shared memory (DSM)

Distributed shared memory is the collection of shared memory and multicomputers.

DSM = SM + MC



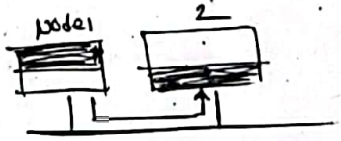
DSM consist of two component.



1. Shared address space.
2. Replication and consistancy of memory objects.

Transparent Remote access:

- Remote access is more expensive compared to local memory access.
- Individual operation can have very low overhead.
- Threads can distinguish between local and remote access.



Advantage:

1. Ease to programming
2. Eases porting of existing code.
3. pointer handling
4. share complex data.
5. No marshalling

Why DSM?

1. shared memory model easiest to program.
2. physical " not possible on multicomputer.
3. DSM emulates shared memory.

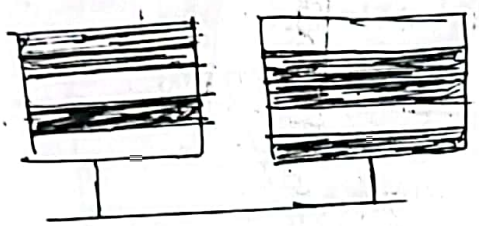
DSM implementation

- Hardware
  multiprocessor
- operating system (OS) with hardware support
  → SCI network cards.
  → SCI maps extended physical address space to remote nodes.
- OS and virtual memory
  → virtual memory.
  → local address space vs large address space

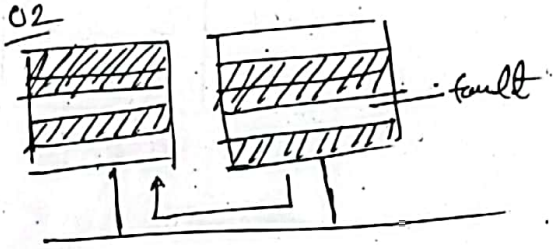
• Middleware

Figure

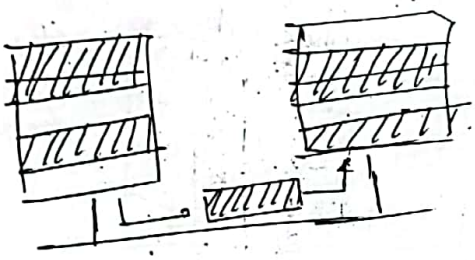
01



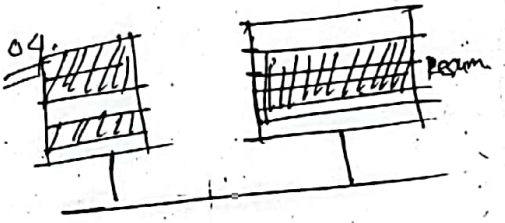
02



03.



04.



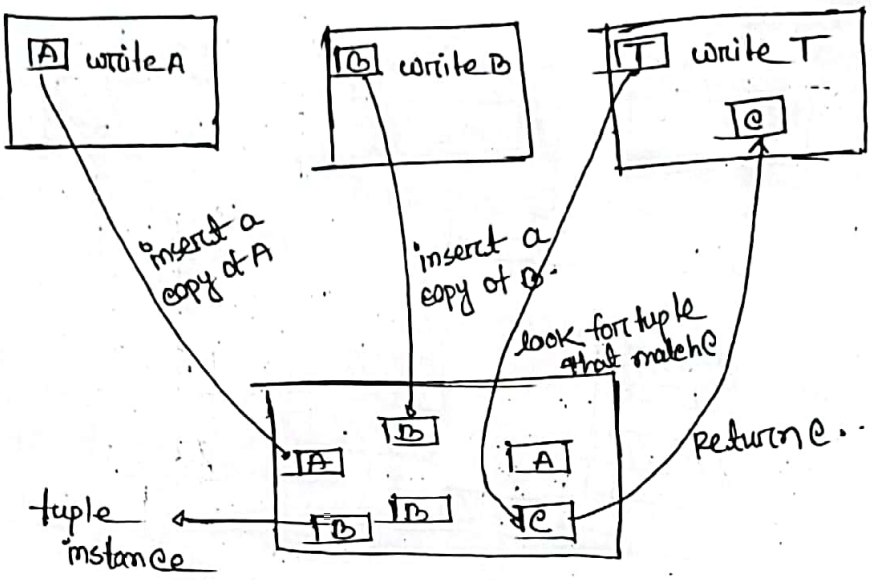
DSM MODEL

It is a form of memory architecture where physically seperated memories can be addressed as one logically shared address space. Here shared does not mean that there is a single centralized memory but the address space is shared. It is annotation, Fine grained and more complex for user.

Shared structures

1. Encapsulate shared data
2. Access only through predefine procedure.
3. Tightly intrigated synchronisation.

Figure:



Application:

1. Scientific parallel computing. Ex- Bioinformatics simulation
2. Graphics.
3. Data server
4. Data storage
5. Multiprocessor } Environment
6. Multicomputer. } Environment
   - supercomputer
   - cluster

Requirements:

1. Transparency: location, migration.

2. Reliability:
computation depend on ability of the data.

3. performance:
Important for transparency.
" " high performance.

4. scalability
Important in wide area.
" " large computation.

5. programmability:
Easy to program.
communication transparency.

—

Distributed File System (DFS)

Requirment ?

DFS is a method of storing and accessing file based in a server on client architecture. In DFS one or more central servers store files that can be accessed, with proper authorization rights by any number of remote clients in the network.

Client and Servers:

→ Client access files and directories

→ server provides ,,

→ server allow clients to perform operation.

→ operation : add , read / write

CHAllenges Of DFS:

1. Transparency

Requirments:

1. Transparency
2. concurrency
3. Replication
4. Fault tolerance
5. Consistency
6. Security
7. Efficiency

- location: a client can not tell where the file is.
- migration: a file can move easily.
- Replication: multiple copies of a file may exist.
- concurrency: multiple client may access.

2. Flexibility:
   server may added on replaced :

3. Dependability:
   - consistancy
   - security
   - Fault tolerance.

4. Scalability:
   Handle increasing number of files.

Atomic transaction
→ A sequence of file manipulation is executed indivisibly.
→ two transaction can never interface.
→ standard for database.
→ Expensive.

Stateless Server: Advantage
- Fault tolerance
- No open/close calls needed.
- No problem if server crashes
- " " if client "
- No limit on opening file.

Stateful:

→ shorter request message.

→ Better performance

→ Read ahead easier.

→ File locking possible.

Caching

A system that pools together the random access memory of multiple networked computers into a single memory network used as a data cache to provide fast access to data.

we can cache in three locations:

1. main memory of server.
2. Disk of the client
3. main memory of client

consistancy.

what is replication? Advantage

Replication:

The multiple copies of files on different servers is the process of storing data in more than one site or no
→ prevent data loss.

Advantages:

1. increase availability of data → protect system against down time
2. Reliability of data → Distribute workload.
3. high performance

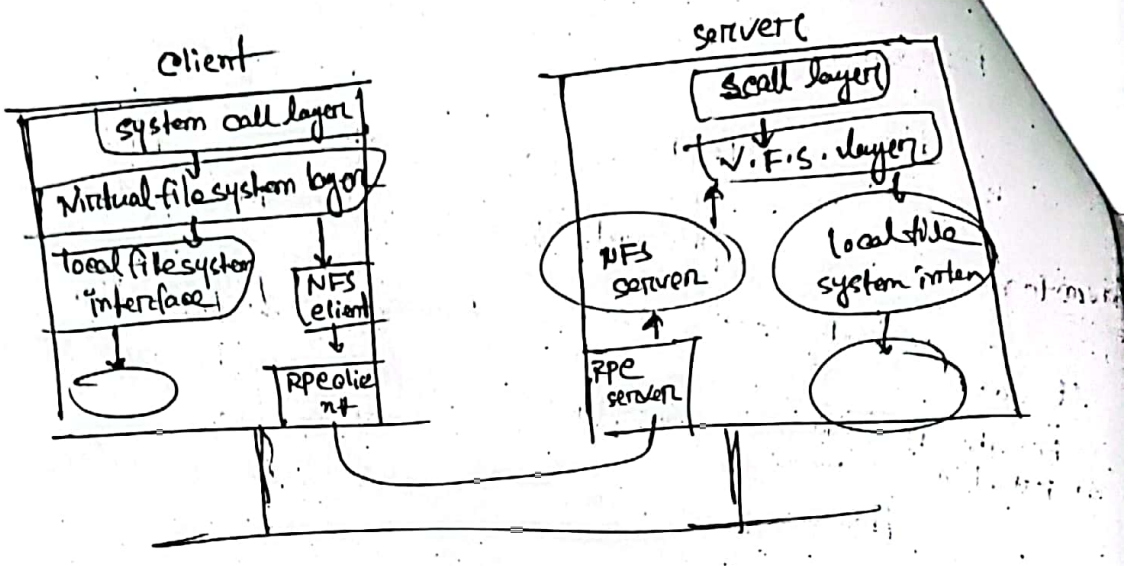
Three design

- Explicit replication: Client write files to multiple ser
- Lazy file " : Server copies file to other serve
- Group " " : Write simultaneously go to a group.

Network file System (NFS)

properties:

- Introduced by sun.
- Multiple client and server.
- stateless servers.
- File locking through seperate server.
- No replication
- caching loca files.
- consistency
- One RPC c communiation



** ANDREW File System (AFS):

properties:

1. Developed as campus wide file system.
2. Scalability.
3. Global name space for file system.
4. API same for UNIX.
5. cache on local disk.

Scalability:

- server serves whole files.
- " invalidate whole cache files.
- client do not validate cache
- little cache traffic

** CODA — Develope as a research project

- successor of Andrew file system.
- support disconnected mobile operations.
- support replication.

** Google File System:

Google file system is a scalable file system created by google. Google file system provides fault tolerance, scalability, availability and performance to large network and connected nodes. It is made up of several storage system built from low cost hardware components.

Motivation:

1. 10+ character
2. 1000+ nodes per cluster.
3. pools of 1000+ clients.

properties:

1. Files split in fixed size chunk of 64 mbytes.
2. chunk store on chunk server
3. chunk replicate on multiple chunk.
4. client interact with the chunk.
5. No explicit caching
6. provide efficiency

CHUBBY

A lock service for loosely coupled distributed system.

Chubby is
- → lock service
- → simple fs
- → Name service
- → synchronisation

Architecture:

cell: 5 replicas
writes: paxos aggrement
reads: local by master
pathname: cell/name
open, close, write delete
lock: release

Fault tolerance

Dependability:

Fault tolerance is the ability of a system to continue operating without interruption when one or more of its component fail.

- **Availability:** system is ready to be used immediately.
- **Reliability:** system can run continuously without failure.
- **Safety:** when a system fails, ~~nothing~~ to operate nothing catastrophic happens.
- **maintainability:** how easily a failed system can be repaired.

AWS Failure 2011

Amazon web services (AWS) fails in April 21, 2011. It was unavailable in US East region for 2 days.

Disadvantages:

1. 13% of volumes in one availability zone got stuck.

2. led to control API errors and ou
in whole region.

3. re-config errors and re-mirroring storm.

4. Network config problem.

5. cp election overload.

6. cp API thread starvation.

Failure

Terminology:

1. failure: A system fails when it does not meets it promises or can not provide its service in specific manner.

2. Errors: part of the system that leads to failure.

3. Fault: The course of an error.

Recursive:

i) failure can be fault
ii) manufacturing fault leds to disk failure.
iii) Disk failure " " database =
iv) Database " " email fail

types of failure: 2 types —

Total failure:

All component in a system fail. It happens in nondistributed system.

partial failure:

One or more components fails. happens in distributed system. some components affected and other component completely uneffected.

Types of fault:

-the causes of error is fault.

1. Transient fault: occur once then disappear.

2. Intermittent fault: occurs, vanishes, reoccurs, vanishes.

3. permanent fault: persist untill faulty component is replaced.

Types of failures:

1. process failure: process proceeds incorrectly.

2. Storage failure: secondary storage is inaccessible.

3. communication failure: link or node failure

4. crash failure:
server halts but working correctly until it halts

5. Fail stop:

6. Fail resume:

7. Fail silent: client donot know server halted.

8. value failure: value wrong

9. Timing failure:

10. Arbitrary failure:

Failure detector System

A failure detector is a computer application or a subsystem that is responsible for the detection of node failure.

(i) synchronous system:
- timout
- sends probs to detect crash failures.

(ii) Asynchronous System:
- timeout gives no guarantee

How fault tolerance ensure in DS?

Fault tolerance

Fault tolerance is the abil... continue operating without inter... ...rence of its component goe... ...a system?

in a part... to D.S.

Goal:

- Automatically recover
- Without seriously af... performance.

Technique:

- prevention: prevent or re...
- prediction: ~~perte~~ predic...
- Masking: hide the ...
- Recovery: restore ... state.

Failure prevention:

make sure faults

- Quality hardware
- " software

Failure prediction:

Deal with expected fault

- Test for error
- Error handling
- Error correcting
- Checksum.

Failure/Fault Recovery

Restoring an erroneous state to an error free state.

- Reclamation of resources
- consistency
- Efficiency

two types

Forward Recovery

⇒ correct state without moving back to previous state

⇒ error must be known in advance.

Backward

⇒ by moving to a back to previous state correct

⇒ High overhead

⇒ re occur

⇒ impossible to roll back

How a fault tolerance assured in a system?

⇒ fault and failure are limited in a part. It needed to provide 3 main feature to D.S.

1. Reliability
2. Availability
3. Security

Naming:

A Name in a distributed system is a string of bits or character that is used to refer to an entity. system manages a wide collection of entities in different kinds! They are—

Files, processors, users, Hosts.

Basic concepts:

Name:

string of bit or character that is used to refer an entity.

Entity:

An entity in DS can be practically anything. An entity such as a network connection may provide operations for sending and receiving data.

Resources, process, user etc.

Address:

An access point is special kind of entity in DS. The name of an access point is called address. An entity can offer more than one access point. access point of the entity may be changed. An address can be special kind of name.

Identifier:

An identifier is a name that uniquely identify entity.

properties:

1. Identifier refer to most one entity.
2. entity refer to by most one identifier.
3. Always refer to same entity.

Difference

SYSTEM ORIENTED Name	VS	HUMAN ORIENTED Name
1. Represent in machine readable form.		1. Variable length character strings.
2. structured or unstructured		2. Usually structured.
3. Easy to store, manipulate and compare.		3. Easy to remember and compare.
4. Hard for remember and human to use		4. Hard for machine to process
5: Example: inode		5. Exam: URL

Name space:

~~Container~~:

container for a set of related names.

Structure options:

- Flat
- Hierarchical
- Tag based

path names:

- sequence of edge label

Aliasing:

- another name for an entity.
- hard link
- soft link.

Naming service:

Naming service provide name space.

Name servers:

- naming service implemented by name servers.
- Implements services operations.

Operations:

- look up
- Add
- remove
- modify

names.

Partitioning

Split (ভাগ) name space over multiple server.

two type:

1. structure:
   - According to graph structure
   - Improve look up performance
   - Rigid structure.

2. Structure free:
   - Flexible
   - Decrease look up performance

DNS

Domain name system is a distributed database system that translates domain name to numerical "internate protocol address. It is the system for converting alphabetic names into numerical "ip address.

Structure:

1. Hierarchical structure
2. Top level domain

3. zone: a directory node.

4. Domain: subtree of global tree.

5. Domain name: path name.

How DNS Wo…

1. Requesting website
2. contact the Recurs…
3. Query the Authoritative…
4. Access the DNS Recor…
5. Final DNS step
6. Authoritative DNS s…
7. Recursive nameserver…

Cloud Computing

ব্যবহারকারীদের Harddrive এর পরিবর্তে internate এ ডেটা ও program সংরক্ষণ ও তার access করে।

A style of computing in which dynamically scalable and virtualize resources are provided as a service over the internate. → is a term refered to storing and accessing data over internate, does not store data in HW, Access data from remote server.

Key characteristics:

1. On demand and self service
2. Broad Network access.
3. automated as needed.
4. pooled resources
5. Elasticity
   - scalability
6. Measured service
7. security.

---

1. on demand networ… access.
2. self service
3. Broad network accessing.
4. Multi-tenancy
5. Resource poolin…
6. Elasticity
7. scalability
8. measured service

Q.

*** Key challanges of clould computing.

1. security and privacy
2. portability
3. Reliable and flexiable
4. cost
5. Downtime
6. lack of resource
7. Management of multi cloud environment.

*** Why we use Distributed algorithm?

Distributed || algo has 2 advantage.

1. For some application no central processor is available to handle the calculation.

2. A large network must forward all measurement data to a single processor there is a communication bottleneck and higher energy drain.

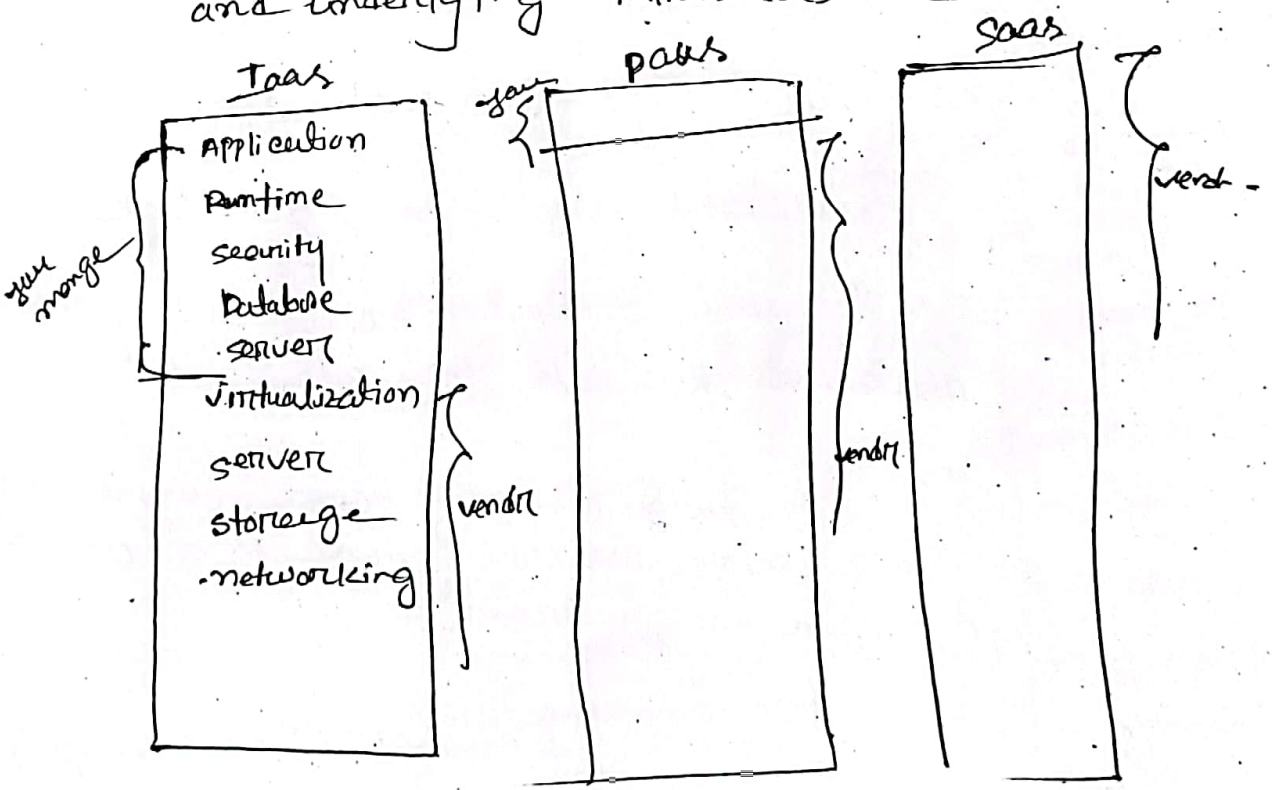
tele communication, scientific computing exte

3. Serverless computing:

Overlapping with paas, serverless computing focus on building app functionality without spending time. It is highly scaleable.

**4. Software as a service (saas):

Method for delivering software applications over the internate. Should provider host and manage the software applications and underlying infrastructure.



Benefit:

=> Flexiable:
1. Flexible provisioning -
2. Add machine on demand.
3. Add storage on demand.

=> Effort:
low barrier to entry

=> speed:
. speed is so good.

=> cost:
. eliminate the capital expense of buying the hardware and software.
. pay for what we use.
. low cost

=> security:
. strengthen security.
. redundancy.
. Data backups.

=> performance:
reduce network latency
greater economies of scale.

=> Reliability:

=> Speed:

Benefit

1. cost saving
2. Flexiable
3. High speed.
4. Backup and restore data.
5. Automatic so integration.
6. Reliability
7. Mobility
8. Quick Def
9. API acce
10. Allow pa use.

Disadvantag

1. performance vary and
2. security
3. Downtime
4. lower bo
5. locks of

Types of cloud Computing

public: open service available to everyone.

private: owned, operated and available to specific organization.

Hybrid: system use some private and some public cloud services.

Fall into four part.

1. Infrastructure as a service (Iaas):

- basic category of cloud computing.

With Iaas one can rent it infrastructure as storage, network, os, server etc.

2. platform as a service (paas):

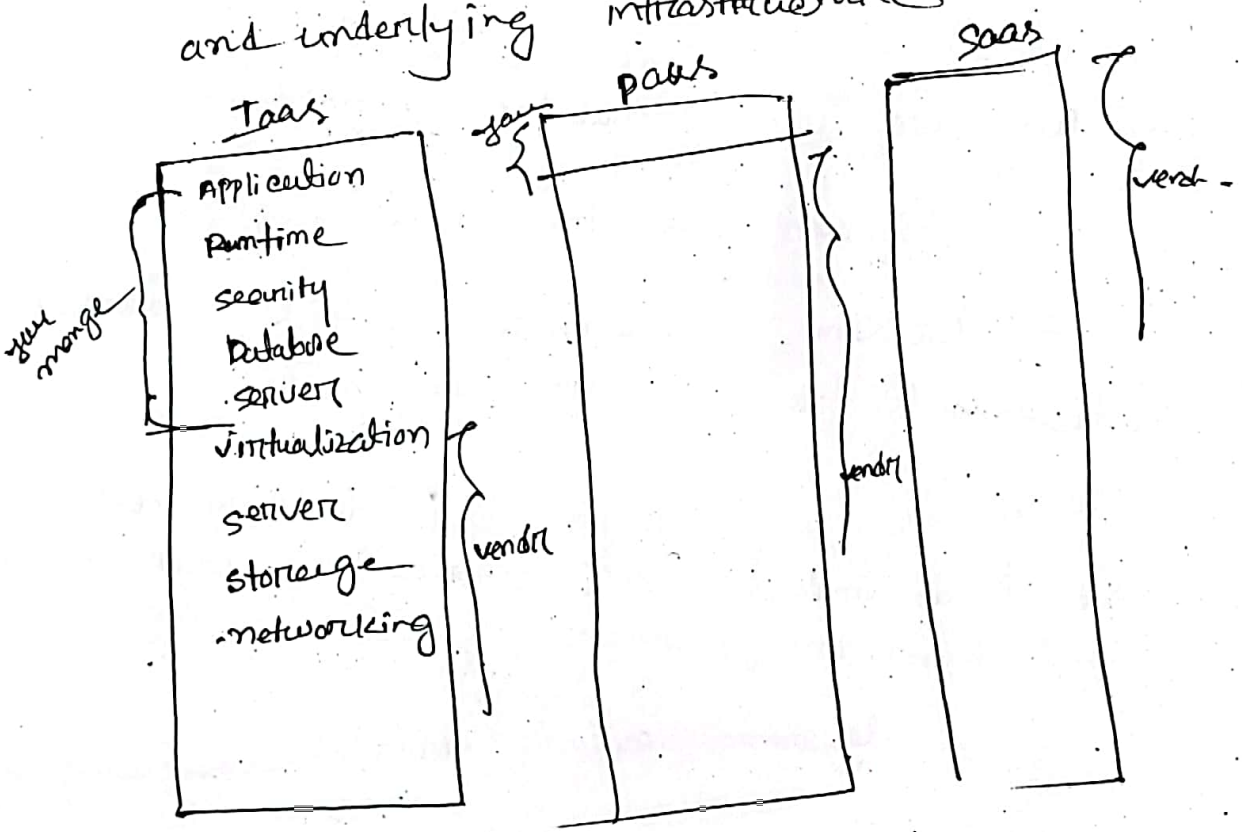
paas supply an environment for developing, testing, delivering and managering software application. It is used to create a mobile apps or web quickly.

9. Serverless computing:

Overlapping with paas, serverless computing focus on building app functionality without spending time. It is highly scaleable.

*** 9. Software as a service (saas):

Method for delivering software applications over the internet. Cloud provider host and manage the software applications and underlying infrastructure.



*** Key challanges of clould computing.

1. security and privacy
2. portability
3. Relicable and flexicable
4. cost
5. Downtime
6. lack of resource
7. Management of multi cloud environment.

*** Why we use Distributed algorithm?

Distributed algo has 2 advantage.

1. For some application no central processor is available to handle the calculation.

2. A large network must forward all measurement data to a single processor there is a communication bottleneck and higher energy drain.

tele communication, scientific computing exte

Bully election Algo

Many distributed algo need to process to act as a coordinator.

Election Algo: technique to pick unique coordinator.

Types: Bully
Ring

Bully:

⇒ Each process has a unique ID.

⇒ process knows the id and address.

⇒ communication reliable.

⇒ select process with highest id.

⇒ If coordinate fail then election have initiates.

⇒ order of msg required with n process

Derivative Function Derivative function:

Soln:

$$S(x) = \frac{1}{1+e^{-x}}$$

$$\frac{d}{dx} \cdot S(x) = \frac{d}{dx} \cdot \left(\frac{1}{1+e^{-x}} \right)$$

Quotient Rule:

$$\frac{d}{dx} f = \frac{(\text{denominator} * \frac{d}{dx} \text{numerator}) - (\text{nume} * \frac{d}{dx} \text{denometer})}{\text{denominator}^2}$$

Then,

$$\frac{d}{dx} S(x) = \frac{d}{dx} \frac{1}{1+e^{-x}}$$ (1 → Nume, $1+e^{-x}$ → denomi)

$$\frac{d}{dx}(Sx) = \frac{(1+e^{-x}) \cdot \frac{d}{dx}(1) - (1 * \frac{d}{dx}(1+e^{-x}))}{(1+e^{-x})^2}$$

$$= \frac{(1+e^{-x}) \cdot 0 - 1 \cdot (-e^{-x})}{(1+e^{-x})^2}$$

$$= \frac{e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1-1+e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1+e^{-x}}{(1+e^{-x})^2} - \frac{1}{(1+e^{-x})^2}$$

Diagram:-