05/01/22

Tutorial 01

Page No.:
Date : / /

Q1 Define the following terms.

a. Distributed System
b. Reliability
c. Scalability

Q-2 Explain with a real time example of Distributed System.

Q-3 Explain different types of transparancies in Distributed Systems

A-1 a) Distributed Systems is set of computers which are capable of functioning independently, are connected to a network to perform similar or different tasks, but act as single processing unit.

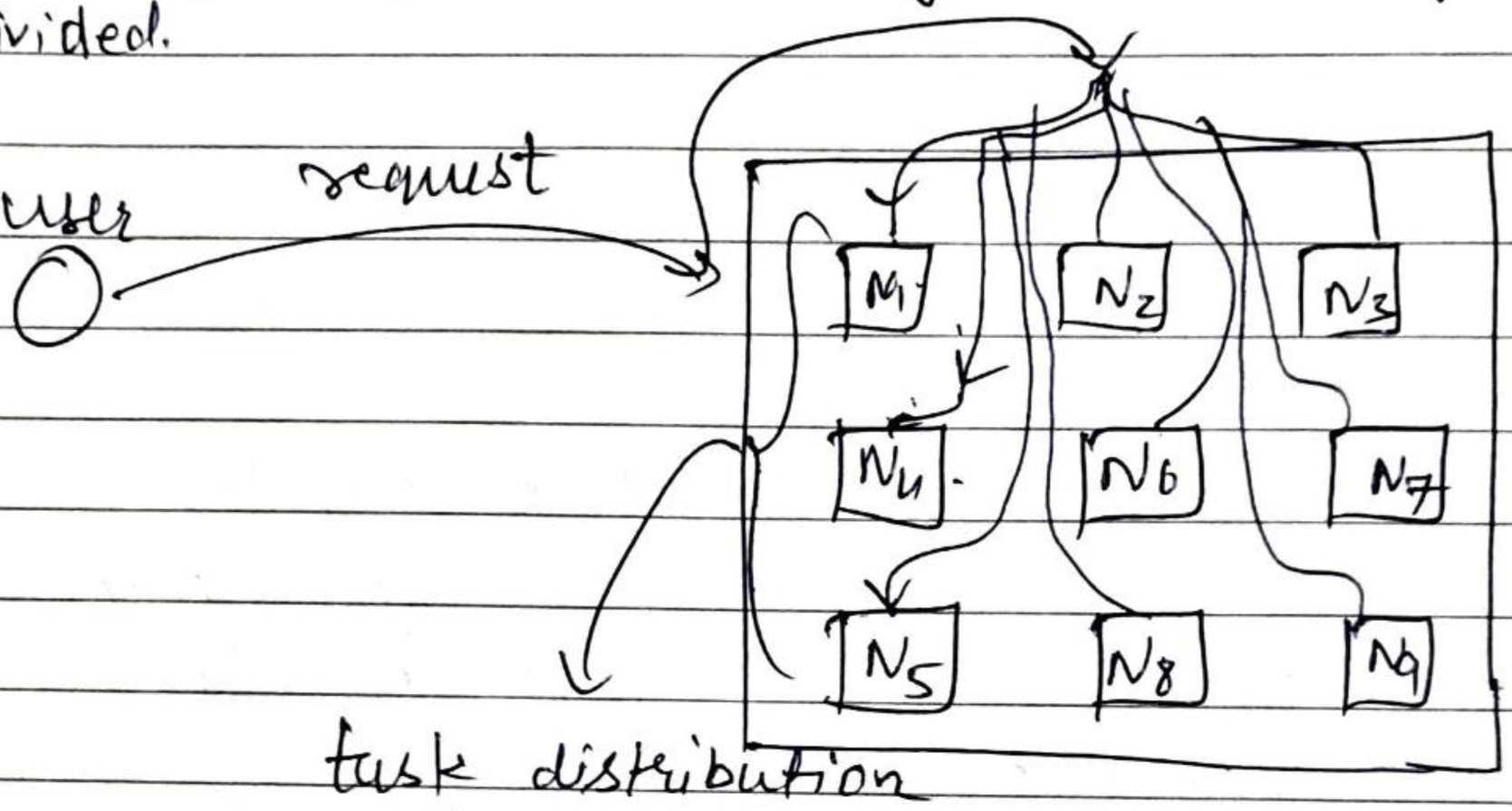
b) Scalibility => it is the ability of distributed system to increase the number independent machines which can be either in terms of memory or processing power, to increase/decrease tasks, without decrease in performace

a) Reliability => It is the ability of distributed system to deliver/process task of user/client even when one or more nodes of distributed systems get corrupted. For this near by nodes must have backup in case of memory or available processor.

A-2 A real time example of Distributed System is google colab, It takes on data from user/client and processes it according to demands of user/client. It takes on multiple requests at a time and requires high processing power which is not possible to process for a single unit.

Date: / /

In this a distributed system is used in which tons of requests are distributed/divided among different nodes in a set of computers.
This is managed by network protocols and after processing are again combined in prescribed order according to protocols through which they are divided.



→ For security, if a node fails, multiple copies are made so user need not request again.
→ A Distributed system act as a single unit even if multiple machines work independently for same task

Q3 Following are types of transparancies:- [Hiding the fact that DS is decentralized]

1) Access transparancies ⇒ It means differences in data representation are hidden.
2) Location transparancies ⇒ It means irrespective of location where data is shared, the task is completed, without disclosing real location. It feels like it works of one place shown or interface.

Page No.:
Date : / /

3). Migration ⇒ If system fails or an error occurs, the resources are moved from one network server to another, but this are hidden ~~movements~~ user.

4.) Relocation ⇒ ~~If a user is migrating~~ If a process is in state or is running, and it is migrated in that state, than it is hidden from user.

5.) Concurrency ⇒ Several users might request at the same time, but it should not affect processing power or latency should not be observable. and proper processing & request's response should be done.

6.) Replication ⇒ To overcome data loss errors and process & respond user efficiently multiple copies are made, which are hidden from user.

7.) Failure ⇒ when a node/machine fails, with backups & available processor units, processing is done, which are hidden from users.