# FCC Notes: UNIT 1 CHPT 1:

Q1. WHAT ARE THE DIFFICULTIES FACED BY ENTERPRISES IN TRADITIONAL COMPUTING?

### 1.2.1 Enterprise Perspective

Enterprises have been valuable consumers of computing since inception. They have always been among the front-runners for adopting computing-based process automation for running day-to-day business activities. Table 1.1 points out the difficulties faced by enterprises with *traditional computing* approach.

Q. WHAT ARE THE DIFFICULTIES FACED BY ENTERPRISES IN TRADITIONAL COMPUTING?

**Table 1.1** Difficulties faced by enterprises in traditional computing

| *Traditional Computing Scenario* | *Problematic facts and related questions* |
|---|---|
| Business without help of computing services is beyond imagination, and the customized software packages manage business activities. Most organizations use ERP packages (implemented by some IT enterprise) to get maximum benefits from regular business operations. | To run enterprise resource planning applications, business organizations need to invest huge volumes of capital to setup the required IT infrastructure. Servers, client terminals, network infrastructure are required, and they to be put together in a proper manner. Moreover, arranging adequate power supply, cooling system and provisioning space also consume a major part of the IT budget.<br>Are there ways to avoid this huge initial investment for computing infrastructural setup? |

| *Traditional Computing Scenario* | *Problematic facts and related questions* |
|---|---|
| Business application package implementation also over-burdens the IT enterprises with many other costs. Setting up infrastructure, installation of OS, device drivers, management of routers, firewalls, proxy servers etc. are all responsibilities of the enterprise in traditional computing approach. | Enterprises (or IT service firms) need to maintain a team of experts (system maintenance team) in order to manage the whole thing. This is a burden for HR management and incurs recurring capital investment (for salaries). Can enterprises get relief from these responsibilities and difficulties? It would help them concentrate fully on the functioning of business applications. |
| Even those IT enterprises whose sole business interest is developing applications are bound to setup computing infrastructure before they start any development work. | This is an extra burden for enterprises who are only interested in application development. They can outsource the management of infrastructure to some third party, but the cost and quality of such services varies quite a bit.<br>Can IT enterprises avert such difficulties? |
| Computing infrastructure requires adequate hardware procurement. This procurement is costly, but it is not a one-time investment. After every few years, existing devices become outdated as more powerful devices appear. | It becomes difficult to compete in the market with outdated hardware infrastructure. Advanced software applications also require upgraded hardware in order to maximize business output.<br>Can this process of upgrading hardware on a regular basis be eliminated from an enterprise's responsibility? |
| It is not unusual to find an updated version of application with new releases that is more advanced and apt to keep up with changing business scenario. | Adopting an updated version of an application requires necessary efforts from subscriber's end. Fresh installation and integration of components need to be done.<br>Can subscribers be relieved of this difficulty of periodically upgrading the applications? |
| Capacity planning of computing resources is a critical task for any organization. Appropriate planning needs time, expertise and budgetary allocation since low resource volume hampers the pace of the performance of applications. | Enterprises generally plan and procure to support the maximum business load that they have anticipated. But average resource demand remains far less, most of the time. This causes resource wastage and increases the recurring cost of business.<br>If this capacity planning task could be made less critical and resource procurement strategy more cost effective? |

Q2. WHAT ARE THE DIFFICULTIES FACED BY INDIVIDUAL USERS IN TRADITIONAL COMPUTING?

with latest available technology etc. have always been the matters of concern.

## Q. WHAT ARE THE DIFFICULTIES FACED BY INDIVIDUAL USERS IN TRADITIONAL COMPUTING?

### 1.2.2 Individual User's Perspective

Individual users have also been consumers of computing since a long time. They use computing for various purposes like editing documents, developing program, playing games, watching videos, accessing Internet etc. Table 1.2 focuses on the troubles individual users of computing usually face with the *traditional computing* approach.

**Table 1.2** Difficulties faced by individual users in traditional computing approach

| *Traditional Computing Scenario* | *Problematic facts and related questions* |
|---|---|
| To work with software applications (like text editor, image editor, programming, games etc.), users first need to procure computing system where these software applications run. | For general users (who don't want to experiment with computer hardware devices), this initial capital investment for setting up computing infrastructure is often more than the software applications they use!<br>If this huge unnecessary investment for procuring hardware components could be avoided? |
| Requirement analysis and procurement of hardware infrastructure are responsibility of the users. But, actual utilization of these resources depends on frequency of user access and the kind of software applications they run over it. | General users are usually not experts of computing systems. They are often misguided and procure unnecessary volume/capacity of hardware resources, most portions of which remain unutilized. This reduces the return on investment (ROI).<br>If this approach could be changed; if users did not have to procure a fixed volume of hardware resource prior to its actual use or demand? |
| A hardware component may fail for many reasons. Maintenance of the hardware infrastructure is the users' responsibility. | Time, cost and the uncertainty are involved in the process.<br>If users could get relief from these responsibilities and difficulties? |
| Computing systems (desktop, laptop etc.) procured by most users are hardly used for few hours daily on an average. | Non-utilization of procured systems results in wastage of resource with regard to total investment.<br>If the hardware resources would be available on payment of usage basis? |
| Software licensing cost needs separate budgetary allocation. Licenses are sold for fixed period of time (usually for one year-duration). | If software is used for 2–5 hour per day on an average during licensing period, it depicts 8 % –20 % utilization of entire investment.<br>If this cost could be reduced? If the licensing fee would be paid on hourly usage basis? |
| Users are burdened with installation and critical customization of software. They also troubleshoot in case the software crashes. | Professional help can be obtained against payment, or users can troubleshoot themselves, thereby investing more time.<br>If users could get relief from these responsibilities and difficulties? |

## Q3. WHAT DO YOU MEAN BY COMPUTING ARE WHAT ARE ITS LAYERS?

Computing refers to the process of using computers and computing devices to perform tasks and solve problems. It encompasses a wide range of activities that involve processing and manipulating data using algorithms, software, and hardware.

that involve processing and manipulating data us
algorithms, software, and hardware.

## 1.4 THREE LAYERS OF COMPUTING

Computers and computing have become an integral part of our daily lives. Different people use different categories of computing facilities. These computing facilities can be segmented into three categories:

- Infrastructure
- Platform
- Application

These three categories of computing facilities form three layers in the basic architecture of computing. Figure 1.1 represents the relationships between these three entities.

### 1.4.1 Infrastructure

The bottom layer or the foundation is the 'computing infrastructure' facility. This includes all physical computing devices or hardware components like the processor, memory, network, storage devices and other hardware appliances. Infrastructure refers to computing resources in their bare-metal form (without any layer of software installed over them, not even the operating system). This layer needs basic amenities like electric supply, cooling system etc.

### 1.4.2 Platform

In computing, platform is the underlying system over which applic
that the platform consists of the physical computing device (hardwa
software where the program or application can run. The term 'com
different abstract levels. It consists of:

- Certain hardware components, only.
- Hardware loaded with an operating system (OS).
- Hardware and OS, additionally, loaded with run-time libraries.

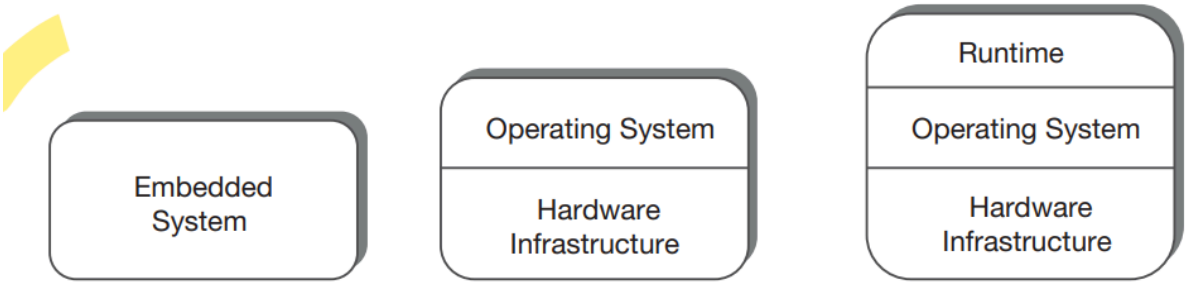



**FIG 1.2:** Computing platforms in different forms

A fully configured physical computer loaded with an operating system is considered as a platform for computing. Different platforms can be installed over the same computing infrastructure. Platform layer provides the platform to execute the applications.

### 1.4.3 Application

Applications (application software) constitute the topmost layer of this layered architecture. This layer generally provides interfaces for interaction with external systems (human or machine) and is accessed by *end users* of computing. A user actually works on the application layer while he or she is going to edit a document, play a game or use the calculator in a computer. At this layer, organizations access enterprise applications using application interfaces to run their business.

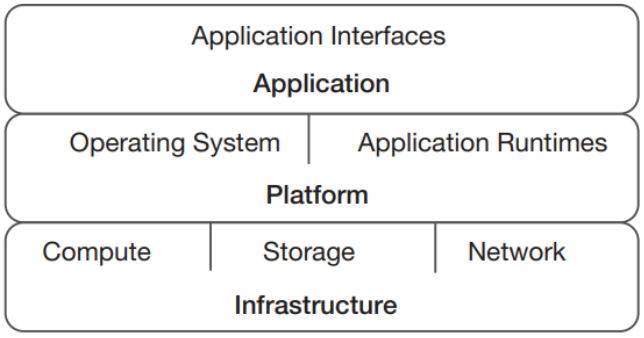



**FIG 1.4:** Elements of three computing layers

## Q4. WHAT DO YOU MEAN BY TRADITIONAL COMPUTING ARE WHAT ARE ITS LAYERS?

Traditional computing refers to the classical approach of performing computations using conventional digital computers that rely on binary bits (0s and 1s) as the basic units of information.

### THREE LAYERS IN TRADITIONAL COMPUTING:

- **Traditional Infrastructure Model:**

The traditional infrastructure model, often referred to as on-premises or conventional infrastructure, is a computing setup in which an organization owns, manages, and maintains its own physical hardware, servers, networking equipment, and software on its premises. This model contrasts with cloud computing, where services are provided over the internet by third-party providers. In the traditional infrastructure model, organizations are responsible for the entire lifecycle of their technology resources, from procurement and installation to maintenance and eventual replacement.

Key features of the traditional infrastructure model include:

1. **Physical Hardware**: Organizations acquire and manage their own hardware, including servers, storage devices, routers, switches, and other networking equipment. This hardware is typically housed in data centers owned or leased by the organization.
2. **Data Center Management**: Organizations are responsible for setting up and maintaining their own data centers, including cooling, power management, physical security, and infrastructure monitoring.
3. **Software Management**: Organizations install and maintain their own software applications, operating systems, and middleware on the hardware. This includes regular updates, patches, and troubleshooting.
4. **Customization and Control**: The traditional infrastructure model allows for a high degree of customization and control over hardware and software configurations, which can be important for certain industries or specific requirements.
5. **Security and Compliance**: Organizations have direct control over security measures and compliance standards for their infrastructure. They can implement security protocols tailored to their needs.

- **Traditional Platform Model:**

A "traditional platform model" refers to an older or historical approach to providing technology platforms for software development, applications, or services.

Characteristics of **Traditional Platform Model are:**

1. **Monolithic Architecture**: Platforms are built using a monolithic architecture, where all components of an application or service are tightly integrated into a single, large codebase. This can make development and maintenance more complex.
2. **On-Premises Deployment**: Platforms are installed and operated on-premises, within an organization's own infrastructure. This contrasts with modern cloud-based platforms that offer remote access and scalability.
3. **Higher Maintenance Burden**: Organizations are responsible for maintaining and upgrading the platform's hardware, software, and infrastructure, which can be resource-intensive.
4. **Higher Costs**: Due to the need to manage physical resources and the associated maintenance, the traditional platform model could have higher overall costs.

- **Traditional Application Model:**

The term "traditional application model" generally refers to the conventional way of designing, developing, and deploying software applications before the advent of newer paradigms and technologies. This model is often in contrast to modern approaches like cloud-native applications, microservices architecture, and containerization. Here are some key characteristics of the traditional application model:

1. **Monolithic Architecture**: Traditional applications are typically built using a monolithic architecture, where all components and functionalities are tightly integrated into a single codebase. This can lead to challenges in scalability, maintainability, and deployment.
2. **On-Premises Deployment**: Applications are installed and run on local servers or physical hardware within an organization's premises. This requires managing the infrastructure and associated maintenance tasks.
3. **Higher Maintenance Effort**: Maintenance tasks, such as applying patches, upgrading libraries, and ensuring security, can be resource-intensive and might require planned downtime.
4. **Higher Costs**: The traditional application model often requires significant upfront hardware and software investments. Ongoing maintenance and support can also contribute to higher costs.
5. **Single Point of Failure**: Since all components are closely tied together, a failure in one part of the application can impact the entire system.

## Q5. WHAT ARE THE MAJOR ASPECTS OF COMPUTING?

■ Infrastructure as a Service
■ Platform as a Service
■ Application or Software as a Service

- **Infrastructure as a Service:**

Infrastructure as a Service (IaaS) is a cloud computing model that provides virtualized computing resources over the internet. In an IaaS model, instead of owning and managing physical hardware, organizations can rent and use virtualized resources from a cloud service provider.

Key characteristics of Infrastructure as a Service include:

1. **Virtualized Resources**: IaaS provides virtualized computing resources, including virtual machines (VMs), storage, and networking components.
2. **Scalability**: IaaS platforms offer the ability to scale resources up or down based on demand. This elasticity helps businesses avoid overprovisioning.
3. **Pay-as-You-Go Pricing**: IaaS follows a pay-as-you-go or consumption-based pricing model. Users are billed based on the actual resources they use, which can lead to cost savings compared to traditional on-premises infrastructure.
4. **Global Accessibility**: IaaS resources can be accessed from anywhere with an internet connection. This enables remote work, disaster recovery, and multi-location collaboration.

- **Platform as a Service:**

Platform as a Service (PaaS) is a cloud computing service model that provides a platform and environment for developers to build, deploy, and manage applications without having to worry about the underlying infrastructure, networking, and server management.

Key characteristics of Platform as a Service include:

1. **Abstraction of Infrastructure**: PaaS abstracts away the complexities of managing and provisioning servers, storage, networking, and other infrastructure components. Developers can focus solely on writing code and building applications.
2. **Development Tools**: PaaS platforms typically provide a variety of development tools, such as programming languages, frameworks, libraries, and integrated development environments (IDEs).
3. **Middleware and Services**: PaaS provides middleware services, such as databases, messaging systems, caching, and authentication.
4. **Multi-Tenancy**: PaaS allows multiple users or teams to work on their applications in isolated environments, ensuring security and resource allocation.
5. **Cost Efficiency**: PaaS can be cost-effective since users pay only for the resources and services they use.

- **Application or Software as a Service:**

Application as a Service (AaaS) or Software as a Service (SaaS) is a cloud computing service model that delivers software applications over the internet on a subscription basis. In this model, users can access and use software applications through web browsers without needing to install or maintain the software locally on their devices.
Key characteristics of Software as a Service include:

1. **Web-Based Access**: SaaS applications are accessed through web browsers, allowing users to use the software from any device with an internet connection. This eliminates the need for local installations and compatibility concerns.
2. **Centralized Management**: The SaaS provider is responsible for managing the software, including updates, patches, security, and performance optimization. Users do not need to worry about these tasks.
3. **Multi-Tenancy**: SaaS applications are designed to serve multiple customers (tenants) from a single instance of the software.
4. **Automatic Updates**: SaaS providers regularly release updates and new features, which are automatically applied to the software without any effort required from users.
5. **Pay-as-You-Go**: Users only pay for the resources and features they need, making SaaS a cost-effective option for businesses of various sizes.

## Q6. WHAT ARE THE INFUENCES BEHIND THE CLOUD SERVICES ADOPTION?

1. **Technological Influences:**

The key technological influences behind cloud service adoption are:

- **Universal Network Connectivity:**

Cloud computing services are generally accessed through high speed network or Internet. Well-connected digital communication network spread across the world is necessary for ubiquitous access to cloud facility. As high speed network communication infrastructure has become available around the world, access to the cloud computing facility from any location has become a reality.

- **High-Performance Computing:**

In traditional approach, high-performance computing (HPC) systems needed specialized hardware components which were costly. Affording HPC was once beyond the imagination of small enterprises and individuals. Cloud computing has made HPC affordable for everyone by aggregating computing power to produce computing performance for executing high performance tasks.

- **Commoditization:**

A product or service turns into a commodity when it becomes marketable and can be interchanged with another product of same type, which should also be available in the market. This is possible when products or services from multiple vendors provide more or less same value to customers, and customers have the option of replacing one product with another product of some other vendor. Cloud offerings from different providers create the same scenario. This commoditization of cloud services has developed irrefutable marketplace for cloud adoption.

2. **Operational or Business Influences:**

- **Low Cost Solution:**

With cloud computing, consumers adopt the philosophy of pay-as-you-use. This turns out more cost effective, since consumers no more need to pay for excess capacity which was very common in traditional computing. This remarkably reduces the IT budget of consumers.

- **Speed or Responsiveness:**

Responsiveness is probably the most important factor in today's business world. Cloud computing represents substantial advantages in terms of responsiveness. The time required to develop, configure, or launch new systems in cloud computing environment is much lesser in comparison with the traditional one.

- **Small Initial Investment:**

Traditional computing approach had a barrier as initial investment to setup computing infrastructure was huge. Even adopting latest technology also meant considerable amount of investment for existing users. Cloud computing eliminates these barriers, as customers need to invest very small capital to start.

- **Less Maintenance Cost:**

The budget allocated for IT services in any organization has a critical influence over the performance and outcome of the business. Studies show that on an average about 70 to 80 percent of any organization's IT budgets goes to the operation and maintenance of existing systems and infrastructure. This cost has drastically reduced in the regime of the cloud computing.

- **Mobility:**

Once there was the era of personal computers when everyone used to access computing of any form using PCs from their desks, be it in a business or for personal use. But those days are far behind. It is difficult to keep pace with today's world if one needs a fixed computer (or laptop etc.) in order to use computing facility. Cloud computing provides full flexibility in this direction. Computing in any capacity can be accessed from a variety of portable devices from any location.

- **Flexibility:**

With cloud computing, it becomes very easy to change computing platform or move towards other technology without significant capital loss and with minimal effort. This flexibility was not available in traditional computing. Cloud services can also grow or shrink as needed. It can expand or shrink with business that provides the higher cost-benefit ratio. When a business needs more computing/IT support they consume more cloud services, when they need less they consume less. Since payment is on usage basis, this elasticity of cloud services provides great flexibility for the consumers.

MCQ's ON PG 43 OF 434

# CHPT 2:

## Q1. WHAT DO U MEAN BY CLIENT-SERVER COMPUTING?

Client-server computing is a model of distributed computing where tasks and responsibilities are divided between two types of entities: clients and servers. This model is commonly used in networking and software architecture to enable efficient sharing of resources and collaboration among multiple computers or devices.

**Characteristics of Client Server Computing:**

The salient points for client server computing are as follows:

- The client server computing works with a system of request and response. The client sends a request to the server and the server responds with the desired information.
- The client and server should follow a common communication protocol so they can easily interact with each other. All the communication protocols are available at the application layer.
- A server can only accommodate a limited number of client requests at a time. So it uses a system based to priority to respond to the requests.

- Denial of Service attacks hindera servers ability to respond to authentic client requests by inundating it with false requests.
- An example of a client server computing system is a web server. It returns the web pages to the clients that requested them.

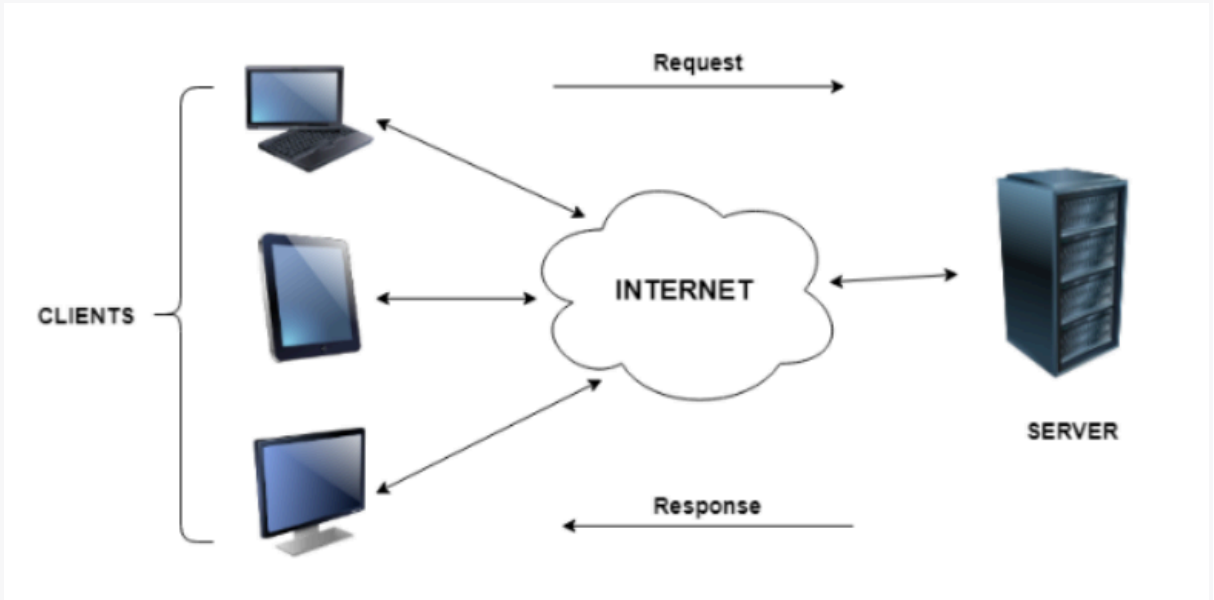
**Advantages of Client Server Computing:**

- All the required data is concentrated in a single place i.e. the server. So it is easy to protect the data and provide authorisation and authentication.
- The server need not be located physically close to the clients. Yet the data can be accessed efficiently.
- It is easy to replace, upgrade or relocate the nodes in the client server model because all the nodes are independent and request data only from the server.
- All the nodes i.e clients and server may not be build on similar platforms yet they can easily facilitate the transfer of data.

**Disadvantages of Client Server Computing:**

- If all the clients simultaneously request data from the server, it may get overloaded. This may lead to congestion in the network.
- If the server fails for any reason, then none of the requests of the clients can be fulfilled. This leads of failure of the client server network.
- The cost of setting and maintaining a client server model are quite high.

Examples of client-server computing are web browsing, emails, online gaming, file sharing and database applications, etc.




## Q2. WHAT DO U MEAN BY PEER-TO-PEER COMPUTING 0R COMMUNICATION?

Peer-to-peer (P2P) computing is a decentralized model of network computing where participants (peers) in the network share resources, data, and services directly with each other without the need for a central server. In a peer-to-peer network, all participants can act both

as clients and servers, contributing their resources and utilizing resources from other peers in the network.

**Characteristics of Peer to Peer Computing:**

- Peer to peer networks are usually formed by groups of a dozen or less computers. These computers all store their data using individual security but also share data with all the other nodes.
- The nodes in peer to peer networks both use resources and provide resources. So, if the nodes increase, then the resource sharing capacity of the peer to peer network increases. This is different than client server networks where the server gets overwhelmed if the nodes increase.
- Since nodes in peer to peer networks act as both clients and servers, it is difficult to provide adequate security for the nodes. This can lead to denial of service attacks.
- Most modern operating systems such as Windows and Mac OS contain software to implement peer to peer networks.

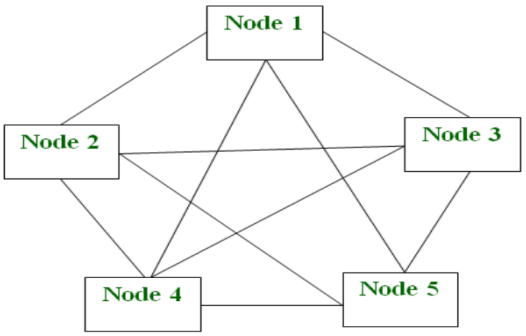
**Advantages of Peer to Peer Computing:**

- Each computer in the peer to peer network manages itself. So, the network is quite easy to set up and maintain.
- In the client server network, the server handles all the requests of the clients. This provision is not required in peer to peer computing and the cost of the server is saved.
- It is easy to scale the peer to peer network and add more nodes. This only increases the data sharing capacity of the system.
- None of the nodes in the peer to peer network are dependent on the others for their functioning.

**Disadvantages of Peer to Peer Computing:**

- It is difficult to backup the data as it is stored in different computer systems and there is no central server.
- It is difficult to provide overall security in the peer to peer network as each system is independent and contains its own data.

Examples of P2P computing are File sharing, Distributed computing, instant messaging, blockchain technology, etc.




P2P Architecture

## Q3. WHAT DO U MEAN BY DISTRIBUTED COMPUTING?

Distributed processing concept emerged as a powerful computing model.

Distributed computing refers to the use of multiple interconnected computers or devices to work together on solving a single problem or performing a task. In distributed computing, these devices, often referred to as nodes, communicate and collaborate over a network to achieve a common goal. The primary objective of distributed computing is to leverage the combined processing power, memory, and resources of multiple machines to perform tasks that might be too large or complex for a single computer to handle efficiently.

Key characteristics of distributed computing include:

**Decentralization**: Distributed computing avoids centralizing all processing on a single machine.

**Resource Sharing**: Nodes in a distributed system share resources such as processing power, memory, storage, and bandwidth, enabling efficient utilization of available resources.

**Concurrency**: Multiple nodes can work concurrently on different parts of a task, leading to improved performance and faster task completion.

**Scalability**: Distributed systems can be designed to scale easily by adding more nodes as needed, which enhances their ability to handle larger workloads.

**Communication**: Effective communication mechanisms between nodes are essential for sharing information, coordinating tasks.

**Advantages and Disadvantages**

Advantages of the Distributed Computing System are:

- **Scalability:** Distributed systems are generally more scalable than centralized systems, as they can easily add new devices or systems to the network to increase processing and storage capacity.
- **Reliability:** Distributed systems are often more reliable than centralized systems, as they can continue to operate even if one device or system fails.

- **Flexibility:** Distributed systems are generally more flexible than centralized systems, as they can be configured and reconfigured more easily to meet changing computing needs.

Disadvantages of the Distributed Computing System are:

- **Complexity:** Distributed systems can be more complex than centralized systems, as they involve multiple devices or systems that need to be coordinated and managed.
- **Security:** It can be more challenging to secure a distributed system, as security measures must be implemented on each device or system to ensure the security of the entire system.
- **Performance:** Distributed systems may not offer the same level of performance as centralized systems, as processing and data storage is distributed across multiple devices or systems.

Distributed computing is employed in various scenarios and applications such as large scale data processing, scientific computing, distributed databases, blockchain networks, IOT, etc.

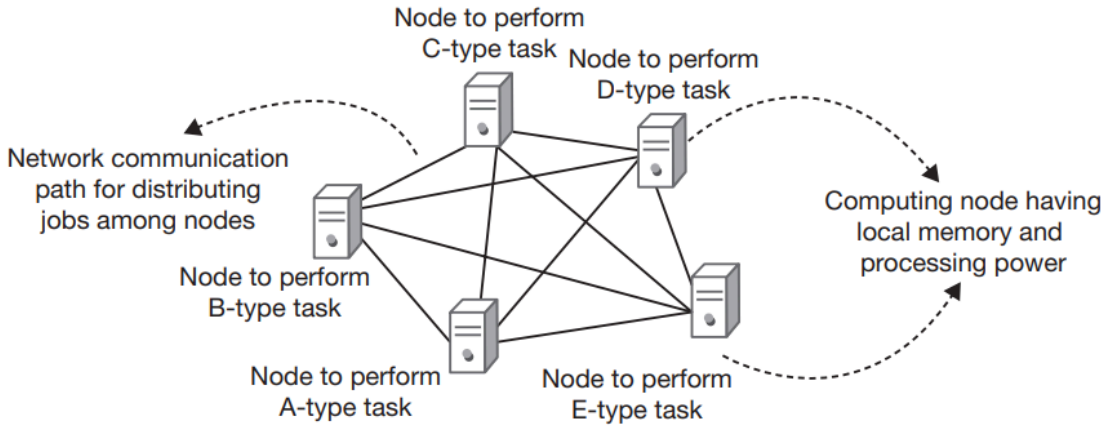



**FIG 2.1:** A model of distributed computing environment

## Q4. WHAT DO U MEAN BY CLUSTER COMPUTING?

The concept of clustering appeared as the next step of evolution in the field of computing. Computing clusters are made of multiple nodes (computers) connected via network which perform similar tasks. Thus, execution of a task can be faster as it can be distributed and executed in parallel across multiple machines inside a cluster. All the nodes of a cluster together give impression of a single system. The cluster computing model emerged as a result of progress in multiple technologies.
In this computing model, a set of computers were reserved to handle specific type of task to make the system more reliable. If any node fails, other nodes in the cluster can handle the load. This was the idea behind resource pool that technologists were trying to implement. Cluster computing introduced the concept of resource pooling. The pools were made of homogeneous computing systems.
With this implementation, reliability was achieved through redundancy.

**Advantages of Cluster Computing :**

**1. High Performance :**

The systems offer better and enhanced performance than that of mainframe computer networks.

**2. Easy to manage :**

Cluster Computing is manageable and easy to implement.

**3. Scalable :**

Resources can be added to the clusters accordingly.

**4 Flexibility :**

It can be upgraded to the superior specification or additional nodes can be added.

**Disadvantages of Cluster Computing :**

**1. High cost :**

It is not so much cost-effective due to its high hardware and its design.

**2. Problem in finding fault :**

It is difficult to find which component has a fault.

**3. More space is needed :**

Infrastructure may increase as more servers are needed to manage and monitor.

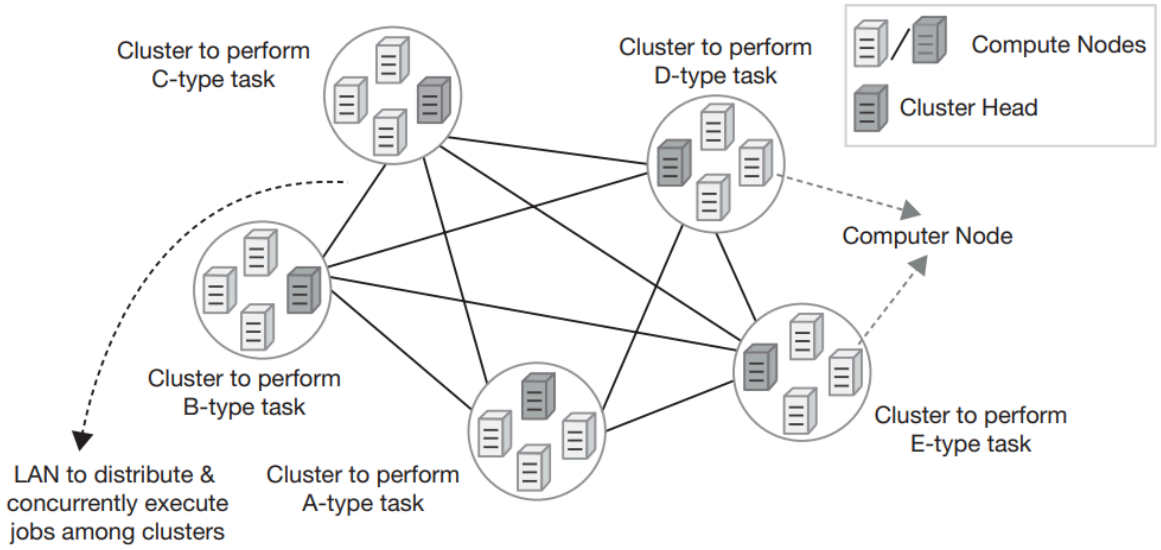



**FIG 2.3:** A cluster computing model

## Q5. WHAT DO U MEAN BY GRID COMPUTING?

**Grid Computing** can be defined as a network of computers working together to perform a task that would rather be difficult for a single machine. All machines on that network work under the same protocol.

Grid computing concept introduced the idea of decentralization of control in distributed computing environment.

Grid Computing is a subset of distributed computing.

Other than decentralization of control, grid computing introduced another important system feature. The computing environment could now be built with heterogeneous computing systems.

**Characteristics of Grid Computing are:**

- Large scale: Grid concept promises to deal with computing resources going up to millions.
- Geographical distribution: Computing resources could be located at distant places.
- Heterogeneity: Computing grid could accommodate resources having different hardware characteristics and using various types of software.
- Resource co-ordination: Resources in a grid could coordinate among themselves to generate aggregated computing capabilities.
- Pervasive access: Resource failure could be handled by granting access to other available resources.
- Unlimited resource addition (scaling): Being a distributed computing model, it allows easy growth of system capacity by adding more resources into an existing system.

**Advantages of Grid Computing:**

1. It is not centralized, as there are no servers required, except the control node which is just used for controlling and not for processing.
2. Multiple heterogeneous machines i.e. machines with different Operating Systems can use a single grid computing network.
3. Tasks can be performed parallelly across various physical locations and the users don't have to pay for them (with money).

**Disadvantages of Grid Computing:**

1. The software of the grid is still in the involution stage.
2. A super-fast interconnect between computer resources is the need of the hour.
3. Licensing across many servers may make it prohibitive for some applications.
4. Many groups are reluctant with sharing resources.
5. Trouble in the control node can come to halt in the whole network.

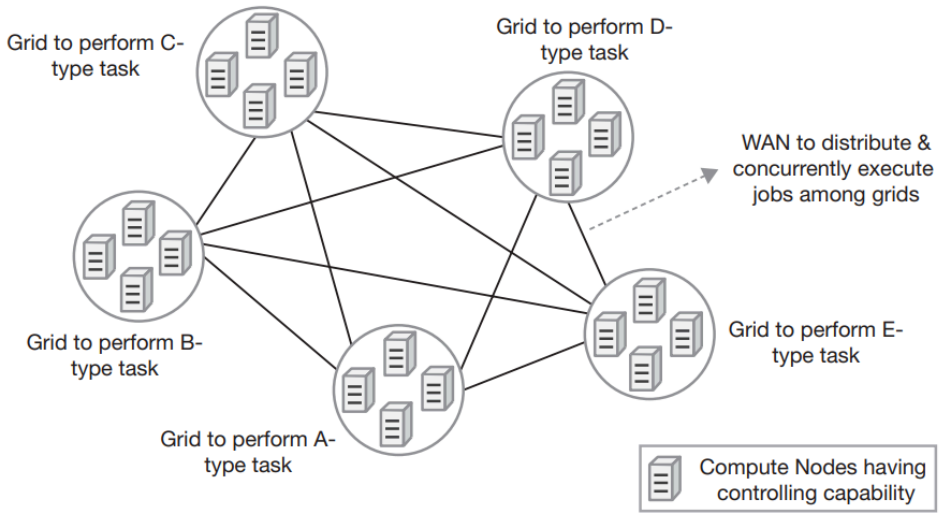



**FIG 2.4:** A grid computing model

## Q6. Differentiate between Cluster and Grid Computing.

**Table 2.1** Characteristics of cluster and grid computing

| *Characteristic* | *Cluster Computing* | *Grid Computing* |
|---|---|---|
| Ownership | Single | Multiple |
| System management | Centralized | Decentralized |
| User management | Centralized | Decentralized |
| Resource Pool Creation | Yes | Yes |
| Resource node type | Homogeneous | Heterogeneous |
| Resource scheduling | Centralized | Decentralized |
| Network connection type* | LAN | LAN/MAN (Metropolitan area network)/WAN |
| System architecture | Single system image | Autonomous, independent nodes |
| Coupling | Tight | Loose |
| Fault tolerance Capability | Low | High |
| Scaling Architecture | Scalable | Scalable, in bigger capacity |
| Real-time Scaling* | No | No |

## Q7. WHAT IS HARDWARE VIRTUALIZATION?

Hardware virtualization, which is also known as server virtualization or simply *virtualization*, is the abstraction of computing resources from the software that uses those resources. In a traditional physical computing environment, software such as an operating system (OS) or enterprise application has direct access to the underlying computer hardware and components, including the processor, memory, storage, etc.
Virtualization brought true essence in the scaling capability of distributed systems. Load could now be shifted to other (and stronger) set of resources without disrupting service. This provides opportunity for real-time system scaling. System with such characteristic is known as scalable system. Scalability is the ability of a system to accommodate growth. Thus, virtualization coupled with resource pooling technique introduced more force and flexibility in the systems.

**Types of hardware virtualization:**

- **Full virtualization:** In a fully virtualized instance, an application would run on top of a guest OS, which would operate on top of the hypervisor and finally the host OS and hardware. Full virtualization creates an environment similar to an OS operating on an individual server. It enables administrators to combine both existing and new systems. This means, to integrate older systems, hardware must be upgraded to match newer systems. Examples of fully virtualized systems include Oracle VM and ESXi.
- **Paravirtualization:** It enables the VM to differ somewhat from the hardware. The hardware isn't necessarily simulated in paravirtualization but uses an application program interface (API) that can modify guest OS's. Paravirtualization can improve performance by decreasing the amount of VMM calls; however, paravirtualization requires the modification of the OS, which also creates a large dependency between the OS and hypervisor that could potentially limit further updates. For example, Xen is a product that can aid in paravirtualization.
- **Hardware-assisted virtualization**: It was first introduced by IBM in 1972 with IBM System/370. Designers soon realized that virtualization functions could be

implemented far more efficiently in hardware rather than software, driving the development of extended command sets for Intel and AMD processors. So, the hypervisor can simply make calls to the processor, which then does the heavy lifting of creating and maintaining VMs.

## Q8. WHAT DO U MEAN BY CLOUD COMPUTING?

Cloud computing is the delivery of different services through the Internet. These resources include tools and applications like data storage, servers, databases, networking, and software. It is a popular option for people and businesses for a number of reasons including cost savings, increased productivity, speed and efficiency, performance, and security.
One major advantage of cloud is its capability of real-time scaling. Unlike grids, computing resources in cloud can be added in real time to meet demand of computing.
Cloud computing evolution has been an outcome based purely on the technological advancements in different fields of computing.
Cloud computing has emerged through evolution and has brought revolution in the field of computing.

**Types of cloud computing:**

- **Public Cloud:** The public cloud provider owns, manages, and assumes all responsibility for the data centers, hardware, and infrastructure on which its customers' workloads run, and it typically provides high-bandwidth network connectivity to ensure high performance and rapid access to applications and data. Public cloud is a multi-tenant environment—the cloud provider's data center infrastructure is shared by all public cloud customers.

- **Private cloud:** Private cloud is a cloud environment in which all cloud infrastructure and computing resources are dedicated to, and accessible by, one customer only. Private cloud combines many of the benefits of cloud computing—including elasticity, scalability. A private cloud is typically hosted on-premises in the customer's data center. But it can also be hosted on an independent cloud provider's infrastructure or built on rented infrastructure
- **Hybrid cloud:** It is a combination of public and private cloud environments. Specifically, and ideally, a hybrid cloud connects an organization's private cloud services and public clouds into a single, flexible infrastructure for running the organization's applications and workloads. The goal of hybrid cloud is to establish a mix of public and private cloud resources.

### Characteristics of Cloud Computing:

**1) Agility:** The cloud **works in a distributed computing environment. It shares resources among users and works very fast.**

**2) High availability and reliability:** The availability of servers is high and more reliable because the **chances of infrastructure failure are minimum**.

**3) High Scalability:** Cloud offers **"on-demand" provisioning of resources on a large scale**, without having engineers for peak loads.

**4) Multi-Sharing:** With the help of cloud computing, **multiple users and applications can work more efficiently** with cost reductions by sharing common infrastructure.

**5) Maintenance:** Maintenance of cloud computing applications is easier, since they **do not need to be installed on each user's computer and can be accessed from different places**. So, it reduces the cost also.

**6) Low Cost:** By using cloud computing, the cost will be reduced because to take the services of cloud computing, **IT company need not to set its own infrastructure** and pay-as-per usage of resources.

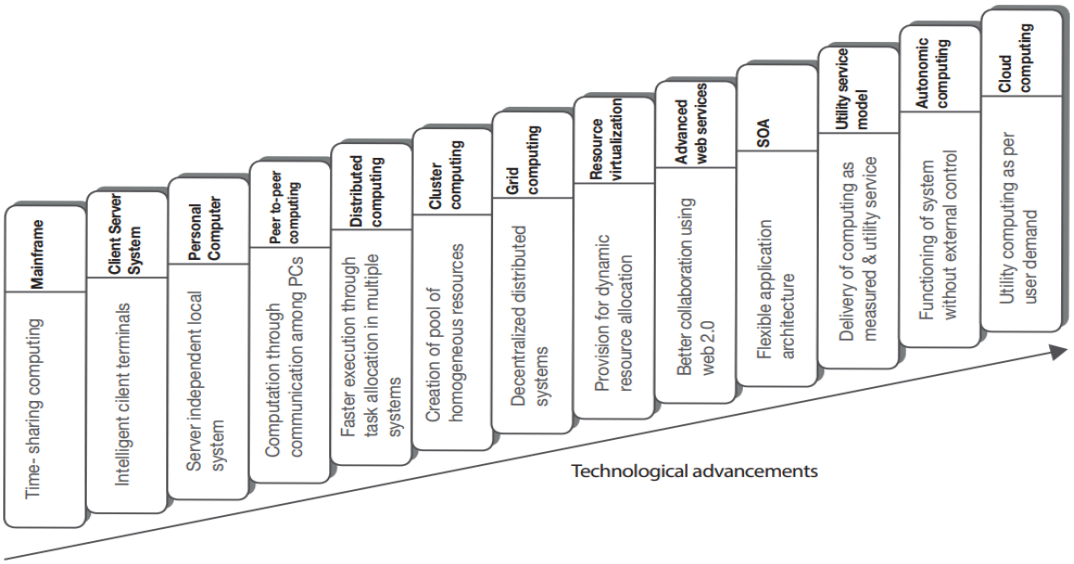



**FIG 2.6:** Technological advancements towards maturity of cloud computing

## Q9. SERVICE ORIENTED ARCHITECTURE (SOA).

Service-Oriented Architecture (SOA) is an architectural approach to designing and developing software systems by organizing them as a collection of loosely coupled and reusable services. In a SOA, software components (services) are designed to perform specific functions and can communicate with each other over a network to accomplish complex tasks.
SOA is more a methodology than a technical approach.
It relies on developing application components as software services as its fundamental principle.
As systems can be developed by integrating services in SOA paradigm, they remain flexible for changes.
This dynamism works well to fit with the need of ever-changing businesses and is considered as an essential foundation for cloud computing.
Adoption of SOA approach in IT system designing makes systems flexible to adapt changes as per business requirements.
The SOA paradigm has been created by using web technologies like XML (Extensible Markup Language), WSDL (Web Services Description Language), SOAP (Simple Object Access Protocol), UDDI (Universal Description, Discovery and Integration).

**Characteristics of Service-Oriented Architecture include:**

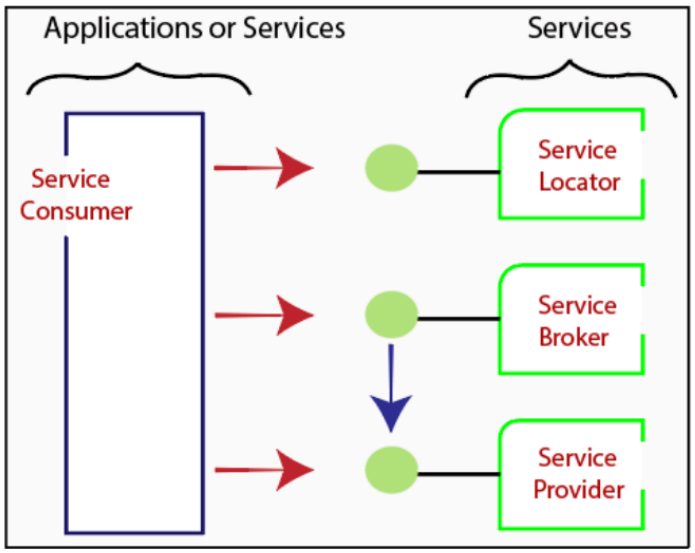
- **Modularity and Reusability:** Services are designed as modular components, each responsible for a specific business function or task. These services can be reused in multiple applications.
- **Loose Coupling:** Services in a SOA are independent entities that communicate through well-defined interfaces (usually using HTTP or SOAP). This loose coupling allows services to be developed, deployed, and maintained independently.
- **Interoperability:** Services can be developed using different programming languages, platforms, and technologies, yet they can still work together seamlessly.
- **Scalability and Flexibility:** As services are designed to be independent, it becomes easier to scale individual services to handle increased demand without affecting the entire system.

**SOA roles:**
The building blocks of a service-oriented architecture are made up of 3 roles.

1. **Service provider:**
   A service provider creates web services and provides them to a service registry. The service provider is responsible for the terms of use of the service.

2. **Service broker or service registry:**
   A service broker or service registry is responsible for providing information about the service to a requester. A broker may be public or private.

3. **Service requester or service consumer:**
   A service requester finds a service in a service broker or service registry and then will connect with the service provider to receive the service.




## Q10. WHAT DO U MEAN BY UTILITY COMPUTING?

The idea OF this model was first presented by John McCarthy, a professor at Massachusetts Institute of Technology (MIT) in 1961 which showed that the computing can be delivered as utility service much like electricity.
This model possesses two important features as the service is available on-demand (as much computing power as required) and the use-basis mode of payment.
Utility computing is a subset of cloud computing, allowing users to scale up and down based on their needs.
For example, a consumer pays his electricity bill as per the number of units consumed, nothing more and nothing less. Similarly, utility computing works on the same concept, which is a pay-per-use model.
It is an uncomplicated, scalable, and cost-effective approach to managing IT needs.
It is a bankable solution toward a rapid digital transformation.

**CHARACTERISTECS OF UTILITY COMPUTING ARE:**

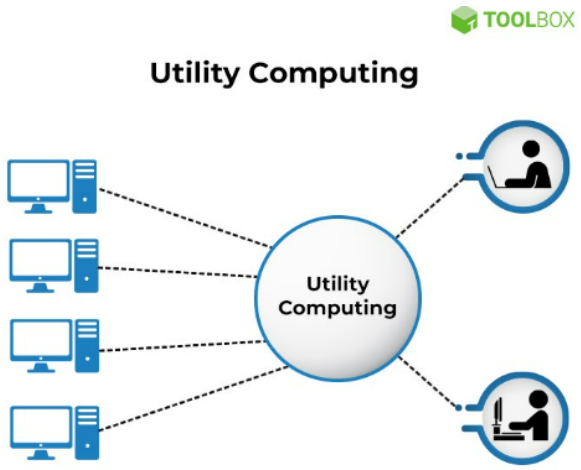
- Scalable computing infrastructure made of heterogeneous resources that can be grown as much as required in real time.
- Single distributed computing environment spread across the globe, empowered by highspeed network.
- Collaborative work facility from different locations, empowered by modern age web service standards.
- Flexible application architecture that is easily modifiable with changing business requirements, empowered by the SOA paradigm.

**Advantages:**

1. **Cost Efficiency:** Utility computing allows organizations to only pay for the resources they use, avoiding the upfront costs of purchasing and maintaining hardware. This can lead to significant cost savings, especially for businesses with varying resource needs.
2. **Scalability:** Resources can be scaled up or down quickly in response to changing demands. This flexibility enables businesses to handle increases in traffic or workload without major investments in infrastructure.
3. **Accessibility:** Resources are accessible from anywhere with an internet connection.

**Disadvantages:**

1. **Dependency on Service Provider:** Organizations relying on utility computing are dependent on the service provider's reliability and performance. Downtime or service outages can impact business operations.
2. **Security and Privacy Concerns:** Storing sensitive data or critical applications on external servers raises security and privacy concerns. Organizations must trust the service provider's security measures.
3. **Limited Control:** Organizations may have limited control over the infrastructure and software stack. Customization options might be constrained by the provider's offerings.
4. **Unpredictable Costs:** While utility computing can offer cost savings, unexpected resource spikes can lead to higher bills. Predicting future costs can be difficult.
5. **Data Transfer Costs:** Moving data in and out of the cloud can incur additional charges, particularly when dealing with large datasets.




## Q11. WHAT DO U MEAN BY AUTONOMIC COMPUTING?

Autonomic computing refers to the ability of an intelligent computing system that can manage itself without any human intervention. These systems can re-configure themselves automatically with changing conditions and protect themselves from any technical failure. Autonomy means the capacity of a system to take its own decisions about its actions. Technologists adopted the idea of autonomic computing from the concept of human nervous system that controls various physiological activities like breathing, sensing etc. without any conscious intervention. The development of autonomic computing system could be possible through application of artificial intelligence (AI) concepts.
The concept of autonomic computing was first presented by IBM in 2001. IBM has defined four areas of automatic computing as shown in Figure.

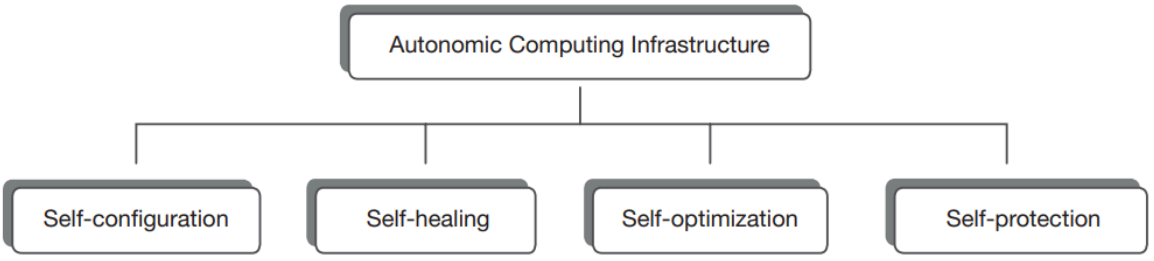



**FIG 2.5:** Characteristics of autonomic computing defined by IBM

Self-configuration is the ability to configure a system based on requirements without any external intervention.
Self-healing is the capability of discovering, diagnosing and correcting errors.
Self-optimization is the ability of a system to automatically control resources for optimal utilization while producing outcome in timely manner with changing business scenarios.
Self-protection means the capacity to identify any occurrence of hostile behaviour (virus infection, denial-of-service attack or unauthorized access etc.) and take corrective actions to make the system less vulnerable.
Development of autonomic computing system has been possible through careful application of artificial intelligence.

Autonomic computing concept has played a very important role in developing cloud computing systems as resource management at cloud data centers.

**Advantages:**

1. It is an open-source.
2. It is an evolutionary technology that adapts itself to new changes.
3. It is optimized hence gives better efficiency and performance thereby taking lesser time in execution.
4. It is very secure and can counter system and security attacks automatically.
5. It has backup mechanisms that allow recovery from system failures and crashes.
6. It reduces the cost of owning (Total Cost of Ownership) such a mechanism as it is less prone to failure and can maintain itself.
7. It can set up itself thereby reducing the time taken in manual setup.

**Disadvantages:**

1. There will always be a possibility of the system crashing or malfunctioning.
2. This would result in an increase in unemployment due to the lesser needs of people after it is implemented.
3. The affordability would be an issue because it would be expensive.
4. It would need people who are very skilled to manage or develop such systems, thereby increasing the cost to the company that employs them.
5. It is dependent on internet speed. Its performance decreases with a decrease in internet speed.
6. It would not be available in rural areas where there are lesser provisions of stable internet connection.

## Q12. COMPARISON BETWEEN CLUSTER, GRID AND CLOUD COMPUTING.

**Table 2.2** Comparison among cluster, grid and cloud computing

| *Cluster Computing* | *Grid Computing* | *Cloud Computing* |
|---|---|---|
| A cluster is normally formed with computers of a single location, otherwise the system becomes complex. | Grid is inherently more distributed by its nature. The computers need not to be in the same geographical location. | It allows total distribution of resources like the grids. Hardware resources are maintained in multiple data centers spread across the globe. |
| Computation job takes place in one administrative domain owned by a single party. | Computation could occur over many administrative domains owned by multiple parties as connected together. | Computing resources of a cloud is usually owned by a single party. But multiple administrative domains can be combined together to perform the job. |
| In a cluster, all computing nodes should have similar hardware systems. That is, the system should be homogeneous in nature. | It can be heterogeneous in nature. The computers that are part of a grid can be made of different hardware architectures. | It can use heterogeneous collection of commodity hardware. |
| It features the centralized task management and scheduling system. | It features the distributed task management and decentralized scheduling. | It features the decentralized task management with more dynamic computing infrastructure. |
| Resources are generally pre-reserved for specific type of task. | Resources are generally pre-reserved for specific type of task. | Resources are not pre-reserved for specific task. Resource utilization is mainly demand-driven. |
| System is not dynamic in nature. Application mobility is not possible. | System is not dynamic in nature. Application mobility is not possible. | It is a dynamic system. Mobility of application is an inherent feature in this system. |
| One whole cluster behaves like a single system. As resources are managed by centralized resource manager, the individual computers can not be operated as separate computers. | Every node is autonomous that is, it has its own resource manager and behaves like an independent entity. So, each computer can be operated independently as distinct computer. | There is no concept of directly accessing any particular physical computing nodes. Underlying computing infrastructure remains hidden from the users. |

MCQ'S ON PG 62 OF 434

# CHPT 3:

## Q1. WHAT DO U MEAN BY METERING AND BILLING IN CLOUD?

Metering and billing in the context of cloud computing refer to the processes by which cloud service providers track and charge customers for their usage of various cloud resources and services. In a cloud environment, resources such as virtual machines, storage, networking, and other services are made available to customers on a pay-as-you-go or subscription basis. Metering and billing mechanisms play a crucial role in ensuring transparency, accurate accounting, and fair pricing for these services.

**Key concepts related to cloud metering and billing include:**

- **Pay-as-You-Go**: This is a common billing model where customers are charged based on the actual amount of resources they use. The more resources they consume, the higher their bill will be.
- **Subscription**: In a subscription-based model, customers commit to a certain level of resource usage over a specified period (e.g., monthly or annually). They pay a fixed amount regardless of whether they fully utilize the allocated resources.
- **Resource Types**: Different types of cloud resources (e.g., compute instances, storage, databases) have different pricing structures based on their capabilities and performance.
- **Resource Usage Reports**: Cloud providers typically provide customers with detailed usage reports that break down their consumption of various resources. This transparency helps customers understand their spending and optimize resource usage.
- **Cost Estimation**: Some cloud platforms offer tools that allow customers to estimate potential costs before provisioning resources. This helps in planning and budgeting.

These mechanisms provide customers with the flexibility to choose the right resources for their needs while controlling costs.

## Q2. WHAT ARE THE BENEFITS OF CLOUD COMPUTING?

I. **Less Acquisition/Purchase Cost:**
In traditional computing, users have to purchase computing resources in significant amount in the beginning. Cloud computing is delivered following the utility service model. Since vendor arranges all necessary resources in this model, subscribers initial investment for acquiring hardware or software drops down drastically. They need not to arrange anything apart from client systems to access cloud services.

II. **Reduced Operational Cost:**
With outsourcing model of utility computing, the cost of running any systems round the clock moves towards the provider's end. Subscribers get rid of the responsibility of system administration, maintenance, and 24 × 7 energy support as well as its cooling support. This is a basis for cost savings because subscribers can use the service by paying very nominal. The provider on the other hand can offer the service at nominal fee to subscribers.

III. **Reduced System Management Responsibility:**
Be it a data center for enterprises or single standalone machine (PC, laptop etc.) for normal users, management of the computing setup (both hardware and software) is an extra headache for consumers of traditional computing. Cloud computing model shifts majority of the infrastructure and other system management tasks towards cloud vendors. Dedicated teams at the vendor's end takes care of all of these activities.

IV. **Use-basis Payment Facility:**
Cloud computing does not charge its subscribers when they do not use it. Even the charge is not fixed; it depends on the duration of usage. Rather, any use is metered and users are charged a reasonable fee according to their consumption.

V. **Unlimited Computing Power and Storage:**

In cloud computing, users can easily access supercomputer like computing power at reasonable cost, if necessary. Earlier in traditional approach, only big corporate could afford high-end computing. Storage is another important issue for users. Cloud provides as much storage as required. It is virtually unlimited which is viewed as a big benefit for users.

VI. **Quality of Service:**
In traditional computing, enterprises often used to outsource major portion of computing related jobs to some third party. Thus, service quality was broadly dependent on the expertise of those third parties or the in-house teams managing it. Whereas in cloud computing, high quality of service (QoS) is ensured as it is provided by renowned computing vendors having well-trained staffs and expertise exclusively in the field of computing.

## Q3. WHAT ARE THE CHALLENGES OF CLOUD COMPUTING?

- **Limited Portability between Cloud Providers:**

Different vendors are coming up with cloud computing facility for public use which is mostly proprietary to various extents. Applications developed on these proprietary clouds are difficult to move to other cloud platform due to vendor lock-in. This problem limits portability of applications.

- **Inter-operability Problem:**
It is the ability of a system to work with other systems. The proprietary issue discussed above not only raises portability problem, it also restricts applications of two different clouds to interoperate with each other. This is known as the problem of interoperability. Subscribers may find two different applications from two different cloud vendors suitable for their requirement.

- **Data Security:**
In cloud computing, users or enterprises need to store data outside their network boundary protected by firewalls. Thus, the trust boundary of enterprises expands up to the external cloud. Security of users data largely depends on the cloud vendors. This may introduce some extent of vulnerabilities to the security of data.

- **Reduced Control over Governance:**
Cloud computing is built and governed by the policies of computing vendor or service provider. Consumers are relieved of the tiring responsibility of managing the computing system. While this turns out as a major benefit, the low control over the governance or authority of computing environment sometimes raises concerns among consumers who used to enjoy full control over selfowned traditional data centers.

- **Multi-Regional Compliance and Legal Issues:**
he privacy or compliance rule generally differs across different legal jurisdictions. The rules for degree of disclosure of personal data to government agencies (in cases of some official investigations) differ from country to country, or even state to state within a country. Situation may arise where the law of the country of a cloud subscriber asks for some data to be disclosed where the law of hosting region of the cloud (that is the region/country of cloud data center) does not allow such disclosure.

Multi-regional legal issues raise concern over information privacy and compliance related problems in cloud computing.

- **Bandwidth Cost:**
  In the current age of Internet, cost of bandwidth is very low at moderate speed of access. But more bandwidth can provide higher speed which is essential for high quality service. While low cost bandwidth may often fulfill requirements of general applications, data intensive applications (those deal with critical and huge volume of data sets) demand higher bandwidth which may add a little more in the total cost of computing.

## Q4. WHAT IS THE ROLE OF A WEB SERVICE?

A web service is the way of establishing communication between two software systems over the internetwork. Web services use standardized way of data exchange since different software systems might be built using different programming languages and run on different platforms. Thus, this standardization is very important so that communication remains independent of programming languages or platforms.

It describes the method of establishing communication between two web-based applications. World Wide Web Consortium (W3C) defines web services as "a software system designed to support interoperable machine-to-machine interaction over a network". Application of web services is an essential part of cloud computing development.

Web services are generally categorized into two different classes:

- Simple Access Object Protocol (SOAP) based
- Representational State Transfer (REST) compliant.

o **SOAP-Based Web Services:**

SOAP-based web services use XML format for communicating messages among web applications as XML is an open format and recognized by all kind of applications. In this approach, HTTP or hyper-text transfer protocol is used for passing messages. The SOAP is originally developed by Microsoft as older Remote Procedure Call (RPC)- based message passing technologies like DCOM (Distributed Component Object Model) or CORBA (Common Object Request Broker Architecture) did not work well with Internet. This was primarily because those technologies relied on binary messaging. On the other hand, the XML format of messaging performs well over Internet. The SOAP was accepted as standard when Microsoft submitted it to the Internet Engineering Task Force (IETF). The rules of SOAP communications are described in Web Services Description Language (WSDL) format. The rules are stored in files with .wsdl extension.

o **REST: Compliant Web Services:**

REST represents a simpler way of communicating messages. SOAP is often considered as complex since creation of the XML structure, REST provides a light weight alternative. REST relies on global identifier to locate the resources. Thus, a separate resource discovery mechanism is not needed. This 'representation' of paths of applications provides the additional power in REST. REST allows many standard formats like XML, JavaScript Object Notation (JSON) or plain text as well as any other agreed upon formats for data exchange. REST is an architecture style for designing networked applications. Here, simple HTTP is used to make calls between machines identified by

their URLs (Uniform Resource Locators) which is simpler than other mechanisms like CORBA, DCOM or SOAP.

## Q5 EXPLAIN SIMPLE OBJECT ACCESS PROTOCOL (SOAP) IN DETAIL.

SOAP (Simple Object Access Protocol) is a message protocol that enables the distributed elements of an application to communicate. SOAP can be carried over a variety of standard protocols, including the web-related Hypertext Transfer Protocol (HTTP).

SOAP was developed as an intermediate language for applications that have different programming languages, enabling these applications to communicate with each other over the internet. SOAP is flexible and independent.

The SOAP is originally developed by Microsoft as older Remote Procedure Call (RPC)-based message passing technologies like DCOM (Distributed Component Object Model) or CORBA (Common Object Request Broker Architecture) did not work well with Internet.

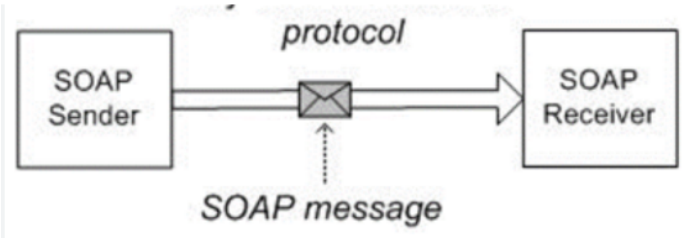
This was primarily because those technologies relied on binary messaging. The SOAP was accepted as standard when Microsoft submitted it to the Internet Engineering Task Force (IETF). The rules of SOAP communications are described in Web Services Description Language (WSDL) format. The rules are stored in files with .wsdl extension.

**Advantages of SOAP:**

- **Platform- and operating system-independent.** SOAP can be carried over a variety of protocols, enabling communication between applications with different programming languages on both Windows and Linux.
- **Works on the HTTP protocol.** Even though SOAP works with many different protocols, HTTP is the default protocol used by web applications.
- **Can be transmitted through different network and security devices.** SOAP can be easily passed through firewalls, where other protocols might require a special accommodation.

**Disadvantages of SOAP:**

- **No provision for passing data by reference.** This can cause synchronization issues if multiple copies of the same object are passed simultaneously.
- **Speed.** The data structure of SOAP is based on XML. XML is largely human-readable, which makes it fairly easy to understand a SOAP message. However, that also makes the messages relatively large compared to the Common Object Request Broker Architecture (CORBA) and its remote procedure call (RPC) protocol that will accommodate binary data. Because of this, CORBA and RPC are faster.
- **Not as flexible as other methods.** Although SOAP is flexible, newer methods, such as RESTful architecture, use XML, JavaScript Object Notation, YAML or any parser needed, which makes them more flexible than SOAP.




## Q6. EXPLAIN REPRESENTATIONAL STATE TRANSFER (REST) IN DETAIL.

REST (***Re**presentational **S**tate **T**ransfer*) was first presented in the year 2000 by ***Roy Fielding*** as an architectural style for ***distributed hypermedia systems.*** REST-compliant or RESTful systems, are ***"stateless"*** and separate a client and a server. A web application developed using REST exposes the data or information about its resources which can be anything that the developer wants. For example, using information exposed to users, clients can create a new user.
REST is an architecture style for designing networked applications. HTTP is used to make calls between machines identified by their URLs (Uniform Resource Locators) which is simpler than other mechanisms like CORBA, DCOM or SOAP.

**Advantages and disadvantages of REST:**
**There are several advantages to using REST. They are as follows:**

- **Resource-based.** A primary benefit of using REST, from both a client and server perspective, is REST interactions are based on constructs which are familiar to anyone accustomed to using HTTP. Employing a resource-based approach, REST defines how developers interact with web services.
- **Communication.** REST-based interactions communicate their status through numerical HTTP status codes. REST APIs use these HTTP status codes to detect errors and ease the API monitoring process. They include the following:

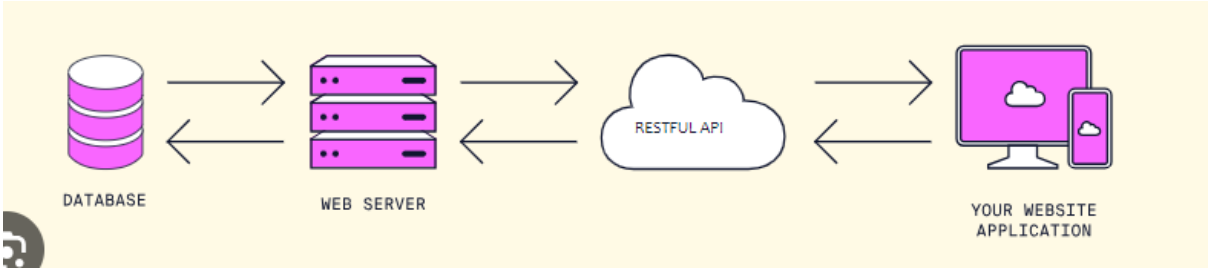
404 error indicates that a requested resource wasn't found;
401 status response code is triggered by an unauthorized request;
200 status response code indicates that a request was successful; and
500 error signals an unrecoverable application fault on the server.

- **Familiarity.** Most developers are already familiar with key elements of the REST architecture, such as Secure Sockets Layer (SSL) encryption and Transport Layer Security (TLS).
- **Language-independent.** When creating RESTful APIs or web services, developers can employ any language that uses HTTP to make web-based requests.
- **Web APIs.** When it comes to caching, RESTful services employ effective HTTP mechanisms.
  For example, by providing many endpoints, a REST API makes it easier for developers to create complex queries that can meet specific deployment needs.

**Disadvantages of REST are as follows:**

- **Architecture.** Developers working with REST frequently encounter limitations due to its architecture design. These include multiplexing multiple requests over a single TCP connection, having different resource requests for each resource file, server request uploads, and long HTTP request headers, which cause delays in webpage loading.

- **Stateless applications.** Since HTTP does not store state-based information between request-response cycles, the client must perform state management tasks. This makes it difficult for programmers to implement server updates without the use of client-side polling or other types of webhooks that send data and executable commands from one app to another.
- **Definition.** Developers generally disagree over defining REST-based designs. As an architectural style, REST lacks a clear reference implementation or a definitive standard that designates whether a specific design can be defined as RESTful. This also leads to uncertainty over whether a given web API conforms to REST-based principles.
- **Data overfetching/underfetching.** RESTful services frequently return large amounts of unusable data combined with relevant information, typically the result of multiple server queries. These inefficiencies also increase the time it takes for a client to return all the required data.




## Q7. DIFFERNCE BETWEEN SOAP AND REST.

| *SOAP* | *REST* |
|---|---|
| It is a message communication protocol that is used to build network application. | It is an architectural style for network application. |
| It uses XML for formatting message. | It uses simpler messaging formats like JSON. It also supports XML. |
| It requires the XML parsing. | Here, the XML parsing can be avoided by using JSON. |
| SOAP defines its own additional security over HTTPS. | It inherits security measures from the underlying transmission protocol as HTTPS. |
| It is a bit heavy. | It is lighter than SOAP and response time is better. |
| It is complex than REST. | It is simple to use and easy to understand. |
| It is recognized as a standard protocol. | It is not an standard, rather an architectural style. |
| SOAP uses services interfaces to expose the business logic. | REST uses Uniform Resource Identifier (URI) to expose business logic. |
| SOAP oriented development is faster since it is supported by many tools. | Development time is longer due to limited tool support. |
| It shows lesser flexibility in controlling resources. | It shows more flexibility in controlling resources. |
| It requires less knowledge in programming. | It requires rather greater knowledge in programming in comparison with SOAP system. |

## Q6. WHAT IS THE ROLE OF AN API?

API (Application Program Interface) is a set of defined functions or methods which is used to compile the application. A programmer can make use of various API tools to make their

program easier and simpler. Also, an API facilitates programmers with an efficient way to develop their software programs. Thus in simpler terms, an API helps two programs or applications to communicate with each other by providing them with the necessary tools and functions.

APIs play important role in cloud computing. When some cloud services are released, corresponding APIs (referred as cloud API) are also released as they are critical for the usefulness and operational success of those services. Cloud services generally provide welldefined APIs for its consumers so that anyone can access and use the capabilities offered to develop application or service.

Cloud APIs expose their features via REST or SOAP.

**Some key roles and functions of APIs:**

- **Interoperability:** APIs enable different software systems, even if they are built using different programming languages or technologies, to work together. This allows for the creation of more comprehensive and feature-rich applications by integrating existing services.
- **Abstraction:** APIs abstract the underlying complexity of a system. Instead of needing to understand the inner workings of a service, developers can interact with it through a well-defined interface, making development more efficient.
- **Reusability:** APIs promote code reuse. Developers can build upon the functionality of existing APIs without needing to reinvent the wheel, saving time and effort.
- **Security:** APIs can offer controlled access to data or services, ensuring that sensitive information remains protected and only accessible to authorized users.
- **Platform Independence:** APIs can be used to build cross-platform applications. For example, a service can provide APIs for both web and mobile applications, enabling consistent functionality across different devices.

## Q7. WHAT DO U MEAN BY UBIQUITIOUS CLOUD?

The term ‘ubiquitous’ means as being everywhere. The idea of ubiquitous computing talks about making computing facility available everywhere and for all the time.

The underlying technologies to support ubiquitous computing include embedded computing devices (electronic chips), networks and Internet among other things. The ubiquitous computing paradigm is also known as pervasive computing. In comparison with desktop computing, interaction with computing environment in ubiquitous computing can happen using any devices having some computing capability of any form. Cloud computing further strengthens the idea of ubiquitous computing. Ubiquitous cloud refers to the use of computing resources spread over geographic locations from any place and any time.

**Key characteristics of the Ubiquitous Cloud include:**

1. **Accessibility:** Cloud services and resources are accessible from any device with an internet connection, such as smartphones, laptops, tablets, IoT devices, and more.
2. **Pervasiveness:** The cloud is not limited to a single physical location or data center. Instead, cloud services are distributed across multiple data centers and regions, providing redundancy and improved availability.

3. **Seamless Integration:** Cloud services are integrated seamlessly into applications and platforms, allowing users to access and use them without requiring specialized knowledge or skills.
4. **Scalability:** Cloud resources can be dynamically scaled up or down based on demand, ensuring optimal performance and resource utilization.
5. **Interconnectivity:** The Ubiquitous Cloud enables different devices and services to communicate and interact with each other, forming a cohesive ecosystem of interconnected devices and applications.
6. **Data Sharing:** Cloud services facilitate easy sharing and synchronization of data across devices, enabling users to access their information from anywhere.

MCQ'S ON PG 80 OF 434

# CHPT 4:

## Q1. WHAT DO U MEAN BY NIST MODEL?

The most appreciated and accepted model of cloud computing was provided by the National Institute of Standards and Technology (NIST) of U.S. The model was published in a document titled as 'NIST Cloud Computing Reference Architecture' by Information Technology Laboratory of NIST in 2011. Following is the statement by the NIST: "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is comprised of five essential characteristics, three service models, and four deployment models."

**Salient points:**

- Cloud computing is a model and not a technology.
- Cloud computing enables the users' access pools of computing resources via network.
- The resources are shared among users and made available on-demand.
- The prime benefit is the ease of use with very little management tensions for the users.

The first two among the four points are self-explanatory. The third point says that no user can hold any resource exclusively unless required for computational or associated tasks. The last point states that the whole thing will basically be managed by a third party referred as provider party and users will simply use it without the responsibility of managing it.
The cloud computing initiative at NIST started in November 2010. The goal of the initiative was to boost the US Government's effort to incorporate cloud computing to enhance the traditional approach of information system maintenance, wherever applicable. NIST being a U.S. government organization did not limit their cloud computing works within the organization only. Rather they created the model with the intention of making it broadly usable and applicable to different organizations across the globe. The NIST model of cloud computing was published by the Information Technology Laboratory at NIST in 2011.

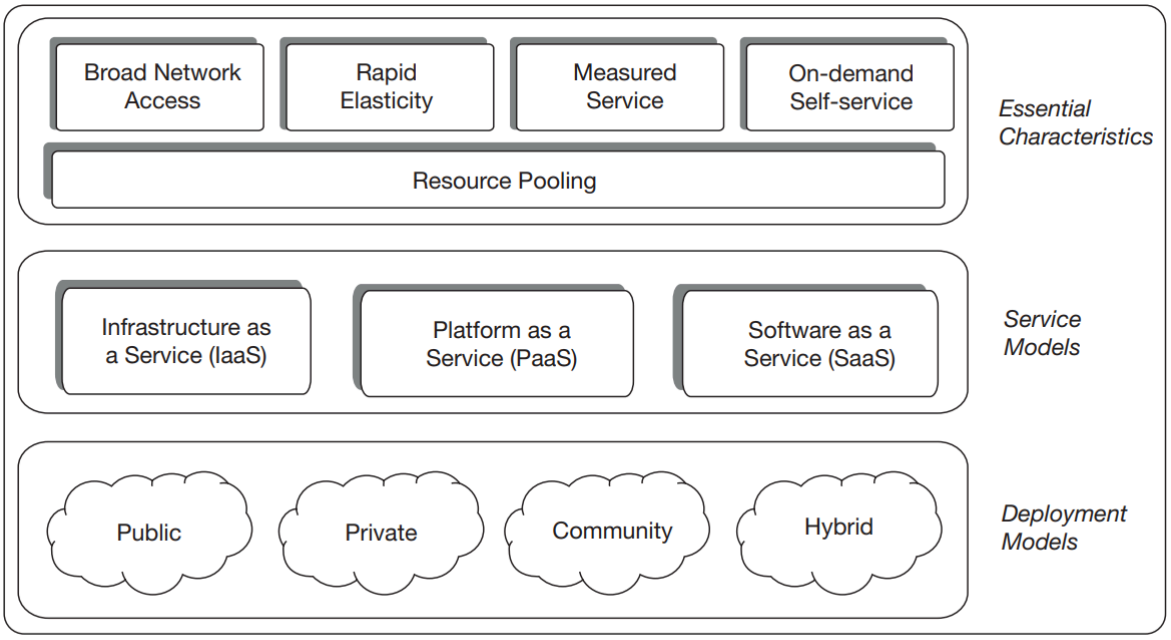



**FIG 4.1:** The NIST cloud computing model

## Q2. WHAT ARE THE ESSENTIAL CHARACTERISTICS OF NIST MODEL?

The NIST model of cloud computing comprises 5 essential characteristics, which differentiates the cloud model from traditional computing approach.

**Essential Characteristics:**

- **On-demand self-service:**
  It is the most attractive feature that users like about this computing model. The on-demand service feature refers to the ability that empowers users to consume the computing facility as much they need at any moment. Being self-service, cloud computing can arrange the on-demand facility for users without any need of human intervention at vendor's end. A user himself/herself can request cloud services as needed through some interface (generally through web forms) and resources become available within seconds. This feature is known as self-service. The self-service interface must be user-friendly in order to be effective and appealing.
- **Resource pooling:**
  Computing requires resources like processor, memory, storage and network. Cloud computing arranges these resources for users at vendor's end. Users can access and use these resources to satisfy their computing needs as and when required. Unlike traditional computing approach where every enterprise or user possesses its own physical computing resources, here pools of computing resources are maintained at remote locations by the provider which is accessed by all of the users. The resource pools must be reasonably large, flexible and capable of supporting many users simultaneously without any failure.
- **Broad network access:**
  Cloud computing provides economic advantage to users as it releases them from the inconvenience of setting-up expensive in-house data centers. Instead, the cloud service facility developed and installed at the provider's end is remotely accessed by users through the network. To serve this purpose, strong network infrastructure has to

be in place for effort-less and fast delivery of the computing services. Thus, high bandwidth communication links spread over the service area are the essential attributes of cloud computing so that users can access computing from any location and anytime.

- **Rapid elasticity:**
  Provisioning of adequate and frequently changing demand of resources for a large number of users is a major technical concern in cloud computing. Provider may not know when and how much of resources users will consume prior to actual demand. But the mechanism should be such that the required volume of resources can be arranged at the time of demand from the users. The computing environment must create an impression of limitless repository of resources to users, and they should be able to consume any volume of resources any time. Again when a user no more uses the resources, those have to be taken back immediately so that there is no wastage of valuable resources through idle possessions. From users' point of view, the system has to be elastic enough. It should be able to grow and shrink according to the requirement. Rapid elasticity refers to this ability of the cloud where a computing system can expand or reduce itself rapidly in accordance with the actual resource requirement at runtime.
- **Measured service:**
  As users use computing services provided by cloud vendor, they must pay for it. In cloud computing model, this payment is determined by measuring the usages of computing resources by a user. Hence, the provider must employ some mechanism to measure the actual consumption by each individual user or organization. This means that the usage of the pooled resources has to be calculated and stated (or billed) to every user based on a metering system. Generally this is done on some known metric such as amount of processing power consumed, use of storage volume, network bandwidth used, number of network transactions etc. Any user is billed based only on the actual consumption of cloud resources or for resources which were allotted to him/her.

## Q3. WHAT DO U MEAN BY MULTI-TENANCY?

Multi-tenancy in simple form implies that a single set of resources can have multiple tenants who are not linked with each other. This statement about multi-tenancy perfectly fits in public cloud environment but does not apply in private deployments with its full essence or ability since all of the users there are internal to a single organization or remains under a single body. This is the reason why multi-tenancy is not mentioned as an essential attribute of cloud computing by NIST. But it is an important characteristic of cloud computing. For instance, Cloud Security Alliance (CSA), an industry working group that studies security issues in cloud computing, identifies multi-tenancy as a key element of cloud model.

**Benefits of multitenant architecture**

- **Multitenancy can save money.** Computing is cheaper at scale, and multitenancy allows resources to be consolidated and allocated efficiently, ultimately saving operational costs. For an individual user, paying for access to a cloud service or a SaaS application is often more cost-effective than running single-tenant hardware and software.

- **Multitenancy enables flexibility.** If you've invested in your own hardware and software, it might reach capacity during times of high demand or sit idle during times of slow demand. A multitentant cloud, on the other hand, can allocate a pool of resources to the users who need it, as their needs scale up and down. As a customer of a public cloud provider, you can access extra capacity when you need it, and not pay for it when you don't.
- **Multitenancy can be more efficient.** Multitenancy reduces the need for individual users to manage infrastructure and handle updates and maintenance. Individual tenants can rely on a central cloud provider, rather than their own teams, to handle those routine chores.

**Some disadvantages of multi-tenancy include the following:**

- Apps delivered by a provider tend to be less flexible than apps in other tenant architectures, such as single-tenancy.
- It is, in general, more complex than single-tenancy because of the additional virtualization and management needed to isolate and secure each tenant.
- Apps need stricter authentication and access controls for security.
- Tenants have to worry about noisy neighbours, which might slow response time for other tenants sharing the same resources.
- Downtime might also be an issue, depending on the provider. Collateral impacts such as one server problem affecting many users can also be an undesirable risk.

## Q4. WHAT DO U MEAN BY REFERENCE ARCHITECTURE?

The NIST cloud reference architecture is a logical extension to the NIST cloud computing definition. The reference architecture was published in September 2011. The reference architecture of NIST does not model system architecture of any particular cloud. Rather it intends to simplify the conception of the operational details of cloud computing. The architecture focusses on 'what' cloud services need to provide but not 'how to' do that. The following Fig. represents the NIST reference-based architecture.

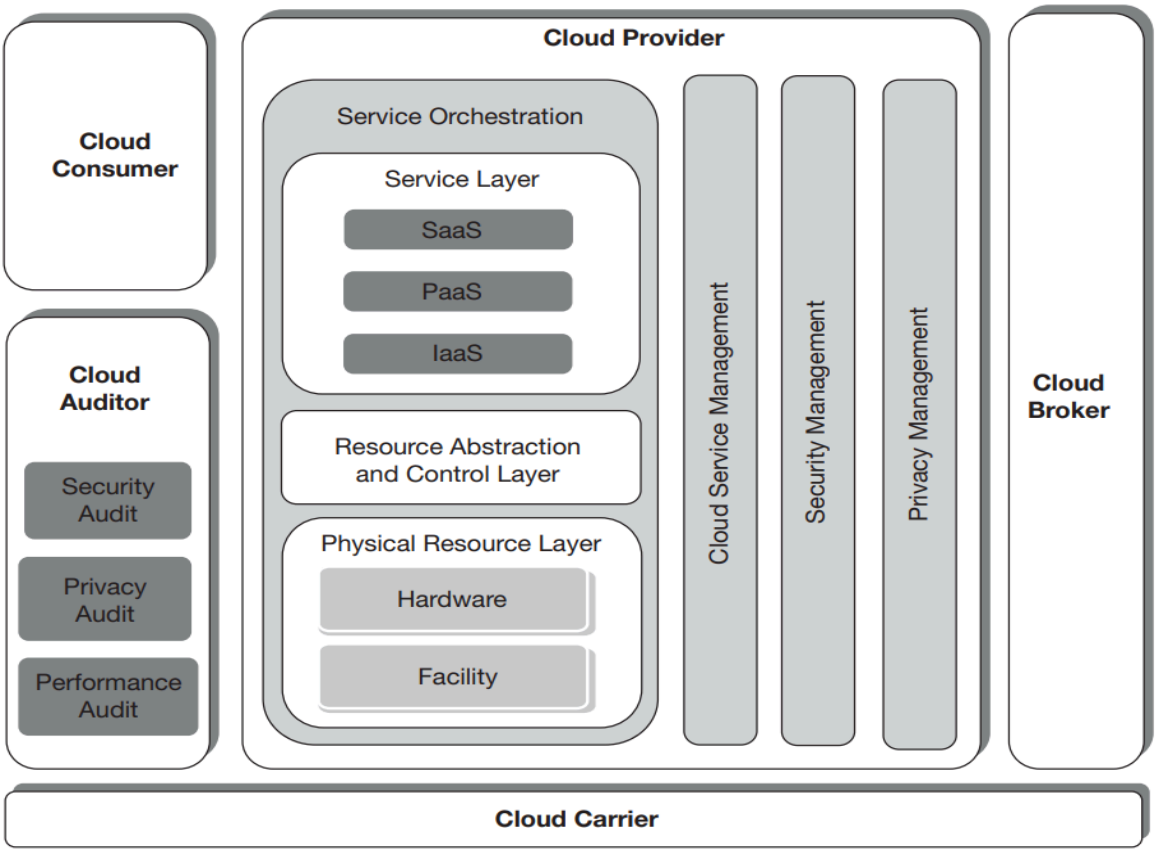



**FIG 4.2:** NIST cloud computing reference architecture[2]

The diagram depicts a generic high-level architecture and represents an actor or role-based model. The five major actors of the model are cloud consumer, cloud provider, cloud broker, cloud auditor and cloud carrier. Along with the actors, the model also identifies their activities and functions. This helps in understanding the responsibilities of the actors.

## Q5. WHAT DO U MEAN BY ACTORS OF THE NIST MODEL?

The NIST cloud computing model describes five major actors as shown in Figure. These actors play key roles in the cloud computing business. Each actor in the reference model is actually an entity; that is, either a person or an organization. The entities perform some tasks by participating in transactions or processes.

1. **Cloud Consumer:**

According to the definition of NIST, 'The cloud consumer is the principal stakeholder for the cloud computing service. A cloud consumer represents a person or an organization that maintains a business relationship with, and uses the service from a cloud provider.'3 The cloud consumer uses cloud service and may be billed for the service by the provider.

2. **Cloud Provider:**

According to NIST, 'A cloud provider is a person or an organization; it is the entity being responsible for making a service available to interested parties. A Cloud Provider acquires and manages the computing infrastructure required for providing the services,...'.3 Here the interested parties who want service from cloud provider are the consumers.

3. **Cloud Auditor:**

The cloud services provided by cloud provider to the cloud consumer must comply to some pre-agreed policies and regulations in terms of performance, security etc. The verification of these agreed conditions can be performed by employing a third-party auditor. The cloud auditor is a party who can conduct independent assessment of cloud services and report it accordingly.

4. **Cloud Broker:**

Usually, there are enormous numbers of service providers and many similar type of services are available from different providers. This may raise confusion among the consumers regarding the uses and management of the services. Moreover, consumers may not be aware about all of the available services and their performances. Even, consumers may find two different services useful from two different providers which would have to be integrated as well. Here comes the role of the cloud broker. According to NIST, 'A cloud broker is an entity that manages the use, performance, and delivery of cloud services and negotiates the relationships between cloud providers and cloud consumers.'3 Consumers can avoid the responsibilities of those complex tasks by requesting services from brokers instead of consuming services from providers directly.

5. **Cloud Carrier:**

Cloud computing services are delivered from cloud provider to cloud consumer either directly or via some cloud broker. Cloud carrier acts as an agent in this delivery process. They are the organizations who provide the connectivity and transport facility of services through their network.

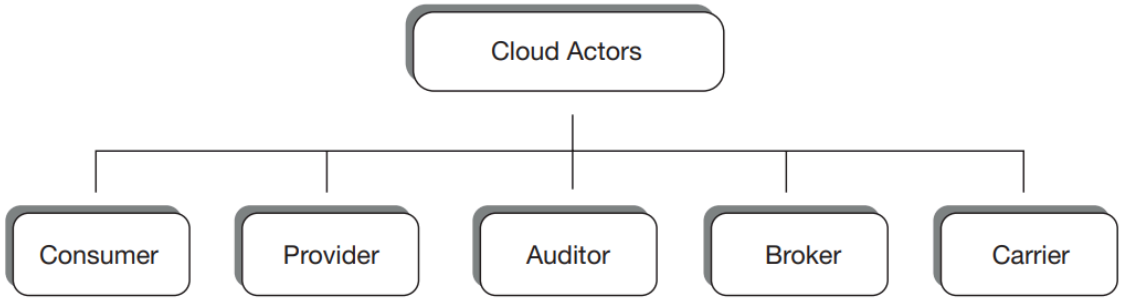



**FIG 4.3:** Actors of NIST cloud computing reference architecture

The role of each actor can be played by a single person; by a group of people or an organization. The actors work in close association with each other.

A cloud consumer may directly request for service to a cloud provider. The provider then delivers the requested services to the consumer. This communication between consumer and provider occurs through the carriers.

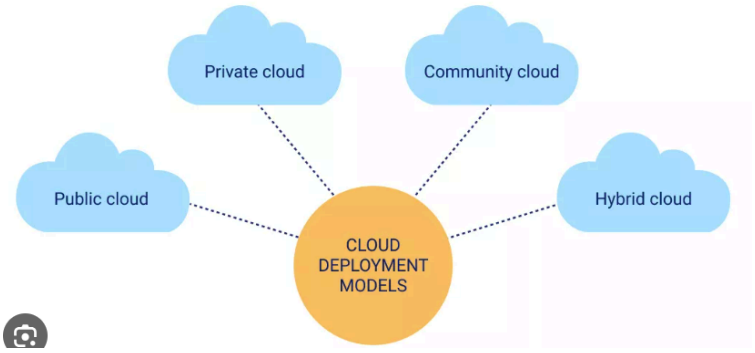
Instead of contacting a cloud provider directly, a cloud consumer also has the option of requesting for services to some cloud brokers. Cloud broker usually integrates the required services from provider and delivers it to the consumer. In this case, the actual cloud providers remain invisible to the cloud consumers.

The broker has been linked with two providers. In such scenario, the cloud broker may create a new service by combining services of those two providers. Cloud broker establishes and eases the interaction between consumers and providers.

## Q6. WHAT DO U MEAN BY CLOUD DEPLOYMENT MODEL?

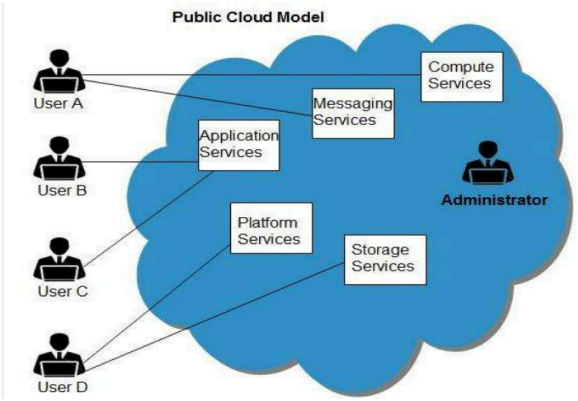
## Q7. WHAT DO U MEAN BY CLOUD AND WHAT ARE THE DIFFERENT TYPES OF CLOUD?

Cloud services can be arranged or deployed in a number of ways. The deployment choice depends on the requirements of the consumer organization. The deployment model describes the utility of a cloud and also specifies its access boundary. The model also indicates the relative location of the cloud with respect to the location of consumer organization.




- **Public Cloud:**

The public cloud deployment model provides the widest range of access to consumers among all cloud deployments. Anyone who subscribes it gets open access to this cloud facility. The consumer can either be an individual user or a group of people representing some organization or an enterprise. Public cloud is also referred as external cloud as physical location-wise it remains external or off-premises and the consumers can then remotely access the service. A public cloud is hosted and managed by some computing vendors who establishes data centers to provide the service to consumers. The consumers under this cloud deployment model are entirely free from any tensions of infrastructure administration and system managementrelated issues. But, at the same time they (consumers) would have low degree of control over the cloud. Amazon Web Services, Google Cloud, Microsoft Azure and Salesforce.com are some of the popular public clouds. Public cloud deployment promotes multi-tenancy at its highest degree. When a large number of consumers dispersed around the world share resources from data center of a single vendor that automatically increases resource utilization rates and decreases vendor's cost of service delivery. Thus for the consumers, the key benefit of using public cloud is its financial advantage. The public cloud providers on the other hand, make advantage of the magnitude of their operation. Being large in volume and business, they can afford state-of-the-art technology and skilled people. This ensures better quality of service.




- **Private Cloud:**

The private cloud deployment does not provide open access to all. It is mainly for organizational use and access to a private cloud deployment is restricted for general public. Private cloud is also referred as internal cloud since it is built to serve internal purpose of the organizations. While public clouds are equally useful for both individual users and organizations, private cloud generally serves the purposes of organizations only. For high-security and critical systems, like systems of defense organizations, private cloud is the suggested approach. While a public cloud cannot physically reside at any consumer's location (physical boundary), private clouds may reside either inside consumer organization's premises (on-premises) or outside (off-premises) at any neutral location. On-premises private clouds physically reside under consumer organization's own physical as well as inside the network boundary. Off-premises private clouds reside outside organization's own network boundary but remains under the control or supervision of the consumer organization. A private cloud may be established and managed by the consumer organization itself or they (the consumer) may outsource the responsibility to some other computing vendor. Figures 4.7 and 4.8 represent on-premises and off-premises private clouds respectively. One major difference of private cloud with public cloud is that any private cloud shares one-to-one relationship with consumer while a public cloud maintains one-to-many relationship. The other differentiating point arises over the ability of consumer to control the cloud.

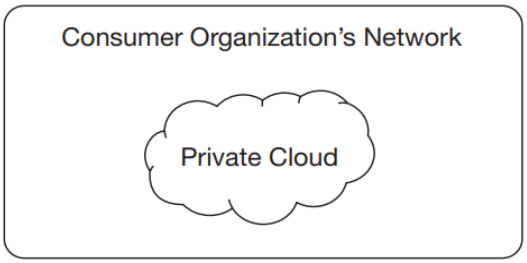



**FIG 4.7:** On-premises private cloud

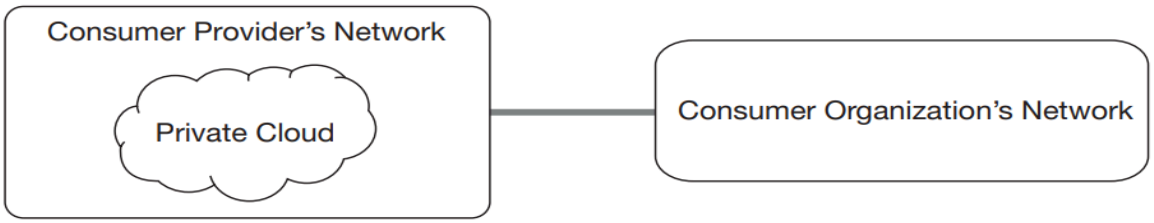
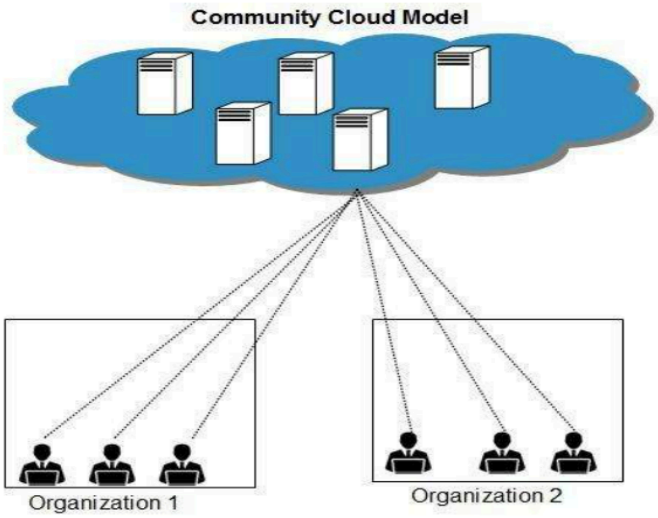



**FIG 4.8:** Off-premises private cloud

- **Community Cloud:**

The community cloud deployment model allows access to a number of organizations or consumers belonging to a community and the model is built to serve some common and specific purpose. It is for the use of some community of people or organizations who share common concerns in business functionalities, security requirements etc. This model allows sharing of infrastructure and resources among multiple consumers belonging to a single community and thus becomes cheaper compared to a private cloud. Community cloud deployment can be on-premises or off-premises. This cloud deployment may be identified as a generalized form of private cloud. While a private cloud is accessible only to one consumer, one community cloud is used by multiple consumers of a community. Thus, this deployment model supports multi-tenancy. The goal of community cloud deployment is to provide the

benefits of public cloud, like multi-tenancy, pay-per-use billing etc. to its consumers along with added level of privacy and security like the private cloud.




- **Hybrid Cloud:**

A hybrid cloud is generally created by combining private or community deployment with public cloud deployment together. This deployment model helps businesses to take advantage of private or community cloud by storing critical applications and data. There at the same time, it provides the cost benefit by keeping shared data and applications on the public cloud. Figure demonstrates a hybrid cloud model combining public cloud with on-premises private cloud. The hybrid cloud can be formed by combining two elements from a set of five different cloud deployments as on-premises private cloud, off-premises private cloud, on-premises community cloud, off-premises community cloud and public cloud, where one among the first four deployments is combined with the last one (public cloud).

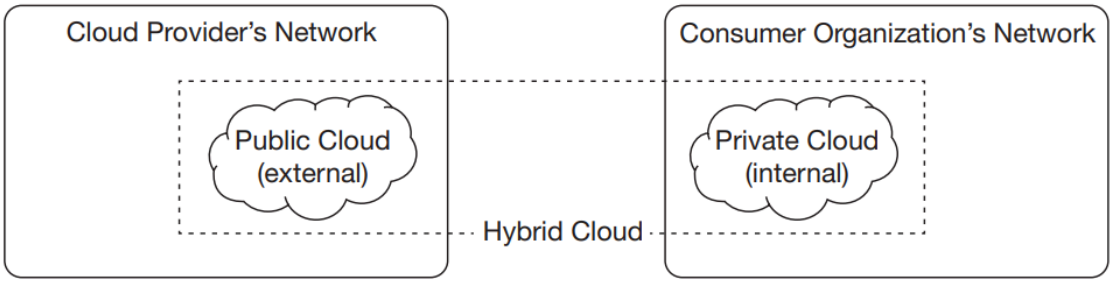



**FIG 4.9:** A hybrid cloud model

## Q8. DIFFERENTIATE BETWEEN PUBLIC AND PRIVATE CLOUD.

**Table 4.2** A comparison between private cloud and public cloud

| *Private Cloud* | *Public Cloud* |
|---|---|
| It can be both of types of on-premises and off-premises. | There cannot be any on-premises public cloud deployment. |
| On-premises private cloud can be delivered over the private network. | It can only be delivered over public network. |
| It does not support multi-tenancy feature for unrelated and external tenants. | It demonstrates multi-tenancy capability with its full ability. |
| The resources are for exclusive use of one consumer (generally an organization). | The resources are shared among multiple consumers. |
| A private cloud facility is accessible to a restricted number of people. | This facility is accessible to anyone. |
| This is for organizational use. | It can be used both by organization and the user. |
| The consumers have important roles to play in management of the cloud. | The management is entirely provider's responsibility. |
| The consumers have more control over the environment. | The consumers have very less control or no control. |
| It provides more confidence regarding security of data as remains under the control of consumer organization's security boundary. | The public cloud deployment often creates concerns regarding security and privacy of data. |
| The cost of computing is more in comparison to public cloud. | This is more economical as multiple unrelated consumers (tenants) share same infrastructure. |
| It is not an ideal scenario for promoting pay-as-you-use philosophy. | It is an ideal model for practicing pay-as-you-use philosophy. |
| The resource may often remain idle resulting in the resource wastage. | The resource utilization is optimum due to the presence of larger number of consumers. |
| It is not so environment friendly like public cloud. | It promotes green computing at its best. |

## Q9. DIFFERENTIATE BETWEEN PRIVATE AND COMMUNITY CLOUD.

| *Private Cloud* | *Community Cloud* |
|---|---|
| One private cloud can be used by one consumer (may be an organization). | One community cloud can be used by one community. |
| Access is restricted among members of a single consumer. | Access is restricted among members of a single community. |
| Resources of a private cloud are for exclusive use of one consumer. | Resources are shared among multiple consumers of a single community. |
| It may reside at consumer's premises. | It may reside at some consumer's premises within the community. |
| The private cloud (on-premises) can be delivered over private network. | To provide access to multiple consumers (of a community), it delivers over public network. |
| It does not support multi-tenancy. | Here multi-tenancy is supported. |
| This is for organizational use. | This one is for use of the communities. |