ENTERPRISE CLOUD CONCEPTS

Unit-3

Important Q & A

1) Explain about Remote Administration System?

Ans) The remote administration system mechanism (Figure 9.1) provides tools and user-interfaces for external cloud resource administrators to configure and administer cloud-based IT resources.
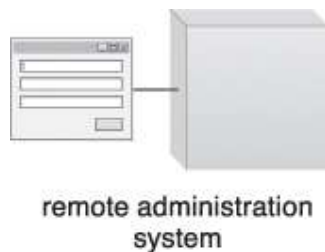

remote administration
system

Figure 9.1 The symbol used in this book for the remote administration system. The displayed user-interface will typically be labeled to indicate a specific type of portal.

A remote administration system can establish a portal for access to administration and management features of various underlying systems, including the resource management, SLA management, and billing management systems described in this chapter (Figure 9.2).
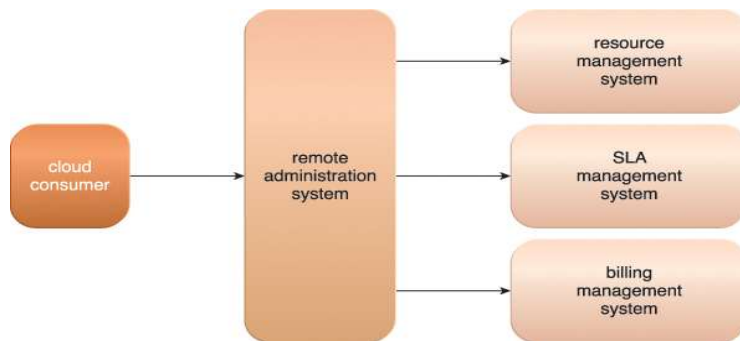

Figure 9.2 The remote administration system abstracts underlying management systems to expose and centralize administration controls to external cloud resource administrators. The system provides a customizable user console, while programmatically interfacing with underlying management systems via their APIs.

The tools and APIs provided by a remote administration system are generally used by the cloud provider to develop and customize online portals that provide cloud consumers with a variety of administrative controls.

The following are the two primary types of portals that are created with the remote administration system:

• Usage and Administration Portal – A general purpose portal that centralizes management controls to different cloud-based IT resources and can further provide IT resource usage reports.
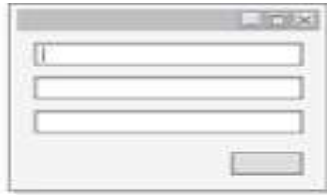

usage and administration portal

• Self-Service Portal – This is essentially a shopping portal that allows cloud consumers to search an up-to-date list of cloud services and IT resources that are available from a cloud provider (usually for lease). The cloud consumer submits its chosen items to the cloud provider for provisioning.

self-service portal

Figure 9.3 illustrates a scenario involving a remote administration system and both usage and administration and self-service portals.
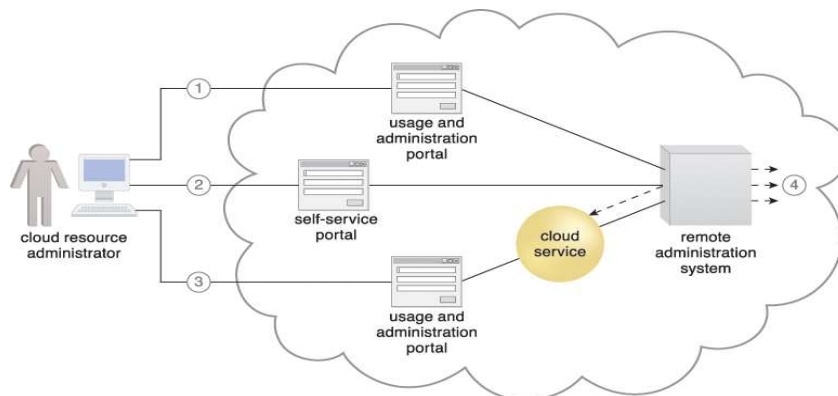

Depending on:

• the type of cloud product or cloud delivery model the cloud consumer is leasing or using from the cloud provider,

• the level of access control granted by the cloud provider to the cloud consumer, and

• further depending on which underlying management systems the remote administration system interfaces with,

...tasks that can commonly be performed by cloud consumers via a remote administration console include:

• configuring and setting up cloud services

• provisioning and releasing IT resource for on-demand cloud services

• monitoring cloud service status, usage, and performance

• monitoring QoS and SLA fulfillment

• managing leasing costs and usage fees

• managing user accounts, security credentials, authorization, and access control

• tracking internal and external access to leased services

• planning and assessing IT resource provisioning

• capacity planning

Additionally, the cloud consumer would be able to further leverage standardized APIs if it is interested in leasing and centrally administering IT resources from multiple cloud providers and/or IT resources residing in cloud and on-premise environments.
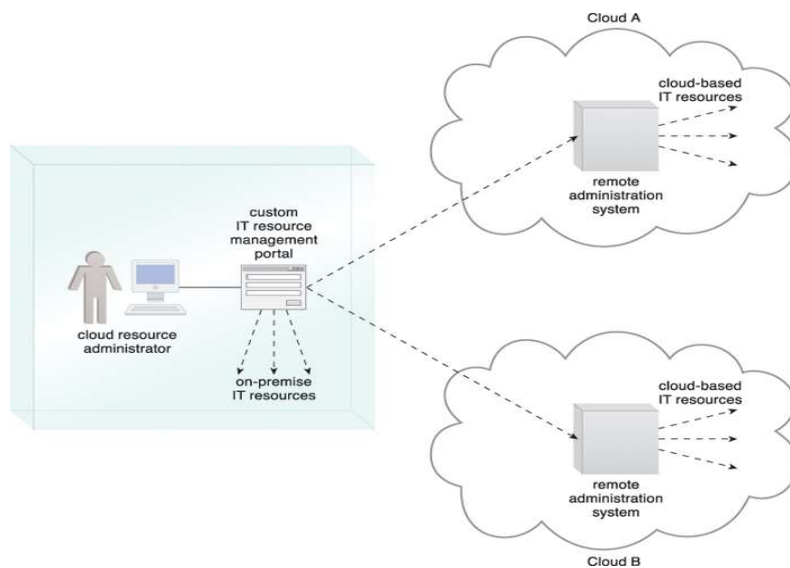
A **Remote Administration System** in cloud computing provides tools and user interfaces for external cloud resource administrators to configure and manage cloud-based IT resources. It centralizes management controls and interfaces with underlying management systems via APIs1.

Key Components:

1. **Customizable User Console**: Provides a centralized interface for managing various cloud resources.
2. **API Integration**: Interfaces with underlying management systems to perform administrative tasks.
3. **Portals**:
    - **Usage and Administration Portal**: Centralizes management controls and provides usage reports.
    - **Self-Service Portal**: Allows cloud consumers to search and request cloud services.

2) What is Resource Management System explain in detail?

Ans) A Resource Management System in cloud computing is designed to efficiently manage, allocate, and monitor cloud resources. This system ensures that resources such as compute power, storage, and network bandwidth are optimally utilized to meet the demands of cloud consumers. Here are the key aspects:

Key Components of a Resource Management System:

1. **Resource Allocation**: Dynamically allocates resources based on the current demand and predefined policies.

2. **Monitoring and Analytics**: Continuously monitors resource usage and provides analytics for informed decision-making.

3. **Load Balancing**: Distributes workloads evenly across available resources to prevent any single resource from being overwhelmed.

4. **Capacity Planning**: Forecasts future resource needs based on historical data and trends.

5. **Automated Scaling**: Automatically scales resources up or down based on the workload, ensuring optimal performance and cost efficiency.
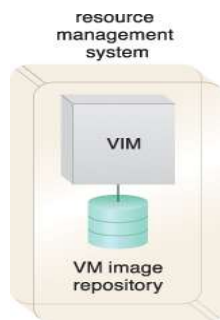

Figure 9.5 A resource management system encompassing a VIM platform and a virtual machine image repository. The VIM may have additional repositories, including one dedicated to storing operational data.
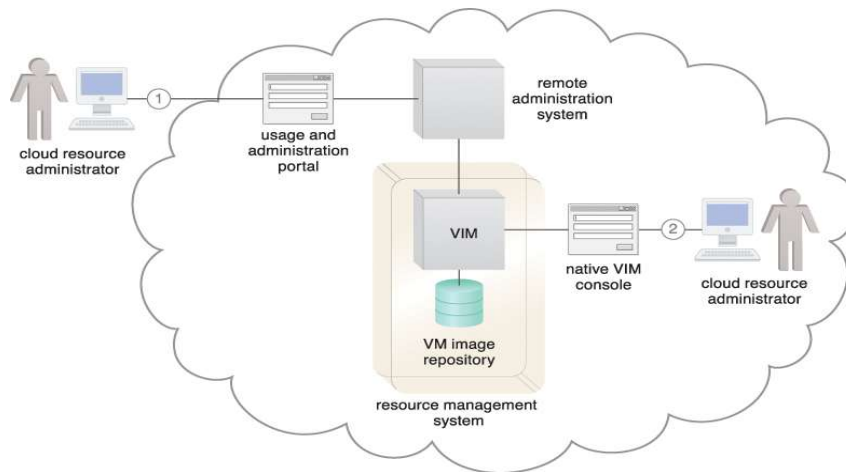
Figure 9.6 The cloud consumer's cloud resource administrator accesses a usage and administration portal externally to administer a leased IT resource (1). The cloud provider's cloud resource administrator uses the native user-interface provided by the VIM to perform internal resource management tasks (2).

3) What is SLA Management System? Give me one example

Ans) The SLA management system mechanism represents a range of commercially available cloud management products that provide features pertaining to the administration, collection, storage, reporting, and runtime notification of SLA data.

A **Service Level Agreement (SLA) Management System** is a tool or platform used to manage and monitor the performance and quality of services provided by a service provider to its customers. It ensures that the agreed-upon service levels are met and helps in maintaining transparency and accountability between the service provider and the customer2.

**Key Features of an SLA Management System:**

1. **Monitoring and Reporting**: Tracks the performance of services against the agreed-upon metrics and generates reports.

2. **Alerts and Notifications**: Sends automated alerts when service levels are not met, allowing for quick resolution.

3. **Contract Management**: Manages the SLA contracts, including terms, conditions, and any changes or updates.

4. **Performance Analysis**: Analyzes service performance data to identify trends and areas for improvement.

5. **Customer Satisfaction**: Measures customer satisfaction and feedback to ensure services meet their expectations.

Example of SLA Management System:

SolarWinds Service Desk is an example of an SLA management tool. It helps IT administrators manage SLA breaches efficiently by setting up automated alerts to notify the appropriate teams when expectations are not being met1. This ensures swift resolution of performance-related problems and maintains customer satisfaction.
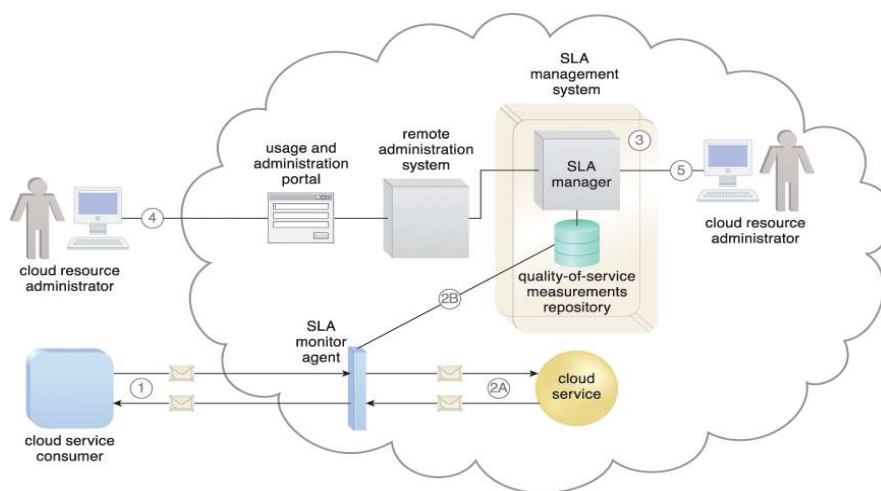

4) Explain about Billing Management System in detail with Case Study Example?

Ans) The billing management system mechanism is dedicated to the collection and processing of usage data as it pertains to cloud provider accounting and cloud consumer billing. Specifically, the billing management system relies on pay-per-use monitors to gather runtime usage data that is stored in a repository that the system components then draw from for billing, reporting, and invoicing purposes (Figures 9.9 and 9.10).

Figure 9.9 A billing management system comprised of a pricing and contract manager and a pay-per-use measurements repository.
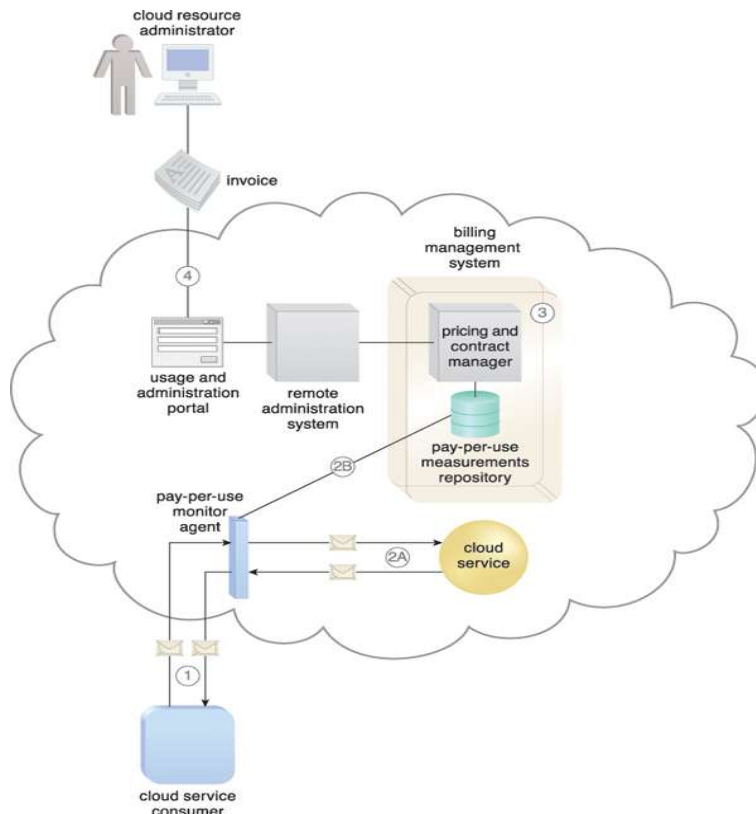
A **Billing Management System (BMS)** in cloud computing is a comprehensive system designed to manage and automate the billing process for cloud services. It collects usage data, processes it, and generates invoices for cloud consumers2. Here's a detailed overview:

Key Components of a Billing Management System:

1. **Usage Data Collection**: Monitors and collects data on the usage of cloud resources.

2. **Pricing and Contract Management**: Defines pricing policies and manages contracts with cloud consumers.

3. **Billing and Invoicing**: Processes usage data to generate accurate invoices.

4. **Reporting and Analytics**: Provides detailed reports and analytics on usage and costs.

5. **Payment Processing**: Manages payment collection and reconciliation

Case Study Example: DTGOV's Custom Billing Management System

DTGOV, a government organization, decided to establish a billing management system to handle custom-defined billable events, such as subscriptions and IT resource volume usage. Here's how they implemented it:

- **Customization**: The system was customized with necessary events and pricing scheme metadata.

- **Usage Data Collection**: Usage events were collected from pay-per-use monitors implemented as extensions to the Virtual Infrastructure Manager (VIM) platform.

- **Billing Process**: The system periodically calculated consolidated cloud service usage fees and generated invoices for cloud consumers.

- **Reporting**: The invoices were provided through the usage and administration portal.

This system allowed DTGOV to efficiently manage billing for various cloud services and improve transparency for cloud consumers.

5) What is Case Study? Explain Case Study with Example in Cloud Computing?

Ans) A case study in cloud computing is a detailed analysis of a project, process, or event that showcases how cloud technology was implemented to solve a specific problem or achieve a particular goal. It typically includes the context, challenges, solutions, and outcomes of the project.
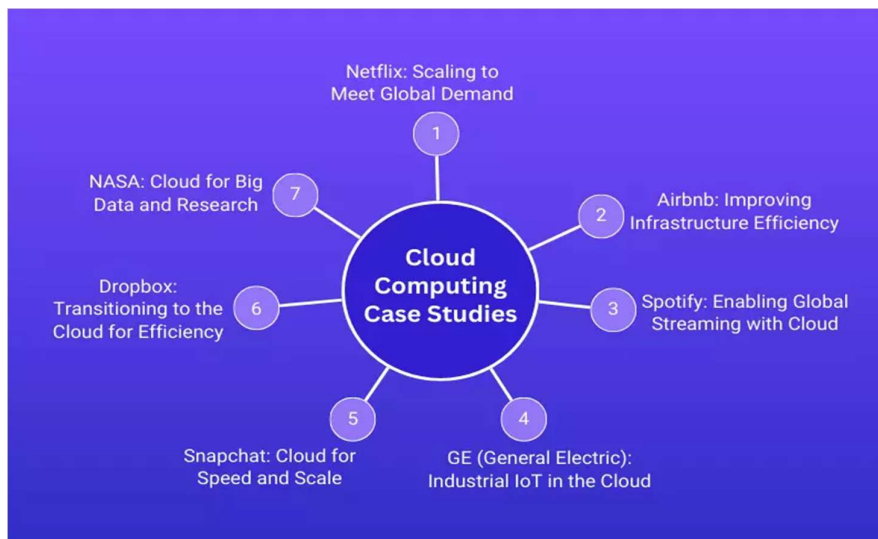
Example of a Cloud Computing Case Study:

Netflix: Scaling to Meet Global Demand

- **Industry**: Entertainment (Streaming)

- **Challenge**: Netflix needed to handle massive amounts of data and ensure smooth streaming for millions of users worldwide.

- **Solution**: Netflix migrated its infrastructure to Amazon Web Services (AWS), which allowed it to scale resources dynamically based on demand.

- **Outcome**: This migration enabled Netflix to handle peak loads efficiently, reduce costs, and improve the overall user experience.

6) What is Workload Distribution Architecture explain its advantages & disadvantages?

Ans) Workload distribution architecture focuses on horizontally scaling IT resources by utilizing identical resources and a load balancer. The load balancer evenly distributes the

workload among available IT assets, such as distributed virtual servers, cloud storage devices, and cloud services. Additional mechanisms that can be part of this architecture include:

- **Cloud Usage Monitor:** Enables runtime tracking and data processing.

- **Audit Monitor:** Monitors the system to fulfill legal requirements.

- **Hypervisor:** Manages workloads and distributes them among virtual hosts.

- **Logical network perimeter:** Isolates cloud consumer network boundaries.

- **Resource clusters:** Support workload balancing between cluster nodes.

- **Resource replication:** Generates new instances of virtualized resources during increased workloads.

Benefits of Workload Distribution Architecture:

- Scalability: Horizontal scaling allows for efficient distribution of workloads, ensuring optimal resource utilization during increased demand.

- Fault tolerance: Distribution of workloads across multiple resources provides fault tolerance and minimizes downtime in case of resource failures.

- Performance optimization: Balancing workloads across resources prevents overload on individual resources, leading to improved performance.

**Disadvantages of Workload Distribution Architecture:**

- Complexity: Implementing workload distribution architecture can be complex, especially when configuring load balancers, monitoring systems, and ensuring resource synchronization.

- Increased latency: Load balancing and workload distribution can introduce some latency, particularly in processing requests involving large datasets or complex applications.
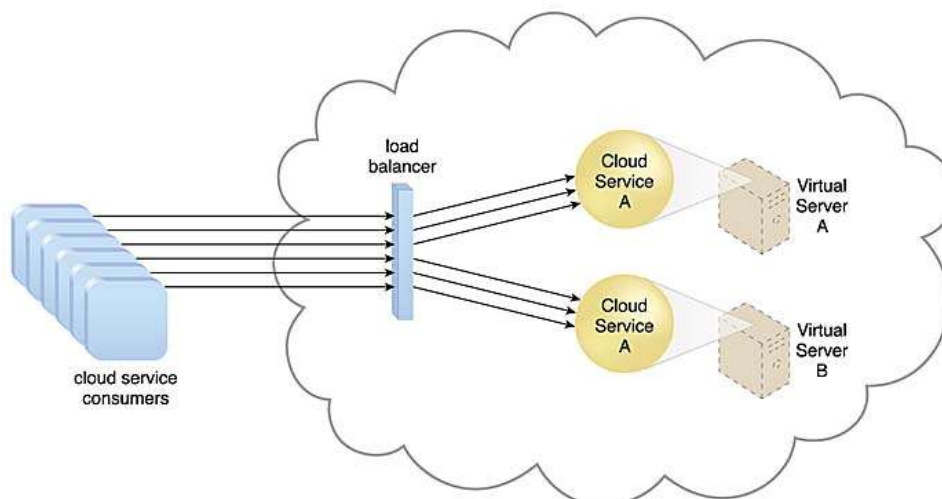
7) Explain about Resource Pooling Architecture? What are its benefits & disadvantages?

Ans) Resource pooling architecture involves grouping identical IT resources into pools, which can include physical and virtual resources. These resources are automatically synchronized and maintained by the system. Examples of resource pools include:

- Physical server pools: Networked servers with pre-installed operating systems and applications.

- Virtual server pools: Configured from pre-chosen templates during provisioning.

- Storage pools: File or block-based storage containers.

- Network pools: Pre-configured network devices like virtual firewalls and switches for redundancy, load balancing, and link aggregation.

- CPU pools: Allocate CPU resources to virtual servers.

- Physical RAM pools: Vertically scale newly provisioned physical servers.
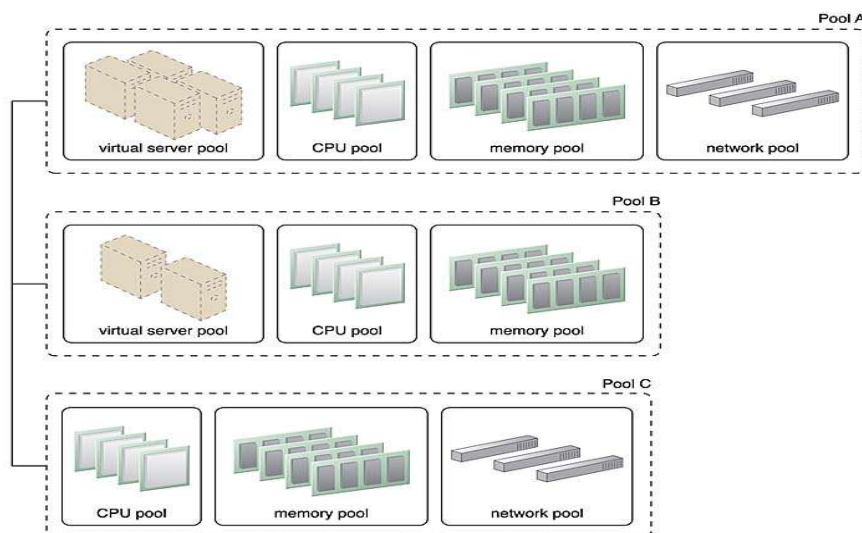
Benefits of Resource Pooling Architecture:

- Resource optimization: Resource pooling ensures efficient utilization of IT resources by dynamically allocating them based on demand.

- Simplified management: The architecture simplifies resource management by automatically synchronizing resources and providing administration tools.

- Scalability: Resource pooling enables both horizontal and vertical scalability, allowing for the allocation of additional resources from appropriate pools as workload increases.

Disadvantages of Resource Pooling Architecture:

- Complexity: Managing and organizing resource pools can be complex, especially in large-scale environments with hierarchical structures.

- Synchronization challenges: Keeping identical resources within pools synchronized and maintaining consistency across multiple instances can be challenging, requiring robust synchronization mechanisms.

8) What is Cloud Computing Architecture explain its Components?

Ans) Cloud Computing Architecture

Architecture of cloud computing is the combination of both SOA (Service Oriented Architecture) and EDA (Event Driven Architecture). Client infrastructure, application, service, runtime cloud, storage, infrastructure, management and security all these are the components of cloud computing architecture.

The cloud architecture is divided into 2 parts, i.e.

1. Frontend
2. Backend
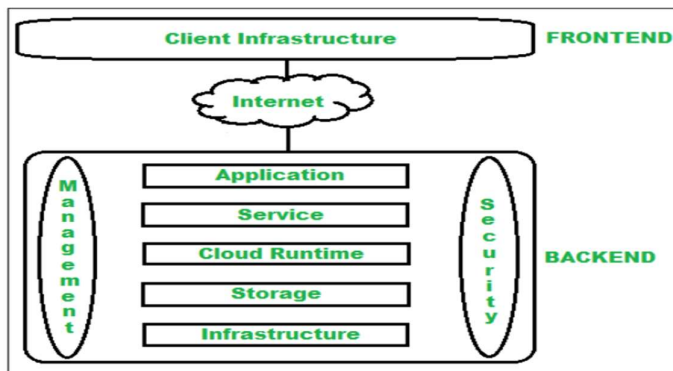
Architecture of Cloud Computing

1. Frontend

Frontend of the cloud architecture refers to the client side of cloud computing system. Means it contains all the user interfaces and applications which are used by the client to access the cloud computing services/resources. For example, use of a web browser to access the cloud platform.

2. Backend

Backend refers to the cloud itself which is used by the service provider. It contains the resources as well as manages the resources and provides security mechanisms. Along with this, it includes huge storage, virtual applications, virtual machines, traffic control mechanisms, deployment models, etc.

Components of Cloud Computing Architecture

1. **Client Infrastructure –** Client Infrastructure is a part of the frontend component. It contains the applications and user interfaces which are required to access the cloud

platform. In other words, it provides a GUI(Graphical User Interface) to interact with the cloud.

2. **Application**: Application is a part of backend component that refers to a software or platform to which client accesses. Means it provides the service in backend as per the client requirement.

3. **Service**: Service in backend refers to the major three types of cloud based services like SaaS, PaaS and IaaS. Also manages which type of service the user accesses.

4. **Runtime Cloud**: Runtime cloud in backend provides the execution and Runtime platform/environment to the Virtual machine.

5. **Storage:** Storage in backend provides flexible and scalable storage service and management of stored data.

6. **Infrastructure:** Cloud Infrastructure in backend refers to the hardware and software components of cloud like it includes servers, storage, network devices, virtualization software etc.

7. **Management:** Management in backend refers to management of backend components like application, service, runtime cloud, storage, infrastructure, and other security mechanisms etc.

8. **Security:** Security in backend refers to implementation of different security mechanisms in the backend for secure cloud resources, systems, files, and infrastructure to end-users.

9. **Internet:** Internet connection acts as the medium or a bridge between frontend and backend and establishes the interaction and communication between frontend and backend.

10. **Database:** Database in backend refers to provide database for storing structured data, such as SQL and NOSQL databases. Example of Databases services include Amazon RDS, Microsoft Azure SQL database and Google CLoud SQL.

11. **Networking:** Networking in backend services that provide networking infrastructure for application in the cloud, such as load balancing, DNS and virtual private networks.

12. **Analytics:** Analytics in backend service that provides analytics capabilities for data in the cloud, such as warehousing, business intelligence and machine learning.

Benefits of Cloud Computing Architecture

- Makes overall cloud computing system simpler.

- Improves data processing requirements.

- Helps in providing high security.

- Makes it more modularized.

- Results in better disaster recovery.

- Scalability.

9) Explain about Cloud Bursting Architecture?

Ans) Cloud Bursting Architecture is a hybrid cloud computing model that allows organizations to use public cloud resources to handle peak loads when their on-premises infrastructure reaches capacity. Here's a detailed overview:

1. **On-Premises Infrastructure**: The primary computing environment where most workloads are processed.

2. **Public Cloud Resources**: External cloud services that are used to handle additional workloads when on-premises capacity is exceeded.

3. **Load Balancer**: Distributes incoming requests between on-premises and cloud resources based on capacity and demand.

4. **Automated Scaling**: Mechanisms that automatically scale resources up or down based on predefined thresholds.

5. **Resource Replication**: Keeps data synchronized between on-premises and cloud environments to ensure consistency.

How It Works:

1. **Capacity Monitoring**: Continuously monitors the usage of on-premises resources.

2. **Threshold Detection**: When resource usage reaches a predefined threshold, the system triggers cloud bursting.

3. **Workload Distribution**: Redirects excess workloads to the public cloud using load balancers.

4. **Resource Scaling**: Public cloud resources are scaled up to handle the additional load.

5. **Synchronization**: Maintains data consistency between on-premises and cloud environments.

6. **Scaling Down**: Once the peak load subsides, resources are scaled down and workloads are shifted back to on-premises infrastructure.
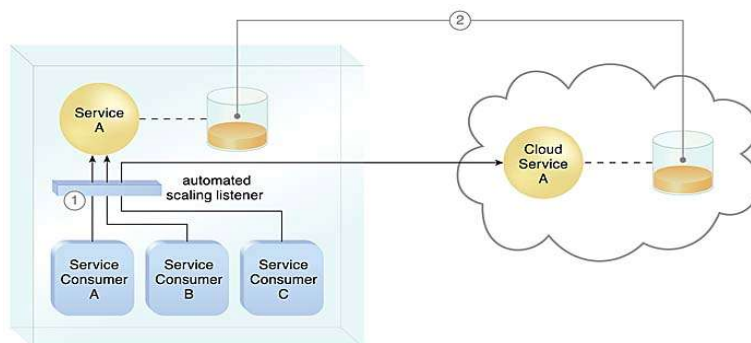
## 10) Explain about Elastic Disk Provisioning Architecture?

Ans) Elastic Disk Provisioning Architecture is a cloud computing model that dynamically allocates storage space to virtual servers based on actual usage rather than pre-allocated fixed disk space. This approach ensures efficient use of storage resources and cost-effective billing for cloud consumers.

Key Components:

1. **Thin Provisioning**: Dynamically allocates storage space as needed, rather than allocating the full amount upfront.

2. **Usage Monitoring**: Continuously monitors actual storage usage to ensure accurate billing.

3. **Dynamic Allocation**: Adjusts storage allocation in real-time based on demand.

4. **Billing Granularity**: Charges cloud consumers based on the exact amount of storage used, rather than pre-allocated space.

Example:

Cloud Provider X uses elastic disk provisioning to manage storage for its virtual servers. When a customer requests a virtual server with three 150 GB disks, the provider allocates the requested space but only charges for the actual used space. If the customer only uses 50 GB of storage, they are billed for 50 GB, not the full 450 GB.
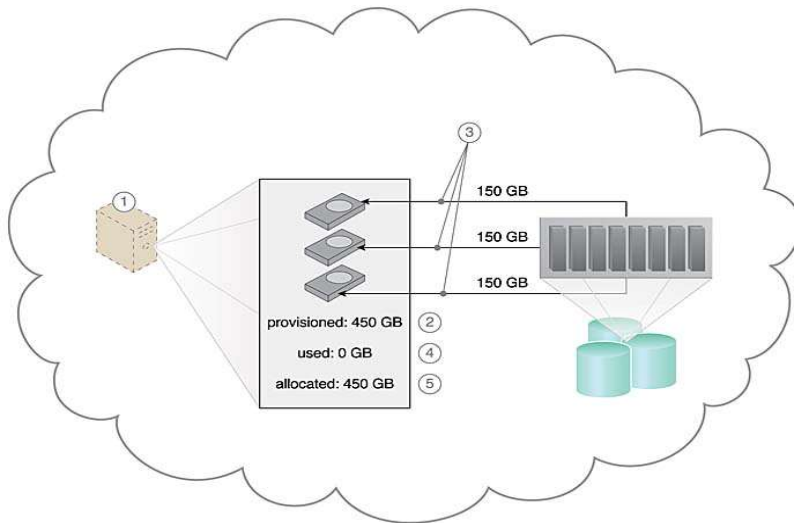
11) What is Dynamic Scalability Architecture? What are its Components & Works?

Ans) Dynamic Scalability Architecture is a cloud computing model designed to handle fluctuating workloads by automatically adjusting the resources based on demand. This architecture ensures that applications remain performant and cost-effective without manual intervention.

Key Components:

1. **Autoscaling**: Automatically adjusts the number of active instances of an application based on current demand.

2. **Load Balancing**: Distributes incoming traffic across multiple instances to ensure even distribution of the load.

3. **Monitoring and Alerts**: Continuously monitors resource usage and triggers scaling actions when predefined thresholds are met.

4. **Resource Provisioning**: Dynamically provisions additional resources (compute, storage, etc.) as needed and de-provisions them when no longer required.

5. **Policy Management**: Allows administrators to set policies that define scaling parameters, such as minimum and maximum resource limits.

How It Works:

1. **Demand Detection**: The system continuously monitors resource usage metrics such as CPU utilization, memory usage, and network traffic.

2. **Threshold Triggering**: When resource usage reaches a predefined threshold, the system triggers an autoscaling event.

3. **Resource Adjustment**: Based on the demand, the system provisions additional resources (e.g., new instances, storage) or de-provisions resources when demand decreases.

4. **Load Distribution**: Load balancers ensure that the traffic is evenly distributed among the available resources.

5. **Monitoring and Reporting**: The system provides real-time monitoring and generates reports on resource usage and performance.