Unit III

Main Features of Cloud Computing

Cloud computing provides **computing resources like servers, storage, and applications over the internet**, allowing users to access them on-demand. Its main features are:

1. **On-Demand Self-Service**
    - Users can **request and use computing resources automatically** whenever needed, without human intervention.
    - Example: Launching a virtual server via a web console.
2. **Scalability (Elasticity)**
    - Cloud resources can **scale up or down automatically** based on demand.
    - Example: An e-commerce website getting more servers during peak shopping hours.
3. **Pay-as-You-Use (Metered Billing)**
    - Users pay **only for the resources they consume**, reducing costs.
    - Example: Paying for storage used in Google Drive or AWS S3.
4. **Broad Network Access**
    - Cloud services are accessible from **any device with internet connectivity**, such as laptops, tablets, or smartphones.
5. **Resource Pooling**
    - Cloud providers **share computing resources among multiple users** dynamically.
    - Example: Multiple virtual machines sharing the same physical server.
6. **High Availability**
    - Cloud services are **always available** due to backup systems, redundancy, and fault-tolerant architectures.
    - Example: Data replication across multiple data centers.
7. **Automatic Updates**
    - Software, security patches, and infrastructure updates are **managed automatically** by the cloud provider.
    - Users don’t need to worry about manual upgrades.

2. Features of Grid Computing

Grid computing is a technology that **connects multiple computers across different locations** to work together as a single system. Its main features are:

1. **Distributed Resources**
    a. Uses computers and resources located in **different geographical locations**.
    b. Enables sharing of processing power, storage, and data across systems.
2. **Resource Sharing**
    a. Shares **CPU, memory, storage, and network resources** among multiple users or applications.
    b. Improves overall system efficiency.
3. **Parallel Processing**
    a. Breaks down **large tasks into smaller sub-tasks** that run simultaneously on multiple computers.
    b. Speeds up computation for complex problems.
4. **Heterogeneous Systems**
    a. Supports a **mix of different hardware, operating systems, and software** across the network.
    b. Ensures flexibility in using existing infrastructure.
5. **High Performance Computing (HPC)**
    a. Suitable for **scientific research, simulations, and complex calculations**.
    b. Leverages combined computing power for faster results.
6. **Decentralized Control**
    a. No single system controls the entire network.
    b. Each participating system contributes resources independently.
7. **Cost-Effective**
    a. Utilizes **existing hardware efficiently**.
    b. Reduces the need to invest in expensive supercomputers.

Programming Support of Google App Engine:

Google App Engine is a **Platform as a Service (PaaS)** in Google Cloud that lets developers build and run applications without managing servers. It provides built-in programming language support, frameworks, and managed services.lications without managing servers. It provides built-in programming language support, frameworks, and managed services.

2. Supported Programming Languages

Definition:
GAE provides **pre-configured environments (runtimes)** for popular programming languages, making it easier to write and run cloud applications.

Standard Environment Languages:

- Python
- Java
- Node.js
- Go
- PHP
- Ruby
- C#

Supported Frameworks

GAE supports popular frameworks for each programming language, which helps in **faster development and easier integration**.

Examples of Framework

Language	Frameworks
Python	Flask, Django
Java	Spring Boot, Java Servlets
Node.js	Express.js
PHP	Laravel, Slim
Ruby	Rails, Sinatra
Go	Native Go frameworks

Built-in Services & APIs

Google App Engine is a **Platform as a Service (PaaS)** that allows developers to build, deploy, and run applications **without managing servers or infrastructure**.

1. Databases

Cloud Datastore / Firestore

Cloud Datastore / Firestore is a **NoSQL, fully managed database** used to store and retrieve application data with high scalability and automatic replication.

Key Point: Suitable for structured and semi-structured data.

Cloud SQL

Cloud SQL is a **managed relational database service** that supports SQL databases such as MySQL, PostgreSQL, and SQL Server.

Key Point: Ideal for applications requiring structured data and SQL queries.

2. Caching

Memcache

Memcache is a **distributed in-memory caching service** that stores frequently accessed data to reduce database load and improve application performance.

Key Point: Provides faster data access and lower latency.

3. Task & Background Processing

Task Queues

Task Queues allow applications to **execute background tasks asynchronously**, such as sending emails or processing data.

Key Point: Helps in handling long-running tasks without delaying user requests.

Cron Jobs

Cron Jobs are used to **schedule tasks at specific time intervals**, such as daily backups or periodic data cleanup.

Key Point: Enables automated and time-based task execution.

4. Networking

Automatic Load Balancing

Automatic load balancing **distributes incoming traffic** across multiple application instances to ensure high availability and performance.

Key Point: Handles traffic spikes without manual intervention.

HTTPS Support

HTTPS support provides **secure communication** between users and applications using SSL/TLS encryption.

Key Point: Protects data during transmission over the internet.

5. Authentication & Security

Identity-Aware Proxy (IAP)

Identity-Aware Proxy (IAP) is a security service that **controls access to applications based on user identity** before granting access.

Key Point: Ensures only authorized users can access applications.

Google IAM (Identity and Access Management)

Google IAM manages **who can access Google Cloud resources and what actions they can perform**.

Key Point: Provides role-based access control and strong security governance.

6. Monitoring & Logging

Cloud Logging

Cloud Logging collects, stores, and analyzes **application and system logs** for debugging and auditing purposes.

Key Point: Helps identify errors and security issues.

Cloud Monitoring

Cloud Monitoring tracks **application performance, uptime, and resource usage** using metrics and alerts.

Key Point: Enables proactive performance management.

Development Tools:

Local Development SDKs

Definition

Local development SDKs are **software development kits** that allow developers to **build and test cloud applications on their local machines** before deploying them to Google Cloud.

Features

- Simulates cloud services locally
- Supports debugging and testing
- Reduces development cost and errors

Benefits

- Faster development
- Early detection of bugs
- Offline development capability

Google Cloud CLI (gcloud)

Definition

Google Cloud CLI, commonly known as **gcloud**, is a **command-line tool** used to manage Google Cloud resources and deploy applications.

Functions

- Deploy applications
- Manage projects and services
- Configure authentication and permissions

Benefits

- Automation friendly
- Faster deployments
- Useful for scripting and DevOps tasks

- **IDE integration** (VS Code, IntelliJ, Eclipse)

IDE Integration

IDE integration allows cloud development tools to be **directly used inside code editors**, improving productivity.

Visual Studio Code (VS Code)

Description

VS Code integrates with Google Cloud through extensions, enabling **cloud project management, debugging, and deployment** from the editor.

Benefits

- Lightweight and fast
- Built-in terminal and Git support
- Easy cloud extension support

IntelliJ IDEA

Description

IntelliJ provides Google Cloud plugins that support **Java-based cloud application development**, debugging, and deployment.

Benefits

- Strong support for Java and Spring
- Advanced code completion and refactoring

Eclipse

Description

Eclipse offers plugins for Google Cloud that support **Java cloud application development** and App Engine deployment.

Benefits

- Widely used in academic environments
- Strong Java EE support

- **CI/CD support** via Cloud Build and GitHub Actions

Ready-to-Use Environment

- No need to install servers
- No need to manage hardware
- Just write code and upload it
- Google takes care of running the application

Built-in Services for Programming

GAE provides services that can be used directly in programs:

- **Database support** (store data)
- **User authentication** (login)
- **Automatic scaling** (handles more users automatically)
- **Load balancing** (shares work among servers)

Easy Development and Deployment

- Programs can be tested locally
- Applications are deployed using simple configuration files
- Supports version control and updates

Programming on Amazon AWS

Amazon Web Services (AWS)

Amazon Web Services is a **cloud computing platform** that allows developers to **run programs and applications without owning physical servers**.

Programming Support in AWS

Supported Programming Languages

AWS supports multiple programming languages such as **Python, Java, C#, JavaScript (Node.js), PHP, Ruby, and Go**, allowing developers to build diverse applications.

AWS Lambda (Serverless Programming)

AWS Lambda is a **serverless computing service** that lets developers run code **without managing servers**, automatically handling scaling and execution.

Amazon EC2 (Elastic Compute Cloud)

Amazon EC2 provides **virtual machines in the cloud** where programmers can install and run applications just like on a physical server.

Database Integration (DynamoDB & RDS)

AWS easily integrates with **DynamoDB (NoSQL database)** and **RDS (relational database service)** for storing and managing application data.

Key Features of AWS

Automatic Scaling

Automatic scaling allows AWS to **increase or decrease resources automatically** based on application demand.

High Security

AWS provides **strong security mechanisms** such as encryption, access control, and monitoring to protect applications and data.

Pay-As-You-Go Pricing

AWS follows a **pay-only-for-what-you-use model**, reducing infrastructure and operational costs.

2. Programming on Microsoft Azure

Microsoft Azure is a cloud computing platform by Microsoft for building and running applications.

Programming Support in Azure

- Supports **C#, .NET, Java, Python, JavaScript, PHP**
- **Azure App Service** for web applications
- **Azure Functions** for serverless programming
- Strong support for **Windows and Visual Studio**

Key Features

- Easy development using Visual Studio
- Automatic scaling and load balancing
- Built-in security and monitoring

3. Simple Comparison

Feature	AWS	Azure
Owner	Amazon	Microsoft
Popular Languages	Python, Java, JS	C#, .NET, Java
Serverless	AWS Lambda	Azure Functions
Best For	General cloud apps	Microsoft-based apps

Emerging Cloud Software Environments

Cloud software environments are platforms that help developers build, deploy, and run applications easily in the cloud. New (emerging) environments focus on **simplicity, speed, and scalability**.

Features of the Cloud environment:

i. Cloud-Native Applications

Cloud-native applications are **applications designed and developed specifically to run in cloud environments**, taking full advantage of cloud scalability, flexibility, and automation.

Microservices

Microservices are an architectural approach where an application is divided into **small, independent services** that communicate through APIs.

Containers

Containers are **lightweight runtime environments** that package applications and their dependencies, ensuring consistent execution across platforms.

APIs (Application Programming Interfaces)

APIs are **interfaces that allow different software services to communicate and exchange data** with each other.

ii. Scalability & Elasticity

Scalability and elasticity refer to the cloud's ability to **adjust resources automatically based on workload demand**.

Automatic Scaling

Automatic scaling is the process where cloud resources **increase or decrease automatically** according to traffic or workload requirements.

2.2 Efficient Resource Utilization

Efficient resource utilization means **optimal use of computing resources** to avoid over-provisioning and reduce costs.

iii. Automation & DevOps

Automation and DevOps focus on **streamlining software development and operations** through tools and automated workflows.

Continuous Integration (CI)

Continuous Integration is a practice where **code changes are frequently integrated, built, and tested automatically**.

Continuous Deployment (CD)

Continuous Deployment is the process of **automatically releasing tested code into production** without manual intervention.

Infrastructure as Code (IaC)

Infrastructure as Code is a method of **managing and provisioning infrastructure using machine-readable configuration files**.

iv. Resilience & Fault Tolerance

Resilience and fault tolerance describe a system's ability to **continue operating despite failures**.

Redundant Services

Redundant services involve **deploying multiple instances of components** to ensure availability if one fails.

Self-Healing Architecture

A self-healing architecture is a system that **automatically detects failures and restores services** without human intervention.

v. Security & Compliance

Security and compliance ensure that cloud systems are **protected against threats** and meet legal and regulatory requirements.

Identity-Based Access

Identity-based access controls **who can access cloud resources based on user identity and roles**

DevSecOps

DevSecOps is the practice of **integrating security into every stage of the DevOps lifecycle**, from development to deployment.

vi. AI & Intelligent Operations

AI and intelligent operations use **artificial intelligence to manage and optimize cloud systems automatically**.

Predictive Analytics

Predictive analytics uses **historical data and machine learning** to forecast resource needs and system behavior.

AIOps (Artificial Intelligence for IT Operations)

AIOps applies **AI and machine learning to automate incident detection, root cause analysis, and system recovery**.

Benefits

Faster development & deployment.
Cost efficiency via pay-as-you-go.
Global accessibility.
Improved reliability & uptime.
Better developer productivity.

Challenges

- Complexity of managing distributed systems.
- Skills gap in cloud-native technologies.
- Security & compliance risks.
- Vendor lock-in in multi-cloud environments.

Applications of Emerging Cloud Software Environment

1. Moving Applications to the Cloud

Moving applications to the cloud requires a structured process to minimize risks and maximize benefits like scalability and cost savings. The migration typically follows key phases: discovery, planning, execution, testing, and optimization.

Common cloud migration approaches

i. Rehost (Lift & Shift)

Rehost, also called **Lift & Shift**, is a cloud migration strategy where an application is **moved to cloud virtual machines exactly as it is**, with **minimal or no code changes**.

Key Point: Fastest and lowest-risk migration approach.

Move the App As-Is to Cloud VMs

This means the application is transferred **without modifying its architecture or design**, running on cloud virtual machines similar to on-premise servers.

Minimal Code Changes

Only **minor configuration changes** (such as IP addresses or storage paths) are made, while application code remains unchanged.

Fastest, Lowest Risk

Rehosting is considered the **quickest and safest** migration option because it avoids major redesign and reduces chances of failure.

ii. Replatform

Replatform is a migration strategy where an application is moved to the cloud with **small optimizations** to use **managed cloud services**, while keeping the core architecture unchanged.

Key Point: Balance between speed and optimization.

Small Optimizations

Includes changes such as:

- Using **managed databases**
- Enabling **auto-scaling**
- Adding **load balancers**

Limited Code Changes

Only **minor application changes** are required to integrate cloud services, without full re-architecture.

Better Performance and Cost than Rehost

Replatforming improves **efficiency, scalability, and cost optimization** compared to simple lift-and-shift.

iii. Replace (SaaS)

Replace is a strategy where an existing application is **discarded and replaced with a cloud-based Software as a Service (SaaS) solution**.

Key Point: No application maintenance or infrastructure management.

Replace Your App with a Cloud-Based Product

Instead of migrating the old system, organizations adopt **ready-made SaaS applications** such as CRM, ERP, or email services.

No Infrastructure to Manage

The cloud provider fully manages **servers, updates, security, and scalability**, allowing users to focus only on business usage.

Microsoft Azure

Microsoft Azure is a **public cloud computing platform** that provides services for computing, storage, networking, databases, AI, and application development, with strong support for **enterprise and hybrid cloud environments**.

Azure – Common Services for Application Migration

1. Compute Services

Azure Virtual Machines

Azure Virtual Machines provide **on-demand virtual servers** in the cloud and are commonly used for **lift-and-shift (rehost) migrations**

2 Azure App Service

Azure App Service is a **fully managed platform** used to host web applications and APIs without managing servers.

3 Azure Kubernetes Service (AKS)

AKS is a **managed container orchestration service** that runs containerized applications using Kubernetes.

4 Azure Functions

Azure Functions is a **serverless computing service** that executes code in response to events without server management.

2. Data Services

1 Azure SQL Database / SQL Managed Instance

These are **managed relational database services** compatible with Microsoft SQL Server.

2 Azure Cosmos DB

Azure Cosmos DB is a **globally distributed NoSQL database** designed for high availability and low latency.

3 Azure Blob Storage

Azure Blob Storage is an **object storage service** for unstructured data such as images, videos, and backups.

3. Networking & Security Services

1 Virtual Network (VNet)

VNet is a **private, isolated network** in Azure that enables secure communication between cloud resources.

2 Azure Load Balancer / Application Gateway

- **Azure Load Balancer** distributes network traffic across virtual machines.
- **Application Gateway** provides load balancing with web application firewall (WAF) support.

3 Azure Active Directory (Entra ID)

Azure Active Directory (Entra ID) is a **cloud-based identity and access management service** for authentication and authorization.

4 Azure Monitor & Log Analytics

These services collect and analyze **performance metrics and logs** for monitoring applications and infrastructure.

4. Azure Migration Tools

1 Azure Migrate

Azure Migrate is a service used for **discovery, assessment, and migration** of on-premise workloads to Azure.

2 Database Migration Service

This service enables **secure and reliable migration of databases** to Azure with minimal downtime.

3 Azure Site Recovery

Azure Site Recovery provides **disaster recovery and near-zero downtime migration** for applications and virtual machines.

5. Best Fit for Azure

.NET / Windows Workloads

Azure is optimized for applications built using **Microsoft .NET and Windows Server technologies**.

Active Directory Integration

Azure integrates seamlessly with **on-premise Active Directory** for identity management.

Microsoft Enterprise Tools

Azure works best for organizations using **Office 365, SharePoint, and other Microsoft products**.

Hybrid Cloud Scenarios

Azure supports **hybrid cloud deployments**, combining on-premise and cloud infrastructure.

Google Cloud Platform (GCP) – Application Services

1. Google Compute Engine

Google Compute Engine provides **scalable virtual machines** for running applications in the cloud.

2. Google App Engine

Google App Engine is a **Platform as a Service (PaaS)** used to build and run cloud-native applications without managing servers.

3. Google Workspace

Google Workspace offers **cloud-based productivity and collaboration tools** such as Gmail, Docs, and Drive.

4. BigQuery / AI Platform

- **BigQuery** is a **serverless data warehouse** for large-scale data analytics.
- **AI Platform** provides tools for **machine learning model training and deployment**.

GCP – Data & Storage Services

1. Cloud SQL

Cloud SQL is a **managed relational database service** for MySQL, PostgreSQL, and SQL Server.

2. Firestore

Firestore is a **fully managed NoSQL document database** designed for real-time applications.

3. Cloud Storage

Cloud Storage is an **object storage service** used for storing files, backups, and media content.

GCP – Networking Services

1. Cloud Load Balancing

Cloud Load Balancing distributes **application traffic across multiple instances globally**.

2. Cloud CDN

Cloud CDN caches content at **edge locations** to reduce latency and improve performance.

GCP – Security Services

1. IAM (Identity and Access Management)

IAM controls **who can access GCP resources and what actions they can perform**.

2. Secret Manager

Secret Manager securely stores and manages **passwords, API keys, and credentials**.

3. Identity-Aware Proxy (IAP)

IAP provides **secure access control** to applications based on user identity.

GCP – DevOps & Monitoring

1. Cloud Build

Cloud Build is a **CI/CD service** that automates building, testing, and deploying applications.

2. Artifact Registry

Artifact Registry stores and manages **container images and build artifacts** securely.

3. Cloud Monitoring & Logging

These services provide **real-time monitoring, logging, and alerting** for cloud applications and infrastructure.

Amazon Cloud Services ,Cloud Applications.

Amazon Web Services (AWS)

Amazon Web Services (AWS) is a **popular cloud platform** that provides online services like servers, storage, databases, and AI tools. It allows companies to **run applications without buying physical hardware**.

Key AWS Services

Amazon EC2

Amazon EC2 provides **virtual computers (servers)** in the cloud where applications can be hosted and run.

Amazon S3

Amazon S3 is **online storage** used to store files, images, videos, and backups safely.

AWS Lambda

AWS Lambda allows developers to **run code without managing servers**. It automatically runs programs when needed.

Amazon RDS

Amazon RDS is a **managed database service** that stores application data and handles backups and maintenance automatically.

Cloud Application Fit on AWS

Elastic Beanstalk

Elastic Beanstalk makes it **easy to deploy applications** without worrying about servers or setup.

App Runner

App Runner helps run **container-based applications** quickly and easily.

AWS Migration Hub

AWS Migration Hub helps organizations **move their applications to AWS** and track the migration process.

Why AWS is Good for Cloud Applications

Scalability

AWS can **increase or decrease resources automatically** based on user demand, which is ideal for web applications.

Pay-As-You-Go Pricing

Users **pay only for the services they use**, which saves money.

Global Regions

AWS has **data centers around the world**, making applications faster and more reliable.

AI & Edge Computing (2025 Focus)

AWS supports **AI features** and **edge computing**, allowing smarter applications and faster processing near users.

Applications:

- **Amazon EC2** – virtual servers for hosting applications
- **AWS Lambda** – serverless computing
- **Amazon S3** – cloud storage
- **Amazon SageMaker** – AI/ML model development

Cloud Applications – Definitions

Cloud Applications

Cloud applications are **software programs designed or migrated to run in a cloud environment**, using cloud infrastructure and services instead of local servers.

Examples of Cloud Applications

SaaS Applications (Software as a Service)

SaaS applications are **fully managed software solutions** delivered over the internet, where users only use the application without managing infrastructure.

Examples: Salesforce, Zoom, Dropbox

PaaS Applications (Platform as a Service)

PaaS applications are built on **cloud platforms that provide development tools, runtime, and infrastructure**, allowing developers to focus on coding.

Examples: Heroku, Google App Engine

IaaS Applications (Infrastructure as a Service)

IaaS applications use **on-demand cloud resources** such as virtual servers, storage, and databases, giving users control over the operating system and applications.

Examples: Virtual machines, cloud storage, cloud databases

Benefits of Cloud Applications

Accessible from Anywhere

Cloud applications can be accessed **from any location using the internet**, supporting remote work and mobility.

Flexible Scaling

Flexible scaling allows cloud applications to **increase or decrease resources automatically** based on user demand.

Reduced Infrastructure Maintenance

The cloud provider handles **hardware, updates, and maintenance**, reducing the workload for organizations.

QUESTION SET:-

1. What is Google App Engine (GAE)?
2. What is a cloud-native application?
3. What is automatic load balancing?
4. Explain the main features of cloud computing.
5. Write a short note on scalability and elasticity.
6. Explain the features of cloud computing in detail.
7. Differentiate between cloud computing and grid computing.
8. What is Google Cloud CLI (gcloud)? Mention its benefits.
9. Explain IDE integration in cloud development.
10. Write a short note on AWS Lambda.
11. What is Google Cloud CLI (gcloud)? Mention its benefits.
12. Describe the features of grid computing.