Unit 6: Cloud platforms and applications

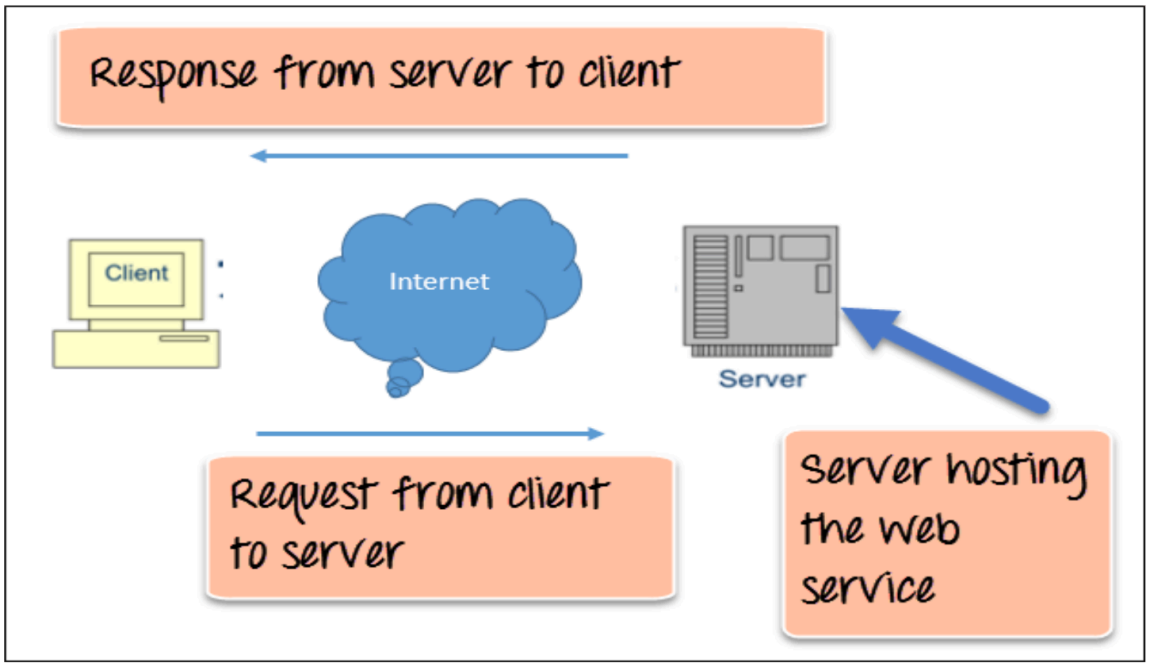
Owner	Saugat Tiwari
Tags	BCA Cloud Computing Notes
Creation Date	@February 3, 2025

6.1 Web Services

Web services are software systems that enable applications to communicate with each other over the internet through standardized communication protocols. This standardization allows different platforms and applications to work together seamlessly.

A web service is a software module that acts as a standardized medium for communication between client and server applications on the World Wide Web. These modules are designed to perform specific tasks and can be discovered and invoked over the network. Once called upon, a web service delivers its functionality to the requesting client.

How WebServices Work?



The diagram shows how web services work: clients send requests to a server that hosts the service.

These requests use remote procedure calls (RPCs) to access the service's methods.

For example: Amazon offers a web service that provides prices for products sold online via amazon.com.

Both .NET and Java can be used to build the front-end since they can communicate with web services.

Characteristics of Web Services

- **Interoperability**: Web services can work across different operating systems, devices, and programming languages.
- **Loose Coupling**: Web services function independently and communicate over the network.
- **Platform and Language Neutrality**: Can be implemented using various technologies (Java, .NET, Python).
- **Reusability**: A single web service can be used in multiple applications.

Why Web Services?

Modern business applications use various programming platforms for web development. While some applications are built in Java, others use .NET, Angular JS, Node.js, and other technologies.

These different applications often need to communicate with each other. However, since they're built using different programming languages, ensuring accurate communication between them can be challenging.

This is where web services play a crucial role. They provide a common platform that enables applications built with different programming languages to communicate effectively with each other.

Types of Web Services

1. **SOAP (Simple Object Access Protocol)**

SOAP is a transport-independent messaging protocol that transfers XML data in the form of SOAP messages. Each message contains an XML document that follows a specific structural pattern, though the content itself can vary. A key advantage of SOAP and web services is their use of HTTP, the standard web protocol, for transmission.

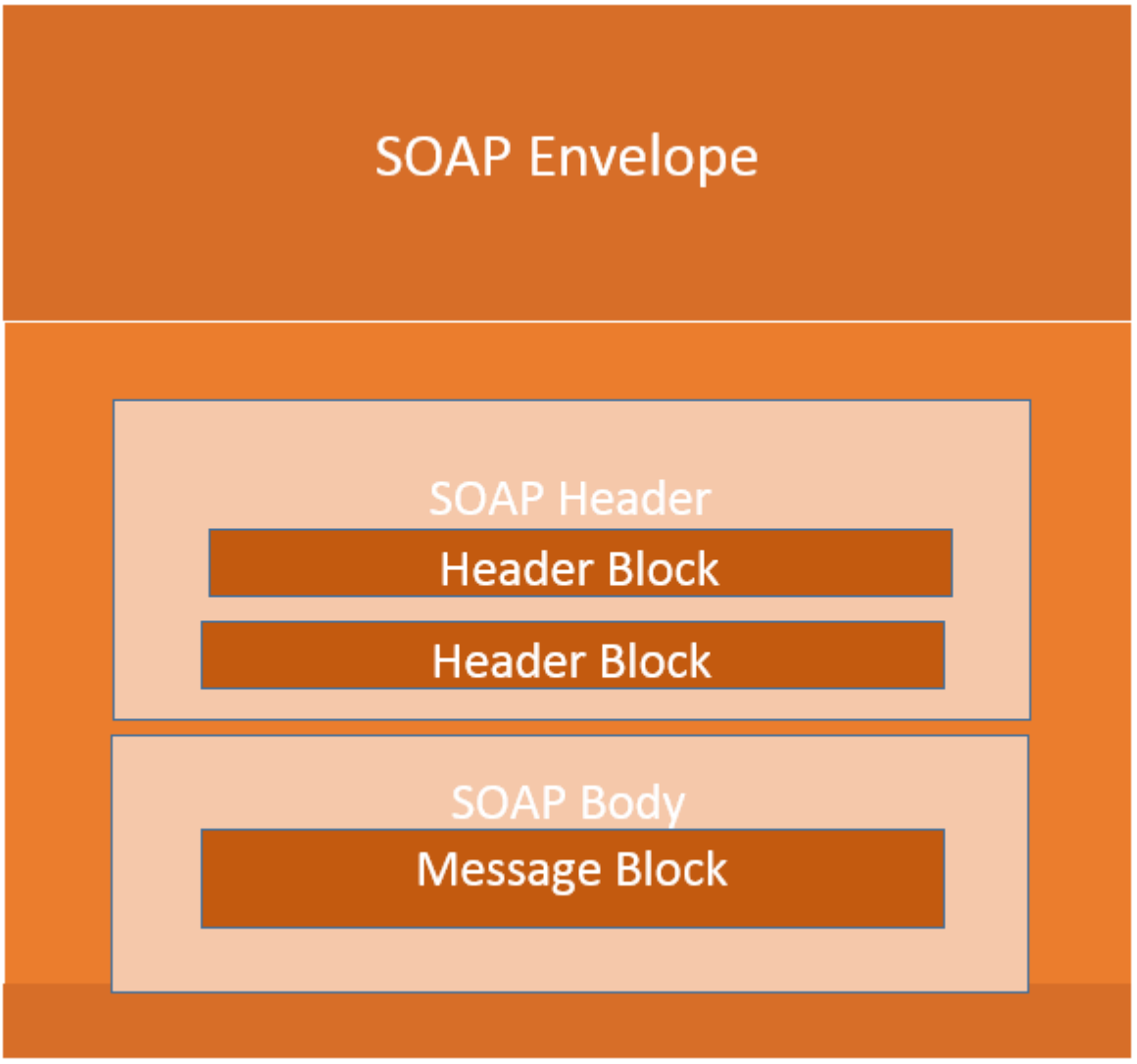
A SOAP message has the following structure:

The root element, called the <Envelope> element, must be present in every SOAP document. This root element is the first element in the XML document.

The Envelope contains two main parts: a header and a body.

The header contains routing data that specifies which client should receive the XML document.

The body contains the actual message.



XML Request Sample:

```
<Envelop xmlns=?http://schemas.xmlsoap.org/soap/envelop/?>
   <Body>
      <getCourseDetailRequest xmlns=?http://udemy.com/course?>
         <id>course1</id>
      <getCourseDetailRequest>
   </Body>
</Envelop>
```

💡**Key Features of SOAP**

- Uses XML-based messaging.
- Provides built-in security (WS-Security).
- Heavier due to strict rules and specifications.
- Example: Used in banking, e-commerce, and enterprise solutions.

2. **REST (Representational State Transfer)**

RESTful REpresentational State Transfer) web services aim to improve web service efficiency by utilizing existing HTTP concepts. Rather than being a protocol, REST is an architectural style.

REST does not enforce a standard message exchange format. While both XML and JSON can be used to build REST services, JSON is the more popular choice. The key concept in REST is a resource, which can be any entity. Each resource is accessed through a Uniform Resource Identifier (URI) and can have multiple representations, such as XML, HTML, and JSON.

💠**The important HTTP methods are:**

GET: Reads a resource

PUT: Updates an existing resource

POST: Creates a new resource

DELETE: Removes a resource

💠**Here are examples of these methods in a social media application:**

POST /users: Creates a new user

GET /users/{id}: Retrieves details of a specific user

GET /users: Retrieves details of all users

DELETE /users: Deletes all users

DELETE /users/{id}: Deletes a specific user

GET /users/{id}/posts/post_id: Retrieves details of a specific post

POST /users/{id}/posts: Creates a post for a user

GET /users/{id}/posts: Retrieves all posts for a user

💠**HTTP also defines the following standard status code:**
404: RESOURCE NOT FOUND
200: SUCCESS
201: CREATED
401: UNAUTHORIZED
500: SERVER ERROR

RESTful Service Constraints:

- The service requires both a producer and consumer
- The service maintains no state between requests

- Service responses must be cacheable
- The interface must be uniform and resource-oriented
- The service architecture must follow a layered design

Advantages of RESTful Web Services:
RESTful web services are platform-independent and can be written in any programming language and executed on any platform. They support multiple data formats including JSON, text, HTML, and XML. These services are faster than SOAP due to their simpler specifications. Additionally, RESTful services are both reusable and language-neutral.

💡**Key Features of REST**

- Uses HTTP methods like GET, POST, PUT, DELETE.
- Can exchange data in XML, JSON, or other formats.
- Lightweight and faster than SOAP.
- Example: Used in social media APIs, e-commerce, and IoT.

Advantages of Web Services

✔️ Facilitates **machine-to-machine communication**.

✔️ **Scalable and flexible** for cloud applications.

✔️ Reduces **integration complexity**.

✔️ Supports **automated workflows** in cloud environments.

6.2 Google App Engine (GAE)

Google App Engine (GAE) is a **Platform-as-a-Service (PaaS)** that allows developers to build, deploy, and manage applications without worrying about infrastructure.

App Engine is a fully managed, serverless platform for developing and hosting web applications at scale. It lets developers choose from several popular programming languages, libraries, and frameworks while handling all server provisioning and scaling automatically.

Operating from Google-managed data centers, App Engine runs applications in secure, sandboxed environments across multiple servers. It automatically scales resources as traffic increases, ensuring optimal performance under varying loads.

The platform primarily supports Go, PHP, Java, Python, Node.js, .NET, and Ruby, with additional language support available through custom runtimes. While the standard environment offers a free tier, the flexible environment is paid-only. Beyond the free tier, users pay for additional storage, bandwidth, and instance hours.

App Engine provides enterprise-grade reliability with a 99.95% uptime SLA for all paid applications. Its robust architecture can withstand multiple datacenter outages without service interruption—demonstrated by the High Replication Datastore's perfect uptime record over a one-year period.

Key Features of Google App Engine

✔️ **Serverless** – No need to manage infrastructure.

✔️ **Auto-scaling** – Adjusts resources based on traffic.

✔️ **Built-in security** – Automatic updates and monitoring.

✔️ **Supports multiple languages** – Java, Python, Go, Node.js.

✔️ **Pay-per-use pricing** – Cost depends on actual resource usage.

Google App Engine Architecture

- **Frontend Instances** – Handle incoming user requests.
- **Backend Services** – Process background tasks.
- **Datastore & Cloud SQL** – Store structured and unstructured data.
- **Task Queues** – Manage background jobs.
- **Memcache** – Improves performance with caching.

Use Cases

- **Web applications** (E-commerce, SaaS applications).
- **Mobile app backends**.
- **Machine Learning applications** (using Google AI tools).
- **IoT and real-time applications**.

6.3 Microsoft Azure Platform

Microsoft Azure is a **comprehensive cloud computing platform** that provides IaaS, PaaS, and SaaS solutions for businesses and developers.

Microsoft Azure, formerly known as Windows Azure, is Microsoft's public cloud computing platform. It provides a wide range of cloud services, including compute, analytics, storage, and networking. Users can select these services to develop and scale new applications or run existing applications in the public cloud. The Azure platform helps businesses manage challenges and meet their organizational goals.

Azure supports all industries—including e-commerce, finance, and many Fortune 500 companies—and is compatible with open-source technologies. It offers four different forms of cloud computing: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), and serverless computing. Microsoft uses a pay-as-you-go pricing model, where subscribers receive monthly bills based only on their actual resource usage.

How does Microsoft Azure work?

After subscribing to Azure, customers gain access to all services in the Azure portal. They can use these services to create cloud-based resources like virtual machines (VM) and databases. Third-party vendors also offer their software through Azure. The pricing for third-party applications typically includes both a subscription fee and usage charges for the underlying infrastructure.

Microsoft offers five levels of customer support:

- Basic
- Developer
- Standard
- Professional Direct
- Premier

Azure's customer support plans differ in scope and pricing. While Basic support comes free with all Azure accounts, other tiers require monthly fees: Developer support at $29, Standard support at $100, and Professional Direct support at $1,000. Microsoft maintains private pricing for its Premier support tier.

What is Microsoft Azure used for?

One of the most popular uses for Microsoft Azure is running virtual machines or containers in the cloud. These compute resources can host infrastructure components like DNS servers, Windows Server services (such as IIS), and third-party applications. Azure also serves as a powerful platform for cloud-based databases. It offers both serverless relational databases like Azure SQL and non-relational databases like NoSQL. The platform is also widely used for backup and disaster recovery, with many organizations utilizing Azure storage as an archive to meet their long-term data retention requirements.

Azure products and services

1. **Compute:** These services enable users to deploy and manage VMs, containers, and batch jobs while supporting remote application access. Compute resources within Azure cloud can be configured with public or private IP addresses, depending on whether the resource needs external accessibility.
2. **Mobile:** These products help developers build cloud applications for mobile devices. They provide notification services, back-end task support, tools for building APIs, and the ability to integrate geospatial context with data.
3. **Web:** These services enable web application development and deployment, providing features for search, content delivery, API management, notifications, and reporting.
4. **Storage:** This category delivers scalable cloud storage for both structured and unstructured data, supporting big data projects, persistent storage, and archival storage.
5. **Analytics:** These services offer distributed analytics and storage capabilities, featuring real-time analytics, big data processing, data lakes, machine learning (ML), business intelligence (BI), IoT data streams, and data warehousing.
6. **Networking:** This group includes virtual networks, dedicated connections, and gateways, along with services for traffic management, diagnostics, load balancing, DNS hosting, and network protection against distributed denial-of-service (DDoS) attacks.
7. **Media and Content Delivery Network (CDN)**: These services provide on-demand streaming, digital rights protection, encoding, media playback, and

content indexing.

8. **Integration:** These services enable server backup, site recovery, and connectivity between private and public clouds.
9. **Identity:** These services ensure only authorized users can access Azure services and help protect encryption keys and other sensitive information in the cloud. They include support for Azure Active Directory and multifactor authentication (MFA).
10. **Internet of Things (IoT):** These services enable users to capture, monitor, and analyze IoT data from sensors and other devices. Features include notifications, analytics, monitoring, and support for coding and execution.
11. **DevOps:** This group provides project and collaboration tools like Azure DevOps—formerly Visual Studio Team Services—that facilitate DevOps software development processes. It includes features for application diagnostics, DevOps tool integrations, and test labs for build testing and experimentation.
12. **Development:** These services help application developers share code, test applications, and track potential issues. Azure supports multiple programming languages—including JavaScript, Python, .NET, and Node.js. The tools also include Azure DevOps, software development kits (SDKs), and blockchain support.
13. **Security:** These products help identify and respond to cloud security threats while managing encryption keys and other sensitive assets.
14. **Artificial Intelligence (AI) and Machine Learning:** This comprehensive suite of services enables developers to integrate AI, machine learning, and cognitive computing capabilities into their applications and data sets.

Key Azure Services

1. **Compute Services**
   - **Azure Virtual Machines** – Run Windows/Linux VMs.
   - **Azure Kubernetes Service (AKS)** – Manage containers.
2. **Storage Services**
   - **Azure Blob Storage** – Store unstructured data.
   - **Azure Files** – Cloud-based file sharing.

3. **Database Services**
    - **Azure SQL Database** – Managed relational database.
    - **Cosmos DB** – NoSQL database for global scalability.
4. **AI & Machine Learning**
    - **Azure Cognitive Services** – Pre-built AI models.
    - **Azure Machine Learning** – Build custom ML models.
5. **Networking & Security**
    - **Azure Virtual Network** – Secure cloud networking.
    - **Azure Active Directory (AAD)** – Identity management.

Benefits of Azure

✔️ **Highly scalable & flexible**.

✔️ **Strong integration** with Microsoft products (Office 365, Windows Server).

✔️ **Enterprise-grade security & compliance**.

✔️ **Multi-cloud and hybrid cloud support**.

Use Cases

- **Hosting enterprise applications** (ERP, CRM).
- **AI & Machine Learning applications**.
- **Big data analytics and IoT**.

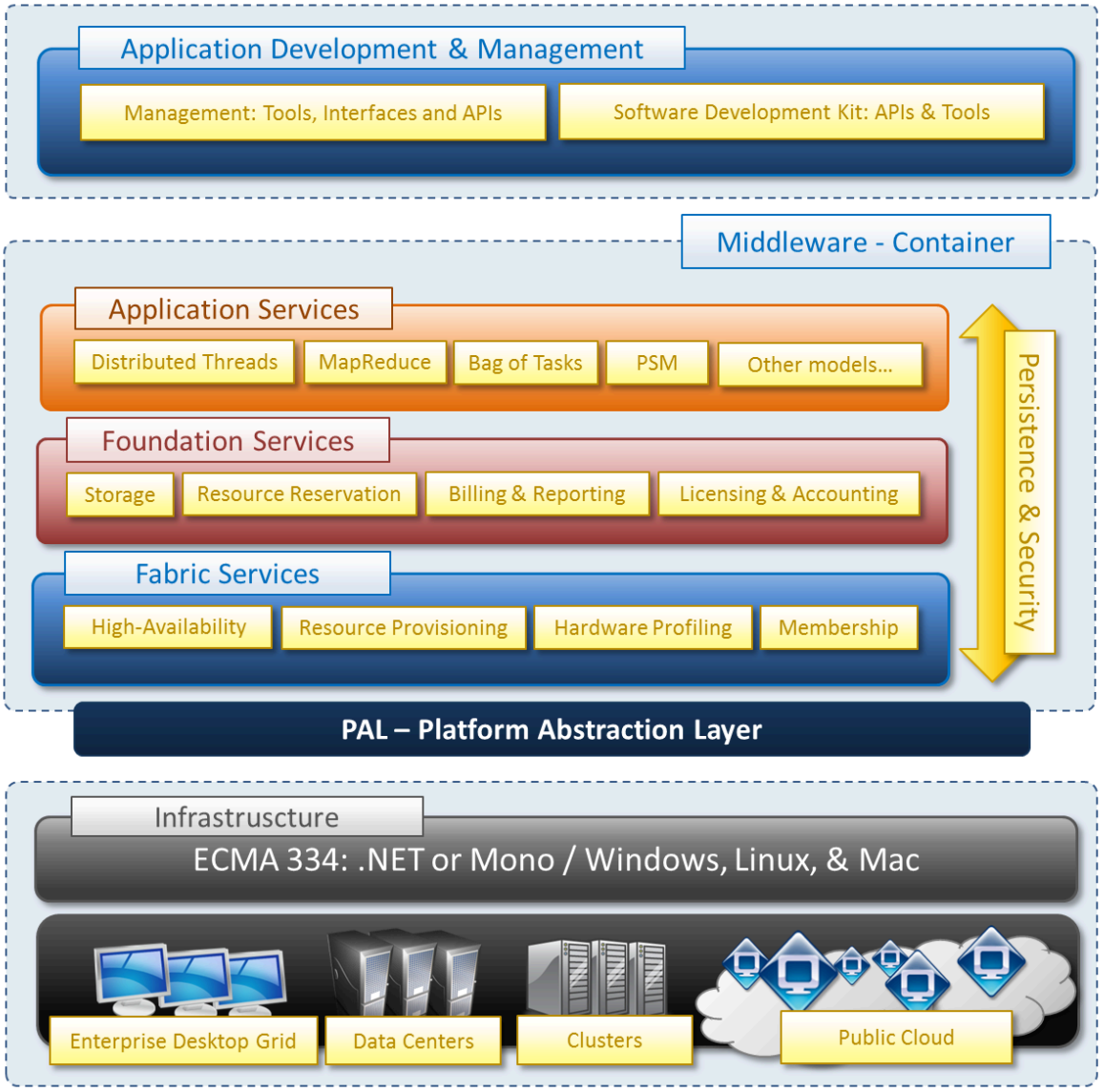
6.4 Aneka: A Multi-Cloud Application Platform

Aneka is a **multi-cloud application platform** developed by **Manjrasoft** that supports hybrid cloud computing.

Aneka is a platform and framework for developing distributed applications in the Cloud. It efficiently utilizes spare CPU cycles from a diverse network of desktop PCs, servers, and datacenters on demand. Through its rich set of APIs, Aneka enables developers to seamlessly access these resources and express application business logic using their preferred programming abstractions. System administrators can use its comprehensive toolset to monitor and control

the deployed infrastructure. The platform works with both public clouds—accessible to anyone via the Internet—and private clouds consisting of nodes with restricted access.

The Aneka-based computing cloud consists of physical and virtualized resources connected through either the Internet or a private intranet. Each resource hosts an Aneka Container instance, which provides the runtime environment for distributed applications. The container manages basic features of each node and coordinates operations across its hosted services. These services are categorized into three types: fabric, foundation, and execution/application services.



1. **Fabric services:**
   These services form the lowest level of the software stack in the Aneka Container, providing access to resource provisioning and monitoring systems.
2. **Foundation services:**
   These core services manage the basic infrastructure of the Aneka Cloud and handle the logical operations of the distributed system. They provide essential support for running distributed applications.
3. **Application services:**
   These services manage application execution and adapt based on the programming model used for developing distributed applications on Aneka.

Features of Aneka

✔️ **Supports multiple programming models** – Task computing, Thread computing, and MapReduce.

✔️ **Works on hybrid cloud environments**.

✔️ **Multi-cloud deployment** – Runs on AWS, Azure, private clouds.

✔️ **Resource scheduling & monitoring**.

Use Cases

- **Big Data processing** – Using MapReduce in cloud environments.
- **Scientific research & simulations**.
- **Cloud-based business applications**.

6.5 Open Challenges in Cloud Computing

Major Challenges

1. **Security & Privacy Concerns**
   - Risk of **data breaches, identity theft, and cyber attacks**.
   - Need for **strong encryption and security policies**.
2. **Data Governance & Compliance**
   - Organizations must comply with **GDPR, HIPAA, and other regulations**.

- Need for **proper data governance frameworks**.

3. **Service Downtime & Reliability**
   - Cloud providers experience **outages**, affecting businesses.
   - Companies must have **disaster recovery plans**.
4. **Vendor Lock-in**
   - Migrating from one cloud provider to another is difficult.
   - **Multi-cloud strategies** help mitigate this risk.
5. **Performance & Latency Issues**
   - **High latency** affects real-time applications.
   - **Edge computing** is a solution to reduce delays.

6.6 Cloud Applications in Different Fields

1. Scientific Applications

Cloud computing is widely used in **scientific research and high-performance computing (HPC)**:

✔️
Genome Sequencing – DNA analysis using cloud-based AI.

✔️ **Climate Modeling** – Simulating climate change patterns.

✔️ **Astronomy & Space Research** – Processing satellite images.

2. Business Applications

Cloud computing plays a key role in **business operations**:

- **Customer Relationship Management (CRM)** – Salesforce, HubSpot.
- **Enterprise Resource Planning (ERP)** – SAP, Oracle ERP Cloud.
- **Cloud-Based Collaboration** – Microsoft 365, Google Workspace.

3. Consumer Applications

Consumers use cloud computing daily in:

- **Cloud Storage** – Google Drive, Dropbox.
- **Streaming Services** – Netflix, YouTube, Spotify.

- **E-commerce Platforms** – Amazon, eBay, Shopify.
- **Social Media & Messaging** – Facebook, Instagram, WhatsApp.