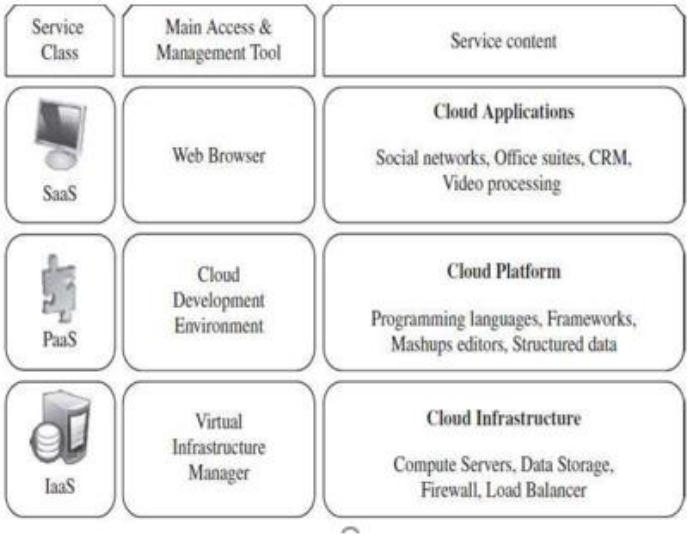
5.2 CLOUD SERVICE MODELS: IAAS, PAAS, SAAS

Cloud computing provides IT resources and services over the internet on a pay-as-you-go basis. The services are categorized mainly into three models:

1. **Infrastructure as a Service (IaaS)**
2. **Platform as a Service (PaaS)**
3. **Software as a Service (SaaS)**



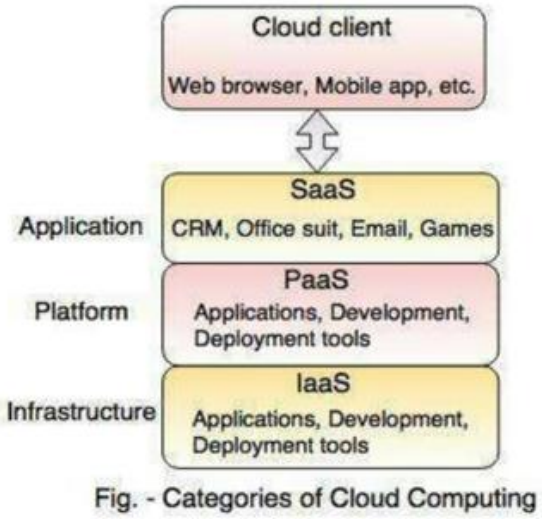


Fig. - Categories of Cloud Computing

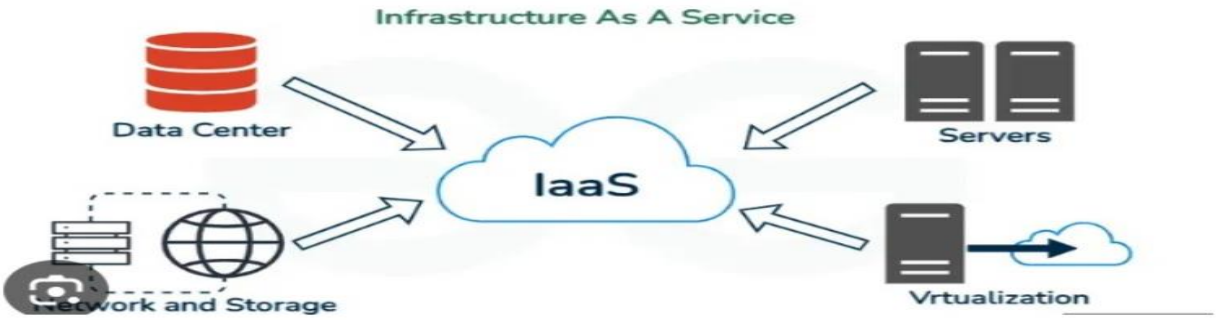
1.Infrastructure as a Service (IaaS)

Definition:

IaaS, or Infrastructure as a Service, is a cloud computing model that provides virtualized computing resources such as **servers, storage, and networking** on demand. Instead of buying and managing physical servers, users rent infrastructure from a cloud provider. Example: Amazon Web Services(AWS), Google Compute Engine(GCE).

virtualization—over the internet on a pay-as-you-go basis. This model allows companies to scale their resources flexibly, reduce capital expenditures, and avoid the complexities of managing physical hardware. Instead of buying and maintaining their own data centers, users can rent these

virtualized resources from a cloud provider and are responsible for installing and maintaining their operating systems and applications.



Working:

- Instead of buying physical hardware, users rent infrastructure from cloud providers (like AWS EC2, Google Compute Engine, Microsoft Azure VM).
- The provider manages hardware, networking, and storage.
- The customer installs and manages operating systems, applications, and middleware.

Example workflow:

- You rent a VM (Virtual Machine).
- You install Linux/Windows OS.
- You set up databases, web servers, or any app you want.
- Provider ensures uptime, power, cooling, and hardware repairs.

Key Features

- Provides virtual machines (VMs) and storage.
- Highly scalable and flexible.
- Pay-as-you-use pricing.
- Supports multiple users through virtualization.
- Self-service provisioning (users can create or delete resources).

Advantages

- Eliminates the cost of buying physical hardware.
- Provides on-demand scalability.
- Complete control over operating systems and applications.
- Suitable for organizations with fluctuating workloads.

Disadvantages

- Users are responsible for managing software and security.
- Requires technical expertise.
- Vendor dependency may exist.

Examples

- Amazon Web Services (AWS EC2, S3)
- Microsoft Azure Virtual Machines
- Google Compute Engine (GCE)
- IBM Cloud Infrastructure

Use Cases

- Hosting websites and applications.
- Storage, backup, and recovery solutions.
- Test and development environments.
- High-performance computing (HPC) tasks.

2. Platform as a Service (PaaS)

Definition

PaaS provides a ready-to-use platform for developers to build, test, and deploy applications without managing infrastructure.

Working:

- Developers write code using the platform tools.

- No need to worry about OS setup, database installation, or server management.
- The provider manages infrastructure + platform, you only focus on coding and deploying.

Example workflow:

- You write a web app in Python.
- Upload the code to a PaaS platform (like Google App Engine, Heroku, AWS Elastic Beanstalk).
- The platform automatically handles scaling, load balancing, and runtime environment.

Key Features

- Pre-installed OS, middleware, and development tools.
- Supports multiple programming languages.
- Automatic scaling and load balancing.
- Integrated database management systems.
- Built-in collaboration tools for developers.

Advantages

- Speeds up application development.
- Reduces complexity of managing infrastructure.
- Enables easy collaboration among development teams.
- Cost-effective for startups and developers.

Disadvantages

- Limited flexibility compared to IaaS.
- Application may depend on a specific platform.
- Migration to another PaaS can be difficult.

Examples

- Google App Engine
- Microsoft Azure App Service
- Heroku

- IBM Cloud Foundry

Use Cases

- Developing mobile or web applications.
- Business intelligence and analytics applications.
- API development and management.
- Continuous integration/continuous deployment (CI/CD).

3. Software as a Service (SaaS)

Definition

SaaS delivers software applications over the internet on a subscription basis. Users can access applications through a web browser without installation.

Working:

- The provider hosts the software and manages everything: infrastructure, platform, updates, and security.
- Users just log in and use the service via web browser or app.

Example workflow:

- You sign in to Gmail or Office 365.
- You don't manage servers, OS, or installation.
- The cloud provider maintains security, updates, and uptime.

Key Features

- It is a Ready-to-use applications.
- It can be access from anywhere (internet-based).
- Automatic software updates and patches.
- Multi-tenant architecture (shared resources among users).
- Subscription-based payment (monthly/annual).

Advantages

- No installation or maintenance required.
- Reduces software piracy.
- Accessible across devices (PC, tablet, mobile).
- Cost-effective for end-users.

Disadvantages

- Limited customization.
- Internet dependency (cannot work offline).
- Data security and privacy concerns.
- Vendor lock-in.

Examples

- Google Workspace (Gmail, Docs, Sheets, Drive)
- Microsoft Office 365
- Zoom
- Dropbox
- Salesforce CRM

Use Cases

- Email and collaboration tools.
- Customer Relationship Management (CRM).
- File storage and sharing.
- Video conferencing and communication.

Comparison: IaaS vs PaaS vs SaaS

Aspect	**IaaS**	**PaaS**	**SaaS**
Definition	Provides virtualized infra	Provides platform for dev	Provides ready-made apps
Users	System admins, IT managers	Developers, programmers	End-users (students, companies)

Aspect	**IaaS**	**PaaS**	**SaaS**
User Control	OS, applications, middleware	Apps and data only	Only usage of software
Provider Control	Hardware, virtualization	Infrastructure + OS	Everything (infra + apps)
Cost	Pay per resource usage	Pay per platform usage	Subscription/License
Examples	AWS EC2, Azure VM, GCE	Heroku, Google App Engine	Gmail, Zoom, Office 365