Cloud vs On-Premises

 Introduction

1. What is On-Premises (On-Prem)?

On-Premises refers to a setup where **all IT infrastructure — servers, storage, networking, and software — is hosted within the organization's own building or data center**.

- The company owns and manages the hardware, software, and network.

- IT staff is responsible for maintenance, updates, security, and backups.

- Data stays physically inside the organization.

Advantages of On-Premises:

- Full control over data and security.

- Highly customizable to business needs.

- Easier compliance with strict regulations like banking, healthcare, or government standards.

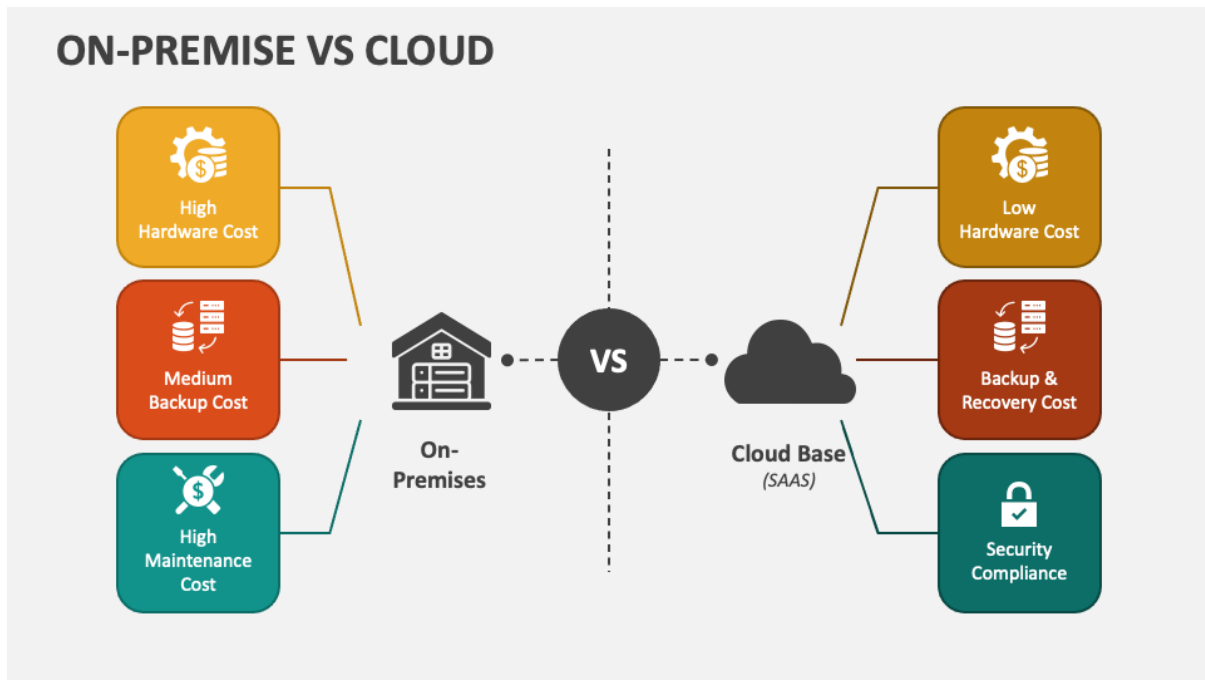**Disadvantages of On-Premises:**

- High upfront cost (buying servers, storage, software licenses).

- Limited scalability — adding new hardware takes time and money.

- Requires IT team for maintenance and troubleshooting.

- Disaster recovery setup is expensive and complex.

Real-World Example:

- A bank hosting customer data in its own data center to comply with regulatory policies.

- Government offices hosting sensitive citizen records internally.

Visual (Flowchart Idea):

User → Company Data Center → Applications & Data

ON-PREMISE VS CLOUD

High Hardware Cost — Low Hardware Cost

Medium Backup Cost — Backup & Recovery Cost

High Maintenance Cost — Security Compliance

On-Premises VS Cloud Base (SAAS)

What is Cloud Computing?

Definition:

Cloud computing is a model where **infrastructure, platforms, or software are hosted by a third-party provider** and accessed via the Internet. Users pay for what they use instead of buying hardware upfront.

Key Features of Cloud:

- Hosted by providers like AWS, Azure, or Google Cloud.

- Accessible from anywhere with an internet connection.

- No need for local servers.

- Can scale resources up or down instantly.

Advantages of Cloud:

1. **Cost Efficiency:** Reduces capital expenditure (CAPEX); pay-as-you-go (OPEX).

2. **Scalability:** Instantly increase or decrease resources as needed.

3. **Maintenance-Free:** Provider handles updates, patches, and hardware.

4. **Global Access:** Employees and customers can access applications remotely.

5. **Quick Deployment:** New services can be deployed in minutes.

Disadvantages of Cloud:

- Less control over physical hardware.

- Dependence on internet connectivity.

- Subscription costs may accumulate over time.

- Data privacy and compliance concerns require additional measures.

Real-World Example:

- A startup hosting a web app on AWS EC2 and storing data on Google Cloud Storage.

- Businesses using Gmail or Microsoft 365 instead of on-prem email servers.

Visual (Flowchart Idea):

User → Internet → Cloud Provider → Applications & Data

Detailed Comparison

Feature	On-Premises	Cloud
Ownership	Company owns hardware	Provider owns infrastructure
Cost Model	High upfront (CAPEX)	Pay-as-you-go (OPEX)
Maintenance	Managed by IT team	Managed by provider
Scalability	Limited by physical hardware	Virtually unlimited
Deployment Speed	Weeks to months	Minutes to hours
Data Security	Full control	Provider-managed; depends on contract
Access	On-site network or VPN	Internet from anywhere

Feature	On-Premises	Cloud
Backup & Recovery	Company manages	Provider may include automated backups
Examples	In-house servers, ERP	AWS, Azure, Google Cloud, SaaS apps

Key Points:

- On-Prem gives full control but is expensive and slow to scale.

- Cloud offers flexibility, speed, and reduced maintenance but less direct control.
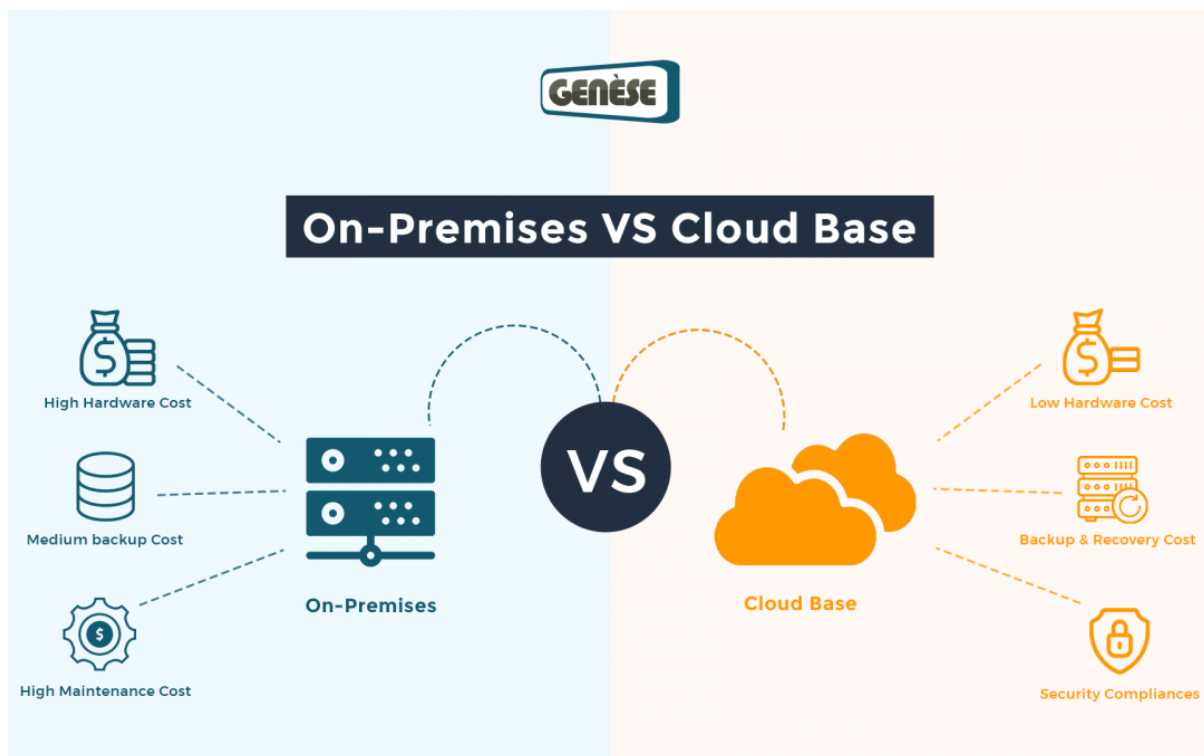
Use Cases & Recommendations

When to Choose On-Premises:

- Businesses needing **full control** over data.

- Companies with **strict compliance or regulatory needs**.

- Organizations that already invested heavily in IT infrastructure.

- Examples: Banks, government agencies, hospitals storing sensitive data.

When to Choose Cloud:

- Startups or small businesses with limited IT budgets.

- Companies needing **fast deployment and scalability**.

- Businesses with a **remote workforce** needing global access.

- Examples: SaaS apps, e-commerce websites, collaborative tools like Google Workspace.

Hybrid Approach:

- Many companies use a **hybrid model**, combining both On-Prem and Cloud.

- Sensitive data remains on-prem, while other applications are moved to the cloud.

- Offers flexibility, security, and scalability.

Conclusion:

- On-Prem is **control-heavy, cost-heavy, and maintenance-heavy**.

- Cloud is **flexible, scalable, and maintenance-light**, ideal for modern businesses.

- Decision depends on **budget, compliance needs, scalability requirements, and IT capabilities**.
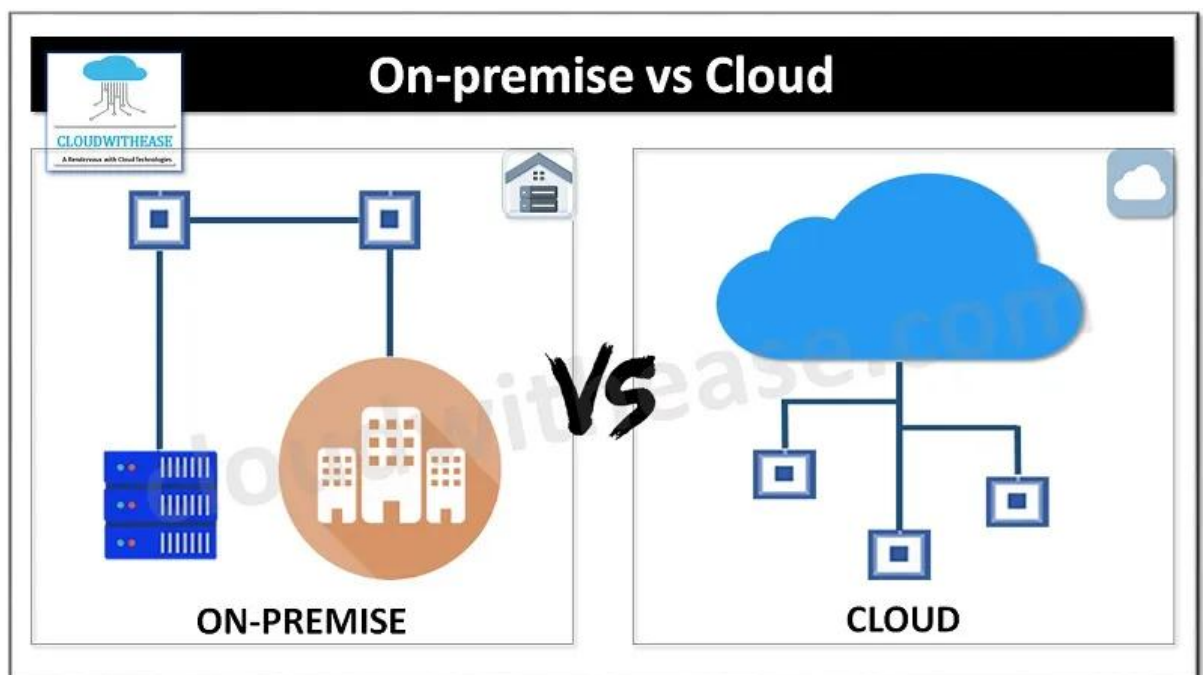
Microsoft Azure and the Role of OS

1. What is Microsoft Azure?

Definition:
Microsoft Azure is a cloud computing platform and service created by Microsoft. It provides a wide range of services such as computing, storage, networking, databases, AI, and analytics, all accessible over the internet.

Key Features of Azure:

- **Compute Services:** Run virtual machines, containers, and app services.

- **Storage Services:** Object storage, file storage, and databases.

- **Networking:** Virtual networks, load balancers, VPN gateways.

- **Security & Identity:** Azure Active Directory, firewalls, security monitoring.

- **AI & Analytics:** Machine learning, cognitive services, data analytics tools.

Advantages:

- **Scalability:** Quickly scale resources up or down.

- **Pay-as-you-go:** Only pay for resources you use.

- **Global Presence:** Azure has data centers worldwide.

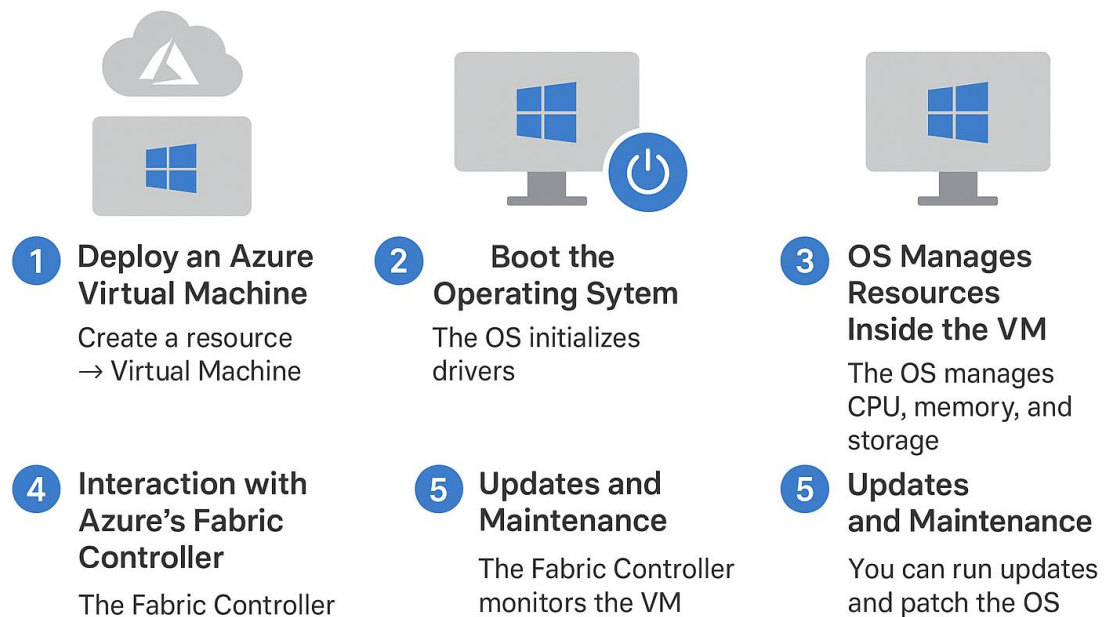- **Integration:** Works seamlessly with Microsoft products like Windows Server, SQL Server, and Office 365.

Example:

- A company can host its website on Azure App Service instead of buying physical servers.

- A startup can use Azure Virtual Machines to run applications without managing hardware.

Visual Concept (Flow):

User → Internet → Azure Cloud → Services (VM, Storage, Database, Apps)

OS Role in an Azure Virtual Machine

1 Deploy an Azure Virtual Machine

Create a resource → Virtual Machine

2 Boot the Operating Sytem

The OS initializes drivers

3 OS Manages Resources Inside the VM

The OS manages CPU, memory, and storage

4 Interaction with Azure's Fabric Controller

The Fabric Controller

5 Updates and Maintenance

The Fabric Controller monitors the VM

5 Updates and Maintenance

You can run updates and patch the OS

2. Role of the Operating System (OS) in Azure

In Azure, the operating system is crucial because it acts as the interface between the physical or virtual hardware and the applications you run. Azure provides multiple ways to work with OS:

a) Virtual Machines (VMs)

- Azure allows you to deploy Windows or Linux virtual machines.

- You can choose the OS you need depending on your application requirements.

- The OS manages resources like CPU, memory, and storage and allows your applications to run efficiently.

Example:

- A Windows Server VM running IIS for web hosting.

- A Linux VM running Apache or Nginx servers.

Visual Concept (Flow):

Azure VM → Operating System (Windows/Linux) → Application → User

b) Platform as a Service (PaaS)

- **When using Azure App Service or Azure Functions, the OS is managed by Azure.**

- **Users don't need to worry about patching, updates, or security of the OS.**

- **Applications run on top of Azure-managed OS environments.**

Example:

- **Deploying a web app on Azure App Service without installing Windows Server or Linux manually.**

Key Point:

- **The OS still exists, but Azure abstracts it so you focus on application development.**

c) Containers and OS

- **Azure supports containers (Docker, Kubernetes), which package applications with minimal OS layers.**

- **The container uses a lightweight OS (or kernel) to run applications efficiently.**

- **OS in this case provides necessary runtime support without full VM overhead.**

Example:

- **Running a Docker container with a Linux base image on Azure Kubernetes Service (AKS).**

d) Security and Updates

- **The OS is responsible for:**

    - **Managing system resources (CPU, memory, disk).**

    - **Controlling network access and user permissions.**

    - **Applying security patches and updates (especially in VMs).**

- **In PaaS or SaaS models, Azure handles most OS-level security tasks for you.**

3. Summary of OS Role in Azure

Aspect	Description
Resource Management	Allocates CPU, memory, disk for apps.
Platform Support	Provides environment for applications to run.
Security & Updates	Applies patches, firewalls, and manages access.
User Control	In IaaS, user controls OS; in PaaS/SaaS, Azure manages OS.
Application Interface	OS acts as a bridge between hardware and applications.

4. Key Takeaways

- **Azure is a comprehensive cloud platform providing IaaS, PaaS, and SaaS services.**

- **OS in Azure can be fully managed by users (VMs) or abstracted by Azure (PaaS/SaaS).**

- **Choosing the right OS depends on your workload, app requirements, and control level needed.**

- **Even in cloud environments, the OS is the backbone that ensures applications run smoothly and securely.**

Different Hosting Techniques
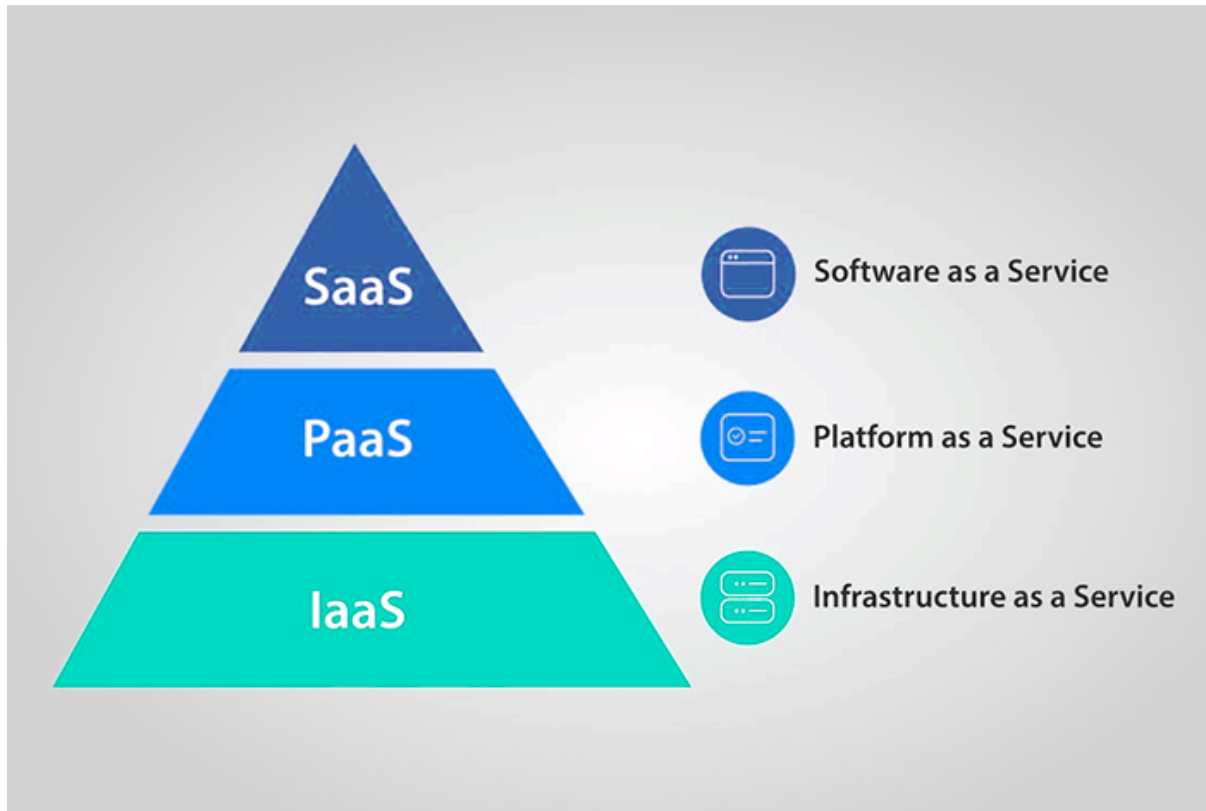
1. Infrastructure as a Service (IaaS)

Definition:
IaaS provides **virtualized computing resources** over the internet. You get access to servers, storage, networking, and virtual machines without managing physical hardware.

Key Features:

- Users manage **applications, data, and OS**.
- Providers manage **hardware, storage, networking, and virtualization**.
- Flexible and scalable.

Advantages:

- No need to buy physical servers.
- Quick provisioning of resources.

- Pay only for what you use.

Disadvantages:

- You must manage OS, apps, and security.

- Requires technical knowledge to configure servers.

Examples: AWS EC2, Microsoft Azure VMs, Google Compute Engine.

Visual Idea:

User → OS & Apps

Provider → Infra (Servers, Storage, Network)

2. Platform as a Service (PaaS)

Definition:
PaaS provides a **platform and environment** to develop, test, and deploy applications. Users focus only on application development while the provider manages the underlying infrastructure.

Key Features:

- Users manage **apps and data**.

- Providers manage **infrastructure, OS, middleware, runtime, and security**.

- Includes development tools and frameworks.

Advantages:

- Faster app development.

- No need to manage servers or middleware.

- Scales automatically with user demand.

Disadvantages:

- Less control over underlying infrastructure.

- Vendor lock-in can be an issue.

Examples: Heroku, Google App Engine, Microsoft Azure App Services.

Visual Idea:

User → Apps & Data

Provider → Infra + OS + Runtime + Middleware

3. Software as a Service (SaaS)

Definition:
SaaS delivers **ready-to-use applications** over the internet. Users don't manage infrastructure, platforms, or software updates — they simply use the application.

Key Features:

- Accessible via browser or app.

- Provider manages everything: infrastructure, updates, security.

Advantages:

- Minimal IT maintenance.

- Quick access to applications.

- Subscription-based cost reduces upfront investment.
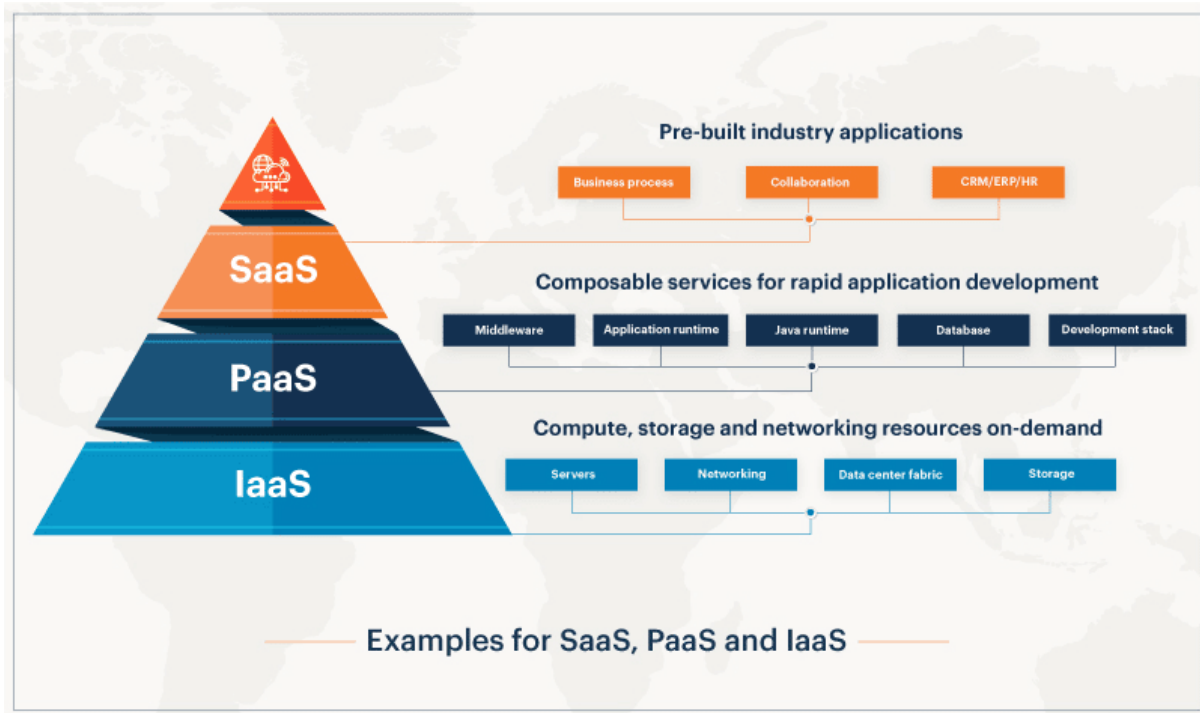
Disadvantages:

- Limited customization.

- Depends on internet connectivity.

Examples: Gmail, Salesforce, Zoom, Dropbox.

Visual Idea:

User → Uses Application

Provider → Infra + Platform + Software

Examples for SaaS, PaaS and IaaS

4. Other Hosting Techniques

a) Hybrid Hosting

- Combines **On-Premises** and **Cloud**.

- Sensitive data remains on-prem, other workloads on cloud.

- Provides **security + scalability**.

b) Dedicated Hosting

- A single physical server dedicated to one organization.

- Full control and performance guarantee.

- More expensive than shared hosting.

c) Shared Hosting

- Multiple customers share a single server.

- Cheapest option, ideal for small websites.

- Limited control and performance may vary.

d) Managed Hosting

- Provider handles setup, maintenance, security, and updates.

- Customer focuses on applications and business logic.

- Can be based on cloud, dedicated, or hybrid servers.

Comparison Table of Cloud Hosting Techniques

Feature	IaaS	PaaS	SaaS
User Manages	OS, Apps, Data	Apps & Data	Only Uses App
Provider Manages	Infra, Storage, Network	Infra + OS + Runtime + Middleware	Infra + Platform + Software
Cost Model	Pay-as-you-go	Pay-as-you-go	Subscription
Examples	AWS EC2, Azure VMs	Heroku, Google App Engine	Gmail, Salesforce, Zoom
Use Case	Flexible server hosting	App development & deployment	End-user ready applications