

## **Cloud Service Types**

Primarily there are three types of cloud service types:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

“As a Service” can be view as “you can rent it for a short time, if you wish” (Example: rent a car.) Also, no commitment, if you wish (start a server at 1:00 pm and close down at 1:00 pm). That is pay for what you use (time, GB, ...) Note: In MS Azure, significant discounts can be obtained by committing for a long time. The cloud provider takes care of buying, developing, and maintaining it.

### **Infrastructure as a Service (IaaS)**

These are essential services of technology:

- Computing
- Storage
- Networking

IaaS Computing:

One example is Azure virtual machines (VM). We will get a Windows/Linux server (computer), and nothing install on it. We can install out own software or third-party software a perform some tasks. Here we pay by the second and have many choices in CPU speeds, RAM, optimizations (optimize for CPU/general purpose).

IaaS Storage:

Azure Storage is one example of this (this is a hard disk/SSD in the cloud). In Azure storage it can give 5 PB of storage capacity and can handle different types of files such as blobs (Binary Large Object), regular files, queues, and storage tables. Note they are very cheap and can also be configured as a data lake.

### IaaS Networking:

Virtual Networking is an example of this, and it does not cost anything. But there are ingress (inward flow of data into a network) and egress (data leaving Azure) bandwidth costs.

### Platform as a Service (PaaS)

On top of the basic infrastructure cloud service providers have an opportunity to provide more. These include a service layer on top of IaaS – computing, storage and networking. The services include Middleware, development tools, database server, and more.

### PaaS Computing:

One example is Azure App Services. Here we simply upload the code (into a Zip file) and configurations to Azure. It runs the code without needing to worry about the VM underneath. It includes additional features such as scaling features, CI/CD, containers, staging and development environments, etc.

### PaaS Storage:

Managed Storage is an example of this. Azure SQL Database can also be considered as another example. When you have a platform as a storage, then you are freed from worrying about the server VM itself or the hard disk itself.

### PaaS Networking:

Azure Front Door (or Load Balancer, Firewall) is an example of this. These are software applications that perform networking tasks.

### Software as a Service (SaaS)

These are Cloud apps/tools such as Office 365, One Drive, Skype. These apps are ready to be used, and you can simply need to set it up and use it.

## Serverless

There are several interpretations. First, you do not manage the servers, and you only handle writing and running the code (that is, servers are provided as a service).

Serverless provided as a consumption-based pricing: you only pay for what you use, no chargers when your app or databased is idle, no need to pre-purchases or reserve resources.

One example is Serverless Azure SQL Database.

Azure SQL Database has multiple pricing options:

- DTU-based (performance teir): Fixed performance levels
- vCore-based: You choose CPU, memory, and storage
- Serverless: Auto-scales CPU & memory based on the usage.

Features of serverless SQL database:

- Azure auto-scale between 0.5 to 80 vCores.
- You are charged per vCore-second and storage used.
- It can pause when idle, cutting costs entirely.

Azure Functions (a way of running codes) is a classic serverless example:

- First 1 million executions per month are free
- After that, you pay per execution and execution time
- Azure handles scaling automatically

Pros of serverless:

- No server management
- Scales automatically (Note: takes time for scaling)
- Cost-efficient for intermittent workloads

Challenges of serverless:

- Cost may vary month-to-month
- Cold starts may cause slight delays (because server may in shutdown state)

Thus, serverless means:

- You focus on code, not infrastructure
- You pay based on actual usage
- It is ideal for event-driven or low-usage workloads

In Azure, following services have some serverless offerings:

- Functions
- Container Apps
- Kubernetes
- SQL Database
- Cosmos DB