Cloud Concepts Study Notes

Andrew Brown

Cloud Computing

"Text-book" definition of **Cloud Computing**:

> *`Cloud Computing` is the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than on a local server or personal computer.*

Cloud computing allows companies or individuals to rent the resources they need for their companies or personal projects like storage space.

Companies no longer have to purchase physical hardware with expensive IT departments.

This has helped a lot of smaller companies launch their projects or services quickly.

Companies can adjust the types of services they need on demand instead of waiting weeks or longer for new servers or other equipment to scale their businesses.

Individuals no longer need a computer with a huge hard drive for photos or files. They can now store these on services like Google Drive.

On-premise vs Cloud Providers

On-premise

- You own the servers
- You hire the IT people
- You pay or rent the real-estate
- You take all the risk.

Cloud Providers

- Someone else owns the servers
- Someone else hires the IT people
- Someone else pays or rents the real-estate
- You are responsible for configuring your cloud services and code, someone else takes care of the rest

Dedicated Server

One physical machine dedicated to single a business

- Runs a single web-app/site.
- Very Expensive, High Maintenance
- High Security

Virtual Private Server

The physical machine is virtualized into sub-machines

- Runs multiple web-apps/sites
- Better Utilization and Isolation of Resources

Shared Hosting

One physical machine, shared by hundred of businesses

- Relies on most tenants under-utilizing their resources.
- Very Cheap, Limited functionality, Poor Isolation

Cloud Hosting

Multiple physical machines that act as one system

- The system is abstracted into multiple cloud services
- Flexible, Scalable, Secure, Cost-Effective, High Configurability

AWS

Amazon calls their cloud provider service **Amazon Web Services** (Commonly referred to just AWS)

AWS was launched in 2006 is the leading **Cloud Service Provider (CSP)** in the world.

A **Cloud Service Provider** - is a company which provides multiple Cloud Services, and those Cloud Services can be chained together to create cloud architectures

- Most commonly through internet-hosted computing, storage, and software services.

Trade-offs and Benefits of Cloud Computing vs On-premise

1st: Trade capital expense for variable expense

- **No upfront-cost:** In the past companies would have to pay for all of data centers and servers
- **Pay On-Demand:** Companies only pay for the resources they consume or use.

2nd: Benefit from massive economics of scale

- AWS and other cloud providers serve a huge market of consumers. The cost of using a cloud service provider like AWS is significantly cheaper because you are sharing the cost with other customers.
- It allows you to have access to resources that would normally be very expensive, but it's a shared cost.

3rd: Stop guessing capacity

- Eliminate guesswork about infrastructure capacity needs.
- **Instead of paying for idle or under-utilized servers,** you can scale up or down to meet the current need.

4th: Increased speed and agility

- Launch resources **within a few clicks in minutes** instead of waiting days or weeks for your IT to implement the solution on- premise.

5th: Stop spending money on running and maintaining data centres

- **Focus on your own customers**, rather than on the heavy lifting of racking, stacking and powering servers.

- If a company doesn't have to build their own data centre they save money on the amount of employees needed.
- Companies save money by not spending money on unnecessary hardware.

6th: Go global in minutes

- Deploy your app in **multiple regions around the world with a few clicks.**
- Provide lower latency and a better experience for your customers at minimal cost.
- In minutes you can quickly run servers in Asia, Europe and North America.

Cloud Computing models

There are three main models of Cloud Computing:

1. SaaS - Software as a Service
2. PaaS- Platform as a Service
3. IaaS- Infrastructure as a Service

IaaS- Infrastructure as a Service (Host)

Most basic building blocks of cloud IT infrastructure. It has the most flexibility and management control of all the different cloud computing models. It is the closest to having a traditional on-premises data center.

Examples:

- Amazon Web Services
- Microsoft Azure
- Google Cloud

PaaS- Platform as a Service (Build)

Provides a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining the infrastructure typically associated with developing and launching an app. Less flexibility than IaaS because of pre-constructed packages.

Examples:

- AWS Elastic Beanstalk
- Windows Azure
- Heroku
- Google App Engine

SaaS- Software as a Service (Consume)

Completed projects managed by a service provider. Easy to use and comes complete with a user interface. Least flexibility

Examples:

- Gmail
- Adobe Creative Cloud

The Shared Responsibility Model

Customer Responsibilities

On-Premise	IaaS	PaaS	SaaS
Applications	Applications	Applications	
Data	Data	Data	
Runtime	Runtime		
Middleware	Middleware		
OS	OS		
Virtualization			
Servers			
Storage			
Networking			

Cloud Development Models

There are three different types of deployment models for Cloud Computing.

Cloud

When you think of **Cloud**, think of **small startups.**

Cloud Deployment

- 100% of IT infrastructure is on the cloud.
- All of a companies applications were migrated to or created on the cloud.
- Helps to remove roadblocks of costly and time consuming procurement processes for on-premises infrastructure (big servers and data centers!).

- Great for small businesses and start-ups.

Hybrid

When you think of **Hybrid**, think of **Fin Tech Companies**

Hybrid Deployment

- Connects on-premises technology with cloud-based resources
- Great for established companies that had a dedicated data center but also wants to migrate processes over to the cloud
- Data is partially on the cloud, and partially in the on-premises
- Popular in the Fin Tech space

On-Premise

hen you think of **On-premise**, think of **large old companies** or companies that are beholden to **Government Regulations**.

On-premises (Private cloud) Deployment

- Use virtualization to deploy resources in their on-premises data centers.
- Resembles traditional IT infrastructure with big servers, data centers, etc.
- Do not get the same benefits of cloud computing (ability to easily scale up and down on demand)
- Company has dedicated resources that are not shared with others (good for security)
- Resources cannot be accessed using the internet
- Typically older large companies or owned by Government organizations

What is Total Cost of Ownership?

Total cost of ownership (TCO) is a financial estimate intended to help buyers and owners determine the direct and indirect costs of a product or system. It is a

management accounting concept that can be used in full cost accounting or even ecological economics where it includes social costs.

- AWS Provides the **TCO Calculator** service to help companies with this calculation

Capital Expenditure (CAPEX)

Spending money upfront on physical infrastructure Deducting that expense from your tax bill over time.

- Server Costs (computers)
- Storage Costs (hard drives)
- Network Costs (Routers, Cables, Switches)
- Backup and Archive Costs
- Disaster Recovery Costs
- Datacenter Costs (Rent, Cooling, Physical Security)
- Technical Personal

With Capital Expenses, you have to guess upfront what you plan to spend

Operational Expenditure (OPEX)

The costs associated with an on-premises data center that has shifted the cost to the service provider. The customer only has to be concerned with non-physical costs.

- Leasing Software and Customizing features
- Training Employees in Cloud Services
- Paying for Cloud Support
- Billing based on cloud metrics eg.
  - compute usage
  - storage usage

With Operation Expenses you can try a product or service without investing in equipment

Cloud Architecture Terminologies

- **Availability** Your ability to ensure service remains available - Highly Available (HA)
- **Scalability** Your ability to grow rapidly or unimpeded
- **Elasticity** Your ability to shrink and grow to meet the demand
- **Fault Tolerance** Your ability to prevent a failure
- **Disaster Recovery** Your ability to recover from failure - Highly Durable (DR)

What is High Availability (HA)?

Your ability for your service to remain available by ensuring there is no single point of failure and/or ensure a certain level of performance

How can you achieve High Availability?

Running your workload across multiple Availability Zones ensures that if 1 or 2 AZs become unavailable your service/applications remain available.

How can High Availability be implemented on AWS?

Using **Elastic Load Balancer** would assist in implementing High Availability

> *A load balancer allows you to evenly distribute traffic to multiple servers in one or more datacenters. If a datacenter or server becomes unavailable (unhealthy) the load balancer will route the traffic to only available datacenters with servers.*

What is High Scalability?

Your ability to increase your capacity based on the increasing demand of traffic, memory and computing power

How is High Scalability defined?

- Vertical Scaling is known as Scaling Up (When Upgrade to a bigger server)
- Horizontal Scaling is known as Scaling Out (When Add more servers of the same size)

What is High Elasticity?

Your ability to automatically increase or decrease your capacity based on the current demand of traffic, memory and computing power

How is Elasticity achieved?

Elasticity relies on Horizontal Scaling. Vertical Scaling is generally hard for traditional architecture so you'll usually only see horizontal scaling described with Elasticity.

Scaling Out — Add more servers of the same size Scaling In — Removing more servers of the same size

How can Elasticity be implemented in AWS?

Auto Scaling Groups (ASG) - are an AWS feature that will automatically add or remove servers based on scaling rules you define.

What is Highly Fault Tolerant?

The ability for your service to ensure there is no single point of failure.

What is a Fail-over?

Fail-overs are when you have a plan to shift traffic to a redundant system in case the primary system fails

How can Fault Tolerance be achieved?

A common example is having a copy (secondary) of your database where all ongoing changes are synced. The Secondary is not in-use until a failover occurs and it becomes the primary database.

How can Fault Tolerance be implemented using AWS?

RDS Multi-AZ - is when you run a duplicate standby database in another Availability Zone in case your primary database fails.

What is High Durability?

Your ability to recover from a disaster and to prevent the loss of data Solutions that recover from a disaster is known as Disaster Recovery (DR)

Questions you should be asking to about your Disaster Recovery procedures:

- Do you have a backup?
- How fast can you restore that backup?
- Does your backup still work?
- How do you ensure current live data is not corrupt?